Lecture Notes
Institute of Bioinformatics, Johannes Kepler University Linz

Theoretical Concepts of Machine Learning

Summer Semester 2014

by Sepp Hochreiter

Institute of Bioinformatics
Johannes Kepler University Linz
A-4040 Linz, Austria

Tel. +43 732 2468 8880
Fax +43 732 2468 9308
http://www.bioinf.jku.at

Literature

- V. N. Vapnik; Statistical Learning Theory, Wiley & Sons, 1998
- Schölkopf, Smola; Learning with kernels, MIT Press, 2002
- S. M. Kay; Fundamentals of Statistical Signal Processing, Prentice Hall, 1993
- M. I. Jordan (ed.); Learning in Graphical Models, MIT Press, 1998 (Original by Kluwer Academic Pub.)
- Duda, Hart, Stork; Pattern Classification; Wiley & Sons, 2001
- C. M. Bishop; Neural Networks for Pattern Recognition, Oxford University Press, 1995
- C. M. Bishop; Pattern Recognition, Oxford University Press, 2008
- A. Dobson; An Introduction to Generalized Linear Models, 2nd edition, ISBN: 1-58488-165-8, Series: Texts in Statistical Science, Chapman & Hall / CRC, Boca Raton, London, New York, Washington D.C., 2002.
- A. C. Rencher and G. B. Schaalje; Linear Models in Statistics, 2nd edition, Wiley, Hoboken, New Jersey, USA, 2008.

Contents

1 Introduction **1**
- 1.1 Machine Learning Introduction . . . 1
- 1.2 Course Specific Introduction . . . 2

2 Generalization Error **5**
- 2.1 Model Quality Criteria . . . 5
- 2.2 Introducing the Generalization Error . . . 6
  - 2.2.1 Definition of the Generalization Error / Risk . . . 6
  - 2.2.2 Empirical Estimation of the Generalization Error . . . 8
    - 2.2.2.1 Test Set . . . 8
    - 2.2.2.2 Cross-Validation . . . 8
- 2.3 Minimal Risk for a Gaussian Classification Task . . . 11

3 Maximum Likelihood **19**
- 3.1 Loss for Unsupervised Learning . . . 19
  - 3.1.1 Projection Methods . . . 19
  - 3.1.2 Generative Model . . . 20
  - 3.1.3 Parameter Estimation . . . 20
- 3.2 Mean Squared Error, Bias, and Variance . . . 21
- 3.3 Fisher Information Matrix, Cramer-Rao Lower Bound, and Efficiency . . . 23
- 3.4 Maximum Likelihood Estimator . . . 25
- 3.5 Properties of Maximum Likelihood Estimator . . . 26
  - 3.5.1 MLE is Invariant under Parameter Change . . . 26
  - 3.5.2 MLE is Asymptotically Unbiased and Efficient . . . 26
  - 3.5.3 MLE is Consistent for Zero CRLB . . . 27
- 3.6 Expectation Maximization . . . 28
- 3.7 Maximum Entropy Estimation . . . 31

4 Noise Models **35**
- 4.1 Gaussian Noise . . . 35
- 4.2 Laplace Noise and Minkowski Error . . . 37
- 4.3 Binary Models . . . 37
  - 4.3.1 Cross-Entropy . . . 38
  - 4.3.2 Logistic Regression . . . 39
  - 4.3.3 (Regularized) Linear Logistic Regression is Strictly Convex . . . 43
  - 4.3.4 Softmax . . . 43

4.3.5 (Regularized) Linear Softmax is Strictly Convex 44

5 Statistical Learning Theory **47**
5.1 Error Bounds for a Gaussian Classification Task 48
5.2 Empirical Risk Minimization 49
5.2.1 Complexity: Finite Number of Functions 49
5.2.2 Complexity: VC-Dimension 51
5.3 Error Bounds 56
5.4 Structural Risk Minimization 60
5.5 Margin as Complexity Measure 61
5.6 Average Error Bounds for SVMs 66

6 Theory of Kernels and Dot Products **69**
6.1 Kernels, Dot Products, and Mercer's Theorem 69
6.2 Reproducing Kernel Hilbert Space 72

7 Optimization Techniques **75**
7.1 Parameter Optimization and Error Minimization 75
7.1.1 Search Methods and Evolutionary Approaches 75
7.1.2 Gradient Descent 77
7.1.3 Step-size Optimization 80
7.1.3.1 Heuristics 80
7.1.3.2 Line Search 83
7.1.4 Optimization of the Update Direction 84
7.1.4.1 Newton and Quasi-Newton Method 84
7.1.4.2 Conjugate Gradient 86
7.1.5 Levenberg-Marquardt Algorithm 90
7.1.6 Predictor Corrector Methods 91
7.1.7 Convergence Properties 92
7.2 On-line Optimization 94
7.3 Convex Optimization 96

8 Bayes Techniques **101**
8.1 Likelihood, Prior, Posterior, Evidence 101
8.2 Maximum A Posteriori Approach 103
8.3 Posterior Approximation 105
8.4 Error Bars and Confidence Intervals 106
8.5 Hyper-parameter Selection: Evidence Framework 108
8.6 Hyper-parameter Selection: Integrate Out 112
8.7 Model Comparison 114
8.8 Posterior Sampling 115

9 Linear Models **117**
9.1 Linear Regression 117
9.1.1 The Linear Model 117
9.1.2 Interpretations and Assumptions 118
9.1.2.1 Interpretations 118

9.1.2.2 Assumptions . . . 120
9.1.3 Least Squares Parameter Estimation . . . 122
9.1.4 Evaluation and Interpretation of the Estimation . . . 123
9.1.4.1 Residuals and Error Variance . . . 123
9.1.4.2 Coefficient of determination . . . 124
9.1.4.3 Outliers and Influential Observations . . . 124
9.1.4.3.1 Outliers. . . . 124
9.1.4.3.2 Influential Observations. . . . 125
9.1.5 Confidence Intervals for Parameters and Prediction . . . 126
9.1.5.1 Normally Distributed Error Terms . . . 126
9.1.5.2 Error Term Distribution Unknown . . . 127
9.1.6 Tests of Hypotheses . . . 128
9.1.6.1 Test for a Set of Variables Equal to Zero . . . 128
9.1.6.2 Test for a Single Variable Equal to Zero . . . 129
9.1.7 Examples . . . 130
9.1.7.1 Hematology Data . . . 130
9.1.7.1.1 Computing Estimates, Confidence Intervals, Tests. . . . 130
9.1.7.2 Carbohydrate Diet Data . . . 134
9.2 Analysis of Variance . . . 137
9.2.1 One Factor . . . 137
9.2.2 Two Factors . . . 139
9.2.3 Examples . . . 144
9.2.3.1 Dried Plant Weights . . . 144
9.2.3.2 Extended Dried Plants . . . 147
9.2.3.3 Two-Factor ANOVA Toy Example . . . 149
9.3 Analysis of Covariance . . . 152
9.3.1 The Model . . . 152
9.3.2 Examples . . . 153
9.3.2.1 Achievement Scores . . . 153
9.3.2.2 Birthweights of Girls and Boys . . . 155
9.4 Mixed Effects Models . . . 160
9.4.1 Approximative Estimator . . . 160
9.4.1.1 Estimator for Beta . . . 161
9.4.1.2 Estimator for u . . . 162
9.4.2 Full Estimator . . . 163
9.5 Generalized Linear Models . . . 163
9.5.1 Logistic Regression . . . 167
9.5.1.1 The Model . . . 167
9.5.1.2 (Regularized) Logistic Regression is Strictly Convex . . . 168
9.5.1.3 Maximizing the Likelihood . . . 168
9.5.2 Multinomial Logistic Regression: Softmax . . . 171
9.5.2.1 The Method . . . 171
9.5.2.2 (Regularized) Softmax is Strictly Convex . . . 172
9.5.3 Poisson Regression . . . 174
9.5.4 Examples . . . 176
9.5.4.1 Birthweight Data: Normal . . . 176

9.5.4.2 Beetle Mortality: Logistic Regression 178
9.5.4.3 Embryogenic Anthers: Logistic Regression 181
9.5.4.4 Toy Example 1: Poisson Regression 186
9.5.4.5 Toy Example 2: Poisson Regression 187
9.5.4.6 Detergent Brand: Poisson Regression 189
9.5.4.7 Tumor Data: Poisson Regression 192
9.5.4.8 Ulcers and Aspirin Use: Logistic Regression 194
9.6 Regularization 198
9.6.1 Partial Least Squares Regression 198
9.6.2 Ridge Regression 199
9.6.3 LASSO 202
9.6.4 Elastic Net 203
9.6.5 Examples 204
9.6.5.1 Example: Ridge Regression, LASSO, Elastic Net 204
9.6.5.2 Example: Diabetes using Least Angle Regression 208
9.6.5.3 Example: Relevant Variable but No Correlation to Response . . 211
9.6.5.4 Example: Irrelevant Variable but High Correlation to Response 212
9.6.5.5 Gas Vapor: Ridge Regression and LASSO 214
9.6.5.6 Chemical Reaction: Ridge Regression and LASSO 217
9.6.5.7 Land Rent: Ridge Regression and LASSO 221

List of Figures

2.1 Cross-validation: The data set is divided into 5 parts. 9
2.2 Cross-validation: For 5-fold cross-validation there are 5 iterations. 9
2.3 Linear transformations of the Gaussian $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. 12
2.4 A two-dimensional classification task where the data for each class are drawn from a Gaussian. 12
2.5 Posterior densities $p(y = 1 \mid \boldsymbol{x})$ and $p(y = -1 \mid \boldsymbol{x})$ as a function of $\boldsymbol{x}$. 14
2.6 x^* is a non-optimal decision point because for some regions the posterior $y = 1$ is above the posterior $y = -1$ but data is classified as $y = -1$. 14
2.7 Two classes with covariance matrix $\boldsymbol{\Sigma} = \sigma^2 \boldsymbol{I}$ each in one (top left), two (top right), and three (bottom) dimensions. 16
2.8 Two classes with arbitrary Gaussian covariance lead to boundary functions which are hyperplanes, hyper-ellipsoids, hyperparaboloids etc. 17

3.1 Projection model, where the observed data $\boldsymbol{x}$ is the input to the model $\boldsymbol{u} = g(\boldsymbol{x}; \boldsymbol{w})$. 19
3.2 Generative model, where the data $\boldsymbol{x}$ is observed and the model $\boldsymbol{x} = g(\boldsymbol{u}; \boldsymbol{w})$ should produce the same distribution as the observed distribution. 20
3.3 The variance of an estimator $\hat{\boldsymbol{w}}$ as a function of the true parameter is shown. . . . 25
3.4 The maximum likelihood problem. 28

4.1 Different noise assumptions lead to different Minkowski error functions. 38
4.2 The sigmoidal function $\frac{1}{1+\exp(-x)}$. 39

5.1 Typical example where the test error first decreases and then increases with increasing complexity. 51
5.2 The consistency of the empirical risk minimization is depicted. 52
5.3 Linear decision boundaries can shatter any 3 points in a 2-dimensional space. . . 53
5.4 Linear decision boundaries cannot shatter any 4 points in a 2-dimensional space. 54
5.5 The growth function is either linear or logarithmic in l. 56
5.6 The error bound is the sum of the empirical error, the training error, and a complexity term. 59
5.7 The bound on the risk, the test error, is depicted. 59
5.8 The structural risk minimization principle is based on sets of functions which are nested subsets $\mathcal{F}_n$. 60
5.9 Data points are contained in a sphere of radius R at the origin. 62
5.10 Margin means that hyperplanes must keep outside the spheres. 62
5.11 The offset b is optimized in order to obtain the largest $\|\boldsymbol{w}\|$ for the canonical form which is $\|\boldsymbol{w}_*\|$ for the optimal value b_*. 64

5.12 Essential support vectors. 67

6.1 Nonlinearly separable data is mapped into a feature space where the data is linear separable. 70
6.2 An example of a mapping from the two-dimensional space into the three-dimensional space. 70

7.1 The negative gradient $-\boldsymbol{g}$ gives the direction of the steepest decent depicted by the tangent on $(R(\boldsymbol{w}), \boldsymbol{w})$, the error surface. 77
7.2 The negative gradient $-\boldsymbol{g}$ attached at different positions on a two-dimensional error surface $(R(\boldsymbol{w}), \boldsymbol{w})$. 78
7.3 The negative gradient $-\boldsymbol{g}$ oscillates as it converges to the minimum. 78
7.4 Using the momentum term the oscillation of the negative gradient $-\boldsymbol{g}$ is reduced. 79
7.5 The negative gradient $-\boldsymbol{g}$ lets the weight vector converge very slowly to the minimum if the region around the minimum is flat. 79
7.6 The negative gradient $-\boldsymbol{g}$ is accumulated through the momentum term. 79
7.7 Length of negative gradient: examples. 80
7.8 The error surface is locally approximated by a quadratic function. 82
7.9 Line search. 84
7.10 The Newton direction $-\boldsymbol{H}^{-1}\,\boldsymbol{g}$ for a quadratic error surface in contrast to the gradient direction $-\boldsymbol{g}$. 85
7.11 Conjugate gradient. 86
7.12 Conjugate gradient examples. 87

8.1 The maximum a posteriori estimator $\boldsymbol{w}_{\mathrm{MAP}}$ is the weight vector which maximizes the posterior $p(\boldsymbol{w} \mid \{\boldsymbol{z}\})$. 104
8.2 Error bars obtained by Bayes technique. 107
8.3 Error bars obtained by Bayes technique (2). 108

9.1 Influential Observations 125
9.2 Carbohydrate vs. Protein 136
9.3 Dobson's dried plant data 145
9.4 Results of ANOVA for dried plant data 146
9.5 Dobson's extended dried plant data 148
9.6 Two-factor ANOVA toy data 150
9.7 Scatter plot achievement scores data 154
9.8 Scatter plot birthweight data 157
9.9 The sigmoid function 167
9.10 Dobson's beetle data 179
9.11 Fitting of Dobson's beetle data 182
9.12 Dobson's embryogenic anther data 183
9.13 Best model for Dobson's embryogenic anther data 185
9.14 Toy Poisson Regression data 186
9.15 Poisson regression for randomized controlled trial 188
9.16 Dobson's malignant melanoma data 193
9.17 Dobson's gastric and duodenal ulcers and aspirin use 195
9.18 LASSO vs. ridge regression 202

9.19 Example correlated explanatory variables 204
9.20 Example of ridge regression 206
9.21 Example of LASSO 207
9.22 Example of elastic net 208
9.23 Example of least angle regression 209
9.24 Example: relevant Variable with no target correlation 213
9.25 Example: irrelevant Variable has high target correlation 214

List of Tables

9.1 ANOVA table . . . 129
9.2 Rencher's hematology data . . . 131
9.3 Dobson's carbohydrate diet data . . . 135
9.4 Data of dried plant weights . . . 144
9.5 Extended dried plant weights . . . 147
9.6 Two-factor ANOVA toy data . . . 150
9.7 Achievement scores data . . . 155
9.8 Birthweight data . . . 156
9.9 Common GLM models . . . 166
9.10 Count data distribution . . . 174
9.11 Dobson's beetle data . . . 179
9.12 Dobson's embryogenic anther data . . . 182
9.13 Poisson regression for randomized controlled trial . . . 187
9.14 Ries & Smith (1963) data . . . 190
9.15 Dobson's malignant melanoma data . . . 193
9.16 Dobson's gastric and duodenal ulcers and aspirin use . . . 195
9.17 Example: relevant Variable with no target correlation . . . 211
9.18 Example: irrelevant Variable has high target correlation . . . 212
9.19 Rencher's gas vapor data . . . 215
9.20 Rencher's chemical reaction data . . . 218
9.21 Rencher's land rent data . . . 222

List of Algorithms

7.1 Line Search 83
7.2 Conjugate Gradient (Polak-Ribiere) 90
9.1 Partial least squares regression 200

Chapter 1

Introduction

1.1 Machine Learning Introduction

This course is part of the curriculum of the master in computer science (in particular the majors "Computational Engineering" and "Intelligent Information Systems") and part of the master in bioinformatics at the Johannes Kepler University Linz.

Machine learning is currently a major research topic at companies like Google, Microsoft, Amazon, Facebook, AltaVista, Zalando, and many more. Applications are found in computer vision (image recognition), speech recognition, recommender systems, analysis of Big Data, information retrieval. Companies that try to mine the world wide web are offering search engines, social networks, videos, music, information, or connecting people use machine learning techniques. Machine learning methods are used to classify and label web pages, images, videos, and sound recordings in web data. They can find specific objects in images and detect a particular music style if only given the raw data. Therefore Google, Microsoft, Facebook are highly interested in machine learning methods. Machine learning methods attracted the interest of companies offering products via the web. These methods are able to identify groups of similar users, to predict future behavior of customers, and can give recommendation of products in which customers will be interested based previous costumer data.

Machine learning has major applications in biology and medicine. Modern measurement techniques in both biology and medicine create a huge demand for new machine learning approaches. One such technique is the measurement of mRNA concentrations with microarrays and sequencing techniques. The measurement data are first preprocessed, then genes of interest are identified, and finally predictions made. Further machine learning methods are used to detect alternative splicing, nucleosome positions, gene regulation, etc. Alongside neural networks the most prominent machine learning techniques relate to support vector machines, kernel approaches, projection method and probabilistic models like latent variable models. These methods provide noise reduction, feature selection, structure extraction, classification / regression, and assist modeling. In the biomedical context, machine learning algorithms categorize the disease subtype or predict treatment outcomes based on DNA characteristics, gene expression profiles. Machine learning approaches classify novel protein sequences into structural or functional classes. For analyzing data of association studies, machine learning methods extract new dependencies between DNA markers (SNP - single nucleotide polymorphisms, SNV - single nucleotide variants, CNV - copy number variations) and diseases (Alzheimer, Parkinson, cancer, multiples sclerosis, schizophrenia or alcohol dependence).

The machine learning course series comprises:

- "Basic Methods of Data Analysis": this course gives a smooth introduction to machine learning with examples in R ; it covers summary statistics (mean, variance), data summary plots (boxplot, violin plot, scatter plot), principal component analysis, independent component analysis, multidimensional scaling (Kruskal's or Sammon's map), locally linear embedding, Isomap, hierarchical clustering, mixture models, k-means, similarity based clustering (affinity propagation), biclustering

- "Machine Learning: Supervised Methods": classification and regression techniques, time series prediction, kernel methods, support vector machines, neural networks, deep learning, deep neural and belief networks, ARMA and ARIMA models, recurrent neural networks, LSTM

- "Machine Learning: Unsupervised Methods": maximum likelihood estimation, maximum a posterior estimation, maximum entropy, expectation maximization, principal component analysis, statistical independence, independent component analysis, factor analysis, mixture models, sparse codes, population codes, kernel PCA, hidden Markov models (factorial HMMs and input-output HMMs), Markov networks and random fields, clustering, biclustering, restricted Boltzmann machines, auto-associators, unsupervised deep neural networks

- "Theoretical Concepts of Machine Learning": estimation theory (unbiased and efficient estimator, Cramer-Rao lower bound, Fisher information matrix), consistent estimator, complexity of model classes (VC-dimension, growth, annealed entropy), bounds on the generalization error, Vapnik and worst case bounds on the generalization error, optimization (gradient based methods and convex optimization), Bayes theory (posterior estimation, error bounds, hyperparameter optimization, evidence framework), theory on linear functions (statistical tests, intervals, ANOVA, generalized linear functions, mixed models)

In this course the most prominent machine learning techniques are introduced and their mathematical basis and derivatives are explained. If the student understands these techniques, then the student can select the methods which best fit to the problem at hand, the student is able to optimize the parameter settings for the methods, the student can adapt and improve the machine learning methods, and the student can develop new machine learning methods.

Most importantly, students should learn how to chose appropriate methods from a given pool of approaches for solving a specific problem. To this end, they must understand and evaluate the different approaches, know their advantages and disadvantages as well as where to obtain and how to use them. In a step further, the students should be able to adapt standard algorithms for their own purposes or to modify those algorithms for particular applications with certain prior knowledge or problem-specific constraints.

1.2 Course Specific Introduction

In this course the theoretical basics of machine learning methods are taught. To motivate the maximum likelihood estimator, first concepts from statistics and estimation theory are introduced. Estimators are characterized by whether they are biased or unbiased, whether they are efficient or not (relies on Cramer-Rao lower bound and the Fisher information matrix), or whether they are

consistent or not. The latter is important for machine learning because one wants to known if more data lead to better results and how many more data items lead to which improvement.

We present basics of the *statistical learning theory* like the *empirical risk minimization*. In the framework of statistical learning theory the complexity of model classes will be defined (VC-dimension, growth, annealed entropy). These complexity concepts are used to derive bounds on the generalization error like Vapnik, Chernoff, covering numbers, and other bounds. Using these bounds, *structural risk minimization* will be introduced. Further topics are optimization, where gradient-based and convex optimization is covered. Then the Bayes framework is introduced which allows to estimate the posterior, derive error bounds for model predictions, optimize hyperparameters e.g. by integrating out the posterior or by the evidence framework. The final part focuses on the theory of linear functions including statistical tests, parameter interval estimation, and ANOVA models. The linear functions are extended to generalized linear functions and to mixed models.

We define quality criteria for selected models in order to pin down a goal for model selection, i.e. learning. In most cases the quality criterion is not computable and we have to find approximations to it. The definition of the quality criterion first focuses on supervised learning.

For unsupervised learning we introduce *maximum likelihood* as quality criterion. In this context we introduce concepts like bias and variance, efficient estimator, and the Fisher information matrix. Next we extend maximum likelihood to supervised learning. Measurement noise determines the error model which in turn determines the quality criterion of the supervised approach. Here also classification methods with binary output can be treated.

A central question in machine learning is: Does learning from examples help in the future? Obviously, learning helps humans to master the environment they live in. But what is the mathematical reason for that? It might be that tasks in the future are unique and nothing from the past helps to solve them. Future examples may be different from examples we have already seen.

Learning on the training data is called “empirical risk minimization” (ERM) in statistical learning theory. ERM results that if the complexity is restricted and the dynamics of the environment does not change, learning helps. “Learning helps” means that with increasing number of training examples the selected model converges to the best model for all future data. Under mild conditions the convergence is uniform and even fast, i.e. exponentially. These theoretical theorems found the idea of learning from data because with finite many training examples a model can be selected which is close to the optimal model for future data. How close is governed by the number of training examples, the complexity of the task including noise, the complexity of the model, and the model class.

To measure the complexity of the model we will introduce the VC-dimension (Vapnik-Chervonenkis). Using model complexity and the model quality on the training set, theoretical bounds on the generalization error, i.e. the performance on future data, will be derived. From these bounds the principle of “structural risk minimization” will be derived to optimize the generalization error through training.

We introduce basic techniques for minimizing the error that is techniques for model selection for a parameterized model class. On-line methods are treated, i.e. methods which do not require a training set but attempt at improving the model (selecting a better model) using only one example at a certain time point.

Chapter 2

Generalization Error

In this chapter we want to define quality criteria for selected models in order to pin down a goal for model selection, i.e. learning. In most cases the quality criterion is not computable and we have to find approximations to it. The definition of the quality criterion first focuses on supervised learning.

In supervised learning the most widely used quality criteria is the performance on future data. Performance on future data is in general expressed via the *generalization error*, the expected error on future data.

2.1 Model Quality Criteria

Learning in machine learning is equivalent to model selection. A model from a set of possible models is chosen and will be used to handle future data.

But what is the best model? We need a quality criterion in order to choose a model. The quality criterion should be such that future data is optimally processed with the model. That would be the most common criterion.

However in some cases the user is not interested in future data but only wants to visualize the current data or extract structures from the current data, where these structures are not used for future data but to analyze the current data. Topics which are related to the later criteria are data visualization, modeling, data compression. But in many cases the model with best visualization, best world explanation, or highest compression rate is the model where rules derived on a subset of the data can be generalized to the whole data set. Here the rest of the data can be interpreted as future data. Another point of view may be to assume that future data is identical with the training set. These considerations allow also to treat the later criteria also with the former criteria.

Some machine learning approaches like Kohonen networks don't possess a quality criterion as a single scalar value but minimize a potential function. Problem is that different models cannot be compared. Some ML approaches are known to converge during learning to the model which really produces the data if the data generating model is in the model class. But these approaches cannot supply a quality criterion and the quality of the current model is unknown.

The performance on future data will serve as our quality criterion which possesses the advantages of being able to compare models and to know the quality during learning which gives in turn a hint when to stop learning.

For supervised data the performance on future data can be measured directly, e.g. for classification the rate of misclassifications or for regression the distance between model output, the prediction, and the correct value observed in future.

For unsupervised data the quality criterion is not as obvious. The criterion cannot be broken down to single examples as in the supervised case but must include all possible data with its probability for being produced by the data generation process. Typical, quality measures are the likelihood of the data being produced by the model, the ratio of between and within cluster distances in case of clustering, the independence of the components after data projection in case of ICA, the information content of the projected data measured as non-Gaussianity in case of projection pursuit, expected reconstruction error in case of a subset of PCA components or other projection methods.

2.2 Introducing the Generalization Error

In this section we define the performance of a model on future data for the supervised case. The performance of a model on future data is called *generalization error*. For the supervised case an error for each example can be defined and then averaged over all possible examples. The error on one example is called *loss* but also *error*. The expected loss is called *risk*.

2.2.1 Definition of the Generalization Error / Risk

We assume that *objects* $x \in \mathcal{X}$ from an object set $\mathcal{X}$ are represented or described by *feature vectors* $\boldsymbol{x} \in \mathbb{R}^d$.

The *training set* consists of l objects $X = \left\{x^1, \ldots, x^l\right\}$ with a characterization $y^i \in \mathbb{R}$ like a label or an associated value which must be predicted for future objects. For simplicity we assume that y^i is a scalar, the so-called *target*. For simplicity we will write $\boldsymbol{z} = (\boldsymbol{x}, y)$ and $Z = X \times \mathbb{R}$.

The *training data* is $\left\{\boldsymbol{z}^1, \ldots, \boldsymbol{z}^l\right\}$ ($\boldsymbol{z}^i = (\boldsymbol{x}^i, y^i)$), where we will later use the *matrix of feature vectors* $\boldsymbol{X} = \left(\boldsymbol{x}^1, \ldots, \boldsymbol{x}^l\right)^T$, the *vector of labels* $\boldsymbol{y} = \left(y^1, \ldots, y^l\right)^T$, and the *training data matrix* $\boldsymbol{Z} = \left(\boldsymbol{z}^1, \ldots, \boldsymbol{z}^l\right)$ ("^{T}" means the transposed of a matrix and here it makes a column vector out of a row vector).

In order to compute the performance on the future data we need to know the future data and need a quality measure for the deviation of the prediction from the true value, i.e. a *loss function*.

The future data is not known, therefore, we need at least the probability that a certain data point is observed in the future. The data generation process has a density $p(\boldsymbol{z})$ at $\boldsymbol{z}$ over its data space. For finite discrete data $p(\boldsymbol{z})$ is the probability of the data generating process to produce $\boldsymbol{z}$. $p(\boldsymbol{z})$ is the *data probability*.

The loss function is a function of the target and the model prediction. The model prediction is given by a function $g(\boldsymbol{x})$ and if the models are parameterized by a parameter vector $\boldsymbol{w}$ the model prediction is a parameterized function $g(\boldsymbol{x}; \boldsymbol{w})$. Therefore the loss function is $L(y, g(\boldsymbol{x}; \boldsymbol{w}))$. Typical loss functions are the *quadratic loss* $L(y, g(\boldsymbol{x}; \boldsymbol{w})) = (y - g(\boldsymbol{x}; \boldsymbol{w}))^2$ or the zero-one

loss function

$$L(y, g(\boldsymbol{x}; \boldsymbol{w})) \;=\; \begin{cases} 0 & \text{for} \quad y \;=\; g(\boldsymbol{x}; \boldsymbol{w}) \\ 1 & \text{for} \quad y \;\neq\; g(\boldsymbol{x}; \boldsymbol{w}) \end{cases} \quad . \tag{2.1}$$

Now we can define the *generalization error* which is the expected loss on future data, also called *risk* R (a functional, i.e. a operator which maps functions to scalars):

$$R(g(.; \boldsymbol{w})) \;=\; \mathrm{E}_{\boldsymbol{z}} \left(L(y, g(\boldsymbol{x}; \boldsymbol{w})) \right) \; . \tag{2.2}$$

The risk for the quadratic loss is called *mean squared error*.

$$R(g(.; \boldsymbol{w})) \;=\; \int_Z L(y, g(\boldsymbol{x}; \boldsymbol{w}))\, p\,(\boldsymbol{z}) \; d\boldsymbol{z} \; . \tag{2.3}$$

In many cases we assume that y is a function of $\boldsymbol{x}$, the *target function* $f(\boldsymbol{x})$, which is disturbed by noise

$$y \;=\; f(\boldsymbol{x}) \;+\; \epsilon \; , \tag{2.4}$$

where ϵ is a noise term drawn from a certain distribution $p_n(\epsilon)$, thus

$$p(y \mid \boldsymbol{x}) \;=\; p_n(y \;-\; f(\boldsymbol{x})) \; . \tag{2.5}$$

Here the probabilities can be rewritten as

$$p\,(\boldsymbol{z}) \;=\; p\,(\boldsymbol{x}) \; p(y \mid \boldsymbol{x}) \;=\; p\,(\boldsymbol{x}) \; p_n(y \;-\; f(\boldsymbol{x})) \; . \tag{2.6}$$

Now the risk can be computed as

$$\begin{aligned} R(g(.; \boldsymbol{w})) \;=\; \int_Z L(y, g(\boldsymbol{x}; \boldsymbol{w}))\, p\,(\boldsymbol{x}) \; p_n(y \;-\; f(\boldsymbol{x}))\, d\boldsymbol{z} \;= \\ \int_X p\,(\boldsymbol{x}) \; \int_{\mathbb{R}} L(y, g(\boldsymbol{x}; \boldsymbol{w}))\, p_n(y \;-\; f(\boldsymbol{x}))\, dy \; d\boldsymbol{x} \; , \end{aligned} \tag{2.7}$$

where

$$\begin{aligned} R(g(\boldsymbol{x}; \boldsymbol{w})) \;=\; \mathrm{E}_{y|\boldsymbol{x}} \left(L(y, g(\boldsymbol{x}; \boldsymbol{w})) \right) \;= \\ \int_{\mathbb{R}} L(y, g(\boldsymbol{x}; \boldsymbol{w}))\, p_n(y \;-\; f(\boldsymbol{x}))\, dy \; . \end{aligned} \tag{2.8}$$

The noise-free case is $y \;=\; f(\boldsymbol{x})$, where $p_n \;=\; \delta$ can be viewed as a Dirac delta-distribution:

$$\int_{\mathbb{R}} h(\boldsymbol{x}) \delta(\boldsymbol{x}) d\boldsymbol{x} \;=\; h(\boldsymbol{0}) \tag{2.9}$$

therefore

$$R(g(\boldsymbol{x}; \boldsymbol{w})) \;=\; L(f(\boldsymbol{x}), g(\boldsymbol{x}; \boldsymbol{w})) \;=\; L(y, g(\boldsymbol{x}; \boldsymbol{w})) \tag{2.10}$$

and eq. (2.3) simplifies to

$$R(g(.; \boldsymbol{w})) \;=\; \int_X p\,(\boldsymbol{x})\, L(f(\boldsymbol{x}), g(\boldsymbol{x}; \boldsymbol{w})) d\boldsymbol{x} \; . \tag{2.11}$$

Because we do not know $p\,(\boldsymbol{z})$ the risk cannot be computed; especially we do not know $p(y \mid \boldsymbol{x})$. In practical applications we have to approximate the risk.

To be more precise $\boldsymbol{w} \;=\; \boldsymbol{w}\,(\boldsymbol{Z})$, i.e. the parameters depend on the training set.

2.2.2 Empirical Estimation of the Generalization Error

Here we describe some methods how to estimate the risk (generalization error) for a certain model.

2.2.2.1 Test Set

We assume that data points $\boldsymbol{z} = (\boldsymbol{x}, y)$ are iid (independent identical distributed) and, therefore also $L(y, g(\boldsymbol{x}; \boldsymbol{w}))$, and $E_{\boldsymbol{z}}(|L(y, g(\boldsymbol{x}; \boldsymbol{w}))|) < \infty$.

The risk is an expectation of the loss function:

$$R(g(.; \boldsymbol{w})) = \mathrm{E}_{\boldsymbol{z}}(L(y, g(\boldsymbol{x}; \boldsymbol{w}))) , \tag{2.12}$$

therefore this expectation can be approximated using the (strong) law of large numbers:

$$R(g(.; \boldsymbol{w})) \approx \frac{1}{m} \sum_{i=l+1}^{l+m} L\left(y^i, g(\boldsymbol{x}^i; \boldsymbol{w})\right) , \tag{2.13}$$

where the set of m elements $\left\{\boldsymbol{z}^{l+1}, \ldots, \boldsymbol{z}^{l+m}\right\}$ is called *test set*.

Disadvantage of the test set method is, that the test set cannot be used for learning because $\boldsymbol{w}$ is selected using the training set and, therefore, $L(y, g(\boldsymbol{x}; \boldsymbol{w}))$ is not iid for training data points. Intuitively, if the loss is low for some training data points then we will expect that the loss will also be low for the following training data points.

2.2.2.2 Cross-Validation

If we have only few data points available we want to use them all for learning and not for estimating the performance via a test set. But we want to estimate the performance for our final model.

We can divide the available data multiple times into training data and test data and average over the result. Problem here is that the test data is overlapping and we estimate with dependent test data points.

To avoid overlapping test data points we divide the training set into n parts (see Fig. 2.1). Then we make n runs where for the ith run part no. i is used for testing and the remaining parts for training (see Fig. 2.2). That procedure is called *n-fold cross-validation*. The *cross-validation risk* $R_{n-\mathrm{cv}}(\boldsymbol{Z})$ is the cumulative loss over all folds used for testing.

A special case of cross-validation is *leave-one-out cross-validation* (LOO CV) where $n = l$ and a fold contains only one element.

The cross-validation risk is a nearly (almost) unbiased estimate for the risk.

Unbiased means that the expected cross-validation error is equal the expected risk, where the expectation is over training sets with l elements.

We will write $\boldsymbol{Z}_l := \boldsymbol{Z}$ as a variable for training sets with l elements. The j fold of an n-fold cross-validation is denoted by $\boldsymbol{Z}^j$ or $\boldsymbol{Z}^j_{l/n}$ to include the number l/n of elements of the fold. The n-fold cross-validation risk is

$$R_{n-\mathrm{cv}}(\boldsymbol{Z}_l) = \frac{1}{n} \sum_{j=1}^{n} \frac{n}{l} \sum_{\boldsymbol{z} \in \boldsymbol{Z}^j_{l/n}} \left(L\left(y, g\left(\boldsymbol{x}; \boldsymbol{w}_j\left(\boldsymbol{Z}_l \setminus \boldsymbol{Z}^j_{l/n}\right)\right)\right)\right) , \tag{2.14}$$

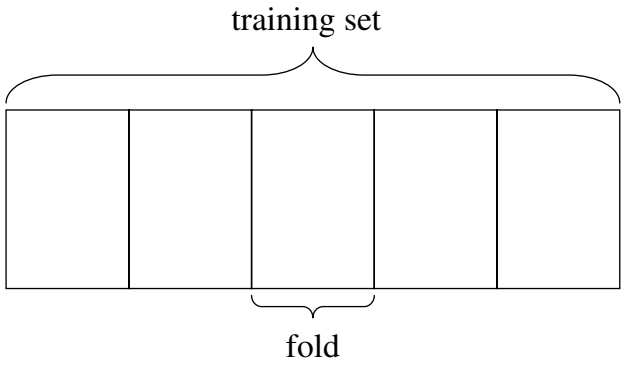

Figure 2.1: Cross-validation: The data set is divided into 5 parts for 5-fold cross-validation — each part is called fold.

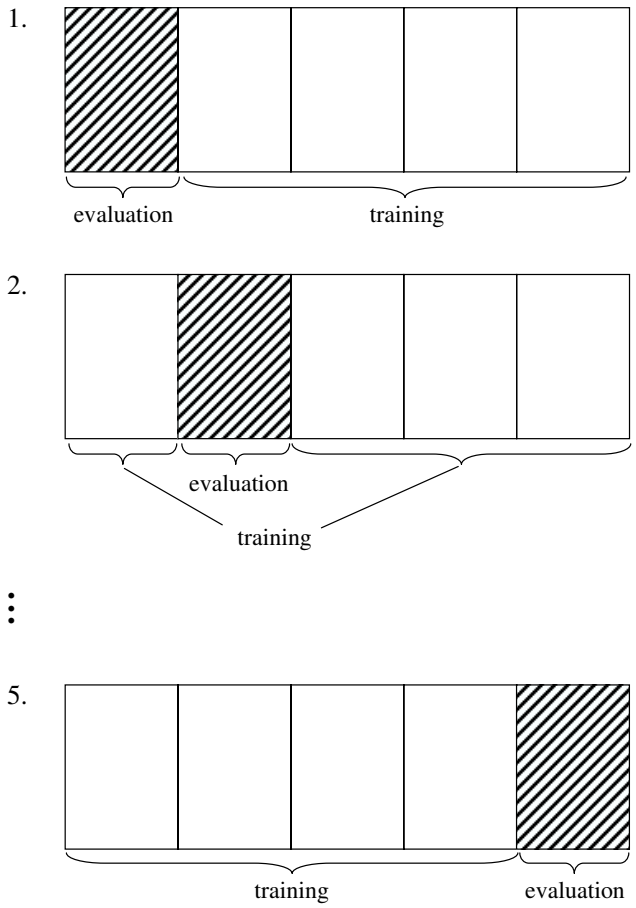

Figure 2.2: Cross-validation: For 5-fold cross-validation there are 5 iterations and in each iteration a different fold is omitted and the remaining folds form the training set. After training the model is tested on the omitted fold. The cumulative error on all folds is the cross-validation error.

where $\boldsymbol{w}_j$ is the model selected when removing the jth fold and

$$R_{n-\mathrm{cv},j}(\boldsymbol{Z}_l) \;=\; \frac{n}{l} \sum_{\boldsymbol{z} \in \boldsymbol{Z}^j_{l/n}} \left(L\left(y, g\left(\boldsymbol{x}; \boldsymbol{w}_j\left(\boldsymbol{Z}_l \setminus \boldsymbol{Z}^j_{l/n}\right)\right)\right)\right) \tag{2.15}$$

is the risk for the jth fold.

The statement that the "cross-validation estimate for the risk is almost unbiased" (Luntz and Brailovsky) means

$$\mathrm{E}_{\boldsymbol{Z}_{l(1-1/n)}}\left(R\left(g\left(.; \boldsymbol{w}\left(\boldsymbol{Z}_{l(1-1/n)}\right)\right)\right)\right) \;=\; \mathrm{E}_{\boldsymbol{Z}_l}\left(R_{n-\mathrm{cv}}\left(\boldsymbol{Z}_l\right)\right) \;. \tag{2.16}$$

The generalization error on training size l without one fold l/n, namely $l - l/n = l(1 - 1/n)$ can be estimated by cross-validation on training data of size l by n-fold cross-validation. For large l the training size l or $l(1-1/n)$ should lead similar results, that is the estimate is almost unbiased.

The following two equations will prove eq. (2.16).

The left hand side of eq. (2.16) can be rewritten as

$$\begin{aligned} &\mathrm{E}_{\boldsymbol{Z}_{l(1-1/n)}}\left(R\left(g\left(.; \boldsymbol{w}\left(\boldsymbol{Z}_{l(1-1/n)}\right)\right)\right)\right) \;= \\ &\mathrm{E}_{\boldsymbol{Z}_{l(1-1/n)} \cup \boldsymbol{z}}\left(L\left(y, g\left(\boldsymbol{x}; \boldsymbol{w}\left(\boldsymbol{Z}_{l(1-1/n)}\right)\right)\right)\right) \;= \\ &\mathrm{E}_{\boldsymbol{Z}_{l(1-1/n)}} \mathrm{E}_{\boldsymbol{Z}_{l/n}} \left(\frac{n}{l} \sum_{\boldsymbol{z} \in \boldsymbol{Z}_{l/n}} \left(L\left(y, g\left(\boldsymbol{x}; \boldsymbol{w}\left(\boldsymbol{Z}_{l(1-1/n)}\right)\right)\right)\right)\right) \;. \end{aligned} \tag{2.17}$$

The second equations stems from the fact that the data points are iid, therefore $\mathrm{E}_{\boldsymbol{z}}\left(f(\boldsymbol{z})\right) \;=\; \frac{1}{k}\sum_{i=1}^{k} \mathrm{E}_{\boldsymbol{z}}\left(f(\boldsymbol{z}^i)\right) \;=\; \mathrm{E}_{\boldsymbol{Z}_k}\left(\frac{1}{k}\sum_{i=1}^{k} f(\boldsymbol{z}^i)\right)$.

The right hand side of eq. (2.16) can be rewritten as

$$\begin{aligned} &\mathrm{E}_{\boldsymbol{Z}_l}\left(R_{n-\mathrm{cv}}\left(\boldsymbol{Z}_l\right)\right) \;= \\ &\mathrm{E}_{\boldsymbol{Z}_l} \left(\frac{1}{n}\sum_{j=1}^{n} \frac{n}{l} \sum_{(\boldsymbol{x},y) \in \boldsymbol{Z}^j_{l/n}} \left(L\left(y, g\left(\boldsymbol{x}; \boldsymbol{w}_j\left(\boldsymbol{Z}_l \setminus \boldsymbol{Z}^j_{l/n}\right)\right)\right)\right)\right) \;= \\ &\frac{1}{n}\sum_{j=1}^{n} \mathrm{E}_{\boldsymbol{Z}_l} \left(\frac{n}{l} \sum_{(\boldsymbol{x},y) \in \boldsymbol{Z}^j_{l/n}} \left(L\left(y, g\left(\boldsymbol{x}; \boldsymbol{w}_j\left(\boldsymbol{Z}_l \setminus \boldsymbol{Z}^j_{l/n}\right)\right)\right)\right)\right) \;= \\ &\mathrm{E}_{\boldsymbol{Z}_{l(1-1/n)}} \mathrm{E}_{\boldsymbol{Z}_{l/n}} \left(\frac{n}{l} \sum_{(\boldsymbol{x},y) \in \boldsymbol{Z}_{l/n}} \left(L\left(y, g\left(\boldsymbol{x}; \boldsymbol{w}\left(\boldsymbol{Z}_{l(1-1/n)}\right)\right)\right)\right)\right) \;. \end{aligned} \tag{2.18}$$

The first equality comes from the fact that sum and integral are interchangeable. Therefore it does not matter whether first the data is drawn and then the different folds are treated or the data is drawn again for treating each fold. The second equality comes from the fact that $\mathrm{E}\left(\boldsymbol{Z}^j_{l/n}\right) \;=\; \mathrm{E}\left(\boldsymbol{Z}_{l/n}\right)$.

Therefore both sides of eq. (2.16) are equal.

The term "almost" addresses the fact that the estimation is made with $l(1-1/n)$ training data using the risk and with l training data using n-fold cross-validation.

However the cross-validation estimate has high variance. The high variance stems from the fact that the training data is overlapping. Also test and training data are overlapping. Intuitively speaking, if data points are drawn which make the task very complicated, then these data points appear in many training sets and at least in one test set. These data points strongly increase the estimate of the risk. The opposite is true for data points which make learning more easy. That means single data points may strongly influence the estimate of the risk.

2.3 Minimal Risk for a Gaussian Classification Task

We will show an example for the optimal risk for a classification task.

We assume that we have a binary classification task where class $y = 1$ data points come from a Gaussian and class $y = -1$ data points come from a different Gaussian.

Class $y = 1$ data points are drawn according to

$$p(\boldsymbol{x} \mid y = 1) \ \propto \ \mathcal{N}\left(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1\right) \tag{2.19}$$

and Class $y = -1$ according to

$$p(\boldsymbol{x} \mid y = -1) \ \propto \ \mathcal{N}\left(\boldsymbol{\mu}_{-1}, \boldsymbol{\Sigma}_{-1}\right) \tag{2.20}$$

where the Gaussian $\mathcal{N}\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$ has density

$$p(\boldsymbol{x}) \ = \ \frac{1}{(2\,\pi)^{d/2}\,|\boldsymbol{\Sigma}|^{1/2}} \ \exp\left(-\frac{1}{2}\,(\boldsymbol{x}-\boldsymbol{\mu})^T\,\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right) \ . \tag{2.21}$$

$\boldsymbol{x}$ is the d-dimensional feature vector, $\boldsymbol{\mu}$ is the mean vector, $\boldsymbol{\Sigma}$ is the $d \times d$-dimensional covariance matrix.

As depicted in Fig. 2.3, the linear transformation $\boldsymbol{A}$ leads to the Gaussian $\mathcal{N}\left(\boldsymbol{A}^T\boldsymbol{\mu}, \boldsymbol{A}^T\boldsymbol{\Sigma}\,\boldsymbol{A}\right)$. All projections $\boldsymbol{P}$ of a Gaussian are Gaussian. A certain transformation $A_w = \boldsymbol{\Sigma}^{-1/2}$ ("whitening") leads to a Gaussian with the identity matrix as covariance matrix. On the other hand each Gaussian with covariance matrix $\boldsymbol{\Sigma}$ can be obtained from a Gaussian with covariance matrix $\boldsymbol{I}$ by the linear transformation $\boldsymbol{\Sigma}^{1/2}$.

Affine transformation (translation and linear transformation) allow to interpret all Gaussians as stemming from a Gaussian with zero mean and the identity as covariance matrix.

At a certain point $\boldsymbol{x}$ in the feature space the probability $p(\boldsymbol{x}, y = 1)$ of observing a point from class $y = 1$ is the probability $p(y = 1)$ of choosing class $y = 1$ multiplied by the Gaussian density for class $y = 1$

$$p(\boldsymbol{x}, y = 1) \ = \ p(\boldsymbol{x} \mid y = 1)\, p(y = 1) \ . \tag{2.22}$$

Fig. 2.4 shows a two-dimensional classification task where the data for each class are drawn from a Gaussian (black: class 1, red: class -1). The discriminant functions are are two hyperbolas forming the optimal decision boundaries.

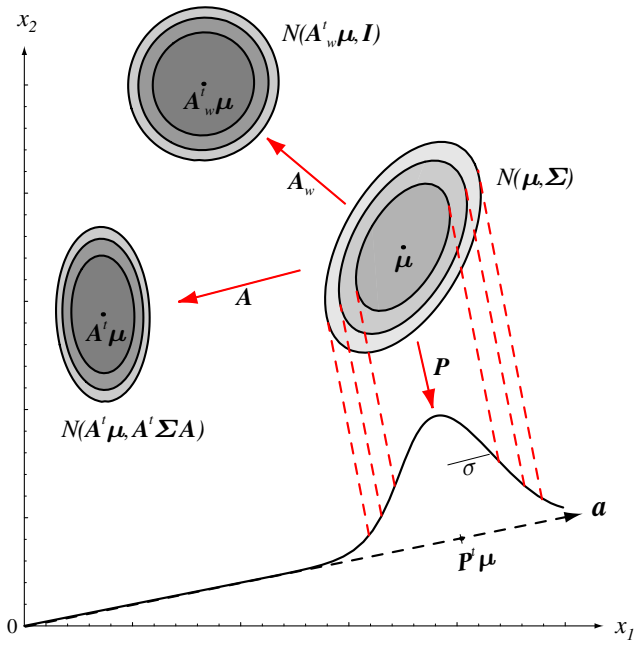

Figure 2.3: Linear transformations of the Gaussian $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. The linear transformation $\boldsymbol{A}$ leads to the Gaussian $\mathcal{N}\left(\boldsymbol{A}^T\boldsymbol{\mu}, \boldsymbol{A}^T\boldsymbol{\Sigma}\,\boldsymbol{A}\right)$. All projections $\boldsymbol{P}$ of a Gaussian are Gaussian. A certain transformation A_w ("whitening") leads to a Gaussian with the identity matrix as covariance matrix. Copyright © 2001 John Wiley & Sons, Inc.

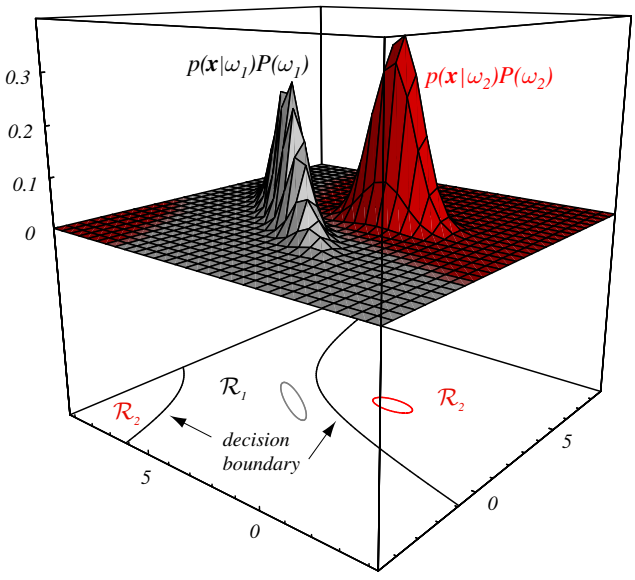

Figure 2.4: A two-dimensional classification task where the data for each class are drawn from a Gaussian (black: class 1, red: class -1). The optimal decision boundaries are two hyperbolas. Here $\omega_1 \equiv y = 1$ and $\omega_2 \equiv y = -1$. In the gray regions $p(y = 1 \mid \boldsymbol{x}) > p(y = -1 \mid \boldsymbol{x})$ holds and in the red regions the opposite holds. Copyright © 2001 John Wiley & Sons, Inc.

The probability of observing a point at $\boldsymbol{x}$ not depending on the class is

$$p(\boldsymbol{x}) \;=\; p(\boldsymbol{x}, y=1) \;+\; p(\boldsymbol{x}, y=-1)\;. \tag{2.23}$$

Here the variable y is "integrated out".

The probability of observing a point from class $y=1$ at $\boldsymbol{x}$ is

$$p(y=1 \mid \boldsymbol{x}) \;=\; \frac{p(\boldsymbol{x} \mid y=1)\, p(y=1)}{p(\boldsymbol{x})}\;. \tag{2.24}$$

This formula is obtained by applying the Bayes rule.

We define the regions of class 1 as

$$X_1 \;=\; \{\boldsymbol{x} \mid g(\boldsymbol{x}) > 0\} \tag{2.25}$$

and regions of class -1 as

$$X_{-1} \;=\; \{\boldsymbol{x} \mid g(\boldsymbol{x}) < 0\}\;. \tag{2.26}$$

and the loss function as

$$L(y, g(\boldsymbol{x}; \boldsymbol{w})) \;=\; \left\{ \begin{array}{lll} 0 & \text{for} & y\, g(\boldsymbol{x}; \boldsymbol{w}) > 0 \\ 1 & \text{for} & y\, g(\boldsymbol{x}; \boldsymbol{w}) < 0 \end{array} \right. . \tag{2.27}$$

The risk of eq. (2.3) is for the zero-one loss

$$\begin{aligned} R(g(.; \boldsymbol{w})) \;=\; & \int_{X_1} p\,(\boldsymbol{x}, y=-1)\; d\boldsymbol{x} \;+\; \int_{X_{-1}} p\,(\boldsymbol{x}, y=1)\; d\boldsymbol{x} \;= \\ & \int_{X_1} p\,(y=-1 \mid \boldsymbol{x})\; p(\boldsymbol{x})\, d\boldsymbol{x} \;+\; \int_{X_{-1}} p\,(y=1 \mid \boldsymbol{x})\; p(\boldsymbol{x})\, d\boldsymbol{x} \;= \\ & \int_X \left\{ \begin{array}{lll} p\,(y=-1 \mid \boldsymbol{x}) & \text{for} & g(\boldsymbol{x}) > 0 \\ p\,(y=1 \mid \boldsymbol{x}) & \text{for} & g(\boldsymbol{x}) < 0 \end{array} \right\} p(\boldsymbol{x})\, d\boldsymbol{x}\;. \end{aligned} \tag{2.28}$$

In the last equation it obvious how the risk can be minimized by choosing the smaller value of $p\,(y=-1 \mid \boldsymbol{x})$ and $p\,(y=1 \mid \boldsymbol{x})$. Therefore, the risk is minimal if

$$g(\boldsymbol{x}; \boldsymbol{w}) \left\{ \begin{array}{lll} > 0 & \text{for} & p\,(y=1 \mid \boldsymbol{x}) \;>\; p\,(y=-1 \mid \boldsymbol{x}) \\ < 0 & \text{for} & p\,(y=-1 \mid \boldsymbol{x}) \;>\; p\,(y=1 \mid \boldsymbol{x}) \end{array} \right. . \tag{2.29}$$

The minimal risk is

$$\begin{aligned} R_{\min} \;=\; & \int_X \min\{p\,(\boldsymbol{x}, y=-1)\,, p\,(\boldsymbol{x}, y=1)\}\; d\boldsymbol{x} \;= \\ & \int_X \min\{p\,(y=-1 \mid \boldsymbol{x})\,, p\,(y=1 \mid \boldsymbol{x})\}\; p(\boldsymbol{x})\; d\boldsymbol{x}\;. \end{aligned} \tag{2.30}$$

Because at each point either class $y=1$ or class $y=-1$ will be misclassified we classify the point as belonging to the class with higher probability. This is demonstrated in Fig. 2.5 where at each position $\boldsymbol{x}$ either the red or the black line determines the probability of misclassification. The

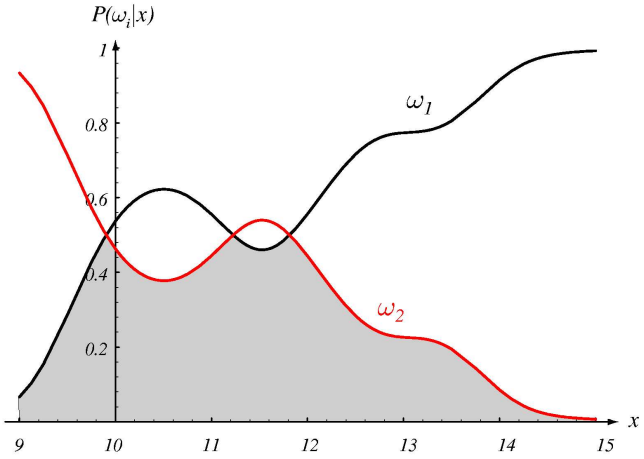

Figure 2.5: Posterior densities $p(y = 1 \mid \boldsymbol{x})$ and $p(y = -1 \mid \boldsymbol{x})$ as a function of $\boldsymbol{x}$. If using the optimal discriminant function the gray region is the integral eq. (2.28) and gives the probability of misclassifying a data point. Modified figure with copyright © 2001 John Wiley & Sons, Inc.

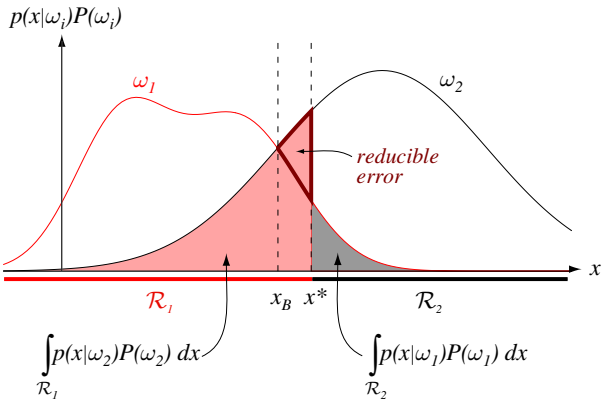

Figure 2.6: x^* is a non-optimal decision point because for some regions the posterior $y = 1$ is above the posterior $y = -1$ but data is classified as $y = -1$. The misclassification rate is given by the filled region. However the misclassification mass in the red triangle can be saved if using as decision point x_B. Copyright © 2001 John Wiley & Sons, Inc.

ratio of misclassification is given by integrating along the curve according to eq. (2.28). The minimal integration value is obtained if one chooses the lower curve as misclassification probability that is classifying the point as belonging to the class of the upper curve.

For a linear classifier there is in one dimension only a point x, the *decision point*, where values larger than x are classified to one class and values smaller than x are classified into the other class. The optimal decision point minimizes the misclassification rate. Fig. 2.6 shows such an example.

We call function g a *discriminant function* if it has a positive value at $\boldsymbol{x}$ and the corresponding data point is classified as belonging to the positive class and vice versa. Such functions are also called *classification functions*. The class estimation $\hat{y}(\boldsymbol{x})$ ($\hat{}$ indicates estimation), i.e. the classifier is

$$\hat{y}(\boldsymbol{x}) \;=\; \text{sign}\; g(\boldsymbol{x})\;. \tag{2.31}$$

A discriminant function which minimizes the future risk is

$$\begin{aligned} g(\boldsymbol{x}) \;&=\; p(y=1 \mid \boldsymbol{x}) \;-\; p(y=-1 \mid \boldsymbol{x}) \;= \\ &\frac{1}{p(\boldsymbol{x})}\left(\, p(\boldsymbol{x} \mid y=1)\, p(y=1) \;-\; p(\boldsymbol{x} \mid y=-1)\, p(y=-1)\,\right)\;, \end{aligned} \tag{2.32}$$

where only the difference in the last brackets matters because $p(\boldsymbol{x}) > 0$. Note, that the optimal discriminant function is not unique because the difference of strict monotone mappings of $p(y = 1 \mid \boldsymbol{x})$ and $p(y = -1 \mid \boldsymbol{x})$ keep the sign of discriminant function and lead to the same classification rule.

Using this fact we take the logarithm to obtain a more convenient discriminant function which also minimizes the future risk:

$$\begin{aligned} g(\boldsymbol{x}) \;&=\; \ln p(y=1 \mid \boldsymbol{x}) \;-\; \ln p(y=-1 \mid \boldsymbol{x}) \;= \\ &\ln \frac{p(\boldsymbol{x} \mid y=1)}{p(\boldsymbol{x} \mid y=-1)} \;+\; \ln \frac{p(y=1)}{p(y=-1)}\;. \end{aligned} \tag{2.33}$$

For our Gaussian case we obtain

$$\begin{aligned} g(\boldsymbol{x}) \;=\; & -\frac{1}{2}\,(\boldsymbol{x}-\boldsymbol{\mu}_1)^T\,\boldsymbol{\Sigma}_1^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_1) \;-\; \frac{d}{2}\ln 2\pi \;- \\ & \frac{1}{2}\,\ln|\boldsymbol{\Sigma}_1| \;+\; \ln p(y=1) \;+ \\ & \frac{1}{2}\,(\boldsymbol{x}-\boldsymbol{\mu}_{-1})^T\,\boldsymbol{\Sigma}_{-1}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_{-1}) \;+\; \frac{d}{2}\ln 2\pi \;+\; \frac{1}{2}\,\ln|\boldsymbol{\Sigma}_{-1}| \;-\; \ln p(y=-1) \;= \\ & -\frac{1}{2}\,(\boldsymbol{x}-\boldsymbol{\mu}_1)^T\,\boldsymbol{\Sigma}_1^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_1) \;-\; \frac{1}{2}\,\ln|\boldsymbol{\Sigma}_1| \;+\; \ln p(y=1) \;+ \\ & \frac{1}{2}\,(\boldsymbol{x}-\boldsymbol{\mu}_{-1})^T\,\boldsymbol{\Sigma}_{-1}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_{-1}) \;+\; \frac{1}{2}\,\ln|\boldsymbol{\Sigma}_{-1}| \;-\; \ln p(y=-1) \;= \\ & -\frac{1}{2}\boldsymbol{x}^T\left(\boldsymbol{\Sigma}_1^{-1} \;-\; \boldsymbol{\Sigma}_{-1}^{-1}\right)\,\boldsymbol{x} \;+\; \boldsymbol{x}^T\left(\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1 \;-\; \boldsymbol{\Sigma}_{-1}^{-1}\boldsymbol{\mu}_{-1}\right) \;- \\ & \frac{1}{2}\,\boldsymbol{\mu}_1^T\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1 + \frac{1}{2}\,\boldsymbol{\mu}_{-1}^T\boldsymbol{\Sigma}_{-1}^{-1}\boldsymbol{\mu}_{-1} \;-\; \frac{1}{2}\,\ln|\boldsymbol{\Sigma}_1| \;+\; \frac{1}{2}\,\ln|\boldsymbol{\Sigma}_{-1}| \;+ \\ & \ln p(y=1) \;-\; \ln p(y=-1) \;= \\ & -\frac{1}{2}\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x} \;+\; \boldsymbol{w}^T\boldsymbol{x} \;+\; b\,. \end{aligned} \tag{2.34}$$

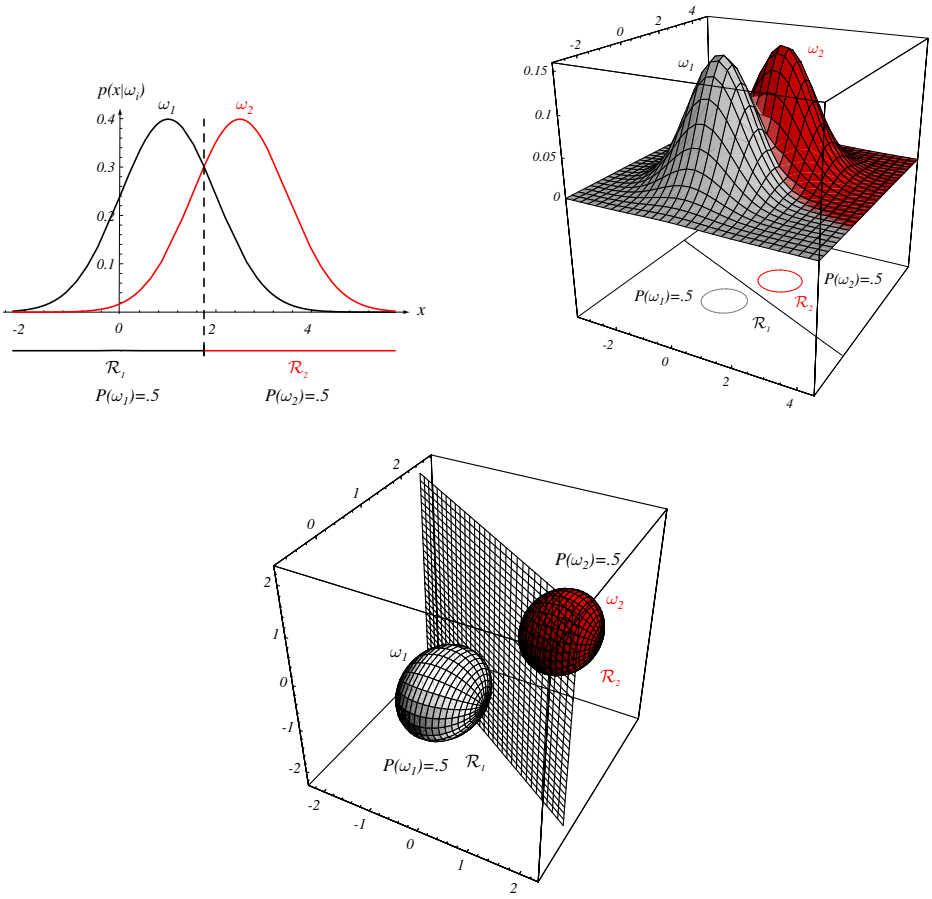

Figure 2.7: Two classes with covariance matrix $\mathbf{\Sigma} = \sigma^2 \boldsymbol{I}$ each in one (top left), two (top right), and three (bottom) dimensions. The optimal boundary is a hyperplane. Copyright © 2001 John Wiley & Sons, Inc.

The function $g(\boldsymbol{x}) = 0$ defines the class boundaries which are hyper-quadrics (hyper-ellipses or hyper-hyperbolas).

If the covariance matrices of both classes are equal, $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_{-1} = \mathbf{\Sigma}$, then the discriminant function is

$$\begin{aligned} g(\boldsymbol{x}) &= \boldsymbol{x}^T \mathbf{\Sigma}^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_{-1}) + \\ &\boldsymbol{\mu}_{-1}^T \mathbf{\Sigma}^{-1} \boldsymbol{\mu}_{-1} - \boldsymbol{\mu}_1^T \mathbf{\Sigma}^{-1} \boldsymbol{\mu}_1 + \ln p(y=1) - \ln p(y=-1) = \\ &\boldsymbol{w}^T \boldsymbol{x} + b \,. \end{aligned} \tag{2.35}$$

The boundary function $g(\boldsymbol{x}) = 0$ is a hyperplane in the d-dimensional space. See examples for $d = 1$, $d = 2$, and $d = 3$ in Fig. 2.7.

For the general case, where $\mathbf{\Sigma}_1 \neq \mathbf{\Sigma}_{-1}$ the boundary functions can be hyperplanes, hyper-ellipsoids, hyper-paraboloids etc. Examples for the 2-dim. case are given in Fig. 2.8.

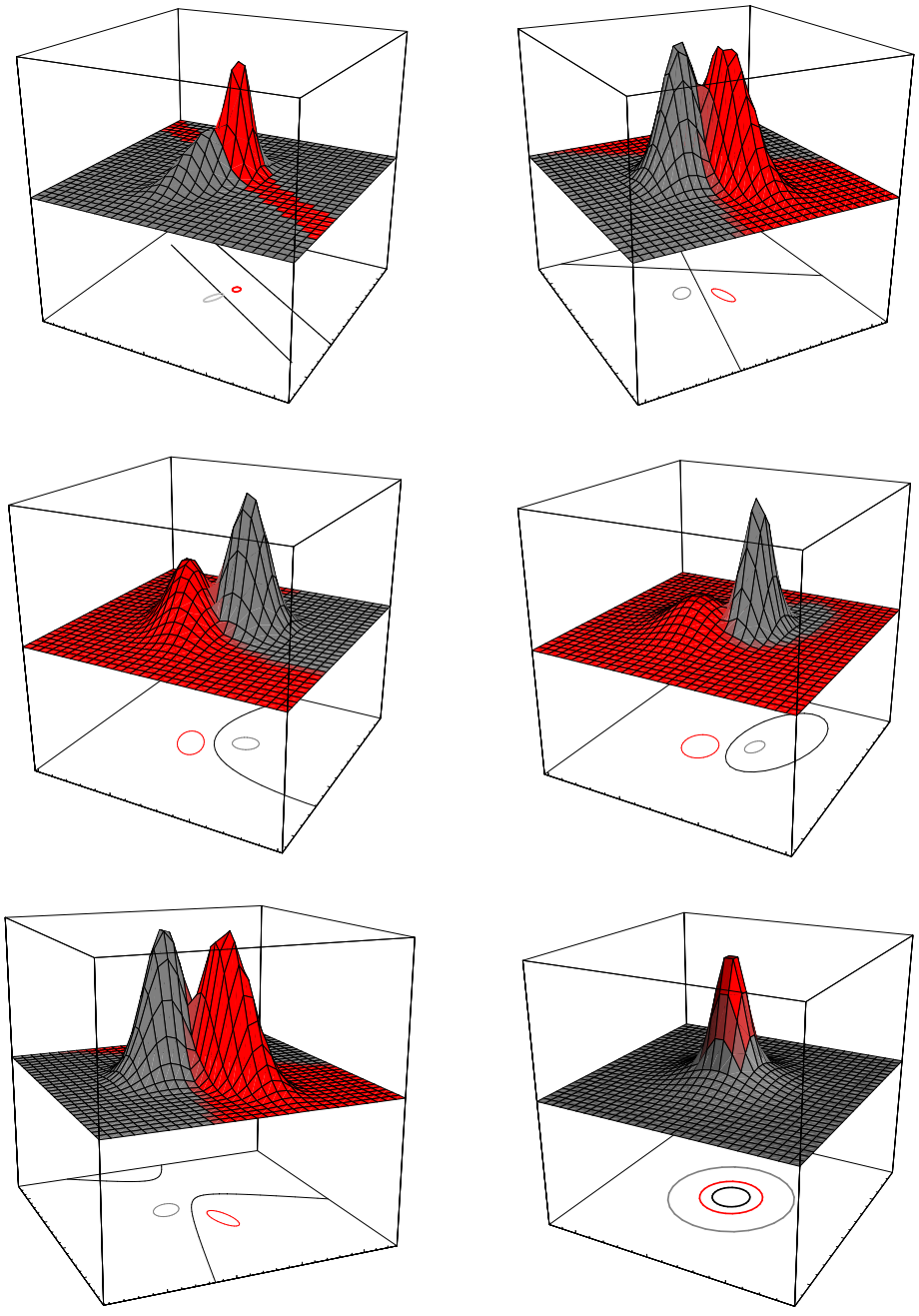

Figure 2.8: Two classes with arbitrary Gaussian covariance lead to boundary functions which are hyperplanes, hyper-ellipsoids, hyperparaboloids etc.

Chapter 3

Maximum Likelihood

In this chapter, we introduce maximum likelihood as quality criterion for unsupervised methods. In this context we introduce concepts like bias and variance, efficient estimator, and the Fisher information matrix.

So far we only considered the supervised task, where we have a label y which should be predicted correctly for future data. We were able to define the loss via the distance between the predicted value and the true value.

For unsupervised tasks defining a loss function and a risk is not as straight forward as in supervised learning.

3.1 Loss for Unsupervised Learning

3.1.1 Projection Methods

Unsupervised tasks include projection of the data into another space in order to fulfill desired requirements. Fig. 3.1 depicts a projection model.

For example with "Principal Component Analysis" (PCA) the data is projected into a lower dimensional space. Here a trade-off between losing information and low dimensionality appears. It is difficult to define a loss function which takes both the dimension and the information loss into account.

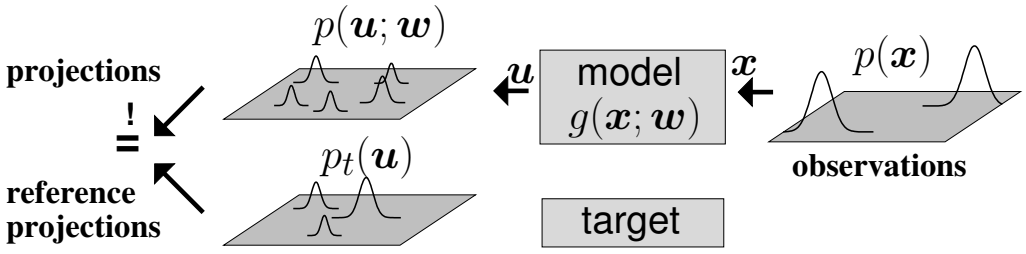

Figure 3.1: Projection model, where the observed data $\boldsymbol{x}$ is the input to the model $\boldsymbol{u} = g(\boldsymbol{x};\boldsymbol{w})$. The model output distribution should match a target distribution or the output distribution should fulfill some constraints. The later can be replaced by the distribution which fulfills the constraints and is closest to the target distribution.

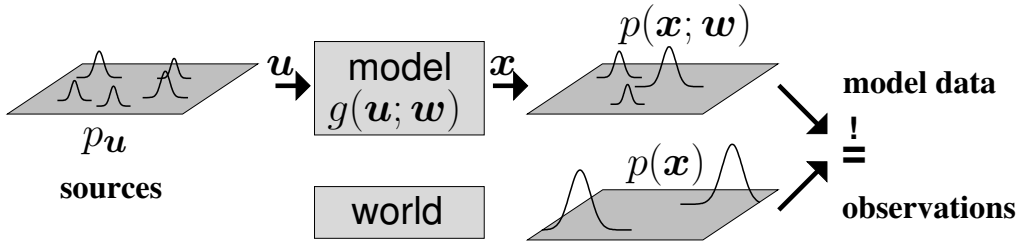

Figure 3.2: Generative model, where the data $\boldsymbol{x}$ is observed and the model $\boldsymbol{x} = g(\boldsymbol{u};\boldsymbol{w})$ should produce the same distribution as the observed distribution. The vectors $\boldsymbol{u}$ are random vectors supplied to the model in order to drive it, i.e. to make it a data generation process.

For example with "Independent Component Analysis" (ICA) the data is projected into a space where the components of the data vectors are mutually independent. As loss function may serve the minimal difference of the actual distribution of the projected data to a factorial distribution (a distribution where the components are independent from each other).

Often only characteristics of a factorial distribution are optimized such as entropy (factorial distribution has maximal entropy under constant variance), cummulants (some are maximal and some are zero for factorial distributions), and others.

For some cases a prototype factorial distribution (e.g. the product of super-Gaussians) is used and the projected data should be aligned to this prototype distribution as good as possible.

For example with "Projection Pursuit" the components have to be maximally non-Gaussian. Here the ideas as for ICA hold, too.

3.1.2 Generative Model

One of the most common unsupervised tasks is to build a *generative model* that is a model which simulates the world and produces the same data as the world. Fig. 3.2 depicts a generative model.

The data generation process of the world is assumed to be probabilistic. Therefore, the observed data of the world are only random examples and we do not want to exactly reproduce the observed data (that can be done by storing the data in data bases). However, we assume that the data generation process produces the data according to some distribution.

The generative model approach attempts to approximate the distribution of the real world data generation process as good as possible. As loss function the distance between the distribution of the data generation process and the model output distribution is suitable.

Generative models include "Factor Analysis", "Latent Variable Models", and Boltzmann Machines.

3.1.3 Parameter Estimation

In another approach to unsupervised learning we assume that we know the model which produces the data but the model is parameterized and the actual parameters are not known. The task of the unsupervised learning method is to estimate the actual parameters.

Here the loss would be the difference between the actual (true) parameter and the estimated parameter. However we have no future data points. But we can evaluate the estimator through the expected difference, where the expectation is made over the training set.

In the next sections we will focus on parameter estimation and will evaluate estimation methods based on expectation over the training set.

3.2 Mean Squared Error, Bias, and Variance

In unsupervised tasks the *training data* is $\{\boldsymbol{z}^1, \ldots, \boldsymbol{z}^l\}$, where $\boldsymbol{z}^i = \boldsymbol{x}^i$ and, therefore, it is $\{\boldsymbol{x}\} = \{\boldsymbol{x}^1, \ldots, \boldsymbol{x}^l\}$ for which we will often simply write $\boldsymbol{X}$ (the matrix of training data).

The true parameter vector is denoted by $\boldsymbol{w}$ and its estimate by $\hat{\boldsymbol{w}}$.

An estimator is *unbiased* if

$$\mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}) = \boldsymbol{w}\,, \tag{3.1}$$

i.e. on the average the estimator will yield the true parameter.

The *bias* is

$$b(\hat{\boldsymbol{w}}) = \mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}) - \boldsymbol{w}\,. \tag{3.2}$$

The *variance* of the estimator is defined as

$$\mathrm{var}(\hat{\boldsymbol{w}}) = \mathrm{E}_{\boldsymbol{X}}\left((\hat{\boldsymbol{w}} - \mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}))^T (\hat{\boldsymbol{w}} - \mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}))\right)\,. \tag{3.3}$$

An evaluation criterion for supervised methods is the *mean squared error* (MSE) as described in the text after eq. (2.2). The MSE in eq. (2.2) was defined as an expectation over future data points.

Here we define the MSE as expectation over the training set, because we deal with unsupervised learning and evaluate the estimator. The MSE gives the expected error as squared distance between the estimated parameter and the true parameter.

$$\mathrm{mse}(\hat{\boldsymbol{w}}) = \mathrm{E}_{\boldsymbol{X}}\left((\hat{\boldsymbol{w}} - \boldsymbol{w})^T (\hat{\boldsymbol{w}} - \boldsymbol{w})\right)\,. \tag{3.4}$$

We can reformulate the MSE:

$$\begin{aligned}\mathrm{mse}(\hat{\boldsymbol{w}}) \;=\; \mathrm{E}_{\boldsymbol{X}}\left((\hat{\boldsymbol{w}} \;-\; \boldsymbol{w})^T\,(\hat{\boldsymbol{w}} \;-\; \boldsymbol{w})\right) \;= \\ \mathrm{E}_{\boldsymbol{X}}\Big(((\hat{\boldsymbol{w}} \;-\; \mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}})) \;+\; (\mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}) \;-\; \boldsymbol{w}))^T \\ ((\hat{\boldsymbol{w}} \;-\; \mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}})) \;+\; (\mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}) \;-\; \boldsymbol{w}))\Big) \;= \\ \mathrm{E}_{\boldsymbol{X}}\Big((\hat{\boldsymbol{w}} \;-\; \mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}))^T\,(\hat{\boldsymbol{w}} \;-\; \mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}})) \;- \\ 2\;(\hat{\boldsymbol{w}} \;-\; \mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}))^T\,(\mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}) \;-\; \boldsymbol{w}) \;+ \\ (\mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}) \;-\; \boldsymbol{w})^T\,(\mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}) \;-\; \boldsymbol{w})\Big) \;= \\ \mathrm{E}_{\boldsymbol{X}}\left((\hat{\boldsymbol{w}} \;-\; \mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}))^T\,(\hat{\boldsymbol{w}} \;-\; \mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}))\right) \;+ \\ (\mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}) \;-\; \boldsymbol{w})^T\,(\mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}) \;-\; \boldsymbol{w}) \;= \\ \mathrm{var}(\hat{\boldsymbol{w}}) \;+\; b^2(\hat{\boldsymbol{w}})\;.\end{aligned} \tag{3.5}$$

where the last but one equality comes from the fact that only $\hat{\boldsymbol{w}}$ depends on $\boldsymbol{X}$ and therefore

$$\begin{aligned}\mathrm{E}_{\boldsymbol{X}}\left((\hat{\boldsymbol{w}} \;-\; \mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}))^T\,(\mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}) \;-\; \boldsymbol{w})\right) \;= \\ (\mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}) \;-\; \mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}))^T\,(\mathrm{E}_{\boldsymbol{X}}(\hat{\boldsymbol{w}}) \;-\; \boldsymbol{w}) \;=\; 0\;.\end{aligned} \tag{3.6}$$

The MSE is decomposed into a variance term $\mathrm{var}(\hat{\boldsymbol{w}})$ and a bias term $b^2(\hat{\boldsymbol{w}})$. The variance has high impact on the performance because large deviations from the true parameter have strong influence on the MSE through the quadratic error term.

Note that averaging linearly reduces the variance. The average is

$$\hat{\boldsymbol{w}}_{a_N} \;=\; \frac{1}{N}\sum_{i=1}^{N}\hat{\boldsymbol{w}}_i\;, \tag{3.7}$$

where

$$\begin{aligned}\hat{\boldsymbol{w}}_i \;&=\; \hat{\boldsymbol{w}}_i\,(\boldsymbol{X}_i) \\ \boldsymbol{X}_i \;&=\; \left\{\boldsymbol{x}^{(i-1)\;l/N\;+\;1},\ldots,\boldsymbol{x}^{i\;l/N}\right\}\;,\end{aligned} \tag{3.8}$$

i.e. $\boldsymbol{X}_i$ is the i-th subset of $\boldsymbol{X}$ and contains l/N elements. The size of the data is l and the examples of $\boldsymbol{X}_i$ range from $(i-1)\;l/N \;+\; 1$ to $i\;l/N$.

The average is unbiased:

$$\mathrm{E}_{\boldsymbol{X}}\left(\hat{\boldsymbol{w}}_{a_N}\right) \;=\; \frac{1}{N}\sum_{i=1}^{N}\mathrm{E}_{\boldsymbol{X}_i}\hat{\boldsymbol{w}}_i \;=\; \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{w} \;=\; \boldsymbol{w}\;. \tag{3.9}$$

The variance is linearly reduced

$$\begin{aligned}\mathrm{covar}_{\boldsymbol{X}}\left(\hat{\boldsymbol{w}}_{a_N}\right) \;=\; \frac{1}{N^2}\sum_{i=1}^{N}\mathrm{covar}_{\boldsymbol{X}_i}\left(\hat{\boldsymbol{w}}_i\right) \;= \\ \frac{1}{N^2}\sum_{i=1}^{N}\mathrm{covar}_{\boldsymbol{X},l/N}\left(\hat{\boldsymbol{w}}\right) \;=\; \frac{1}{N}\mathrm{covar}_{\boldsymbol{X},l/N}\left(\hat{\boldsymbol{w}}\right)\;,\end{aligned} \tag{3.10}$$

where $\mathrm{covar}_{\boldsymbol{X},l/N}\left(\hat{\boldsymbol{w}}\right)$ is the estimator with l/N training points.

We used the facts:

$$\begin{aligned} &\mathrm{covar}_{\boldsymbol{X}}(\boldsymbol{a} + \boldsymbol{b}) = \\ &\mathrm{covar}_{\boldsymbol{X}}(\boldsymbol{a}) + \mathrm{covar}_{\boldsymbol{X}}(\boldsymbol{b}) + 2\,\mathrm{covar}_{\boldsymbol{X}}(\boldsymbol{a}, \boldsymbol{b}) \\ &\mathrm{covar}_{\boldsymbol{X}}(\lambda\,\boldsymbol{a}) = \lambda^2\,\mathrm{covar}_{\boldsymbol{X}}(\boldsymbol{a})\,. \end{aligned} \tag{3.11}$$

For averaging it is important that the training sets $\boldsymbol{X}_i$ are independent from each other and do not overlap. Otherwise the estimators are dependent and the covariance terms between the estimators do not vanish.

One approach to find an optimal estimator is to construct from all unbiased estimators the one with minimal variance, which is called Minimal Variance Unbiased (MVU) estimator.

A MVU estimator does not always exist. However there are methods to check whether a given estimator is a MVU estimator.

3.3 Fisher Information Matrix, Cramer-Rao Lower Bound, and Efficiency

In the following we will define a lower bound, the *Cramer-Rao Lower Bound* for the variance of an unbiased estimator. That induces a lower bound on the MSE of an estimator.

We need the *Fisher information matrix* $\boldsymbol{I}_F$ to define this lower bound. The Fisher information matrix $\boldsymbol{I}_F$ for a parameterized model is

$$\boldsymbol{I}_F(\boldsymbol{w}):\ \left[\boldsymbol{I}_F(\boldsymbol{w})\right]_{ij} = \mathrm{E}_{p(\boldsymbol{x};\boldsymbol{w})}\left(\frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial w_i}\,\frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial w_j}\right), \tag{3.12}$$

and $\left[\boldsymbol{A}\right]_{ij} = A_{ij}$ selects the ijth element of a matrix and

$$\begin{aligned} &\mathrm{E}_{p(\boldsymbol{x};\boldsymbol{w})}\left(\frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial w_i}\,\frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial w_j}\right) = \\ &\int \frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial w_i}\,\frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial w_j}\,p(\boldsymbol{x};\boldsymbol{w})\,d\boldsymbol{x}\,. \end{aligned} \tag{3.13}$$

If the density function $p(\boldsymbol{x};\boldsymbol{w})$ satisfies

$$\forall_{\boldsymbol{w}}:\ \mathrm{E}_{p(\boldsymbol{x};\boldsymbol{w})}\left(\frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}}\right) = \boldsymbol{0} \tag{3.14}$$

then the Fisher information matrix is

$$\boldsymbol{I}_F(\boldsymbol{w}):\ \boldsymbol{I}_F(\boldsymbol{w}) = -\,\mathrm{E}_{p(\boldsymbol{x};\boldsymbol{w})}\left(\frac{\partial^2 \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}\,\partial \boldsymbol{w}}\right). \tag{3.15}$$

The last equation follows from the fact that

$$0 \;=\; \mathrm{E}_{p(\boldsymbol{x};\boldsymbol{w})}\left(\frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}} d\boldsymbol{x}\right) \;= \tag{3.16}$$

$$\int_X \frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}}\, p(\boldsymbol{x};\boldsymbol{w})\, d\boldsymbol{x}$$

$$\Longrightarrow \;\; \frac{\partial}{\partial \boldsymbol{w}} \int_X \frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}}\, p(\boldsymbol{x};\boldsymbol{w})\, d\boldsymbol{x} \;=\; \mathbf{0} \tag{3.17}$$

$$\Longrightarrow \;\; \int_X \left(\frac{\partial^2 \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}\, \partial \boldsymbol{w}}\, p(\boldsymbol{x};\boldsymbol{w}) + \right. \tag{3.18}$$

$$\left. \frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}} \frac{\partial p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}}\right) d\boldsymbol{x} \;=\; \mathbf{0}$$

$$\Longrightarrow \;\; \int_X \left(\frac{\partial^2 \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}\, \partial \boldsymbol{w}}\, p(\boldsymbol{x};\boldsymbol{w}) + \right. \tag{3.19}$$

$$\left. \frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}}\, \frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}}\, p(\boldsymbol{x};\boldsymbol{w})\right) d\boldsymbol{x} \;=\; \mathbf{0}$$

$$\Longrightarrow \;\; -\,\mathrm{E}_{p(\boldsymbol{x};\boldsymbol{w})}\left(\frac{\partial^2 \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}\, \partial \boldsymbol{w}}\right) \;= \tag{3.20}$$

$$\mathrm{E}_{p(\boldsymbol{x};\boldsymbol{w})}\left(\frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}}\, \frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}}\right) .$$

The Fisher information gives the amount of information that an observable random variable $\boldsymbol{x}$ carries about an unobservable parameter $\boldsymbol{w}$ upon which the parameterized density function $p(\boldsymbol{x};\boldsymbol{w})$ of $\boldsymbol{x}$ depends.

Theorem 3.1 (Cramer-Rao Lower Bound (CRLB))
Assume that

$$\forall_{\boldsymbol{w}} : \;\; \mathrm{E}_{p(\boldsymbol{x};\boldsymbol{w})}\left(\frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}}\right) \;=\; \mathbf{0} \tag{3.21}$$

and that the estimator $\hat{\boldsymbol{w}}$ is unbiased.

Then,

$$\mathrm{covar}(\hat{\boldsymbol{w}}) \;-\; \boldsymbol{I}_F^{-1}(\boldsymbol{w}) \tag{3.22}$$

is positive definite:

$$\mathrm{covar}(\hat{\boldsymbol{w}}) \;-\; \boldsymbol{I}_F^{-1}(\boldsymbol{w}) \;\geq\; \mathbf{0}\,. \tag{3.23}$$

An unbiased estimator attains the bound in that $\mathrm{covar}(\hat{\boldsymbol{w}}) \;=\; \boldsymbol{I}_F^{-1}(\boldsymbol{w})$ *if and only if*

$$\frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}} \;=\; \boldsymbol{A}(\boldsymbol{w})\;(\boldsymbol{g}(\boldsymbol{x}) \;-\; \boldsymbol{w}) \tag{3.24}$$

for some function $\boldsymbol{g}$ and square matrix $\boldsymbol{A}(\boldsymbol{w})$. In this case the MVU estimator is

$$\hat{\boldsymbol{w}} \;=\; \boldsymbol{g}(\boldsymbol{x}) \quad \textit{with} \;\; \mathrm{covar}(\hat{\boldsymbol{w}}) \;=\; \boldsymbol{A}^{-1}(\boldsymbol{w})\,. \tag{3.25}$$

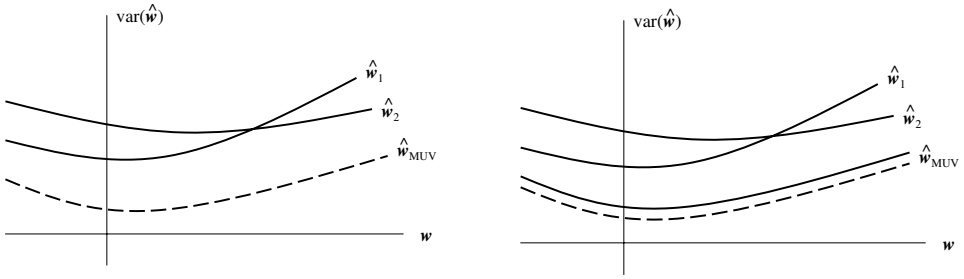

Figure 3.3: The variance of an estimator $\hat{\boldsymbol{w}}$ as a function of the true parameter is shown. We show three estimators: $\hat{\boldsymbol{w}}_1$, $\hat{\boldsymbol{w}}_2$, and the minimal variance unbiased $\hat{\boldsymbol{w}}_{\mathrm{MVU}}$. Left: The MVU estimator $\hat{\boldsymbol{w}}_{\mathrm{MVU}}$ is efficient because it reaches the CRLB. Right: The MVU estimator $\hat{\boldsymbol{w}}_{\mathrm{MVU}}$ does not reach the CRLB and is not efficient.

Note that

$$\left[\mathrm{covar}(\hat{\boldsymbol{w}}) \; - \; \boldsymbol{I}_F^{-1}(\boldsymbol{w})\right]_{ii} \; \geq \; 0 \, , \tag{3.26}$$

therefore

$$\mathrm{var}(\hat{w}_i) \; = \; [\mathrm{covar}(\hat{\boldsymbol{w}})]_{ii} \; \geq \; \left[\boldsymbol{I}_F^{-1}(\boldsymbol{w})\right]_{ii} \, . \tag{3.27}$$

An estimator is said to be *efficient* if it reaches the CRLB. It is efficient in that it efficiently makes use of the data and extracts information to estimate the parameter.

A MVU estimator may or may not be efficient. This means it could have minimum variance but without reaching the CRLB as depicted in Fig. 3.3.

3.4 Maximum Likelihood Estimator

In many cases of parameter estimation with unsupervised learning the MVU estimator is unknown or does not exist.

A very popular estimator is the Maximum Likelihood Estimator (MLE) on which almost all practical estimation tasks are based. The popularity stems from the fact that it can be applied to a broad range of problems and it approximates the MVU estimator for large data sets. The MLE is even asymptotically efficient and unbiased. That means the MLE does everything right and this efficiently if enough data is available.

The likelihood $\mathcal{L}$ of the data set $\{\boldsymbol{x}\} \; = \; \{\boldsymbol{x}^1, \ldots, \boldsymbol{x}^l\}$ is

$$\mathcal{L}(\{\boldsymbol{x}\}; \boldsymbol{w}) \; = \; p(\{\boldsymbol{x}\}; \boldsymbol{w}) \, , \tag{3.28}$$

i.e. the probability of the model $p(\boldsymbol{x}; \boldsymbol{w})$ to produce the data set. However the set $\{\boldsymbol{x}\}$ has zero measure and therefore the density at the data set $\{\boldsymbol{x}\}$ must be used.

For iid data sampling the likelihood is

$$\mathcal{L}(\{\boldsymbol{x}\}; \boldsymbol{w}) \; = \; p(\{\boldsymbol{x}\}; \boldsymbol{w}) \; = \; \prod_{i=1}^{l} p(\boldsymbol{x}^i; \boldsymbol{w}) \, . \tag{3.29}$$

Instead of maximizing the likelihood $\mathcal{L}$ the log-likelihood $\ln \mathcal{L}$ is maximized or the negative log-likelihood $-\ln \mathcal{L}$ is minimized. The logarithm transforms the product of the iid data sampling into a sum:

$$-\ln \mathcal{L}(\{\boldsymbol{x}\}; \boldsymbol{w}) = -\sum_{i=1}^{l} \ln p(\boldsymbol{x}^i; \boldsymbol{w}) \,. \tag{3.30}$$

To motivate the use of the density in the likelihood one can assume that if $p(\boldsymbol{x}^i; \boldsymbol{w})$ is written actually $p(\boldsymbol{x}^i; \boldsymbol{w})\, d\boldsymbol{x}$ is meant, which gives the probability of observing $\boldsymbol{x}$ in a region of volume $d\boldsymbol{x}$ around $\boldsymbol{x}^i$. In this case the likelihood gives the probability of the model to produce similar data points as $\{\boldsymbol{x}\}$, where similar means data points in a volume $d\boldsymbol{x}$ around the actual observed data points.

However the fact that the MLE is so popular is based on its simple use and its properties (given in the next section) especially that it is optimal for the number of training points going to infinity.

3.5 Properties of Maximum Likelihood Estimator

In the next subsections different properties of the MLE are given. First, MLE is invariant under parameter change. Then, most importantly, the MLE is asymptotically unbiased and efficient, i.e. asymptotically optimal. Finally, the MLE is consistent for zero CRLB.

3.5.1 MLE is Invariant under Parameter Change

Theorem 3.2 (Parameter Change Invariance)
Let g be a function changing the parameter $\boldsymbol{w}$ into parameter $\boldsymbol{u}$: $\boldsymbol{u} = g(\boldsymbol{w})$, then

$$\hat{\boldsymbol{u}} = g(\hat{\boldsymbol{w}}) \,, \tag{3.31}$$

where the estimators are MLE. If g changes $\boldsymbol{w}$ into different $\boldsymbol{u}$ then $\hat{\boldsymbol{u}} = g(\hat{\boldsymbol{w}})$ maximizes the likelihood function

$$\max_{\boldsymbol{w}:\boldsymbol{u}=g(\boldsymbol{w})} p(\{\boldsymbol{x}\}; \boldsymbol{w}) \,. \tag{3.32}$$

This theorem is important because for some models parameter changes simplify the expressions for the densities.

3.5.2 MLE is Asymptotically Unbiased and Efficient

Note, that an estimator $\hat{\boldsymbol{w}} = \hat{\boldsymbol{w}}(\boldsymbol{X})$ changes its properties with the size l of the training set $\boldsymbol{X}$. For example for a reasonable estimator the variance should decrease with increasing l.

The maximum likelihood estimator is *asymptotically unbiased*

$$\mathrm{E}_{p(\boldsymbol{x};\boldsymbol{w})}\left(\hat{\boldsymbol{w}}\right) \overset{l\to\infty}{\to} \boldsymbol{w} \tag{3.33}$$

and it is *asymptotically efficient*

$$\operatorname{covar}(\hat{\boldsymbol{w}}) \overset{l\to\infty}{\to} \operatorname{CRLB} . \tag{3.34}$$

These properties are derived from the following theorem

Theorem 3.3 (MLE Asymptotic Properties)
If $p(\boldsymbol{x};\boldsymbol{w})$ *satisfies*

$$\forall_{\boldsymbol{w}} : \ \mathrm{E}_{p(\boldsymbol{x};\boldsymbol{w})}\left(\frac{\partial \ln p(\boldsymbol{x};\boldsymbol{w})}{\partial \boldsymbol{w}} d\boldsymbol{x}\right) = \mathbf{0} \tag{3.35}$$

then the MLE which maximizes $p(\{\boldsymbol{x}\};\boldsymbol{w})$ *is asymptotically distributed according to*

$$\hat{\boldsymbol{w}} \overset{l\to\infty}{\propto} \mathcal{N}\left(\boldsymbol{w}, \boldsymbol{I}_F^{-1}(\boldsymbol{w})\right) , \tag{3.36}$$

where $\boldsymbol{I}_F(\boldsymbol{w})$ *is the Fisher information matrix evaluated at the unknown parameter* $\boldsymbol{w}$*.*

This quite general theorem is the basis of the asymptotic optimal properties of the MLE.

However for practical applications the number l is finite and the performance of the MLE is not known.

For example consider the general linear model

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{w} + \boldsymbol{\epsilon} , \tag{3.37}$$

where $\boldsymbol{\epsilon} \propto \mathcal{N}(\mathbf{0}, \boldsymbol{C})$ is an additive Gaussian noise vector.

Then the MLE is

$$\hat{\boldsymbol{w}} = \left(\boldsymbol{A}^T\boldsymbol{C}^{-1}\boldsymbol{A}\right)^{-1} A^T\boldsymbol{C}^{-1}\boldsymbol{x} . \tag{3.38}$$

which is also efficient and, therefore, MVU. The density of $\hat{\boldsymbol{w}}$ is

$$\hat{\boldsymbol{w}} \propto \mathcal{N}\left(\boldsymbol{w} , \left(\boldsymbol{A}^T\boldsymbol{C}^{-1}\boldsymbol{A}\right)^{-1}\right) . \tag{3.39}$$

Note for factor analysis which will be considered later also $\boldsymbol{C}$ has to be estimated.

3.5.3 MLE is Consistent for Zero CRLB

A estimator is said to be *consistent* if

$$\hat{\boldsymbol{w}} \overset{l\to\infty}{\to} \boldsymbol{w} , \tag{3.40}$$

i.e. for large training sets the estimator approaches the true value.

Later – in the empirical risk minimization treatment by V. Vapnik in 5.2 – we need a more formal definition for consistency as

$$\lim_{l\to\infty} p\left(|\hat{\boldsymbol{w}} - \boldsymbol{w}| > \epsilon\right) = 0 . \tag{3.41}$$

The MLE is consistent if the CRLB is zero. This follows directly from the fact that MLE is asymptotically unbiased and efficient, i.e. the variance will approach zero.

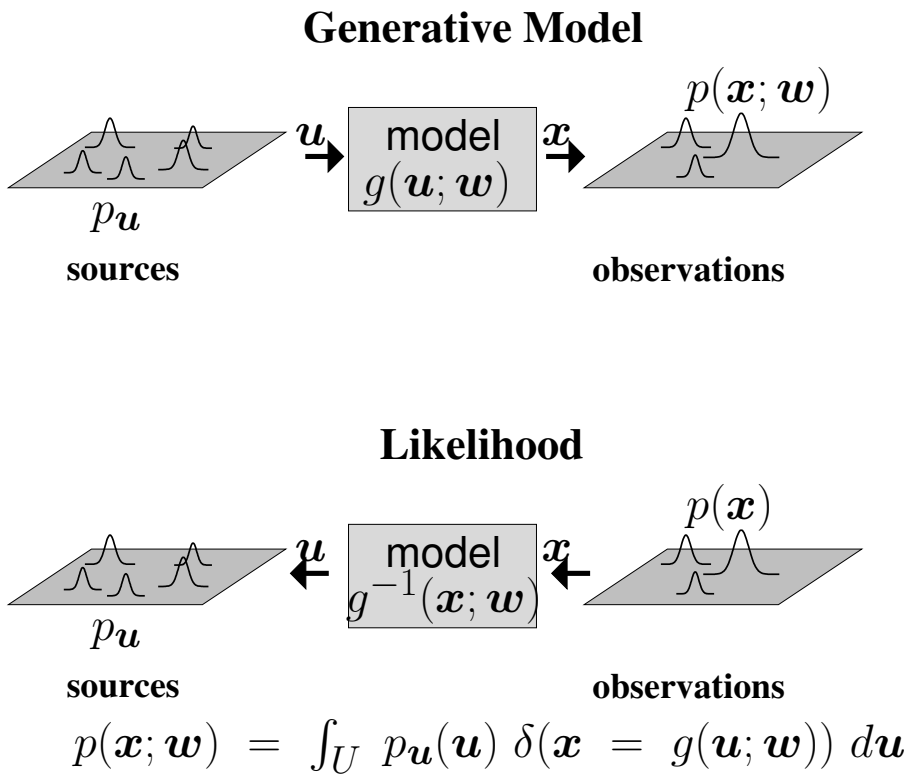

Figure 3.4: The maximum likelihood problem. Top: the generative model which produces data points. Bottom: in order to compute the likelihood for $\boldsymbol{x}$ all points $\boldsymbol{u}$ which are mapped to $\boldsymbol{x}$ must be determined and then multiplied with the probability $p_{\boldsymbol{u}}(\boldsymbol{u})$ that $\boldsymbol{u}$ is observed in the model.

3.6 Expectation Maximization

The likelihood can be maximized by gradient descent methods as will be described in Section 7.1. However the likelihood must be expressed analytically to obtain the derivatives. For some models the likelihood cannot be computed analytically because of hidden states of the model, of a many-to-one output mapping of the model, or of non-linearities. As depicted in Fig. 3.4, to compute the likelihood the inverse of a function – more precise, all elements $\boldsymbol{u}$ which are mapped to a certain observed point $\boldsymbol{x}$ – must be computed in order to obtain the likelihood that the point is generated by the model. If g is highly nonlinear, then the integral which determines the likelihood is difficult to compute analytically. To guess the likelihood numerically is difficult as the density of the model output at a certain point in space must be estimated.

The variables $\boldsymbol{u}$ in Fig. 3.4 can be interpreted as unobserved variables, i.e. *hidden variables* or *latent variables*. For models with hidden variables the likelihood is determined by all possible values of the hidden variables which can produce output $\boldsymbol{x}$.

For many models the joint probability $p(\{\boldsymbol{x}\}, \boldsymbol{u}; \boldsymbol{w})$ of the hidden variables $\boldsymbol{u}$ and observations $\{\boldsymbol{x}\}$ is easier to compute than the likelihood of the observations. If we can also estimate $p(\boldsymbol{u} \mid \{\boldsymbol{x}\}; \boldsymbol{w})$ of the hidden variables $\boldsymbol{u}$ using the parameters $\boldsymbol{w}$ and given the observations $\{\boldsymbol{x}\}$ then we can apply the Expectation Maximization (EM) algorithm.

Let us assume we have an estimation $Q(\boldsymbol{u} \mid \{\boldsymbol{x}\})$ for $p(\boldsymbol{u} \mid \{\boldsymbol{x}\}; \boldsymbol{w})$, which is some density with respect to $\boldsymbol{u}$. The following inequality is the basis for the EM algorithm:

$$\begin{aligned} \ln \mathcal{L}(\{\boldsymbol{x}\}; \boldsymbol{w}) \;&=\; \ln p(\{\boldsymbol{x}\}; \boldsymbol{w}) \;= \\ &\ln \int_U p(\{\boldsymbol{x}\}, \boldsymbol{u}; \boldsymbol{w})\, d\boldsymbol{u} \;= \\ &\ln \int_U \frac{Q(\boldsymbol{u} \mid \{\boldsymbol{x}\})}{Q(\boldsymbol{u} \mid \{\boldsymbol{x}\})} p(\{\boldsymbol{x}\}, \boldsymbol{u}; \boldsymbol{w})\, d\boldsymbol{u} \;\geq \\ &\int_U Q(\boldsymbol{u} \mid \{\boldsymbol{x}\}) \;\ln \frac{p(\{\boldsymbol{x}\}, \boldsymbol{u}; \boldsymbol{w})}{Q(\boldsymbol{u} \mid \{\boldsymbol{x}\})}\, d\boldsymbol{u} \;= \\ &\int_U Q(\boldsymbol{u} \mid \{\boldsymbol{x}\}) \;\ln p(\{\boldsymbol{x}\}, \boldsymbol{u}; \boldsymbol{w})\, d\boldsymbol{u} - \\ &\int_U Q(\boldsymbol{u} \mid \{\boldsymbol{x}\}) \;\ln Q(\boldsymbol{u} \mid \{\boldsymbol{x}\})\, d\boldsymbol{u} \;= \\ &\mathcal{F}(Q, \boldsymbol{w})\;. \end{aligned} \tag{3.42}$$

where the "$\geq$" is the application of Jensen's inequality. Jensen's inequality states that the value of a convex function of an integral is smaller or equal to the integral of the convex function applied to the integrand. Therefore a convex function of an expectation is smaller or equal to the expectation of the convex function. Here the expectation with respect to $Q(\boldsymbol{u} \mid \{\boldsymbol{x}\})$ is used and the fact that $-\ \ln$ is a convex function.

Above inequality states that $\mathcal{F}(Q, \boldsymbol{w})$ is a lower bound to the log-likelihood $\ln \mathcal{L}(\{\boldsymbol{x}\}; \boldsymbol{w})$.

The EM algorithm is an iteration between two steps, the "E"-step and the "M"-step:

$$\begin{aligned} &\textbf{E-step:} \\ &Q_{k+1} \;=\; \arg\max_{Q}\; \mathcal{F}(Q, \boldsymbol{w}_k) \\ &\textbf{M-step:} \\ &\boldsymbol{w}_{k+1} \;=\; \arg\max_{\boldsymbol{w}}\; \mathcal{F}(Q_{k+1}, \boldsymbol{w})\;. \end{aligned} \tag{3.43}$$

It is important to note that in the E-step the maximal Q is

$$\begin{aligned} Q_{k+1}(\boldsymbol{u} \mid \{\boldsymbol{x}\}) \;&=\; p(\boldsymbol{u} \mid \{\boldsymbol{x}\}; \boldsymbol{w}_k) \\ \mathcal{F}(Q_{k+1}, \boldsymbol{w}_k) \;&=\; \ln \mathcal{L}(\{\boldsymbol{x}\}; \boldsymbol{w}_k)\;. \end{aligned} \tag{3.44}$$

This means that the maximal Q is the posterior or the hidden variables $p(\boldsymbol{u} \mid \{\boldsymbol{x}\}; \boldsymbol{w}_k)$ using the current parameters $\boldsymbol{w}_k$. Furthermore, the lower bound F is equal to the log-likelihood with the current parameters $\boldsymbol{w}_k$. **The bound is thight and reaches the log-likelihood.**

To see the last statements:

$$p(\boldsymbol{u}, \{\boldsymbol{x}\}; \boldsymbol{w}_k) \;=\; p(\boldsymbol{u} \mid \{\boldsymbol{x}\}; \boldsymbol{w}_k)\; p(\{\boldsymbol{x}\}; \boldsymbol{w}_k)\;, \tag{3.45}$$

therefore

$$\begin{aligned}\mathcal{F}(Q, \boldsymbol{w}) \;=\; \int_U Q(\boldsymbol{u} \mid \{\boldsymbol{x}\}) \;\ln \frac{p(\{\boldsymbol{x}\}, \boldsymbol{u}; \boldsymbol{w})}{Q(\boldsymbol{u} \mid \{\boldsymbol{x}\})}\, d\boldsymbol{u} \;= \\ \int_U Q(\boldsymbol{u} \mid \{\boldsymbol{x}\}) \;\ln \frac{p(\boldsymbol{u} \mid \{\boldsymbol{x}\}; \boldsymbol{w})}{Q(\boldsymbol{u} \mid \{\boldsymbol{x}\})}\, d\boldsymbol{u} \;+\; \ln p(\{\boldsymbol{x}\}; \boldsymbol{w}) \;= \\ -\int_U Q(\boldsymbol{u} \mid \{\boldsymbol{x}\}) \;\ln \frac{Q(\boldsymbol{u} \mid \{\boldsymbol{x}\})}{p(\boldsymbol{u} \mid \{\boldsymbol{x}\}; \boldsymbol{w})}\, d\boldsymbol{u} \;+\; \ln \mathcal{L}(\{\boldsymbol{x}\}; \boldsymbol{w})\end{aligned} \tag{3.46}$$

The expression $\int_U Q(\boldsymbol{u} \mid \{\boldsymbol{x}\}) \;\ln \frac{Q(\boldsymbol{u}|\{\boldsymbol{x}\})}{p(\boldsymbol{u}|\{\boldsymbol{x}\};\boldsymbol{w})}\, d\boldsymbol{u}$ is the Kullback-Leibler divergence $D_{\mathrm{KL}}(Q \parallel p)$ between $Q(\boldsymbol{u} \mid \{\boldsymbol{x}\})$ and $p(\boldsymbol{u} \mid \{\boldsymbol{x}\}; \boldsymbol{w})$. The Kullback-Leibler divergence $\mathrm{KL}(p_1, p_2)$ is defined as

$$D_{\mathrm{KL}}(p_1 \parallel p_2) \;=\; \int_U p_1(\boldsymbol{u}) \;\ln \frac{p_1(\boldsymbol{u})}{p_2(\boldsymbol{u})}\, d\boldsymbol{u} \tag{3.47}$$

and the *cross entropy* as

$$-\int_U p_1(\boldsymbol{u}) \;\ln p_2(\boldsymbol{u})\, d\boldsymbol{u}\,. \tag{3.48}$$

The Kullback-Leibler divergence is always greater than or equal to zero:

$$D_{\mathrm{KL}}(p_1 \parallel p_2) \;\geq\; 0 \tag{3.49}$$

because

$$\begin{aligned}0 \;=\; \ln 1 \;=\; \ln \int_U p_2(\boldsymbol{u})\, d\boldsymbol{u} \;= \\ \ln \int_U p_1(\boldsymbol{u})\, \frac{p_2(\boldsymbol{u})}{p_1(\boldsymbol{u})}\, d\boldsymbol{u} \;\geq \\ \int_U p_1(\boldsymbol{u}) \;\ln \frac{p_2(\boldsymbol{u})}{p_1(\boldsymbol{u})}\, d\boldsymbol{u} \;=\; -\,D_{\mathrm{KL}}(p_1 \parallel p_2)\,.\end{aligned} \tag{3.50}$$

Thus, if $D_{\mathrm{KL}}(Q \parallel p) \;=\; 0$ then $\mathcal{F}(Q, \boldsymbol{w}_k)$ is maximized because the Kullback-Leibler divergence, which enters the equation with a negative sign, is minimal. We have $Q(\boldsymbol{u} \mid \{\boldsymbol{x}\}) \;=\; p(\boldsymbol{u} \mid \{\boldsymbol{x}\}; \boldsymbol{w})$ and obtain

$$\mathcal{F}(Q, \boldsymbol{w}) \;=\; \ln \mathcal{L}(\{\boldsymbol{x}\}; \boldsymbol{w})\,. \tag{3.51}$$

In the M-step only the expression $\int_U Q_{k+1}(\boldsymbol{u} \mid \{\boldsymbol{x}\}) \;\ln p(\{\boldsymbol{x}\}, \boldsymbol{u}; \boldsymbol{w})\, d\boldsymbol{u}$ must be considered because the other term (the entropy of Q_{k+1}) is independent of the parameters $\boldsymbol{w}$.

The EM algorithm can be interpreted as:

- E-step: Tighten the lower bound to equality: $\mathcal{F}(Q, \boldsymbol{w}) \;=\; \ln \mathcal{L}(\{\boldsymbol{x}\}; \boldsymbol{w})\,.$
- M-step: Maximize the lower bound which is at the equality and therefore increase the likelihood. This might lead to a lower bound which is no longer tight.

The EM algorithm increases the lower bound because in both steps the lower bound is maximized.

Can it happen that maximizing the lower bound may decrease the likelihood? No! At the beginning of the M-step we have $\mathcal{F}(Q_{k+1}, \boldsymbol{w}_k) = \ln \mathcal{L}(\{\boldsymbol{x}\}; \boldsymbol{w}_k)$, and the E-step does not change the parameters $\boldsymbol{w}$:

$$\begin{aligned} \ln \mathcal{L}(\{\boldsymbol{x}\}; \boldsymbol{w}_k) \;=\; \mathcal{F}(Q_{k+1}, \boldsymbol{w}_k) \;\leq \\ \mathcal{F}(Q_{k+1}, \boldsymbol{w}_{k+1}) \;\leq\; \mathcal{F}(Q_{k+2}, \boldsymbol{w}_{k+1}) \;=\; \ln \mathcal{L}(\{\boldsymbol{x}\}; \boldsymbol{w}_{k+1}) \;, \end{aligned} \tag{3.52}$$

where the first "$\leq$" is from the M-step which gives w_{k+1} and the second "$\leq$" from the E-step which gives Q_{k+2}.

The EM algorithms will later be derived for hidden Markov models, mixture of Gaussians, factor analysis, independent component analysis, etc.

3.7 Maximum Entropy Estimation

A *maximum entropy probability distribution* is the distribution with maximal entropy given a class of distributions. If any prior knowledge is missing except that a distribution a certain class, then maximum entropy distribution should be chosen because

- it has minimal prior assumptions on the distribution and
- physical systems converge over time to maximal entropy configurations which makes it the most likely observed solution.

The *principle of maximum entropy* was first expounded by E.T. Jaynes in 1957, where he emphasized a natural correspondence between statistical mechanics and information theory.

For discrete random variables $p_k = p(x = x_k)$, the entropy of is defined as

$$H \;=\; -\sum_{k \geq 1} p_k \log p_k \;. \tag{3.53}$$

We assume $p_k log p_k = 0$ for $p_k = 0$. For continuous random variables x with probability density $p(x)$, the entropy is

$$H \;=\; -\int_{-\infty}^{\infty} p(x) \log p(x)\; dx \;, \tag{3.54}$$

where we set $p(x) log p(x) = 0$ for $p(x) = 0$.

The **normal distribution** $N(\mu, \sigma^2)$ has maximum entropy among all real-valued distributions with mean μ and standard deviation σ. Normality imposes the minimal prior assumptions given the first two moments. The **uniform distribution** on the interval $[a, b]$ is the maximum entropy distribution among all continuous distributions which are supported in the interval $[a, b]$. The **exponential distribution** with mean $1/\lambda$ is the maximum entropy distribution among all continuous distributions supported in $[0, \infty]$ that have a mean $1/\lambda$.

Theorem 3.4 (Boltzmann's Theorem: Discrete)
Suppose $S = \{x_1, x_2, \ldots\}$ is a (finite or infinite) discrete subset of the reals and n functions $f_1, \ldots, f_n$ and n numbers $a_1, \ldots, a_n$ are given. Let be C the class of all discrete random variables X which are supported on S and which satisfy the n conditions:

$$\mathrm{E}(f_j(X)) \;=\; a_j \quad \textit{for } j = 1, \ldots, n \tag{3.55}$$

If there exists a member of C which assigns positive probability to all members of S and if there exists a maximum entropy distribution for C, then this distribution has the following shape:

$$\Pr(X = x_k) \;=\; c\, \exp\left(\sum_{j=1}^{n} \lambda_j\, f_j(x_k)\right) \quad \textit{for } k = 1, 2, \ldots\,, \tag{3.56}$$

where the constants c and λ_j have to be determined so that the sum of the probabilities is 1 and the above conditions for the expected values are satisfied.

Conversely, if constants c and λ_j as above can be found, then the above distribution is indeed the maximum entropy distribution for class C.

Theorem 3.5 (Boltzmann's Theorem: Continuous)
Suppose S is a closed subset of the real numbers $\mathbb{R}$ and n measurable functions $f_1, \ldots, f_n$ and n numbers $a_1, \ldots, a_n$ are given. Let be C the class of all continuous random variables which are supported on S and which satisfy the n expected value conditions:

$$\mathrm{E}(f_j(X)) \;=\; a_j \quad \textit{for } j = 1, \ldots, n \tag{3.57}$$

If there is a member in C whose density function is positive everywhere in S, and if there exists a maximal entropy distribution for C, then its probability density $p(x)$ has the following shape:

$$p(x) \;=\; c\, \exp\left(\sum_{j=1}^{n} \lambda_j\, f_j(x)\right) \quad \textit{for all } x \in S\,, \tag{3.58}$$

where the constants c and λ_j have to be determined so that the integral of $p(x)$ over S is 1 and the above conditions for the expected values are satisfied.

Conversely, if constants c and λ_j like this can be found, then $p(x)$ is indeed the density of the (unique) maximum entropy distribution for class C.

This theorem is proved with the calculus of variations and Lagrange multipliers.

Not all classes of distributions contain a maximum entropy distribution. A class may contain distributions of arbitrarily large entropy (e.g. the class of all continuous distributions on $\mathbb{R}$ with mean 0 but arbitrary standard deviation). Or the entropies of distributions from a class are bounded from above but there is no distribution which attains the maximal entropy (e.g. the class of all continuous distributions X on $\mathbb{R}$ with $\mathrm{E}(X) = 0$ and $\mathrm{E}(X^2) = \mathrm{E}(X^3) = 1$. The expected value restrictions for the class C may force the probability distribution to be zero in certain subsets of S. In that case Boltzmann's theorem does not apply, but S can be shrinked.

SOLUTION: DISCRETE CASE
We require our probability distribution p to satisfy

$$\sum_{i=1}^{n} p(x_i)\ f_k(x_i)\ =\ F_k \qquad k = 1, \ldots, m\ . \tag{3.59}$$

Furthermore, the probability must sum to one:

$$\sum_{i=1}^{n} p(x_i)\ =\ 1\ . \tag{3.60}$$

The probability distribution with maximum information entropy subject to these constraints is

$$p(x_i)\ =\ \frac{1}{Z(\lambda_1, \ldots, \lambda_m)}\ \exp\left(\lambda_1 f_1(x_i)\ +\ \cdots\ +\ \lambda_m f_m(x_i)\right)\ . \tag{3.61}$$

This distribution is called the *Gibbs distribution* in statistical mechanics. The normalization constant Z is determined by

$$Z(\lambda_1, \ldots, \lambda_m)\ =\ \sum_{i=1}^{n} \exp\left(\lambda_1 f_1(x_i)\ +\ \cdots\ +\ \lambda_m f_m(x_i)\right)\ , \tag{3.62}$$

and is called the *partition function*. The value of the Lagrange multipliers λ_k are determined by

$$F_k\ =\ \frac{\partial}{\partial \lambda_k} \log Z(\lambda_1, \ldots, \lambda_m)\ . \tag{3.63}$$

These m simultaneous equations may not have a closed form solution but can be solved numerically.

The derivatives are:

$$\begin{aligned} \frac{\partial}{\partial \lambda_k} \log Z(\lambda_1, \ldots, \lambda_m)\ &=\ \frac{1}{Z(\lambda_1, \ldots, \lambda_m)}\ \sum_{i=1}^{n} f_k(x_i) \exp\left(\lambda_1 f_1(x_i)\ +\ \cdots\ +\ \lambda_m f_m(x_i)\right) \\ &=\ \sum_{i=1}^{n} p(x_i)\ f_k(x_i)\ . \end{aligned} \tag{3.64}$$

SOLUTION: CONTINUOUS CASE
Instead of the entropy, we use the Kullback-Leibler divergence of m from p

$$D_{\mathrm{KL}}(p \parallel m)\ =\ -\ \int p(x) \log \frac{p(x)}{m(x)}\, dx\ , \tag{3.65}$$

where $m(x)$ is proportional to the limiting density of discrete points and is assumed to be known. The Kullback-Leibler divergence generalizes the entropy since for constant m the Kullback-Leibler divergence is the entropy plus a constant.

We require our probability density function p to satisfy

$$\int p(x)\ f_k(x)\, dx\ =\ F_k \qquad k = 1, \ldots, m\ . \tag{3.66}$$

The density must integrate to one:

$$\int p(x)\, dx \;=\; 1\,. \tag{3.67}$$

The probability density function p with maximum D_{KL} subject to these constraints is

$$p(x) \;=\; \frac{1}{Z(\lambda_1, \ldots, \lambda_m)}\, m(x)\, \exp\left(\lambda_1 f_1(x) \;+\; \cdots \;+\; \lambda_m f_m(x)\right) \tag{3.68}$$

with the partition function Z given by

$$Z(\lambda_1, \ldots, \lambda_m) \;=\; \int m(x)\, \exp\left(\lambda_1 f_1(x) \;+\; \cdots \;+\; \lambda_m f_m(x)\right)\, dx\,. \tag{3.69}$$

Again the values of the Lagrange multipliers λ_k are determined by:

$$F_k \;=\; \frac{\partial}{\partial \lambda_k} \log Z(\lambda_1, \ldots, \lambda_m)\,. \tag{3.70}$$

The derivatives are the contraints:

$$\begin{aligned} \frac{\partial}{\partial \lambda_k} \log Z(\lambda_1, \ldots, \lambda_m) \;&=\; \frac{1}{Z(\lambda_1, \ldots, \lambda_m)} \int f_k(x) \exp\left(\lambda_1 f_1(x) \;+\; \cdots \;+\; \lambda_m f_m(x)\right)\, dx \\ &=\; \int p(x)\; f_k(x)\, dx\,. \end{aligned} \tag{3.71}$$

Here $m(x)$ canceled out.

The invariant measure function $m(x)$ is actually the prior density function representing the "lack of relevant information". The prior density $m(x)$ cannot be determined by the principle of maximum entropy. It can be determined by the principle of transformation groups or marginalization theory.

If x takes values only in (a, b), then the maximum entropy probability density function is

$$p(x) \;=\; Z \cdot m(x)\,, \qquad a < x < b\,, \tag{3.72}$$

where Z is a normalization constant.

Chapter 4

Noise Models

In this section we will make a connection between unsupervised and supervised learning in terms of the quality measure. Towards this end we introduce additional noise on the targets, that means we do not know the exact values of the targets. If we know the noise distribution then we can look for the most likely target. Therefore we can apply maximum likelihood to supervised learning. Supervised learning will be treated as an unsupervised maximum likelihood approach using an error model. The kind of measurement noise determines the error model which in turn determines the quality criterion of the supervised approach. Here also classification methods with binary output can be treated.

4.1 Gaussian Noise

We consider the case of Gaussian target noise and a simple linear model:

$$\boldsymbol{s} \;=\; \boldsymbol{X}\,\boldsymbol{w} \tag{4.1}$$

$$\boldsymbol{y} \;=\; \boldsymbol{s} \;+\; \boldsymbol{\epsilon} \;=\; \boldsymbol{X}\,\boldsymbol{w} \;+\; \boldsymbol{\epsilon}\,, \tag{4.2}$$

where $\boldsymbol{s}$ is the true signal, $\boldsymbol{X}$ is the observed data, $\boldsymbol{w}$ is the parameter vector, y is the observed target, and $\boldsymbol{\epsilon}$ is the Gaussian noise vector with zero mean and covariance $\boldsymbol{\Sigma}$. Note, that the covariance $\boldsymbol{\Sigma}$ is the noise distribution for each measurement or observation $\boldsymbol{x}$.

The value $\boldsymbol{y} \;-\; \boldsymbol{X}\,\boldsymbol{w}$ is distributed according to the Gaussian, therefore the likelihood of $(\boldsymbol{y}, \boldsymbol{X})$ is

$$\mathcal{L}((\boldsymbol{y}, \boldsymbol{X}); \boldsymbol{w}) \;= \tag{4.3}$$
$$\frac{1}{(2\,\pi)^{d/2}\,|\boldsymbol{\Sigma}|^{1/2}}\;\exp\left(-\frac{1}{2}\,(\boldsymbol{y} \;-\; \boldsymbol{X}\,\boldsymbol{w})^T\,\boldsymbol{\Sigma}^{-1}(\boldsymbol{y} \;-\; \boldsymbol{X}\,\boldsymbol{w})\right)\,.$$

The log-likelihood is

$$\ln \mathcal{L}((\boldsymbol{y}, \boldsymbol{X}); \boldsymbol{w}) \;= \tag{4.4}$$
$$-\;\frac{d}{2}\ln(2\,\pi) \;-\; \frac{1}{2}\ln|\boldsymbol{\Sigma}| \;-\frac{1}{2}\,(\boldsymbol{y} \;-\; \boldsymbol{X}\,\boldsymbol{w})^T\,\boldsymbol{\Sigma}^{-1}(\boldsymbol{y} \;-\; \boldsymbol{X}\,\boldsymbol{w})\,.$$

To maximize the log-likelihood we have to minimize

$$(\boldsymbol{y} \;-\; \boldsymbol{X}\,\boldsymbol{w})^T\,\boldsymbol{\Sigma}^{-1}(\boldsymbol{y} \;-\; \boldsymbol{X}\,\boldsymbol{w})\,, \tag{4.5}$$

because other terms are independent of $\boldsymbol{w}$.

The minimum of this term, called *least square criterion*, is the *linear least square* estimator.

Multiplying out the criterion gives

$$\begin{aligned}&(\boldsymbol{y} \; - \; \boldsymbol{X} \; \boldsymbol{w})^T \; \boldsymbol{\Sigma}^{-1} (\boldsymbol{y} \; - \; \boldsymbol{X} \; \boldsymbol{w}) \; = \\ &\boldsymbol{y}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{y} \; - \; 2 \boldsymbol{y}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{X} \boldsymbol{w} \; + \; \boldsymbol{w}^T \boldsymbol{X}^T \boldsymbol{\Sigma}^{-1} \; \boldsymbol{X} \; \boldsymbol{w}\end{aligned} \tag{4.6}$$

and the derivative with respect to $\boldsymbol{w}$ is

$$- \; 2 \boldsymbol{X}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{y} \; + \; 2 \boldsymbol{X}^T \boldsymbol{\Sigma}^{-1} \; \boldsymbol{X} \; \boldsymbol{w} \; , \tag{4.7}$$

which are called the Wiener-Hopf equations (correlation between features $\boldsymbol{X}$ and target $\boldsymbol{y}$ should be equal to the correlation between features $\boldsymbol{X}$ and model prediction $\boldsymbol{X} \; \boldsymbol{w}$). Setting the derivative to zero gives the least square estimator

$$\hat{\boldsymbol{w}} \; = \; \left(\boldsymbol{X}^T \boldsymbol{\Sigma}^{-1} \; \boldsymbol{X} \right)^{-1} \boldsymbol{X}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{y} \tag{4.8}$$

The minimal least square criterion is

$$\boldsymbol{y}^T \left(\boldsymbol{\Sigma}^{-1} \; - \; \boldsymbol{\Sigma}^{-1} \boldsymbol{X} \left(\boldsymbol{X}^T \boldsymbol{\Sigma}^{-1} \; \boldsymbol{X} \right)^{-1} \boldsymbol{X}^T \boldsymbol{\Sigma}^{-1} \right) \; \boldsymbol{y} \; . \tag{4.9}$$

In many cases the noise covariance matrix is

$$\boldsymbol{\Sigma}^{-1} \; = \; \frac{1}{\sigma} \; \boldsymbol{I} \; , \tag{4.10}$$

which means that for each observation we have the same noise assumption.

We obtain

$$\hat{\boldsymbol{w}} \; = \; \left(\boldsymbol{X}^T \boldsymbol{X} \right)^{-1} \boldsymbol{X}^T \boldsymbol{y} \; , \tag{4.11}$$

where $\left(\boldsymbol{X}^T \boldsymbol{X} \right)^{-1} \boldsymbol{X}^T$ is known as the pseudo inverse or Moore-Penrose inverse of $\boldsymbol{X}$. The minimal value is

$$\frac{1}{\sigma} \; \boldsymbol{y}^T \left(\boldsymbol{I} \; - \; \boldsymbol{X} \left(\boldsymbol{X}^T \boldsymbol{X} \right)^{-1} \boldsymbol{X}^T \right) \; \boldsymbol{y} \; . \tag{4.12}$$

Note that we can derive the squared error criterion also for other models $\boldsymbol{g}(\boldsymbol{X}; \boldsymbol{w})$ instead of the linear model $\boldsymbol{X} \boldsymbol{w}$. However, in general the estimator must be selected by using an optimization technique which minimizes the least square criterion

$$(\boldsymbol{y} \; - \; \boldsymbol{g}(\boldsymbol{X}; \boldsymbol{w}))^T \; \boldsymbol{\Sigma}^{-1} (\boldsymbol{y} \; - \; \boldsymbol{g}(\boldsymbol{X}; \boldsymbol{w})) \; . \tag{4.13}$$

These considerations are the basis for the least square fit and also for the mean squared error as a risk function. These approaches to loss and risk are derived from maximum likelihood and Gaussian noise assumptions. Therefore the mean squared error in 2.2 can be justified by Gaussian noise assumption.

4.2 Laplace Noise and Minkowski Error

Even if the Gaussian noise assumption is the most widely used noise model, other noise models may be more adequate for certain problems.

For example the loss function

$$\|\boldsymbol{y} \, - \, \boldsymbol{g}(\boldsymbol{X}; \boldsymbol{w})\|_1 \tag{4.14}$$

corresponds to Laplace noise assumption. Or for one dimension

$$|\boldsymbol{y} \, - \, \boldsymbol{g}(\boldsymbol{x}; \boldsymbol{w})| \ . \tag{4.15}$$

For one dimensional output we obtain for the Laplacian noise model

$$p(y \, - \, g(\boldsymbol{x}; \boldsymbol{w})) \, = \, \frac{\beta}{2} \exp\left(- \, \beta \, |y \, - \, g(\boldsymbol{x}; \boldsymbol{w})|\right) \tag{4.16}$$

with loss function

$$|y \, - \, g(\boldsymbol{x}; \boldsymbol{w})| \, . \tag{4.17}$$

For the Minkowski error

$$|y \, - \, g(\boldsymbol{x}; \boldsymbol{w})|^r \tag{4.18}$$

the corresponding noise model is

$$p(y \, - \, g(\boldsymbol{x}; \boldsymbol{w})) \, = \, \frac{r \, \beta^{1/r}}{2 \, \Gamma(1/r)} \exp\left(- \, \beta \, |y \, - \, g(\boldsymbol{x}; \boldsymbol{w})|^r\right) \ , \tag{4.19}$$

where Γ is the gamma function

$$\begin{aligned} \Gamma(a) \, &= \, \int_0^\infty \, u^{a-1} \, e^{-u} \, du \\ \Gamma(n) \, &= \, (n-1)! \, . \end{aligned} \tag{4.20}$$

For $r < 2$ large errors are down-weighted compared to the quadratic error and vice versa for $r > 2$. That means for data with outliers $r < 2$ may be an appropriate choice for the noise model. See examples of error functions in Fig. 4.1.

If random large fluctuations of the output are possible then $r < 2$ should be used in order to give these fluctuations more probability in the noise model and down-weight their influence onto the estimator.

4.3 Binary Models

Above noise considerations do not hold for binary y as used for classification. If the class label is disturbed then y is assigned to the opposite class.

Also for binary y the likelihood approach can be applied.

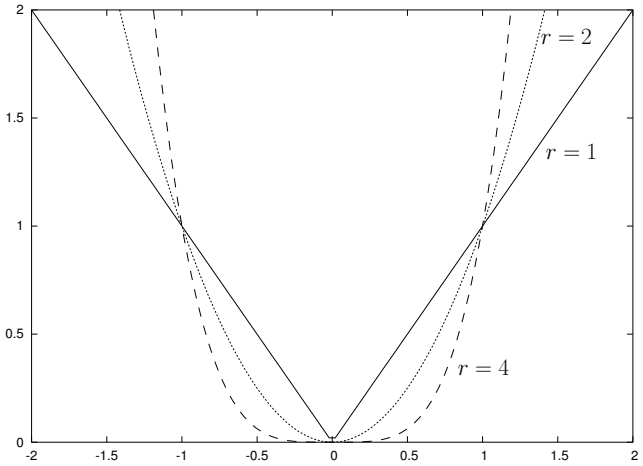

Figure 4.1: Different noise assumptions lead to different Minkowski error functions: $r = 1$ (Laplace noise), $r = 2$ (Gaussian noise), and $r = 4$.

4.3.1 Cross-Entropy

For a classification problem with K classes, we assume that the model output is a probability:

$$g_k(\boldsymbol{x}; \boldsymbol{w}) \;=\; p(\boldsymbol{y} = \boldsymbol{e}_k \mid \boldsymbol{x}) \;. \tag{4.21}$$

and that

$$\boldsymbol{y} \;\in\; \{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_K\} \;. \tag{4.22}$$

If $\boldsymbol{x}$ is in the k-th class then $\boldsymbol{y} \;=\; (0, \ldots, 0, 1, 0, \ldots, 0)$, where the "1" is at position k in the vector $\boldsymbol{y}$.

The likelihood of iid data is

$$\mathcal{L}(\{\boldsymbol{z}\}; \boldsymbol{w}) \;=\; p(\{\boldsymbol{z}\}; \boldsymbol{w}) \;=\; \prod_{i=1}^{l} \prod_{k=1}^{K} p(\boldsymbol{y}^i \;= \boldsymbol{e}_k \mid \boldsymbol{x}^i; \boldsymbol{w})^{\left[\boldsymbol{y}^i\right]_k} p(\boldsymbol{x}^i) \;= \tag{4.23}$$

$$\prod_{i=1}^{l} p(\boldsymbol{x}^i) \prod_{k=1}^{K} p(\boldsymbol{y}^i \;= \boldsymbol{e}_k \mid \boldsymbol{x}^i; \boldsymbol{w})^{\left[\boldsymbol{y}^i\right]_k}$$

because

$$\prod_{k=1}^{K} p(\boldsymbol{y}^i \;= \boldsymbol{e}_k \mid \boldsymbol{x}^i; \boldsymbol{w})^{\left[\boldsymbol{y}^i\right]_k} \;=\; p(\boldsymbol{y}^i \;= \boldsymbol{e}_r \mid \boldsymbol{x}^i; \boldsymbol{w}) \text{ for } \boldsymbol{y}^i \;=\; \boldsymbol{e}_r \;. \tag{4.24}$$

The log-likelihood is

$$\ln \mathcal{L}(\{\boldsymbol{z}\}; \boldsymbol{w}) \;=\; \sum_{k=1}^{K} \sum_{i=1}^{l} \left[\boldsymbol{y}^i\right]_k \ln p(\boldsymbol{y}^i \;= \boldsymbol{e}_k \mid \boldsymbol{x}^i; \boldsymbol{w}) \;+\; \sum_{i=1}^{l} \ln p(\boldsymbol{x}^i) \;. \tag{4.25}$$

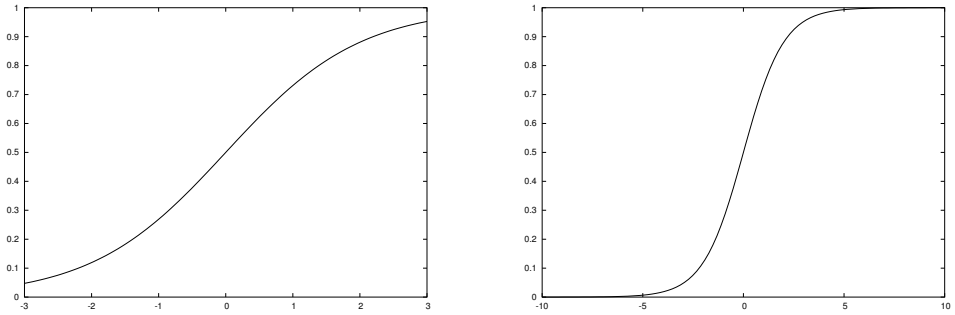

Figure 4.2: The sigmoidal function $\frac{1}{1+\exp(-x)}$.

Therefore the criterion

$$\sum_{k=1}^{K} \sum_{i=1}^{l} \left[\boldsymbol{y}^i\right]_k \ln p(\boldsymbol{y}^i \ = \boldsymbol{e}_k \mid \boldsymbol{x}^i; \boldsymbol{w}) \tag{4.26}$$

is a natural loss function which is called *cross entropy*. Note that $\left[\boldsymbol{y}^i\right]_k$ is the observed probability $p(\boldsymbol{y}^i \ = \boldsymbol{e}_k)$ which is one if $\boldsymbol{y}^i \ = \boldsymbol{e}_k$ and zero otherwise.

Therefore above formula is indeed the cross entropy as defined in eq. (3.48) for discrete distributions.

4.3.2 Logistic Regression

A function g mapping $\boldsymbol{x}$ onto $\mathbb{R}$ can be transformed into a probability by the sigmoidal function

$$\frac{1}{1 \ + \ e^{-\ g(\boldsymbol{x};\boldsymbol{w})}} \tag{4.27}$$

which is depicted in Fig. 4.2.

Note that

$$1 \ - \ \frac{1}{1 \ + \ e^{-\ g(\boldsymbol{x};\boldsymbol{w})}} \ = \ \frac{e^{-\ g(\boldsymbol{x};\boldsymbol{w})}}{1 \ + \ e^{-\ g(\boldsymbol{x};\boldsymbol{w})}} \ . \tag{4.28}$$

We set

$$p(y \ = \ 1 \mid \boldsymbol{x}; \boldsymbol{w}) \ = \ \frac{1}{1 \ + \ e^{-\ g(\boldsymbol{x};\boldsymbol{w})}} \tag{4.29}$$

and

$$p(y \ = \ 0 \mid \boldsymbol{x}; \boldsymbol{w}) \ = \ \frac{e^{-\ g(\boldsymbol{x};\boldsymbol{w})}}{1 \ + \ e^{-\ g(\boldsymbol{x};\boldsymbol{w})}} \ . \tag{4.30}$$

We obtain

$$g(\boldsymbol{x}; \boldsymbol{w}) \ = \ \ln \left(\frac{p(y \ = \ 1 \mid \boldsymbol{x})}{1 \ - \ p(y \ = \ 1 \mid \boldsymbol{x})} \right) \ . \tag{4.31}$$

According to eq. (3.30) the log-likelihood is

$$\ln \mathcal{L}(\{\boldsymbol{z}\}; \boldsymbol{w}) \;=\; \sum_{i=1}^{l} \ln p(\boldsymbol{z}_i; \boldsymbol{w}) \;=\; \sum_{i=1}^{l} \ln p(y^i, \boldsymbol{x}^i; \boldsymbol{w}) \;= \tag{4.32}$$
$$\sum_{i=1}^{l} \ln p(y^i \mid \boldsymbol{x}^i; \boldsymbol{w}) \;+\; \sum_{i=1}^{l} \ln p(\boldsymbol{x}^i) \;.$$

Therefore maximum likelihood maximizes

$$\sum_{i=1}^{l} \ln p(y^i \mid \boldsymbol{x}^i; \boldsymbol{w}) \tag{4.33}$$

Next we will consider the derivative of the log-likelihood. First we will need some algebraic properties:

$$\frac{\partial}{\partial w_j} \ln p(y = 1 \mid \boldsymbol{x}^i; \boldsymbol{w}) \;=\; \frac{\partial}{\partial w_j} \ln \frac{1}{1 \;+\; e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}} \;= \tag{4.34}$$
$$\left(1 \;+\; e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}\right) \left(- \frac{e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}}{\left(1 \;+\; e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}\right)^2} \right) \frac{\partial\, g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j} \;=$$
$$-\; \frac{e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}}{1 \;+\; e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}} \frac{\partial\, g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j} \;=\; -\, p(y \;=\; 0 \mid \boldsymbol{x}^i; \boldsymbol{w}) \frac{\partial\, g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j}$$

and

$$\frac{\partial}{\partial w_j} \ln p(y = 0 \mid \boldsymbol{x}^i; \boldsymbol{w}) \;=\; \frac{\partial}{\partial w_j} \ln \frac{e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}}{1 \;+\; e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}} \;= \tag{4.35}$$
$$\frac{1 \;+\; e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}}{e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}} \left(\frac{e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}}{1 \;+\; e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}} \;-\; \frac{e^{-\; 2\; g(\boldsymbol{x}^i; \boldsymbol{w})}}{\left(1 \;+\; e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}\right)^2} \right) \frac{\partial\, g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j} \;=$$
$$\frac{1}{1 \;+\; e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}} \frac{\partial\, g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j} \;=\; p(y \;=\; 1 \mid \boldsymbol{x}^i; \boldsymbol{w}) \frac{\partial\, g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j}$$

We can rewrite the likelihood as

$$\sum_{i=1}^{l} \ln p(y^i \mid \boldsymbol{x}^i; \boldsymbol{w}) \;= \tag{4.36}$$
$$\sum_{i=1}^{l} y^i \ln p(y = 1 \mid \boldsymbol{x}^i; \boldsymbol{w}) \;+\; \sum_{i=1}^{l} (1 \;-\; y^i) \;\ln p(y = 0 \mid \boldsymbol{x}^i; \boldsymbol{w})$$

which gives for the derivative

$$\begin{aligned}
&\frac{\partial}{\partial w_j} \sum_{i=1}^{l} \ln p(y^i \mid \boldsymbol{x}^i; \boldsymbol{w}) \;= \\
&\sum_{i=1}^{l} y^i \frac{\partial}{\partial w_j} \ln p(y = 1 \mid \boldsymbol{x}^i; \boldsymbol{w}) + \\
&\sum_{i=1}^{l} (1 \;-\; y^i) \; \frac{\partial}{\partial w_j} \ln p(y = 0 \mid \boldsymbol{x}^i; \boldsymbol{w}) \;= \\
&\sum_{i=1}^{l} - \, y^i \; p(y \;=\; 0 \mid \boldsymbol{x}^i; \boldsymbol{w}) \; \frac{\partial \; g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j} + \\
&\sum_{i=1}^{l} (1 \;-\; y^i) \; p(y \;=\; 1 \mid \boldsymbol{x}^i; \boldsymbol{w}) \; \frac{\partial \; g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j} \;= \\
&\sum_{i=1}^{l} \big(- \, y^i \; (1 \;-\; p(y \;=\; 1 \mid \boldsymbol{x}^i; \boldsymbol{w})) \\
&(1 \;-\; y^i) \; p(y \;=\; 1 \mid \boldsymbol{x}^i; \boldsymbol{w}) \big) \; \frac{\partial \; g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j} \;= \\
&\sum_{i=1}^{l} \big(p(y \;=\; 1 \mid \boldsymbol{x}^i; \boldsymbol{w}) \;-\; y^i \big) \; \frac{\partial \; g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j} \,,
\end{aligned} \tag{4.37}$$

where

$$p(y \;=\; 1 \mid \boldsymbol{x}^i; \boldsymbol{w}) \;=\; \frac{1}{1 \;+\; e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}} \tag{4.38}$$

For computing the maximum the derivatives have to be set to zero

$$\forall_j : \;\; \sum_{i=1}^{l} \big(p(y \;=\; 1 \mid \boldsymbol{x}^i; \boldsymbol{w}) \;-\; y^i \big) \; \frac{\partial \; g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j} \;=\; 0 \,. \tag{4.39}$$

Note that the derivatives are products between the prediction error

$$\big(p(y \;=\; 1 \mid \boldsymbol{x}^i; \boldsymbol{w}) \;-\; y^i \big) \tag{4.40}$$

and the derivatives of the function g.

This is very similar to the derivative of the quadratic loss function in the regression case, where we would have
$\big(g(\boldsymbol{x}^i; \boldsymbol{w}) \;-\; y^i \big)$ instead of $\big(p(y \;=\; 1 \mid \boldsymbol{x}^i; \boldsymbol{w}) \;-\; y^i \big)$.

If a neural network h with a sigmoid output unit, the mean squared error as objective function, and the class labels as target is trained, then the derivative is

$$\begin{aligned}
&\forall_j : \;\; \sum_{i=1}^{l} \big(h(\boldsymbol{x}^i; \boldsymbol{w}) \;-\; y^i \big) \; \frac{\partial \; h(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j} \;= \\
&\big(p(y \;=\; 1 \mid \boldsymbol{x}^i; \boldsymbol{w}) \;-\; y^i \big) \; h(\boldsymbol{x}^i; \boldsymbol{w}) \, (1 \;-\; h(\boldsymbol{x}^i; \boldsymbol{w})) \; \frac{\partial \; g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j} \,,
\end{aligned} \tag{4.41}$$

where

$$h(\boldsymbol{x}^i; \boldsymbol{w}) \;=\; p(y \;=\; 1 \mid \boldsymbol{x}^i; \boldsymbol{w}) \;=\; \frac{1}{1 \;+\; e^{-\; g(\boldsymbol{x}^i; \boldsymbol{w})}} \tag{4.42}$$

$$\frac{\partial\, h(\boldsymbol{x}^i; \boldsymbol{w})}{\partial g(\boldsymbol{x}^i; \boldsymbol{w})} \;=\; h(\boldsymbol{x}^i; \boldsymbol{w})\; (1 \;-\; h(\boldsymbol{x}^i; \boldsymbol{w}))\;. \tag{4.43}$$

Therefore the gradient for logistic regression and neural networks differs only in the factor $h(\boldsymbol{x}^i; \boldsymbol{w})\; (1 \;-\; h(\boldsymbol{x}^i; \boldsymbol{w}))$. The effect of the factor is that the neural network does not push the output towards 1 or 0.

Alternative formulation with $y \in \{+1, -1\}$

We now give an alternative formulation of logistic regression with $y \in \{+1, -1\}$.

We remember

$$p(y \;=\; 1 \mid \boldsymbol{x}; \boldsymbol{w}) \;=\; \frac{1}{1 \;+\; e^{-\; g(\boldsymbol{x}; \boldsymbol{w})}} \tag{4.44}$$

and

$$p(y \;=\; -1 \mid \boldsymbol{x}; \boldsymbol{w}) \;=\; \frac{e^{-\; g(\boldsymbol{x}; \boldsymbol{w})}}{1 \;+\; e^{-\; g(\boldsymbol{x}; \boldsymbol{w})}} \;=\; \frac{1}{1 \;+\; e^{\; g(\boldsymbol{x}; \boldsymbol{w})}}\;. \tag{4.45}$$

Therefore we have

$$-\ln p(y \;=\; y^i \mid \boldsymbol{x}^i; \boldsymbol{w}) \;=\; \ln \left(1 \;+\; e^{-\; y^i\; g(\boldsymbol{x}^i; \boldsymbol{w})}\right) \tag{4.46}$$

and the objective which is minimized to find the maximum likelihood solution is

$$L \;=\; -\; \sum_{i=1}^{l} \ln p(y^i \mid \boldsymbol{x}^i; \boldsymbol{w}) \;=\; \sum_{i=1}^{l} \ln \left(1 \;+\; e^{-\; y^i\; g(\boldsymbol{x}^i; \boldsymbol{w})}\right) \tag{4.47}$$

The derivatives of the objective with respect to the parameters are

$$\frac{\partial L}{\partial w_j} \;=\; -\; \sum_{i=1}^{l} y^i\; \frac{\partial\; g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j}\; \frac{e^{-\; y^i\; g(\boldsymbol{x}^i; \boldsymbol{w})}}{1 \;+\; e^{-\; y^i\; g(\boldsymbol{x}^i; \boldsymbol{w})}} \;= \tag{4.48}$$

$$-\; \sum_{i=1}^{l} y^i\; \frac{\partial\; g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial w_j}\; \left(1 \;-\; p(y^i \mid \boldsymbol{x}; \boldsymbol{w})\right)\;.$$

The last equation is similar to eq. (4.37).

In matrix notation we have

$$\frac{\partial L}{\partial \boldsymbol{w}} \;=\; -\; \sum_{i=1}^{l} y^i\; \left(1 \;-\; p(y^i \mid \boldsymbol{x}; \boldsymbol{w})\right)\; \frac{\partial\; g(\boldsymbol{x}^i; \boldsymbol{w})}{\partial \boldsymbol{w}}\;. \tag{4.49}$$

4.3.3 (Regularized) Linear Logistic Regression is Strictly Convex

Following Jason D. M. Rennie, we show that linear Logistic Regression is strictly convex.

In the linear case we have

$$g(\boldsymbol{x}^i; \boldsymbol{w}) \;=\; \boldsymbol{w}^T \boldsymbol{x}^i \;. \tag{4.50}$$

For labels $y \in \{+1, -1\}$ we have

$$\frac{\partial L}{\partial w_j} \;=\; -\sum_{i=1}^{l} y^i x_{ij} \;\left(1 \;-\; p(y^i \mid \boldsymbol{x}; \boldsymbol{w})\right) \;. \tag{4.51}$$

The second order derivatives of the objective L that is minimized are

$$H_{jk} \;=\; \frac{\partial L}{\partial w_j \, \partial w_k} \;=\; \sum_{i=1}^{l} \left(y^i\right)^2 \; x_{ij} \; x_{ik} \; p(y^i \mid \boldsymbol{x}; \boldsymbol{w}) \; \left(1 \;-\; p(y^i \mid \boldsymbol{x}; \boldsymbol{w})\right) \;, \tag{4.52}$$

where $\boldsymbol{H}$ is the Hessian.

Since $p(1-p) \geq 0$ for $p \leq 1$, we can define

$$\rho_{ij} \;=\; x_{ij} \; \sqrt{p(y^i \mid \boldsymbol{x}; \boldsymbol{w}) \; (1 \;-\; p(y^i \mid \boldsymbol{x}; \boldsymbol{w}))} \,. \tag{4.53}$$

The bilinear form of the Hessian with a vector $\boldsymbol{a}$ is

$$\begin{aligned}
&\boldsymbol{a}^T \; \boldsymbol{H} \; \boldsymbol{a} \;=\; \sum_{i=1}^{l} \sum_{j=1}^{d} \sum_{k=1}^{d} x_{ij} \; x_{ik} \; a_j \; a_k \; p(y^i \mid \boldsymbol{x}; \boldsymbol{w}) \; \left(1 \;-\; p(y^i \mid \boldsymbol{x}; \boldsymbol{w})\right) \;= \\
&\sum_{i=1}^{l} \sum_{j=1}^{d} \; a_j \; x_{ij} \; \sqrt{p(y^i \mid \boldsymbol{x}; \boldsymbol{w}) \; (1 \;-\; p(y^i \mid \boldsymbol{x}; \boldsymbol{w}))} \\
&\sum_{k=1}^{d} \; a_k \;\; x_{ik} \; \sqrt{p(y^i \mid \boldsymbol{x}; \boldsymbol{w}) \; (1 \;-\; p(y^i \mid \boldsymbol{x}; \boldsymbol{w}))} \;= \\
&\sum_{i=1}^{l} \left(\boldsymbol{a}^T \; \boldsymbol{\rho}_i\right) \left(\boldsymbol{a}^T \; \boldsymbol{\rho}_i\right) \;=\; \sum_{i=1}^{l} \left(\boldsymbol{a}^T \; \boldsymbol{\rho}_i\right)^2 \;\geq\; 0 \,.
\end{aligned} \tag{4.54}$$

Since we did not make any restriction on $\boldsymbol{a}$, the Hessian is positive definite.

Adding a term like $\frac{1}{2} \; \boldsymbol{w}^T \; \boldsymbol{w}$ to the objective for regularization, then the Hessian of the objective is strict positive definite.

4.3.4 Softmax

For multi-class problems logistic regression can be generalized by Softmax.

We assume K classes with $y \in \{1, \ldots, K\}$ and the probability of $\boldsymbol{x}$ belonging to class k is

$$p(y \;=\; k \mid \boldsymbol{x}; g_1, \ldots, g_K, \boldsymbol{w}_1, \ldots, \boldsymbol{w}_K) \;=\; \frac{e^{\;g_k(\boldsymbol{x};\boldsymbol{w}_k)}}{\sum_{j=1}^{K} e^{\;g_j(\boldsymbol{x};\boldsymbol{w}_j)}} \tag{4.55}$$

which gives a multinomial distribution across the classes.

The objective which is minimized in order to maximize the likelihood is

$$L \;=\; -\sum_{i=1}^{l} \ln p(y = y^i \mid \boldsymbol{x}^i; \boldsymbol{w}) \;=\; \sum_{i=1}^{l} \ln \left(\sum_{j=1}^{K} e^{g_j(\boldsymbol{x};\boldsymbol{w}_j)} \right) \;-\; g_{y^i}(\boldsymbol{x}; \boldsymbol{w}_{y^i}) \;. \tag{4.56}$$

In the following we set

$$p(y \;=\; k \mid \boldsymbol{x}; g_1, \ldots, g_K, \boldsymbol{w}_1, \ldots, \boldsymbol{w}_K) \;=\; p(k \mid \boldsymbol{x}; \boldsymbol{W}) \;. \tag{4.57}$$

The derivatives are

$$\frac{\partial L}{\partial w_{kn}} \;=\; \sum_{i=1}^{l} \frac{\partial\; g_k(\boldsymbol{x}^i; \boldsymbol{w}_k)}{\partial w_{kn}} p(k \mid \boldsymbol{x}^i; \boldsymbol{W}) \;-\; \delta_{y^i=k} \sum_{i=1}^{l} \frac{\partial\; g_k(\boldsymbol{x}^i; \boldsymbol{w}_k)}{\partial w_{kn}} \;. \tag{4.58}$$

4.3.5 (Regularized) Linear Softmax is Strictly Convex

Following Jason D. M. Rennie, we show that linear Softmax is strictly convex.

In the linear case we have

$$g_k(\boldsymbol{x}^i; \boldsymbol{w}_k) \;=\; \boldsymbol{w}_k^T \boldsymbol{x}^i \tag{4.59}$$

or in vector notation

$$\boldsymbol{g}(\boldsymbol{x}^i; \boldsymbol{W}) \;=\; \boldsymbol{W}^T \boldsymbol{x}^i \;. \tag{4.60}$$

The derivatives are

$$\frac{\partial L}{\partial w_{kn}} \;=\; \sum_{i=1}^{l} x_{in} p(k \mid \boldsymbol{x}^i; \boldsymbol{W}) \;-\; \delta_{y^i=k} \sum_{i=1}^{l} x_{in} \;. \tag{4.61}$$

To compute the second order derivatives of the objective, we need the derivatives of the probabilities with respect to the parameters:

$$\begin{aligned} \frac{\partial p(v \mid \boldsymbol{x}^i; \boldsymbol{W})}{\partial w_{vm}} \;&=\; x_{im}\, p(k \mid \boldsymbol{x}^i; \boldsymbol{W}) \;\big(1 \;-\; p(k \mid \boldsymbol{x}^i; \boldsymbol{W})\big) \\ \frac{\partial p(k \mid \boldsymbol{x}^i; \boldsymbol{W})}{\partial w_{vm}} \;&=\; x_{im}\, p(k \mid \boldsymbol{x}^i; \boldsymbol{W})\, p(v \mid \boldsymbol{x}^i; \boldsymbol{W})) \;. \end{aligned} \tag{4.62}$$

The second order derivatives of L with respect to the components of the parameter vector $\boldsymbol{w}$ are

$$H_{kn,vm} = \frac{\partial L}{\partial w_{kn} \partial w_{vm}} = \tag{4.63}$$
$$\sum_{i=1}^{l} x_{in}\, x_{im}\, p(k \mid \boldsymbol{x}^i; \boldsymbol{W})\, \left(\delta_{k=v}\, (1 - p(k \mid \boldsymbol{x}^i; \boldsymbol{W})) - \right.$$
$$\left. (1 - \delta_{k=v}) p(v \mid \boldsymbol{x}^i; \boldsymbol{W})\right) .$$

Again we define a vector $\boldsymbol{a}$ with components a_{uj} (note, the double index is considered as single index so that a matrix is written as vector).

We consider the bilinear from

$$\boldsymbol{a}^T \boldsymbol{H}\, \boldsymbol{a} = \tag{4.64}$$
$$\sum_{k,n} \sum_{v,m} \sum_{i} a_{kn}\, a_{vm}\, x_{in}\, x_{im}\, p(k \mid \boldsymbol{x}^i; \boldsymbol{W})\, \left(\delta_{k=v}\, (1 - p(k \mid \boldsymbol{x}^i; \boldsymbol{W})) - \right.$$
$$\left. (1 - \delta_{k=v})\, p(v \mid \boldsymbol{x}^i; \boldsymbol{W})\right) =$$
$$\sum_{k,n} \sum_{i} a_{kn}\, x_{in}\, p(k \mid \boldsymbol{x}^i; \boldsymbol{W}) \sum_{m} x_{im} \left(a_{km} - \sum_{v} a_{vm} p(v \mid \boldsymbol{x}^i; \boldsymbol{W})\right) =$$
$$\sum_{i} \sum_{n} x_{in} \sum_{k} a_{kn}\, p(k \mid \boldsymbol{x}^i; \boldsymbol{W}) \sum_{m} x_{im} \left(a_{km} - \sum_{v} a_{vm} p(v \mid \boldsymbol{x}^i; \boldsymbol{W})\right) =$$
$$\sum_{i} - \left\{ \left(\sum_{n} x_{in} \sum_{k} a_{kn}\, p(k \mid \boldsymbol{x}^i; \boldsymbol{W})\right) \left(\sum_{m} x_{im} \sum_{v} a_{vm} p(v \mid \boldsymbol{x}^i; \boldsymbol{W})\right) \right\} +$$
$$\left\{ \sum_{n} x_{in} \sum_{k} a_{kn}\, p(k \mid \boldsymbol{x}^i; \boldsymbol{W}) \sum_{m} x_{im}\, a_{km} \right\} =$$
$$\sum_{i} - \left\{ \left(\sum_{n} x_{in} \sum_{k} a_{kn}\, p(k \mid \boldsymbol{x}^i; \boldsymbol{W})\right)^2 \right\} +$$
$$\left\{ \sum_{k} p(k \mid \boldsymbol{x}^i; \boldsymbol{W}) \left(\sum_{n} x_{in}\, a_{kn}\right) \left(\sum_{m} x_{im}\, a_{km}\right) \right\} =$$
$$\sum_{i} - \left\{ \left(\sum_{n} x_{in} \sum_{k} a_{kn}\, p(k \mid \boldsymbol{x}^i; \boldsymbol{W})\right)^2 \right\} +$$
$$\left\{ \sum_{k} p(k \mid \boldsymbol{x}^i; \boldsymbol{W}) \left(\sum_{n} x_{in}\, a_{kn}\right)^2 \right\} .$$

If for each summand of the sum over i

$$\sum_{k} p(k \mid \boldsymbol{x}^i; \boldsymbol{W}) \left(\sum_{n} x_{in}\, a_{kn}\right)^2 - \left(\sum_{k} p(k \mid \boldsymbol{x}^i; \boldsymbol{W}) \sum_{n} x_{in}\, a_{kn}\right)^2 \tag{4.65}$$
$$\geq 0$$

holds, then the Hessian $\boldsymbol{H}$ is positive semidefinite. This holds for arbitrary number of samples as each term corresponds to a sample.

In the last inequality the $p(k \mid \boldsymbol{x}^i; \boldsymbol{W})$ can be viewed as a multinomial distribution over k. The terms $\sum_n x_{in}\, a_{kn}$ can be viewed as functions depending on k.

In this case $\sum_k p(k \mid \boldsymbol{x}^i; \boldsymbol{W}) \left(\sum_n x_{in}\, a_{kn}\right)^2$ is the second moment and the squared expectation is $\left(\sum_k\ p(k \mid \boldsymbol{x}^i; \boldsymbol{W}) \sum_n x_{in}\, a_{kn}\right)^2$. Therefore the left hand side of inequality (4.65) is the second central moment, which is larger than zero.

Alternatively inequality (4.65) can be proven by applying Jensen's inequality with the square function as a convex function.

We have proven that the Hessian $\boldsymbol{H}$ is positive semidefinite.

Adding a term like $\frac{1}{2}\ \sum_k \boldsymbol{w}_k^T\ \boldsymbol{w}_k$ to the objective for regularization, then the Hessian of the objective is strictly positive definite.

Chapter 5

Statistical Learning Theory

A central question in machine learning is: Does learning from examples help in the future? Obviously, learning helps humans to master the environment they live in. But what is the mathematical reason for that? It might be that tasks in the future are unique and nothing from the past helps to solve them. Future examples may be different from examples we have already seen.

Learning on the training data is called "empirical risk minimization" (ERM) in statistical learning theory. ERM results that if the complexity is restricted and the dynamics of the environment does not change, learning helps. "Learning helps" means that with increasing number of training examples the selected model converges to the best model for all future data. Under mild conditions the convergence is uniform and even fast, i.e. exponentially. These theoretical theorems found the idea of learning from data because with finite many training examples a model can be selected which is close to the optimal model for future data. How close is governed by the number of training examples, the complexity of the task including noise, the complexity of the model, and the model class.

To measure the complexity of the model we will introduce the VC-dimension (Vapnik-Chervonenkis).

Using model complexity and the model quality on the training set, theoretical bounds on the generalization error, i.e. the performance on future data, will be derived. From these bounds the principle of "structural risk minimization" will be derived to optimize the generalization error through training.

In this section we address the question whether learning from training data that means selecting a model based on training examples is useful for processing future data. Is a model which explains the training data an appropriate model of the world, i.e. also explains new data?

We will see that how useful a model selected based on training data is is determined by its complexity. We will introduce the VC-dimension as complexity measure.

A main issue in statistical learning theory is to derive error bounds for the generalization error. The error bound is expressed as the probability of reaching a certain error rate on future data if the model is selected according to the training data.

Finally from the error bounds it will be seen that model complexity and training data mismatch of the model must be simultaneously minimized. This principle is called "structural risk minimization".

This statistical learning theory is based on two simple principles (1) the uniform law of large

numbers (for inductive interference, i.e. the empirical risk minimization) and (2) complexity constrained models (structural risk minimization).

A first theoretical error bound on the mean squared error was given as the bias-variance formulation in eq. (3.5). The bias term corresponds to training data mismatch of the model whereas the variance term corresponds to model complexity. Higher model complexity leads to more models which fit the training data equally good, therefore the variance is larger. However the bias-variance formulation was derived for the special case of mean squared error. We will generalize this formulation in this section. Also the variance term will be expressed as model complexity for which measurements are available.

First we will start with some examples of error bounds.

5.1 Error Bounds for a Gaussian Classification Task

We revisit the Gaussian classification task from Section 2.3.

The minimal risk is given in eq. (2.30) as

$$R_{\min} \;=\; \int_X \min\{p\left(\boldsymbol{x}, y=-1\right), p\left(\boldsymbol{x}, y=1\right)\}\, d\boldsymbol{x}\;, \tag{5.1}$$

which can be written as

$$\begin{aligned} &R_{\min} \;= \\ &\int_X \min\{p\left(\boldsymbol{x} \mid y=-1\right)\; p(y=-1)\;,\; p\left(\boldsymbol{x} \mid y=1\right)\; p(y=1)\}\, d\boldsymbol{x}\;. \end{aligned} \tag{5.2}$$

For transforming the minimum into a continuous function we will use the inequality

$$\forall_{a,b>0}:\; \forall_{0\leq\beta\leq 1}:\;\; \min\{a,b\} \;\leq\; a^{\beta}\, b^{1-\beta}\;. \tag{5.3}$$

To proof this inequality, without loss of generality we assume $a \geq b$ and have to show that $b \;\leq\; a^{\beta}\, b^{1-\beta}$. This is equivalent to $b \;\leq\; (a/b)^{\beta}\, b$ which is valid because $(a/b)^{\beta} \;\geq\; 1$.

Now we can bound the error by

$$\begin{aligned} &\forall_{0\leq\beta\leq 1}:\;\; R_{\min} \;\leq\; \left(p(y=1)\right)^{\beta}\; \left(p(y=-1)\right)^{1-\beta} \\ &\int_X \left(p\left(\boldsymbol{x} \mid y=1\right)\right)^{\beta}\; \left(p\left(\boldsymbol{x} \mid y=-1\right)\right)^{1-\beta}\; d\boldsymbol{x}\;. \end{aligned} \tag{5.4}$$

Up to now we only assumed a two class problem and did not make use of the Gaussian assumption.

The Gaussian assumption allows to evaluate above integral analytically:

$$\int_X \left(p\left(\boldsymbol{x} \mid y=1\right)\right)^{\beta}\; \left(p\left(\boldsymbol{x} \mid y=-1\right)\right)^{1-\beta}\; d\boldsymbol{x} \;=\; \exp(-\,v(\beta))\;, \tag{5.5}$$

where

$$\begin{aligned} v(\beta) \;&=\; \frac{\beta(1-\beta)}{2} \\ &\left(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1\right)^T\; \left(\beta\,\boldsymbol{\Sigma}_1 \;+\; (1-\beta)\,\boldsymbol{\Sigma}_2\right)^{-1} \left(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1\right) \\ &+\; \frac{1}{2} \ln \frac{\left|\beta\,\boldsymbol{\Sigma}_1 \;+\; (1-\beta)\,\boldsymbol{\Sigma}_2\right|}{\left|\boldsymbol{\Sigma}_1\right|^{\beta}\; \left|\boldsymbol{\Sigma}_2\right|^{1-\beta}}\;. \end{aligned} \tag{5.6}$$

The *Chernoff bound* is obtained by maximizing $v(\beta)$ with respect to β and substituting this β into eq. (5.4).

The optimization has to be done in a one dimensional space which is very efficient.

The bound obtained by setting $\beta = \frac{1}{2}$ is called the *Bhattacharyya bound*:

$$v(1/2) \;=\; \frac{1}{4}\,(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^T\;(\boldsymbol{\Sigma}_1 \;+\; \boldsymbol{\Sigma}_2)^{-1}\,(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1) \tag{5.7}$$
$$+\;\frac{1}{2}\ln\frac{\left|\frac{\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2}{2}\right|}{\sqrt{|\boldsymbol{\Sigma}_1|\;|\boldsymbol{\Sigma}_2|}}\;.$$

5.2 Empirical Risk Minimization

The empirical risk minimization principle states that if the training set is explained by the model then the model generalizes to future examples.

In the following considerations we need to restrict the complexity of the model class in order to obtain statements about the empirical error.

empirical risk minimization: minimize training error

5.2.1 Complexity: Finite Number of Functions

In this subsection we give an intuition why complexity matters. We restrict the definition of the complexity to the number M of functions from which the model can be selected.

First we are interested on the difference between the training error, *the empirical risk*, and the test error, *the risk*.

In eq. (2.2) the risk has been defined as

$$R(g) \;=\; \mathrm{E}_{\boldsymbol{z}}\left(L(y, g(\boldsymbol{x}))\right)\;. \tag{5.8}$$

We define the empirical risk R_{emp} analog to the cross-validation risk in eq. (2.14) as

$$R_{\mathrm{emp}}(g, \boldsymbol{Z}) \;=\; \frac{1}{l}\sum_{i=1}^{l} L\left(y^i, g\left(\boldsymbol{x}^i\right)\right)\;. \tag{5.9}$$

We will write $R_{\mathrm{emp}}(g, l)$ instead of $R_{\mathrm{emp}}(g, \boldsymbol{Z})$ to indicate the size of the training set.

We assume that we chose our model g from a finite set of functions

$$\{g_1, \ldots, g_M\}\;. \tag{5.10}$$

The difference between the empirical risk R_{emp} and the risk R can be different for each of the functions g_i. A priori we do not know which function g_i will be selected by the training procedure, therefore we will consider the worst case that is the maximal distance between R_{emp} and R on the set of functions:

$$\max_{j=1,\ldots,M} \|R_{\mathrm{emp}}(g_j, l) \;-\; R(g_j)\|\;. \tag{5.11}$$

We now consider the probability that the difference is large than ϵ:

$$
\begin{aligned}
& p\left(\max_{j=1,\ldots,M} \|R_{\text{emp}}(g_j, l) \; - \; R(g_j)\| \; > \; \epsilon\right) \; \leq \\
& \sum_{j=1}^{M} p\left(\|R_{\text{emp}}(g_j, l) \; - \; R(g_j)\| \; > \; \epsilon\right) \; \leq \\
& M \; 2 \; \exp\left(-\, 2\, \epsilon^2 \; l\right) \; = \; 2 \; \exp\left(\left(\frac{\ln M}{l} \; - \; 2\, \epsilon^2\right) \; l\right) \; = \; \delta \, ,
\end{aligned}
\tag{5.12}
$$

where the first inequality "$\leq$" comes from the fact that $p(a \text{ OR } b) \; \leq \; p(a) + p(b)$ (this is called the "union bound") and the second inequality "$\leq$" is as special case of Hoeffding's inequality for each g_j if R are within $[0, 1]$. Hoeffding's inequality bounds the difference between empirical mean (the average) and the expectation. For R are within $[0, a]$ the bound would be $\exp\left(-\, 2\, \epsilon^2 \; (1/a^2) \; l\right)$ instead of $\exp\left(-\, 2\, \epsilon^2 \; l\right)$. The one-sided Hoeffding's inequality is

$$
p(\mu_l \; - \; s \; > \; \epsilon) \; < \; \exp\left(-\, 2\, \epsilon^2 \; l\right) \; , \tag{5.13}
$$

where μ_l is the empirical mean of the true value s for l trials.

Above last equation is valid for a two-sided bound. For a one-sided bound, we obtain

$$
\epsilon(l, M, \delta) \; = \; \sqrt{\frac{\ln M \; - \; \ln(\delta)}{2\, l}} \; . \tag{5.14}
$$

The value $\epsilon(l, M, \delta)$ is a complexity term depending on the number l of training examples, the number of possible functions M, and the confidence $(1 - \delta)$.

Theorem 5.1 (Finite Set Error Bound)
With probability of at least $(1 \; - \; \delta)$ *over possible training sets with* l *elements and for* M *possible functions we have*

$$
R(g) \; \leq \; R_{\text{emp}}(g, l) \; + \; \epsilon(l, M, \delta) \; . \tag{5.15}
$$

Fig. 5.1 shows the relation between the test error $R(g)$ and the training error as a function of the complexity. The test error R first decreases and then increases with increasing complexity. The training error decreases with increasing complexity. The test error R is the sum of training error and a complexity term. At some complexity point the training error decreases slower than the complexity term increases – this is the point of the optimal test error.

In order that $\epsilon(l, M, \delta)$ converges to zero with increasing l we must assure that

$$
\frac{\ln M}{l} \; \overset{l \to \infty}{\to} \; 0 \; . \tag{5.16}
$$

Because M is finite this expression is true.

However in most machine learning applications models are chosen from an infinite set of functions. Therefore we need another measure for the complexity instead the measure based on M, the number of functions.

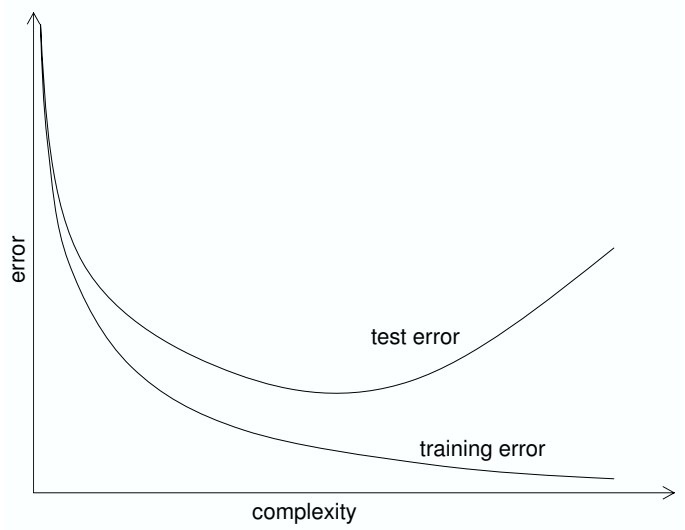

Figure 5.1: Typical example where the test error first decreases and then increases with increasing complexity. The training error decreases with increasing complexity. The test error, the risk, is the sum of training error and a complexity term. At some complexity point the training error decreases slower than the complexity term increases – this is the point of the optimal test error.

5.2.2 Complexity: VC-Dimension

The main idea in this subsection is that on a given training set only a finite number of functions can be distinguished from one another. For example in a classification task all discriminant functions g which lead to the same classification function sign$g(.)$ build one equivalence class.

We will again use parametric models $g(.; \boldsymbol{w})$ with parameter vector $\boldsymbol{w}$.

We first want to answer the following question. Does minimizing the empirical risk with respect to parameter vector (e.g. minimizing the training error) converge to the best solution with increasing training set size, i.e. do we select better models with larger training sets? This question asks whether the empirical risk minimization (ERM) is consistent or not.

We first have to define the parameter $\hat{\boldsymbol{w}}_l$ which minimizes the empirical risk as

$$\hat{\boldsymbol{w}}_l \;=\; \arg\min_{\boldsymbol{w}} R_{\text{emp}}(g(.; \boldsymbol{w}), l) \;. \tag{5.17}$$

The ERM is *consistent* if

$$R(g(.; \hat{\boldsymbol{w}}_l)) \;\stackrel{l\to\infty}{\to}\; \inf_{\boldsymbol{w}} \; R(g(.; \boldsymbol{w})) \tag{5.18}$$

$$R_{\text{emp}}(g(.; \hat{\boldsymbol{w}}_l), l) \;\stackrel{l\to\infty}{\to}\; \inf_{\boldsymbol{w}} \; R(g(.; \boldsymbol{w})) \tag{5.19}$$

hold, where the convergence is in probability.

The ERM is consistent if it generates sequences of $\hat{\boldsymbol{w}}_l,\; l = 1, 2, \ldots$, for which both the risk evaluated with the function parameterized by $\hat{\boldsymbol{w}}_l$ and the empirical risk evaluated with the same function converge in probability to the minimal possible risk given the parameterized functions. The consistency is depicted in Fig. 5.2.

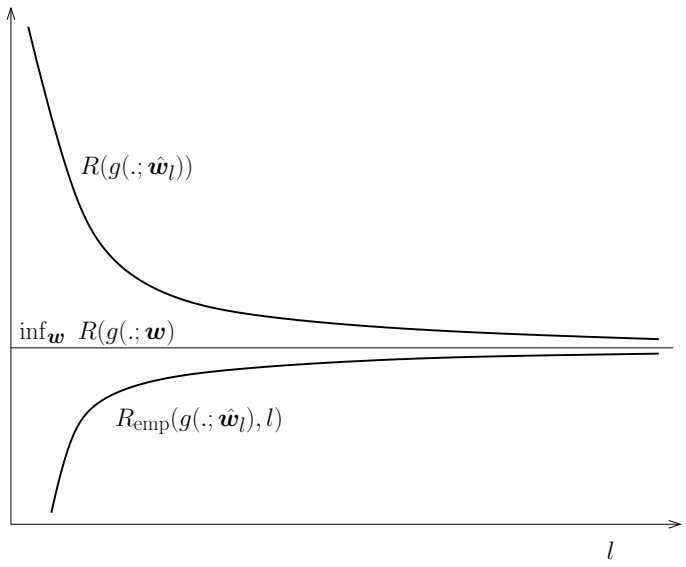

Figure 5.2: The consistency of the empirical risk minimization is depicted. The risk $R(g(.;\hat{\boldsymbol{w}}_l))$ of the optimal training parameter $\hat{\boldsymbol{w}}_l$ and the empirical risk $R_{\text{emp}}(g(.;\hat{\boldsymbol{w}}_l), l)$ for the optimal training parameter $\hat{\boldsymbol{w}}_l$ converge to the minimal possible risk $\inf_{\boldsymbol{w}}\ R(g(.;\boldsymbol{w}))$.

The ERM is *strictly consistent* if for all

$$\Lambda(c) \ = \ \left\{ \boldsymbol{w} \mid \boldsymbol{z} = (\boldsymbol{x}, y),\ \int L(y, g(\boldsymbol{x};\boldsymbol{w}))p(\boldsymbol{z})d\boldsymbol{z} \ \geq \ c \right\} \tag{5.20}$$

the convergence

$$\inf_{\boldsymbol{w}\in\Lambda(c)} \ R_{\text{emp}}(g(.;\boldsymbol{w}), l) \ \stackrel{l\to\infty}{\to} \ \inf_{\boldsymbol{w}\in\Lambda(c)} \ R(g(.;\boldsymbol{w})) \tag{5.21}$$

holds, where the convergence is in probability.

The convergence holds for all subsets of functions where the functions with risk smaller that c are removed.

In the following we only focus on strict consistency and mean "strictly consistent" if we write "consistent".

The maximum likelihood method for a set of densities with $0 \ < \ a \ \leq \ p(\boldsymbol{x};\boldsymbol{w}) \ \leq \ A \ < \ \infty$ is (strictly) consistent if

$$\begin{aligned} &\exists_{\boldsymbol{w}_1} : \\ &\inf_{\boldsymbol{w}} \ \frac{1}{l} \ \sum_{i=1}^{l} \ (-\ \ln p(\boldsymbol{x}_i;\boldsymbol{w})) \ \stackrel{l\to\infty}{\to} \ \inf_{\boldsymbol{w}} \ \int_X (-\ \ln p(\boldsymbol{x};\boldsymbol{w}))\ p(\boldsymbol{x};\boldsymbol{w}_1)\ d\boldsymbol{x}\ . \end{aligned} \tag{5.22}$$

If above convergence takes place for just one specific density $p(\boldsymbol{x};\boldsymbol{w}_1)$ then maximum likelihood is consistent and convergence occurs for all densities of the set.

Under what conditions is the ERM consistent?

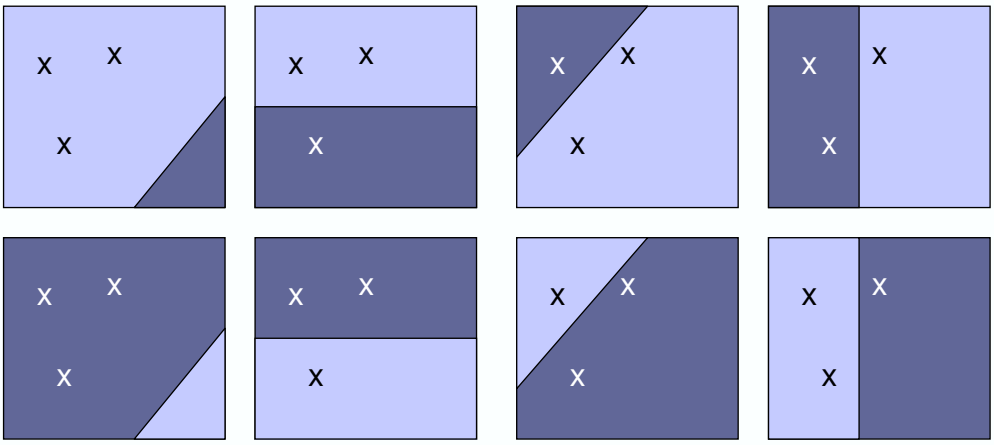

Figure 5.3: Linear decision boundaries can shatter any 3 points in a 2-dimensional space. Black crosses are assigned to -1 and white to +1.

In order to express these conditions we have to introduce new capacity measures: number of points to be shattered, the entropy, the annealed entropy, the growth function, and finally the VC-dimension.

For the complexity measure we first restrict ourselves to classification. Regression can be approximated through classification by dividing the output range into intervals of length ϵ and defining for each interval a class.

How many possibilities exist to label the input data $\boldsymbol{x}^i$ by binary labels $y^i \in \{-1, 1\}$? Clearly each binary vector of length l represents a labeling, therefore we obtain 2^l labelings.

Is our model class complex enough to produce any labeling vector based on the inputs? Not all model classes can do that. Therefore we can define as a complexity measure the number of data points a model class can assign all binary vectors. Assigning all possible binary vectors is called *shattering* the points. Fig. 5.3 depicts the shattering of 3 points in 2-dimensional space. Fig. 5.4 shows a specific labeling of 4 points in a 2-dimensional space which cannot be represented by a linear function. The complexity of linear functions in a 2-dimensional space is that they can shatter 3 points.

The maximum number of points a function class can shatter will be introduced as the *VC-dimension*.

However we will do it more formally.

The *shattering coefficient* of a function class $\mathcal{F}$ with respect to inputs x^i, $1 \leq i \leq l$ is the cardinality of $\mathcal{F}$ if restricted to the l input vectors x^i, $1 \leq i \leq l$ (on input vectors distinguishable functions in $\mathcal{F}$). The shattering coefficient is denoted by

$$N_{\mathcal{F}}(\boldsymbol{x}^1, \ldots, \boldsymbol{x}^l) \; . \tag{5.23}$$

The *entropy of a function class* is

$$H_{\mathcal{F}}(l) \; = \; \mathrm{E}_{(\boldsymbol{x}^1, \ldots, \boldsymbol{x}^l)} \; \ln N_{\mathcal{F}}(\boldsymbol{x}^1, \ldots, \boldsymbol{x}^l) \; . \tag{5.24}$$

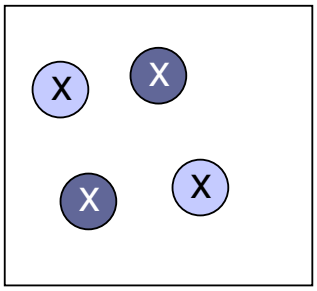

Figure 5.4: Linear decision boundaries *cannot* shatter any 4 points in a 2-dimensional space. Black crosses are assigned to -1 and white to +1, this label assignment cannot be represented by a linear function.

The *annealed entropy of a function class* is

$$H_{\mathcal{F}}^{\text{ann}}(l) \;=\; \ln \mathrm{E}_{(\boldsymbol{x}^1,\ldots,\boldsymbol{x}^l)}\, N_{\mathcal{F}}(\boldsymbol{x}^1,\ldots,\boldsymbol{x}^l)\;. \tag{5.25}$$

Until now we defined entropies, which are based on a probability measure on the observations in order to have a well-defined expectation.

The next definition avoids any probability measure. The *growth function of a function class* is

$$G_{\mathcal{F}}(l) \;=\; \ln \sup_{(\boldsymbol{x}^1,\ldots,\boldsymbol{x}^l)} N_{\mathcal{F}}(\boldsymbol{x}^1,\ldots,\boldsymbol{x}^l)\;. \tag{5.26}$$

Note that

$$H_{\mathcal{F}}(l) \;\leq\; H_{\mathcal{F}}^{\text{ann}}(l) \;\leq\; G_{\mathcal{F}}(l)\;, \tag{5.27}$$

where the first inequality comes from Jensen's inequality and the second is obvious as the supremum is larger than or equal to the expectation.

Theorem 5.2 (Sufficient Condition for Consistency of ERM)
If

$$\lim_{l\to\infty} \frac{H_{\mathcal{F}}(l)}{l} \;=\; 0 \tag{5.28}$$

then ERM is consistent.

For the next theorem we need to define what fast rate of convergence means. Fast rate of convergence means exponential convergence. ERM has a *fast rate of convergence* if

$$p\left(\sup_{\boldsymbol{w}} |R(g(.;\boldsymbol{w})) \;-\; R_{\text{emp}}(g(.;\boldsymbol{w}),l)| \;>\; \epsilon\right) \;<\; b\,\exp\left(-\,c\,\epsilon^2\,l\right) \tag{5.29}$$

holds true.

Theorem 5.3 (Sufficient Condition for Fast Rate of Convergence of ERM)
If

$$\lim_{l\to\infty} \frac{H_{\mathcal{F}}^{\text{ann}}(l)}{l} \;=\; 0 \tag{5.30}$$

then ERM has a fast rate of convergence.

The last two theorems were valid for a given probability measure on the observations. The probability measure enters the formulas via the expectation E. The growth function however does not use a probability measure.

Theorem 5.4 (Consistency of ERM for Any Probability)
The condition

$$\lim_{l\to\infty} \frac{G_{\mathcal{F}}(l)}{l} \;=\; 0 \tag{5.31}$$

is necessary and sufficient for the ERM to be consistent and also ensures a fast rate of convergence.

As can be seen from above theorems the growth function is very important as it is valid for arbitrary distributions of $\boldsymbol{x}$.

We define d_{VC} as the largest integer for which $G_{\mathcal{F}}(l) \;=\; l \ln 2$ holds:

$$d_{\text{VC}} \;=\; \max_{l}\{l \mid G_{\mathcal{F}}(l) \;=\; l \ln 2\}\,. \tag{5.32}$$

If the maximum does not exists then we set $d_{\text{VC}} \;=\; \infty$. The value d_{VC} is called the *VC-dimension* of the function class $\mathcal{F}$. The name VC-dimension is an abbreviation of Vapnik-Chervonenkis dimension. The VC-dimension d_{VC} is the maximum number of vectors that can be shattered by the function class $\mathcal{F}$.

Theorem 5.5 (VC-Dimension Bounds the Growth Function)
The growth function is bounded by

$$G_{\mathcal{F}}(l) \;\begin{cases} = & l \,\ln 2 & \text{if } l \;\leq\; d_{\text{VC}} \\ \leq & d_{\text{VC}} \left(1 \;+\; \ln \frac{l}{d_{\text{VC}}}\right) & \text{if } l \;>\; d_{\text{VC}} \end{cases}\,. \tag{5.33}$$

Fig. 5.5 depicts the statement of this theorem, that the growth function $G_{\mathcal{F}}(l)$ is either linear in l or logarithmic in l.

It follows immediately that a function class with finite VC-dimension is consistent and converges fast.

However the VC-dimension allows to derive bounds on the risk as we already have shown for function classes with finite many functions.

We now want to give examples for the VC dimension of some function classes.

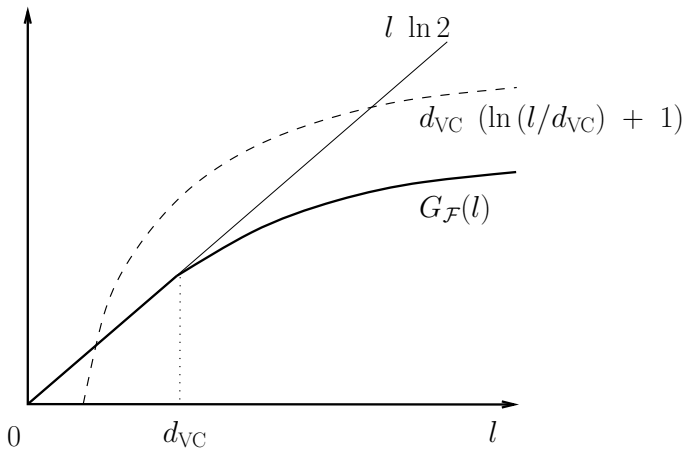

Figure 5.5: The growth function is either linear or logarithmic in l.

- *Linear functions*: The VC dimension of linear discriminant functions $g(\boldsymbol{x}; \boldsymbol{w}) \ = \ \boldsymbol{w}^T \boldsymbol{x}$ is $d_{\mathrm{VC}} \ = \ d$, where d is the dimension of the input space. The VC dimension of linear discriminant functions $g(\boldsymbol{x}; \boldsymbol{w}) \ = \ \boldsymbol{w}^T \boldsymbol{x} \ + \ b$ with offset b is

$$d_{\mathrm{VC}} \ = \ d \ + \ 1 \ . \tag{5.34}$$

- *Nondecreasing nonlinear one-dimensional functions*: The VC dimension of discriminant functions in 1D of the form $\sum_{i=1}^{k} \left| a_i \ x^i \right| \ \mathrm{sign} x \ + \ a_0$ is one. These functions are nondecreasing in x, therefore they can shatter only one point: $d_{\mathrm{VC}} \ = \ 1$. The VC-dimension is independent of number of parameters.

- *Nonlinear one-dimensional functions*: The VC dimension of discriminant functions in 1D of the form $\sin(w \ z)$ defined on $[0, 2\pi]$ is infinity: $d_{\mathrm{VC}} \ = \ \infty$. This can be seen because there exist l points $x^1, \ldots, x^l$ for which a w_0 exists for which $\sin(w_0 \ z)$ is a discriminant function.

- *Neural Networks:* $d_{\mathrm{VC}} \ \leq \ 2 \ W \ \log_2(e \ M)$ for multi-layer perceptions, where M are the number of units, W is the number of weights, e is the base of the natural logarithm (Baum & Haussler 89, Shawe-Taylor & Anthony 91). $d_{\mathrm{VC}} \ \leq \ 2 \ W \ \log_2(24 \ e \ W \ D)$ according to Bartlett & Williamson (1996) for inputs restricted to $[-D; D]$.

In the following subsections we will report error bound which can be expressed by the VC-dimension instead of the growth function, which allows to compute the actual bounds for some functions classes.

5.3 Error Bounds

The idea of deriving the error bounds is to define the set of distinguishable functions. This set has cardinality of $N_{\mathcal{F}}$, the number of different separations of inputs.

Now we can proceed as in 5.2.1, where we had a finite set of functions. In eq. (5.12) we replace the maximum by the sum over all functions. For finite many functions the supremum reduces to the maximum and we can proceed as in eq. (5.12) and use $N_{\mathcal{F}}$ as the number of functions in the class.

Another trick in the proof of the bounds is to use two half-samples and their difference

$$p\left(\sup_{\boldsymbol{w}}\left|\frac{1}{l}\sum_{i=1}^{l}L\left(y^i,g\left(\boldsymbol{x}^i;\boldsymbol{w}\right)\right)\;-\;\frac{1}{l}\sum_{i=l+1}^{2l}L\left(y^i,g\left(\boldsymbol{x}^i;\boldsymbol{w}\right)\right)\right|\;>\;\epsilon\;-\;\frac{1}{l}\right)\;\geq$$
$$\frac{1}{2}\,p\left(\sup_{\boldsymbol{w}}\left|\frac{1}{l}\sum_{i=1}^{l}L\left(y^i,g\left(\boldsymbol{x}^i;\boldsymbol{w}\right)\right)\;-\;R(g(.;\boldsymbol{w})\right|\;>\;\epsilon\right)$$

The probability that the difference of half samples exceeding a threshold is a bound on the probability that the difference of one half-sample to the risk exceeds a threshold ("symmetrization"). The symmetrization step reduces the risk which is defined on all possible samples to finite sample size. The difference on the half-samples counts how the loss on the first sample half differs from the second sample half.

Above considerations clarify why in the following bounds values derived from $N_{\mathcal{F}}$ (finite distinguishable functions) will appear and appear with arguments $2l$ (two half-samples).

Before we report the bounds, we define the minimal possible risk and its parameter:

$$\boldsymbol{w}_0\;=\;\arg\min_{\boldsymbol{w}}R(g(.;\boldsymbol{w})) \tag{5.35}$$
$$R_{\min}\;=\;\min_{\boldsymbol{w}}R(g(.;\boldsymbol{w}))\;=\;R(g(.;\boldsymbol{w}_0))\,. \tag{5.36}$$

Theorem 5.6 (Error Bound)
With probability of at least $(1\;-\;\delta)$ over possible training sets with l elements, the parameter $\boldsymbol{w}_l$ (more precisely $\boldsymbol{w}_l=\boldsymbol{w}\left(\boldsymbol{Z}_l\right)$) which minimizes the empirical risk we have

$$R(g(.;\boldsymbol{w}_l))\;\leq\;R_{\text{emp}}(g(.;\boldsymbol{w}_l),l)\;+\;\sqrt{\epsilon(l,\delta)}\,. \tag{5.37}$$

With probability of at least $(1\;-\;2\delta)$ the difference between the optimal risk and the risk of $\boldsymbol{w}_l$ is bounded by

$$R(g(.;\boldsymbol{w}_l))\;-\;R_{\min}\;<\;\sqrt{\epsilon(l,\delta)}\;+\;\sqrt{\frac{-\ln\delta}{l}}\,. \tag{5.38}$$

Here $\epsilon(l,\delta)$ can be defined for a specific probability as

$$\epsilon(l,\delta)\;=\;\frac{8}{l}\;(H_{\mathcal{F}}^{\text{ann}}(2l)\;+\;\ln(4/\delta)) \tag{5.39}$$

or for any probability as

$$\epsilon(l,\delta)\;=\;\frac{8}{l}\;(G_{\mathcal{F}}(2l)\;+\;\ln(4/\delta)) \tag{5.40}$$

where the later can be expressed though the VC-dimension d_{VC}

$$\epsilon(l,\delta)\;=\;\frac{8}{l}\;(d_{\text{VC}}\,(\ln(2l/d_{\text{VC}})\;+\;1)\;+\;\ln(4/\delta))\,. \tag{5.41}$$

The complexity measures depend all on the ratio $\frac{d_{\mathrm{VC}}}{l}$, the VC-dimension of the class of function divided by the number of training examples.

The bound above is from [Schölkopf and Smola, 2002], whereas an older bound from Vapnik is

$$R(g(.;\boldsymbol{w}_l)) \;\leq\; R_{\mathrm{emp}}(g(.;\boldsymbol{w}_l),l) + \frac{\epsilon(l,\delta)}{2}\left(1 \;+\; \sqrt{1 \;+\; \frac{R_{\mathrm{emp}}(g(.;\boldsymbol{w}_l),l)}{\epsilon(l,\delta)}}\right) . \tag{5.42}$$

It can be seen that the complexity term decreases with $\frac{1}{\sqrt{l}}$. If we have zero empirical risk then the bound on the risk decreases with $\frac{1}{\sqrt{l}}$.

Later in Section 5.6 we will see a bound on the *expected* risk which decreases with $\frac{1}{l}$ for the method of support vector machines.

The bound on the risk for the parameter $\boldsymbol{w}_l$ which minimized the empirical risk has again the form

$$R \;\leq\; R_{\mathrm{emp}} \;+\; \mathrm{complexity} \, . \tag{5.43}$$

This sum is depicted in Fig. 5.6, where it is also shown that the complexity term increases with the VC-dimension whereas the empirical error decreases with increasing VC-dimension.

Note that we again arrived at a bound which is similar to the bias-variance formulation from eq. (3.5), where the means squared error was expressed as bias term and a variance term. Bias corresponds to R_{emp} and variance to the complexity term. With increasing complexity of a function class the number of solutions with the same training error increases, that means the variance of the solutions increases.

In many practical cases the bound is not useful because only for large number of training examples l the bound gives a nontrivial value (trivial values are for example that the misclassification rate is smaller than or equal to 1). In Fig. 5.7 the bound is shown as being far above the actual test error. However in many practical cases the minimum of the bound is close (in terms of complexity) to the minimum of the test error.

For regression instead of the shattering coefficient *covering numbers* can be used. The *ε-covering number of $\mathcal{F}$ with respect to metric d* is $\mathcal{N}(\varepsilon,\mathcal{F},d)$ which is defined as the smallest number which ε-cover $\mathcal{F}$ using metric d. Usually the metric d is the distance of the function on the data set $\boldsymbol{X}$. For example the maximum norm on $\boldsymbol{X}$, that is the distance of two functions is the maximal difference of these two on the set $\boldsymbol{X}$, defines the covering number $\mathcal{N}(\varepsilon,\mathcal{F},\boldsymbol{X}_\infty)$ The *ε-growth function* is defined as

$$G(\varepsilon,\mathcal{F},l) \;=\; \ln \sup_{\boldsymbol{X}} \mathcal{N}(\varepsilon,\mathcal{F},\boldsymbol{X}_\infty) \, . \tag{5.44}$$

We obtain similar bounds on the generalization error like

$$R(g(.;\boldsymbol{w}_l)) \;\leq\; R_{\mathrm{emp}}(g(.;\boldsymbol{w}_l),l) \;+\; \sqrt{\epsilon(\varepsilon,l,\delta)} \, , \tag{5.45}$$

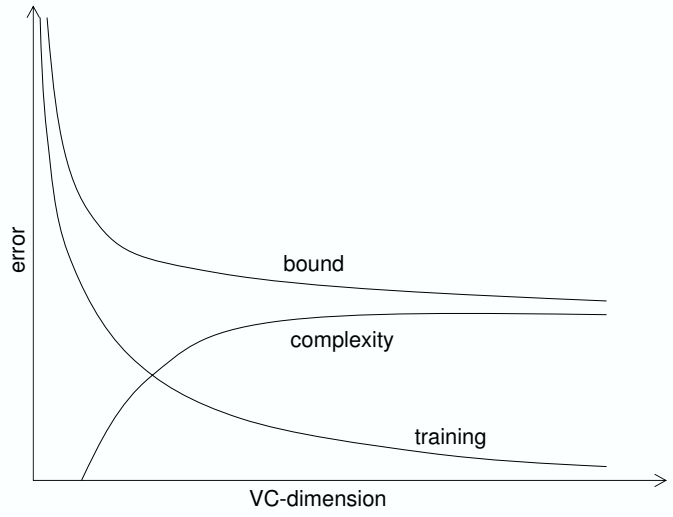

Figure 5.6: The error bound is the sum of the empirical error, the training error, and a complexity term. The complexity increases with increasing VC dimension whereas the training error decreases with increasing complexity.

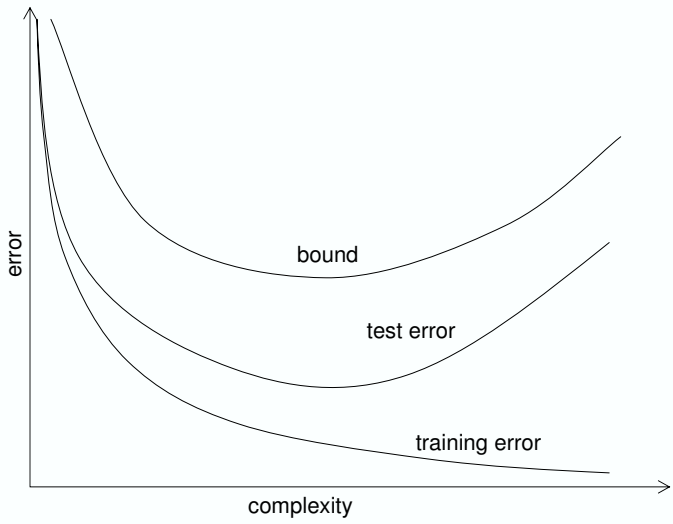

Figure 5.7: The bound on the risk, the test error, is depicted. However the bound can be much larger than the test error because it is valid for any distribution of the input.

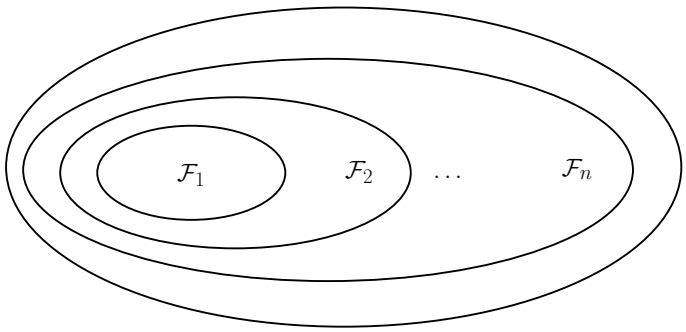

Figure 5.8: The structural risk minimization principle is based on sets of functions which are nested subsets $\mathcal{F}_n$.

where

$$\epsilon(\varepsilon, l, \delta) \;=\; \frac{36}{l}\;(\ln(12\,l) \;+\; G(\varepsilon/6, \mathcal{F}, l) \;-\; \ln\delta)\;. \tag{5.46}$$

Instead to minimizing the empirical risk it would be better to minimize the risk or at least a bound on them.

5.4 Structural Risk Minimization

The Structural Risk Minimization (SRM) principle minimizes the guaranteed risk that is a bound on the risk instead of the empirical risk alone.

In the SRM a nested set of function classes is introduced:

$$\mathcal{F}_1 \;\subset\; \mathcal{F}_2 \;\subset\; \ldots \;\subset\; \mathcal{F}_n \;\subset\; \ldots \;\;, \tag{5.47}$$

where class $\mathcal{F}_n$ possesses VC-dimension d_{VC}^n and

$$d_{\mathrm{VC}}^1 \;\leq\; d_{\mathrm{VC}}^2 \;\leq\; \ldots\; d_{\mathrm{VC}}^n \;\leq\; \ldots \quad , \tag{5.48}$$

holds.

One realization of the SRM principle is the *minimum description length* [Rissanen, 1978] or *minimum message length* [Wallace and Boulton, 1968] principle. In the minimum description length principle a sender transmits a model and the inputs $\boldsymbol{x}^1, \boldsymbol{x}^2, \ldots, \boldsymbol{x}^l$ and the receiver has to recover the labels $y^1, y^2, \ldots, \boldsymbol{y}^l$ from the model and the inputs. If the model does not supply the exact y from the input x then the sender has also to transmit the error. Goal is to minimize the transmission costs, i.e. the description length.

For fixed l the error is R_{emp} and if the model complexity corresponds to the number of bits to describe it, then the risk R is analog to the transmission costs:

$$\mathrm{transmissioncosts} \;=\; R_{\mathrm{emp}} \;+\; \mathrm{complexity}\;. \tag{5.49}$$

Minimizing the transmissions costs is equivalent to minimizing the risk for appropriate error (coding the error) and appropriate model coding which defines the complexity.

If the model codes main structures in the data, then for many training examples (assume large l) the error description can be reduced. If however the model codes specific values for one or few training points which may even correspond to random noise then it should not be transmitted via the model. Transmitting the specific values through the error would be more efficient in terms of bits than coding these values into the model. That means the model should contain rules which apply to as many data points as possible whereas data point specific variations or noise should be contained in the error.

Is the SRM principle consistent? How fast does it converge?

The SRM is always consistent and even supplies a bound on the rate of convergence. That means the SRM procedure converges to the best possible solution with probability one as the number of examples l goes to infinity.

The asymptotic rate of convergence is

$$r(l) \;=\; |R^n_{\min} \;-\; R_{\min}| \;+\; \sqrt{\frac{d^n_{\mathrm{VC}} \,\ln l}{l}}\,, \tag{5.50}$$

where $R^n_{\min}$ is the minimal risk of the function class $\mathcal{F}_n$. Asymptotic rate of convergence means that

$$p\left(\lim_{l\to\infty} \sup\; r^{-1}(l) \;\left|R\left(g\left(.\,;\,\boldsymbol{w}_l^{\mathcal{F}_n}\right)\right) \;-\; R_{\min}\right| \;<\; \infty\right) \;=\; 1\,. \tag{5.51}$$

We assume that $n \;=\; n(l)$ increases with the number of training examples so that for large enough training set size $|R^n_{\min} \;-\; R_{\min}| \;\overset{l\to\infty}{\to}\; 0$.

If the optimal solution belongs to some class $\mathcal{F}_n$ then the convergence rate is

$$r(l) \;=\; O\left(\sqrt{\frac{\ln l}{l}}\right)\,. \tag{5.52}$$

5.5 Margin as Complexity Measure

The VC-dimension can be bounded by different restrictions on the class of functions. The most famous restriction is that the zero isoline of the discriminant function (the boundary function), provided it separates the classes properly, has maximal distance γ (this distance will later be called "margin") to all data points which are contained in a sphere with radius R. Fig. 5.9 depicts such a discriminant function. Fig. 5.10 gives an intuition why a margin reduces the number of hyperplanes and therefore the VC-dimension.

The linear discriminant functions $\boldsymbol{w}^T\boldsymbol{x} \;+b$ can be scaled (scaling $\boldsymbol{w}$ and b) and give the same classification function sign $\left(\boldsymbol{w}^T\boldsymbol{x} \;+b\right)$. Of the class of discriminant functions leading to the same classification function we can choose one representative.

The representative is chosen with respect to the training data and is called *the canonical form* w.r.t. the training data $\boldsymbol{X}$. In the canonical form $\boldsymbol{w}$ and b are scaled that

$$\min_{i=1,\ldots,l} \left|\boldsymbol{w}^T\boldsymbol{x}^i \;+\; b\right| \;=\; 1\,. \tag{5.53}$$

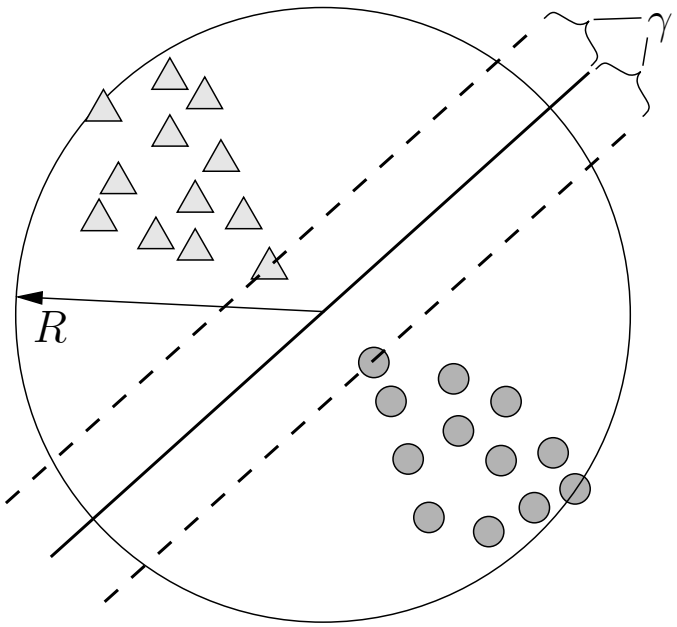

Figure 5.9: Data points are contained in a sphere of radius R at the origin. A linear discriminant function with a boundary having distance γ to all data points is depicted.

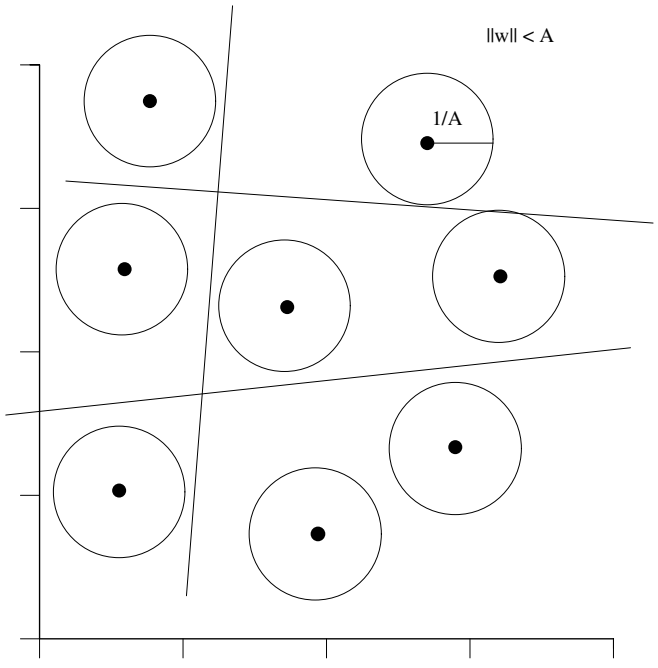

Figure 5.10: Margin means that hyperplanes must keep outside the spheres. Therefore the possible number of hyperplanes is reduced. Copyright © 1997 [Osuna et al., 1997].

Theorem 5.7 (Margin Bounds VC-dimension)
The class of classification functions sign $\left(\boldsymbol{w}^T\boldsymbol{x} + b\right)$*, where the discriminant function* $\boldsymbol{w}^T\boldsymbol{x} + b$ *is in its canonical form versus* $\boldsymbol{X}$ *which is contained in a sphere of radius* R*, and where* $\|\boldsymbol{w}\| \leq \frac{1}{\gamma}$ *satisfy*

$$d_{\mathrm{VC}} \leq \frac{R^2}{\gamma^2} \,. \tag{5.54}$$

This gives with the fact from eq. (5.34)

$$d_{\mathrm{VC}} \leq \min\{\left[\frac{R^2}{\gamma^2}\right], d\} + 1 \,, \tag{5.55}$$

where $[.]$ is the floor of a real number.

Remark: The VC-dimension is defined for a model class and should not depend on the training set.

If at least one data point exists for which the discriminant function $\boldsymbol{w}^T\boldsymbol{x} + b$ is positive and at least one data point exists for which it is negative, then we can optimize b and re-scale $\|\boldsymbol{w}\|$ in order to obtain the smallest $\|\boldsymbol{w}\|$ for discriminant function in the canonical form.

This gives the tightest bound $\frac{1}{\gamma}$ on $\|\boldsymbol{w}\|$ and therefore the smallest VC-dimension.

The optimization of b leads to the result that there exists a data point, without loss of generalization we denote it by $\boldsymbol{x}^1$, for which $\boldsymbol{w}^T\boldsymbol{x}^1 + b = 1$, and a data point without loss of generalization we denote it by $\boldsymbol{x}^2$, for which $\boldsymbol{w}^T\boldsymbol{x}^2 + b = -1$.

Fig. 5.11 depicts the situation.

To see above result, we consider (without loss of generalization) the case that the distance to the negative class is larger than 1:

$$\boldsymbol{w}^T\boldsymbol{x}^1 + b = 1 \tag{5.56}$$

and

$$\boldsymbol{w}^T\boldsymbol{x}^2 + b = -1 - \delta \,, \tag{5.57}$$

where $\delta > 0$, then

$$\boldsymbol{w}^T(\boldsymbol{x}^1 - \boldsymbol{x}^2) = 2 + \delta \,. \tag{5.58}$$

We set

$$\boldsymbol{w}_*^T = \frac{2}{2 + \delta}\,\boldsymbol{w}^T \tag{5.59}$$

which gives

$$\|\boldsymbol{w}_*\| < \|\boldsymbol{w}\| \,. \tag{5.60}$$

Further we set

$$b_* = 1 - \frac{2}{2 + \delta}\,\boldsymbol{w}^T\boldsymbol{x}^1 \,. \tag{5.61}$$

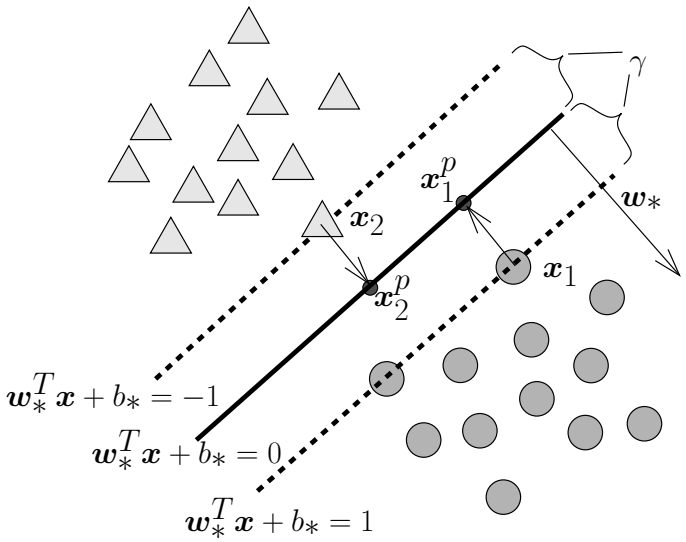

Figure 5.11: The offset b is optimized in order to obtain the largest $\|\boldsymbol{w}\|$ for the canonical form which is $\|\boldsymbol{w}_*\|$ for the optimal value b_*. Now there exist points $\boldsymbol{x}^1$ and $\boldsymbol{x}^2$ with $\boldsymbol{w}_*^T\boldsymbol{x}^1 + b_* = 1$ and $\boldsymbol{w}_*^T\boldsymbol{x}^2 + b_* = -1$. The distance of $\boldsymbol{x}^1$ to the boundary function $\boldsymbol{w}_*^T\boldsymbol{x}^2 + b_* = 0$ is $\gamma = \frac{1}{\|\boldsymbol{w}_*\|}$. In the figure also $\left(\boldsymbol{x}^1\right)^p$ ($\left(\boldsymbol{x}^2\right)^p$), the projection of $\boldsymbol{x}^1$ ($\boldsymbol{x}^2$) onto the boundary functions is depicted.

We obtain

$$\boldsymbol{w}_*^T\boldsymbol{x}^1 + b_* = \frac{2}{2+\delta}\,\boldsymbol{w}^T\boldsymbol{x}^1 + 1 - \frac{2}{2+\delta}\,\boldsymbol{w}^T\boldsymbol{x}^1 = 1. \tag{5.62}$$

and

$$\begin{aligned}\boldsymbol{w}_*^T\boldsymbol{x}^2 + b_* &= \frac{2}{2+\delta}\,\boldsymbol{w}^T\boldsymbol{x}^2 + 1 - \frac{2}{2+\delta}\,\boldsymbol{w}^T\boldsymbol{x}^1 = \\ -\frac{2}{2+\delta}\,\boldsymbol{w}^T\left(\boldsymbol{x}^1 - \boldsymbol{x}^2\right) + 1 &= -\frac{2}{2+\delta}(2+\delta) + 1 = -1\,.\end{aligned} \tag{5.63}$$

We can generalize above example. Without loss of generalization assume that

$$\begin{aligned}\boldsymbol{x}^1 &= \arg\min_{\boldsymbol{x}^i:\, y^i=1}\{\boldsymbol{w}^T\boldsymbol{x}^i\} \text{ and} \\ \boldsymbol{x}^2 &= \arg\max_{\boldsymbol{x}^i:\, y^i=-1}\{\boldsymbol{w}^T\boldsymbol{x}^i\}\,.\end{aligned} \tag{5.64}$$

Then we set

$$\boldsymbol{w}_*^T = \frac{2}{\boldsymbol{w}^T\left(\boldsymbol{x}^1 - \boldsymbol{x}^2\right)}\,\boldsymbol{w} \tag{5.65}$$

which gives

$$\|\boldsymbol{w}_*\| < \|\boldsymbol{w}\| \tag{5.66}$$

because both $\boldsymbol{x}^1$ and $\boldsymbol{x}^2$ have at least a distance of 1 to the boundary functions which guarantees that $\boldsymbol{w}^T\,(\boldsymbol{x}^1\,-\,\boldsymbol{x}^2)\,>\,2$. Further we set

$$b_* \,=\, \frac{2}{\boldsymbol{w}^T\,(\boldsymbol{x}^1\,-\,\boldsymbol{x}^2)}\,\left(-\,\frac{1}{2}\,\boldsymbol{w}^T\,(\boldsymbol{x}^1\,+\,\boldsymbol{x}^2)\right)\,= \\ -\,\frac{\boldsymbol{w}^T\,(\boldsymbol{x}^1\,+\,\boldsymbol{x}^2)}{\boldsymbol{w}^T\,(\boldsymbol{x}^1\,-\,\boldsymbol{x}^2)}\,. \tag{5.67}$$

This gives

$$\boldsymbol{w}_*^T\boldsymbol{x}^1\,+\,b_*\,=\,\frac{2}{\boldsymbol{w}^T\,(\boldsymbol{x}^1\,-\,\boldsymbol{x}^2)}\left(\boldsymbol{w}^T\boldsymbol{x}^1\,-\,\frac{1}{2}\boldsymbol{w}^T(\boldsymbol{x}^1\,+\,\boldsymbol{x}^2)\right)\,= \\ \frac{2}{\boldsymbol{w}^T\,(\boldsymbol{x}^1\,-\,\boldsymbol{x}^2)}\left(\frac{1}{2}\boldsymbol{w}^T\boldsymbol{x}^1\,-\,\frac{1}{2}\boldsymbol{w}^T\boldsymbol{x}^2\right)\,=\,1 \tag{5.68}$$

and similarly

$$\boldsymbol{w}_*^T\boldsymbol{x}^2\,+\,b_*\,=\,-1\,. \tag{5.69}$$

We see that

$$\boldsymbol{w}_*^T(\boldsymbol{x}^1\,-\,\boldsymbol{x}^2)\,=\,\boldsymbol{w}_*^T\boldsymbol{x}^1\,+\,b_*\,-\,\boldsymbol{w}_*^T\boldsymbol{x}^2\,+\,b_*\,=\,2\,. \tag{5.70}$$

For $\|\boldsymbol{w}\|\,=\,\alpha\,\|\boldsymbol{w}_*\|$ with $0<\alpha<1$ we would obtain $\boldsymbol{w}^T(\boldsymbol{x}^1\,-\,\boldsymbol{x}^2)\,<\,2$ and either $\boldsymbol{x}^1$ or $\boldsymbol{x}^2$ is closer than 1 to the boundary function which contradicts that the discriminant function is in the canonical form.

Therefore the optimal $\boldsymbol{w}_*$ and b_* are unique.

We want to compute the distance of $\boldsymbol{x}^1$ to the boundary function. The projection of $\boldsymbol{x}^1$ onto the boundary function is $\left(\boldsymbol{x}^1\right)^p\,=\,\boldsymbol{x}^1\,-\,\alpha\boldsymbol{w}_*$ and fulfills

$$\boldsymbol{w}_*^T\left(\boldsymbol{x}^1\right)^p\,+\,b_*=\,0 \\ \Rightarrow\;\boldsymbol{w}_*^T\left(\boldsymbol{x}^1\,-\,\alpha\boldsymbol{w}_*\right)+b_*\,= \\ \boldsymbol{w}_*^T\boldsymbol{x}^1\,-\,\alpha\,\|\boldsymbol{w}_*\|^2\,+\,b_*\,=\,1\,-\,\alpha\,\|\boldsymbol{w}_*\|^2\,=\,0 \\ \Rightarrow\;\alpha\,=\,\frac{1}{\|\boldsymbol{w}_*\|^2} \\ \Rightarrow\;\left(\boldsymbol{x}^1\right)^p\,=\,\boldsymbol{x}^1\,-\,\frac{1}{\|\boldsymbol{w}_*\|^2}\boldsymbol{w}_* \tag{5.71}$$

The distance of $\boldsymbol{x}^1$ to the boundary function is

$$\left\|\boldsymbol{x}^1\,-\,\left(\boldsymbol{x}^1\right)^p\right\|\,=\,\left\|\boldsymbol{x}^1\,-\,\boldsymbol{x}^1\,-\,\frac{1}{\|\boldsymbol{w}_*\|^2}\boldsymbol{w}_*\right\|\,=\frac{1}{\|\boldsymbol{w}_*\|}\,=\,\gamma\,. \tag{5.72}$$

Similar the distance of $\boldsymbol{x}^2$ to the boundary function is

$$\frac{1}{\|\boldsymbol{w}_*\|}\,=\,\gamma\,. \tag{5.73}$$

Theorem 5.8 (Margin Error Bound)
The classification functions sign $\left(\boldsymbol{w}^T\boldsymbol{x} + b\right)$ *are restricted to* $\|\boldsymbol{w}\| \leq \frac{1}{\gamma}$ *and* $\|\boldsymbol{x}\| < R$*. Let* ν *be the fraction of training examples which have a margin (distance to* $\boldsymbol{w}^T\boldsymbol{x} + b = 0$ *) smaller than* $\frac{\rho}{\|\boldsymbol{w}\|}$.

With probability of at least $(1 - \delta)$ *of drawing* l *examples, the probability to misclassify a new example is bounded from above by*

$$\nu + \sqrt{\frac{c}{l}\left(\frac{R^2}{\rho^2\,\gamma^2}\ln^2 l + \ln(1/\delta)\right)}, \tag{5.74}$$

where c *is a constant.*

The probability $(1 - \delta)$ is the confidence in drawing appropriate training samples whereas the bound is on the probability of drawing a test example. The bound is from Bartlett and Shawe-Taylor.

Again the bound is of the order $\nu + \frac{1}{\sqrt{l}}$.

In the next chapter we will introduce support vector machines as the method for structural risk minimization, where the margin is maximized.

5.6 Average Error Bounds for SVMs

In this section we show average error bounds for support vector machines (SVMs). The average error bounds are tighter than the worst case bounds as factors $\frac{1}{\sqrt{l}}$ in the worst case bounds are now of the order $\frac{1}{l}$ in average bounds.

The following theorems are proved by using the Leave-One-Out Cross Validation (LOO CV) estimator which was shown in Subsection 2.2.2.2 to be almost unbiased. The complexity of the SVM is described by the margin which in turn can be expressed through the support vector weights $\alpha_i > 0$.

Essential support vectors are the support vectors for which the solution changes if they are removed from the training set. Fig. 5.12 shows essential and non-essential support vectors for a two-dimensional problem.

We denote the number of essential support vectors by k_l and r_l the radius of the sphere which contains all essential support vectors.

First we note that $k_l \leq d+1$ (because $(d+1)$ points in general position are enough to define a hyperplane which has equal distance to all points).

Now we can give bounds for the expected risk $ER(g(.;\boldsymbol{w}_l))$, where the expectation is taken over the training set of size l and a test data point.

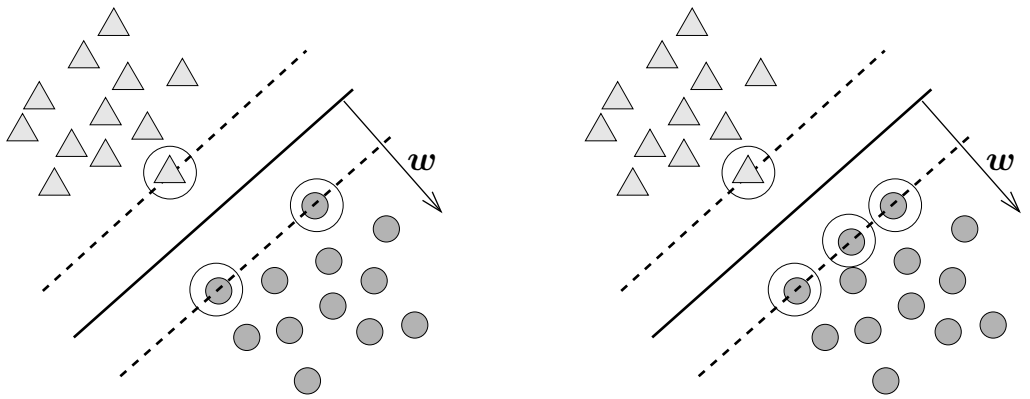

Figure 5.12: Essential support vectors. Left: all support vectors (the circled points) are essential. Right: the three support vectors on the right hand side line are not essential because removing one of them does not change the solution.

Theorem 5.9 (Average Bounds for SVMs) *For the expected risk with above definitions*

$$ER(g(.;\boldsymbol{w}_l)) \;\leq\; \frac{Ek_{l+1}}{l\;+\;1} \tag{5.75}$$

$$ER(g(.;\boldsymbol{w}_l)) \;\leq\; \frac{d+1}{l\;+\;1} \tag{5.76}$$

$$ER(g(.;\boldsymbol{w}_l)) \;\leq\; \frac{E\left(\frac{r_{l+1}}{\gamma_{l+1}}\right)^2}{l\;+\;1} \tag{5.77}$$

$$ER(g(.;\boldsymbol{w}_l)) \;\leq\; \frac{E\min\left\{k_{l+1}\;,\;\left(\frac{r_{l+1}}{\gamma_{l+1}}\right)^2\right\}}{l\;+\;1} \tag{5.78}$$

$$ER(g(.;\boldsymbol{w}_l)) \;\leq\; \frac{E\left(\left(k^*_{l+1}\right)^2\sum_{i^*}\alpha_{i^*}\;+\;m\right)}{l\;+\;1} \tag{5.79}$$

$$C \leq r_l^{-2}: \;\; ER(g(.;\boldsymbol{w}_l)) \;\leq\; \frac{E\left(\left(k_{l+1}\right)^2\sum_i\alpha_i\right)}{l\;+\;1}\;, \tag{5.80}$$

where i^* are the support vectors with $0 \;<\; \alpha_{i^*} \;<\; C$ and m is the number of support vectors with $\alpha_i \;=\; C$.

All of above bounds are based on the leave-one-out cross validation estimator and its property to be almost unbiased.

It is important to note that we obtain for the expected risk a bound of the order $\frac{1}{l}$ whereas we saw in the theoretical sections bounds (worst case bounds) for the risk of $\frac{1}{\sqrt{l}}$. Interesting to know would be the variance of the expected risk.

Chapter 6

Theory of Kernels and Dot Products

6.1 Kernels, Dot Products, and Mercer's Theorem

A function which produces a scalar out of two vectors is called *kernel* k. For example $k\left(\boldsymbol{x}^i, \boldsymbol{x}^j\right) = \left(\left(\boldsymbol{x}^i\right)^T \boldsymbol{x}^j + 1\right)^3$. Certain kernels represent the mapping of vectors into a feature space and a dot product in this space.

The idea of using kernels is to map the feature vectors $\boldsymbol{x}$ by a nonlinear function $\boldsymbol{\Phi}$ into a feature space:

$$\boldsymbol{\Phi}: \ \mathbb{R}^d \to \mathbb{R}^{d_\Phi}, \ \boldsymbol{x} \mapsto \boldsymbol{x}_\phi, \ \boldsymbol{x}_\phi = \boldsymbol{\Phi}(\boldsymbol{x}) \ . \tag{6.1}$$

In this feature space we can apply linear methods for which the theory holds. Afterwards we can project the results back into the original space. Fig. 6.1 depicts the feature mapping. In the feature space the data is assumed to be linear separable . The result can be transferred back into the original space.

For example consider

$$\boldsymbol{\Phi}(\boldsymbol{x}) = \left(x_1^2 \ , \ x_2^2 \ , \ \sqrt{2}\, x_1 \, x_2\right) \ . \tag{6.2}$$

This is a mapping from a two-dimensional space into a three-dimensional space. The four data points $\boldsymbol{x}^1 = (1,1), \boldsymbol{x}^2 = (1,-1), \boldsymbol{x}^3 = (-1,1), \boldsymbol{x}^4 = (-1,-1)$ with labels $y^1 = -1, y^2 = 1, y^3 = 1, y^4 = -1$ are not separable in the two-dimensional space. Their images are

$$\begin{aligned}
\boldsymbol{\Phi}(\boldsymbol{x}^1) &= (1, 1, \sqrt{2}) \\
\boldsymbol{\Phi}(\boldsymbol{x}^2) &= (1, 1, -\sqrt{2}) \\
\boldsymbol{\Phi}(\boldsymbol{x}^3) &= (1, 1, -\sqrt{2}) \\
\boldsymbol{\Phi}(\boldsymbol{x}^4) &= (1, 1, \sqrt{2}) \ ,
\end{aligned}$$

which are linearly separable in the three-dimensional space. Fig. 6.2 shows the mapping into the three-dimensional space.

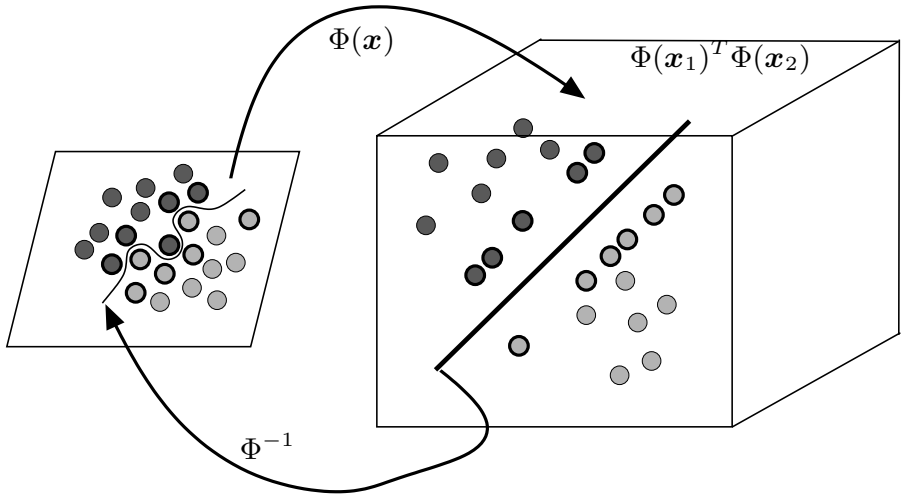

Figure 6.1: Nonlinearly separable data is mapped into a feature space where the data is linear separable. The "support vectors" (support vector machine) in feature space are marked by thicker borders. These vectors as well as the boundary function are shown in the original space where the linear boundary function becomes a nonlinear boundary function.

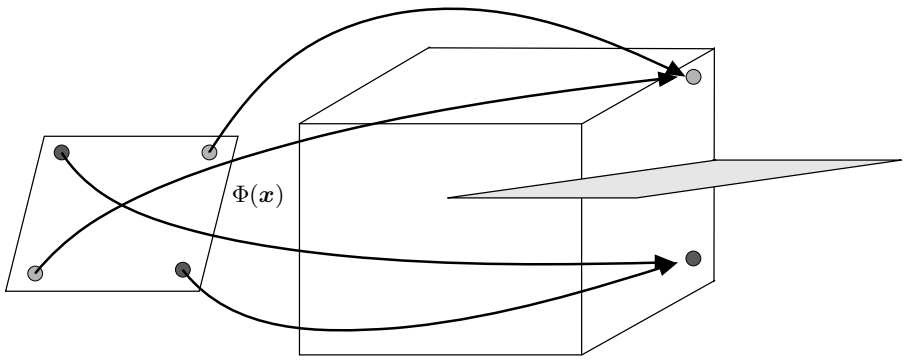

Figure 6.2: An example of a mapping from the two-dimensional space into the three-dimensional space. The data points are not linearly separable in the two-dimensional space but in the three-dimensional space.

We write in the following x_{mn} for $(\boldsymbol{x}^m)_n$, the j-th component of the vector $\boldsymbol{x}^i$. The dot product in the three-dimensional space is

$$\begin{aligned}&\boldsymbol{\Phi}^T(\boldsymbol{x}^i)\boldsymbol{\Phi}(\boldsymbol{x}^j) \;=\; x_{i1}^2\, x_{j1}^2 \;+\; x_{i2}^2\, x_{j2}^2 \;+\; 2\, x_{i1}\, x_{i2}\, x_{j1}\, x_{j2} \;=\\ &(x_{i1}\, x_{j1} \;+\; x_{i2}\, x_{j2})^2 \;=\; \left(\left(\boldsymbol{x}^i\right)^T \boldsymbol{x}^j\right)^2 .\end{aligned}$$

In another example we can map the two-dimensional vectors into a 9-dimensional space by

$$\begin{aligned}\boldsymbol{\Phi}(\boldsymbol{x}) \;=\; &\Big(x_1^3\,,\; x_2^3\,,\; \sqrt{3}\, x_1^2\, x_2\,,\; \sqrt{3}\, x_2^2\, x_1\,, \\ &\sqrt{3}\, x_1^2\,,\; \sqrt{3}\, x_2^2\,,\; \sqrt{6}\, x_1\, x_2\,,\; \sqrt{3}\, x_1\,,\; \sqrt{3}\, x_2\Big)\;.\end{aligned} \tag{6.3}$$

The dot product in the 9-dimensional space is

$$\begin{aligned}&\boldsymbol{\Phi}^T(\boldsymbol{x}^i)\boldsymbol{\Phi}(\boldsymbol{x}^j) \;= \\ &x_{i1}^3\, x_{j1}^3 \;+\; x_{i2}^3\, x_{j2}^3 \;+ \\ &3\, x_{i1}^2\, x_{i2}\, x_{j1}^2\, x_{j2} \;+\; 3\, x_{i2}^2\, x_{i1}\, x_{j2}^2\, x_{j1} \;+ \\ &3\, x_{i1}^2\, x_{j1}^2 \;+\; 3\, x_{i2}^2\, x_{j2}^2 \;+ \\ &6\, x_{i1}\, x_{i2}\, x_{j1}\, x_{j2} \;+\; 3\, x_{i1}\, x_{j1} \;+\; 3\, x_{i2} x_{j2} \;= \\ &=\; \left(\left(\boldsymbol{x}^i\right)^T \boldsymbol{x}^j \;+\; 1\right)^3 \;-1\,.\end{aligned}$$

Therefore mapping into the feature space and dot product in this space can be unified by $\left(\left(\boldsymbol{x}^i\right)^T \boldsymbol{x}^j \;+\; 1\right)^3$. A function which produces a scalar out of two vectors is called *kernel* k. In our example we have $k\left(\boldsymbol{x}^i, \boldsymbol{x}^j\right) \;=\; \left(\left(\boldsymbol{x}^i\right)^T \boldsymbol{x}^j \;+\; 1\right)^3$.

Certain kernels represent the mapping of vectors into a feature space and a dot product in this space. The following theorem characterizes functions which build a dot product in some space.

Theorem 6.1 (Mercer)
Let the kernel k be symmetric and from $L_2(X \times X)$ defining a Hilbert-Schmidt operator

$$T_k(f)(\boldsymbol{x}) \;=\; \int_X k(\boldsymbol{x}, \boldsymbol{x}')\; f(\boldsymbol{x}')\; d\boldsymbol{x}'\;. \tag{6.4}$$

If T_k is positive semi-definite, i.e. for all $f \in L_2(X)$

$$\int_{X\times X} k(\boldsymbol{x}, \boldsymbol{x}')\; f(\boldsymbol{x})\; f(\boldsymbol{x}')\; d\boldsymbol{x}\; d\boldsymbol{x}' \;\geq\; 0\,, \tag{6.5}$$

then T_k has eigenvalues $\lambda_j \;\geq\; 0$ with associated eigenfunctions $\psi_j \in L_2(X)$. Further

$$(\lambda_1, \lambda_2, \ldots) \in \ell_1 \tag{6.6}$$

$$k(\boldsymbol{x}, \boldsymbol{x}') \;=\; \sum_j \lambda_j\; \psi_j(\boldsymbol{x})\; \psi_j(\boldsymbol{x}')\;, \tag{6.7}$$

where ℓ_1 is the space of vectors with finite one-norm and the last sum converges absolutely and uniformly for almost all $\boldsymbol{x}$ and $\boldsymbol{x}'$.

The sum may be an infinite sum for which the eigenvalues converge to zero. In this case the feature space is an infinite dimensional space.

Here "for almost all" means "except for a set with zero measure", i.e. single points may lead to an absolute and uniform convergence. That the convergence is "absolutely and uniformly" is important because the sum can be resorted and derivative and sum can be exchanged.

Note that if X is a compact interval $[a, b]$ and k is continuous then eq. (6.5) is equivalent to positive definiteness of k. A kernel k is *positive semi-definite* if for all l, all $\boldsymbol{x}^1, \ldots, \boldsymbol{x}^l$, and all $\alpha_i, 1 \leq i \leq l$

$$\sum_{i,j=1,1}^{l,l} \alpha_i \, \alpha_j \, k(\boldsymbol{x}^i, \boldsymbol{x}^j) \; \geq \; 0 \, . \tag{6.8}$$

Using the gram matrix $\boldsymbol{K}$ with $K_{ij} = k(\boldsymbol{x}^i, \boldsymbol{x}^j)$ and the vector $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_l)$ this is

$$\boldsymbol{\alpha}^T \, \boldsymbol{K} \, \boldsymbol{\alpha} \; \geq \; 0 \, . \tag{6.9}$$

6.2 Reproducing Kernel Hilbert Space

Reproducing kernel Hilbert spaces are an important theoretical tool for proofing properties of kernel methods.

Hilbert spaces can be defined by reproducing kernels Aronszajn [1950] (see also Stefan Bergman in the 1950s). Later other work on reproducing kernel Hilbert spaces were published Berlinet and Thomas [2004], Kimeldorf and Wahba [1971], Wahba [1990], Cucker and Smale [2002]. See for this section `http://en.wikipedia.org/wiki/Reproducing_kernel_Hilbert_space`.

X is a set and H a Hilbert space of complex-valued functions on X. We say that H is a reproducing kernel Hilbert space if every linear map of the form (the evaluation at x)

$$L_x : f \mapsto f(x) \tag{6.10}$$

from H to the complex numbers is continuous for any x in X.

Theorem 6.2 (Riesz representation theorem)
Let H^* denote H's dual space, consisting of all continuous linear functionals from H into complex numbers $\mathbb{C}$. If x is an element of H, then the function ϕ_x defined by

$$\phi_x(y) \; = \; \langle y, x \rangle \quad \forall y \in H \; , \tag{6.11}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product of the Hilbert space, is an element of H^*. The Riesz representation theorem states that every element of H^* can be written uniquely in this form, that is the mapping

$$\Phi : \; H \to H^*, \quad \Phi(x) = \phi_x \tag{6.12}$$

is an isometric (anti-) isomorphism.

We now apply this theorem to kernels. Mercer's theorem stated that a positive definite kernel k is a dot product:

$$k(x, y) \;=\; \langle x, y \rangle \;. \tag{6.13}$$

Therefore the kernel k defines a Hilbert space on X. In contrast to this definition, we now derive the kernel from Riesz representation theorem.

Theorem 6.2 implies that for every x in X there exists an element k_x of H with the property that:

$$f(x) \;=\; \langle f,\, k_x \rangle \quad \forall f \in H \;. \tag{6.14}$$

This function k_x is called the "point-evaluation functional" at the point x.

Since H is a function space, k_x is a function which can be evaluated at every point. This allows us to define the kernel as the function $k : X \times X \to \mathbb{C}$ by

$$k(x, y) \;\stackrel{\text{def}}{=}\; k_x(y) \;. \tag{6.15}$$

The function k is called the reproducing kernel for the Hilbert space H. k is completely determined by H because Theorem 6.2 guarantees that for every x in X, the element k_x satisfying Eq. (6.14) is unique. H is called "reproducing kernel Hilbert space" (RKHS).

The next theorem states that every symmetric, positive definite kernel uniquely defines an RKHS.

Theorem 6.3 (Moore-Aronszajn)
Suppose k is a symmetric, positive definite kernel on a set X. Then there is a unique Hilbert space of functions on X for which k is a reproducing kernel.

Proof.
Define, for all x in X, $k_x = k(x, \cdot)$. Let H_0 be the linear span of $\{k_x : \; x \in X\}$. Define an inner product on H_0 by

$$\left\langle \sum_{j=1}^{n} b_j k_{y_j}, \sum_{i=1}^{m} a_i k_{x_i} \right\rangle \;=\; \sum_{i=1}^{m} \sum_{j=1}^{n} \overline{a_i} b_j k(y_j, x_i) \;. \tag{6.16}$$

The symmetry of this inner product follows from the symmetry of k and the non-degeneracy follows from the fact that k is positive definite.

Let H be the completion of H_0 with respect to this inner product. Then H consists of functions of the form

$$f(x) \;=\; \sum_{i=1}^{\infty} a_i k_{x_i}(x) \;, \tag{6.17}$$

where $\sum_{i=1}^{\infty} a_i^2 k(x_i, x_i) < \infty$. The fact that the above sum converges for every x follows from the Cauchy-Schwartz inequality.

We confirm the RKHS property Eq. (6.14):

$$\langle f, k_x \rangle \;=\; \left\langle \sum_{i=1}^{\infty} a_i k_{x_i}, k_x \right\rangle \;=\; \sum_{i=1}^{\infty} a_i k(x_i, x) \;=\; f(x)\,. \tag{6.18}$$

Therefore we have

$$f(x) \;=\; \sum_{i=1}^{\infty} a_i k(x_i, x)\,. \tag{6.19}$$

which is for example the model class of support vector machines.

To prove uniqueness, let G be another Hilbert space of functions for which k is a reproducing kernel. For any x and y in X, Eq. (6.14) implies that

$$\langle k_x, k_y \rangle_H \;=\; k(x, y) \;=\; \langle k_x, k_y \rangle_G\,. \tag{6.20}$$

By linearity,

$$\langle \cdot, \cdot \rangle_H \;=\; \langle \cdot, \cdot \rangle_G \tag{6.21}$$

on the span of $\{k_x : \; x \in X\}$. Then $G = H$ by the uniqueness of the completion. **End Proof.**

Properties of the reproducing kernel and the RKHS:

- **reproducing property**:

$$k(x, y) \;=\; \langle k(x, \cdot), k(y, \cdot) \rangle\,. \tag{6.22}$$

- **orthonormal sequences, kernel expansion**: If $\{\phi_k\}_{k=1}^{\infty}$ is an orthonormal sequence such that the closure of its span is equal to H, then

$$k\,(x, y) \;=\; \sum_{k=1}^{\infty} \phi_k\,(x)\,\phi_k\,(y)\;. \tag{6.23}$$

- **Moore-Aronszajn Theorem 6.3**: every symmetric, positive definite kernel defines a unique reproducing kernel Hilbert space.

For machine learning, the model class

$$f(x) \;=\; \sum_{i=1}^{\infty} a_i k_{x_i}(x) \;=\; \sum_{i=1}^{\infty} a_i k(x_i, x) \tag{6.24}$$

is of importance. Since all models of the model class can be represented by the RKHS, this Hilbert space is convenient to proof properties of models or model selection methods.

Chapter 7

Optimization Techniques

7.1 Parameter Optimization and Error Minimization

We focus on a model class of parameterized models with parameter vector $\boldsymbol{w}$. Goal is to find or to select the optimal model. The optimal model is the one which optimizes the objective function. To find the optimal model, we have to search in the parameter space.

The objective function is defined by the problem to solve. In general the objective function includes the empirical error and a term which penalizes complexity. *The goal is to find the model from the model class which optimizes the objective function.*

7.1.1 Search Methods and Evolutionary Approaches

In principle any search algorithm can be used. The simplest way is *random search*, where randomly a parameter vector is selected and then evaluated – the so far best solution is kept.

Another method would be *exhaustive search*, where the parameter space is searched systematically.

These two methods will find the optimal solution for a finite set of possible parameters. They do not use any dependency between objective function and parameter space. For example, if the objective function should be minimized and is 1 for every parameter $\boldsymbol{w}$ except for one parameter $\boldsymbol{w}_{\mathrm{opt}}$ which gives 0. That is a singular solution.

In general there are dependencies between objective function and parameters. These dependencies should be utilized to find the model which optimizes the objective function.

The first dependency is that good solutions are near good solutions in parameter space even if the objective is not continuous and not differentiable. In this case a *stochastic gradient* can be used to locally optimize the objective function. In the context of *genetic algorithms* a stochastic gradient corresponds to mutations which are small. With "small" we mean that every component, e.g. "gene" is mutated only slightly or only one component is mutated at all. In general a stochastic gradient tests solutions which are similar to the current best solution or current best solutions. Finally an update of the current best solution is made sometimes by combining different parameter changes which improved the current best solution.

Another dependency is that good solutions share properties of their parameter vector which are independent of other components of the parameter vector. For example if certain dependencies

between specific parameters guarantee good solutions and these specific parameters do not influence other parameters. In this case *genetic algorithms* can utilize these dependencies through the "crossover mutations" where parts of different parameter vectors are combined. Important is that components are independent of other components, i.e. the solutions have different building blocks which can improve the objective independently of other building blocks.

Besides the genetic algorithms there are other evolutionary strategies to optimize models. These strategies include *genetic programming*, *swarm algorithms*, *ant algorithms*, *self-modifying policies*, *Optimal Ordered Problem Solver*, etc. Sometimes the model class is predefined by the search algorithm or the model class is not parameterized. The latter even modify their own search strategy.

All these methods have to deal with local optima of the objective function. To reduce the problem of optima, genetic algorithms search in general at different locations in the parameter space simultaneously. Also other approaches search in parallel at different locations in parameter space.

To overcome the problem of local optima *simulated annealing* (SA) has been suggested. SA can be shown to find the global solution with probability one if the annealing process is slow enough. SA probabilistic jumps from the current state into another state where the probability is given by the objective function. The objective function is transformed to represent an energy function, so that SA jumps into energetically favorable states. The energy function follows the Maxwell-Boltzmann distribution and the sampling is similar to the Metropolis-Hastings algorithm. A global parameter, the temperature, determines which jumps are possible. At the beginning large jumps even into energetically worse regions are possible due to high temperature whereas with low temperature only small jumps are possible and energetically worse regions are avoided. Therefore the parameter space is at the beginning scanned for favorable regions which later are searched in more detail.

Advantages of these methods are that they

- can deal with discrete problems and non-differentiable objective functions and
- are very easy to implement.

Disadvantages of these methods are that they

- are computationally expensive for large parameter spaces and
- depend on the representation of the model.

To see the computational load of stochastic gradient or genetic algorithms assume that the parameter vector has W components. If we only check whether these components should be increased or decreased, we have 2^W decisions to make. When the amount of change also matters then this number of candidates will be soon infeasible to check only to make one improvement step.

The dependency between the parameter space and the objective function which will be of interest in the following is that the objective function is differentiable with respect to the parameters.

Therefore we can make use of gradient information in the search of the (local) minimum.

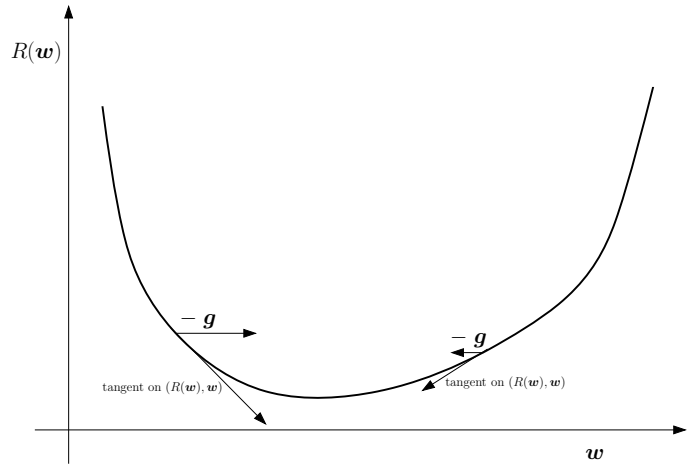

Figure 7.1: The negative gradient $-\boldsymbol{g}$ gives the direction of the steepest decent depicted by the tangent on $(R(\boldsymbol{w}), \boldsymbol{w})$, the error surface.

Problems like that were already treated in Subsection 7.3. However we focused on convex optimization. There only one minimum exists. Further we treated linear and quadratic objectives. The idea of Lagrange multipliers carries over to constraint optimization in general: assume that the problem is locally convex.

7.1.2 Gradient Descent

Assume that the objective function $R(g(.; \boldsymbol{w}))$ is a differentiable function with respect to the parameter vector $\boldsymbol{w}$. The function g is the model. For simplicity, we will write $R(\boldsymbol{w}) = R(g(.; \boldsymbol{w}))$.

The gradient is

$$\frac{\partial R(\boldsymbol{w})}{\partial \boldsymbol{w}} = \nabla_{\boldsymbol{w}} R(\boldsymbol{w}) = \left(\frac{\partial R(\boldsymbol{w})}{\partial w_1}, \ldots, \frac{\partial R(\boldsymbol{w})}{\partial w_W} \right)^T , \tag{7.1}$$

for a W-dimensional parameter vector $\boldsymbol{w}$.

For the gradient of $R(\boldsymbol{w})$ we use the vector

$$\boldsymbol{g} = \nabla_{\boldsymbol{w}} R(\boldsymbol{w}) . \tag{7.2}$$

The negative gradient is the direction of the steepest descent of the objective. The negative gradient is depicted in Fig. 7.1 for a one-dimensional error surface and in Fig. 7.2 for a two-dimensional error surface.

Because the gradient is valid only locally (for nonlinear objectives) we only go a small step in the negative gradient direction if we want to minimize the objective function. The step-size is controlled by $0 < \eta$, the *learning rate*.

The gradient descent update is

$$\Delta \boldsymbol{w} = -\eta \, \nabla_{\boldsymbol{w}} R(\boldsymbol{w}) \tag{7.3}$$

$$\boldsymbol{w}^{\text{new}} = \boldsymbol{w}^{\text{old}} + \Delta \boldsymbol{w} . \tag{7.4}$$

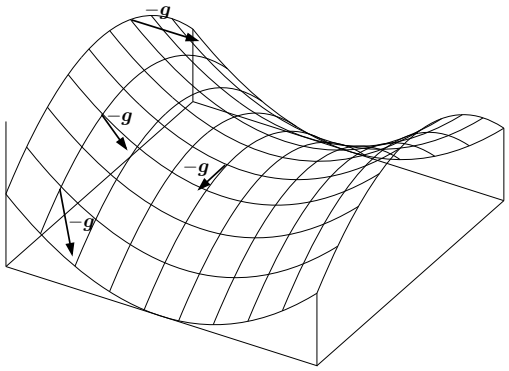

Figure 7.2: The negative gradient $-\boldsymbol{g}$ attached at different positions on a two-dimensional error surface $(R(\boldsymbol{w}), \boldsymbol{w})$. Again the negative gradient gives the direction of the steepest descent.

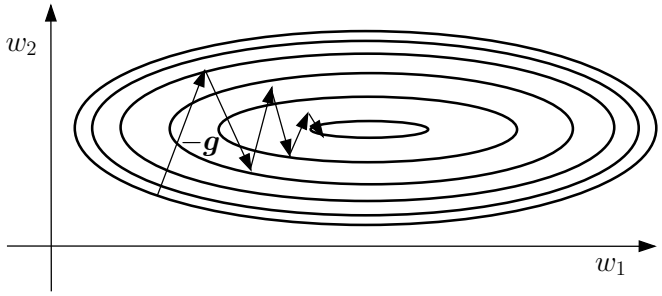

Figure 7.3: The negative gradient $-\boldsymbol{g}$ oscillates as it converges to the minimum.

Momentum Term. Sometimes gradient descent oscillates or slows down because the error surface (the objective function) has a flat plateau. To avoid oscillation and to speed up gradient descent a momentum term can be used. The oscillation of the gradient is depicted in Fig. 7.3 and the reduction of oscillation through the momentum term in Fig. 7.4.

Another effect of the momentum term is that in flat regions the gradients pointing in the same directions are accumulated and the learning rate is implicitly increased. The problem of flat regions is depicted in Fig. 7.5 and the speed up of the convergence in flat regions in Fig. 7.6.

The gradient descent update with momentum term is

$$\Delta^{\text{new}}\boldsymbol{w} \;=\; -\,\eta\,\nabla_{\boldsymbol{w}} R(\boldsymbol{w}) \tag{7.5}$$

$$\boldsymbol{w}^{\text{new}} \;=\; \boldsymbol{w}^{\text{old}} \;+\; \Delta^{\text{new}}\boldsymbol{w} \;+\; \mu\,\Delta^{\text{old}}\boldsymbol{w} \tag{7.6}$$

$$\Delta^{\text{old}}\boldsymbol{w} \;=\; \Delta^{\text{new}}\boldsymbol{w}\,, \tag{7.7}$$

where $0 \;\leq\; \mu \;\leq\; 1$ is the *momentum parameter* or *momentum factor*.

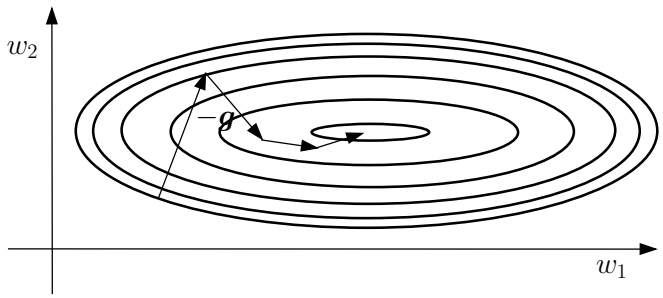

Figure 7.4: Using the momentum term the oscillation of the negative gradient $-\boldsymbol{g}$ is reduced because consecutive gradients which point in opposite directions are superimposed. The minimum is found faster.

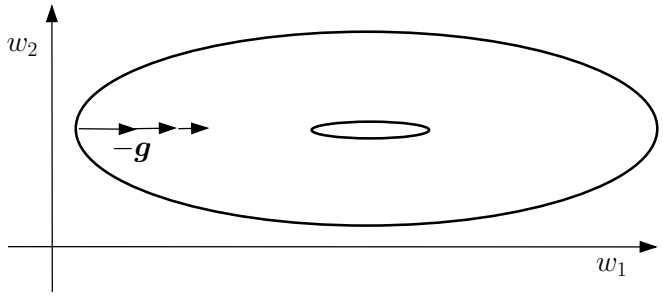

Figure 7.5: The negative gradient $-\boldsymbol{g}$ lets the weight vector converge very slowly to the minimum if the region around the minimum is flat.

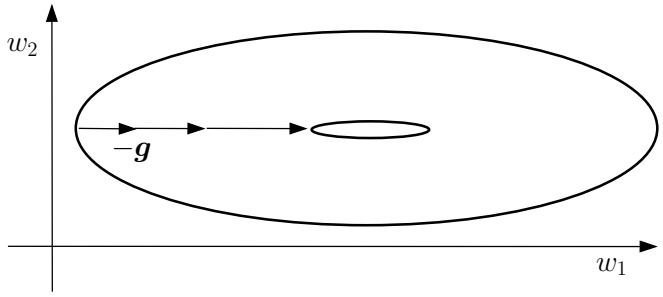

Figure 7.6: The negative gradient $-\boldsymbol{g}$ is accumulated through the momentum term because consecutive gradients point into the same directions. The minimum is found faster.

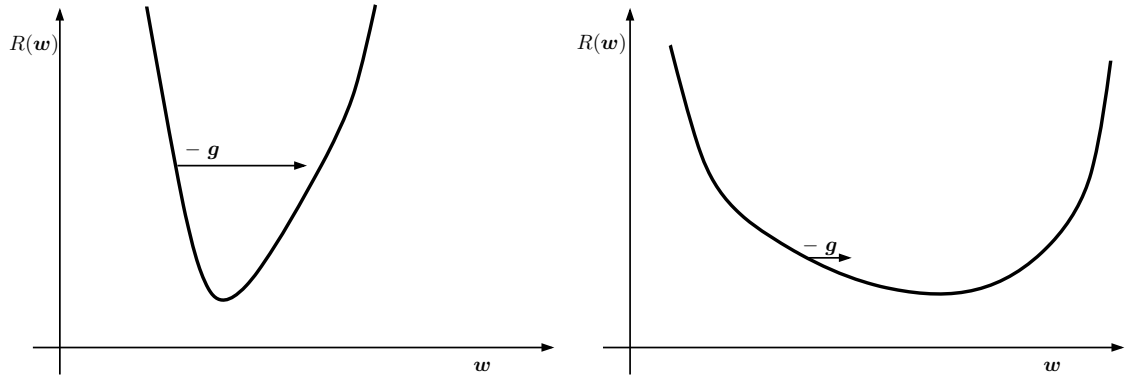

Figure 7.7: Left: the length of the negative gradient $-\boldsymbol{g}$ is large because of the steep descent. However a small gradient step would be optimal because of the high curvature at the minimum. Right: the length of the negative gradient $-\boldsymbol{g}$ is small because of the flat descent. However a large gradient step could be performed because the low curvature ensures that the minimum is not jumped over.

7.1.3 Step-size Optimization

Instead of choosing a fixed step-size the step-size should be adjusted to the curvature of the error surface. For a flat curve the step-size could be large whereas for high curvature and steep minima the step-size should be small. In Fig. 7.7 the

7.1.3.1 Heuristics

Learning rate adjustments. The learning rate is adjusted: if the change of the risk

$$\Delta R \;=\; R(\boldsymbol{w} \,+\, \Delta\boldsymbol{w}) \;-\; R(\boldsymbol{w}) \tag{7.8}$$

is negative then the learning rate is increased and otherwise decreased:

$$\eta^{\text{new}} \;=\; \begin{cases} \rho\,\eta^{\text{old}} & \text{if} \quad \Delta R \;\leq\; 0 \\ \sigma\,\eta^{\text{old}} & \text{if} \quad \Delta R \;>\; 0 \end{cases} \;, \tag{7.9}$$

where $\rho > 1$ and $\sigma < 1$, e.g. $\rho = 1.1$ and $\sigma = 0.5$. Here ρ is only slightly larger than one but sigma is much smaller than one in order to reduce too large learning rates immediately.

Largest eigenvalue of the Hessian. Later in eq. (7.76) we will see that the largest eigenvalue $\lambda_{\max}$ of the Hessian, $\boldsymbol{H}$ given as

$$H_{ij} \;=\; \frac{\partial^2 R(\boldsymbol{w})}{\partial w_i\,\partial w_j} \;, \tag{7.10}$$

bounds the learning rate:

$$\eta \;\leq\; \frac{2}{\lambda_{\max}} \;. \tag{7.11}$$

The idea is to efficiently infer the largest eigenvalue of the Hessian in order to find a maximal learning rate. The maximal eigenvalue of the Hessian can be determined by matrix iteration. Let $(\boldsymbol{e}_1, \ldots, \boldsymbol{e}_W)$ be the from largest to smallest sorted eigenvectors of the Hessian with according eigenvalues $(\lambda_1, \ldots, \lambda_W)$ and let $\boldsymbol{a} \;=\; \sum_{i=1}^{W} \alpha_i \, \boldsymbol{e}_i$, then

$$\boldsymbol{H}^s \, \boldsymbol{a} \;=\; \sum_{i=1}^{W} \lambda_i^s \, \alpha_i \, \boldsymbol{e}_i \;\approx\; \lambda_1^s \, \alpha_1 \, \boldsymbol{e}_1 \; . \tag{7.12}$$

Normalizing $\boldsymbol{H}^s \, \boldsymbol{a}$ and multiplication with $\boldsymbol{H}$ gives both a hint how well $\boldsymbol{e}_1$ is approximated and how large λ_1 may be.

The method depends on how time intensive it is to obtain the Hessian or the product of the Hessian with a vector. How the Hessian is computed depends on the model. For quadratic problems the Hessian is directly given and for neural networks there exist different methods to approximate the Hessian or to compute the product of the Hessian with a vector.

Individual learning rate for each parameter.

We have for parameter w_i

$$\Delta w_i \;=\; -\,\eta_i \, g_i \; , \tag{7.13}$$

where g_i is the i-th component of the gradient $\boldsymbol{g}$: $g_i \;=\; \left[\frac{\partial R(\boldsymbol{w})}{\partial \boldsymbol{w}}\right]_i$. If the parameters would be independent from each other, then individual learning rates are justified. However for dependent parameter, as long as all $\eta_i > 0$ the error decreases.

The *delta-delta rule* adjusts the local learning parameter according to

$$\Delta \eta_i \;=\; \gamma \, g_i^{\text{new}} \, g_i^{\text{old}} \; . \tag{7.14}$$

The delta-delta rule determines whether the local gradient information changes. If it changes then the learning rate is decreased otherwise it is increased. In a flat plateau the learning rate is increased and in steep regions the learning rate is decreased.

This rule was improved to the *delta-bar-delta rule*:

$$\Delta \eta_i \;=\; \begin{cases} \kappa & \text{if} \quad \bar{g}_i^{\text{old}} \, g_i^{\text{new}} \;>\; 0 \\ -\,\phi \, g_i^{\text{new}} & \text{if} \quad \bar{g}_i^{\text{old}} \, g_i^{\text{new}} \;\leq\; 0 \end{cases} \; , \tag{7.15}$$

where

$$\bar{g}_i^{\text{new}} \;=\; (1 \;-\; \theta) \, g_i^{\text{new}} \;+\; \theta \, \bar{g}_i^{\text{old}} \; . \tag{7.16}$$

$\bar{g}_i$ is an exponentially weighted average of the values of g_i. That means instead of the old gradient information an average is used to determine whether the local gradient directions changes.

Big disadvantage of the delta-bar-delta rule that it has many hyper-parameters: θ, κ, ϕ and μ if a momentum term is included as well.

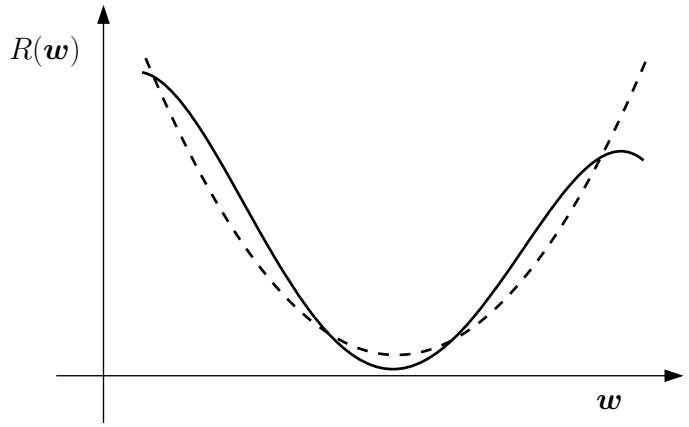

Figure 7.8: The error surface (the solid curve) is locally approximated by a quadratic function (the dashed curve).

Quickprop. This method was developed in the context of neural networks, therefore the name "quickprop" which reminds on the name "back-propagation", the most popular method for computing the gradient in neural networks.

Here again the parameters are treated individually. The *quickprop* learning rule is

$$\Delta^{\text{new}} w_i \;=\; \frac{g_i^{\text{new}}}{g_i^{\text{old}} - g_i^{\text{new}}}\, \Delta^{\text{old}} w_i \;. \tag{7.17}$$

Let us assume that $R(\boldsymbol{w})$ is a function of one variable $R(w_i)$ then the Taylor expansion is

$$\begin{aligned} R(w_i \,+\, \Delta w_i) \;&=\; R(w_i) \,+\, \frac{\partial R(\boldsymbol{w})}{\partial w_i}\, \Delta w_i \,+ \\ \frac{1}{2}\, \frac{\partial^2 R(\boldsymbol{w})}{(\partial w_i)^2}\, (\Delta w_i)^2 \,+\, O\left((\Delta w_i)^3\right) \;&=\; \\ R(w_i) \,+\, g_i\, \Delta w_i \,+\, \frac{1}{2}\, g_i'\, (\Delta w_i)^2 \,+\, O\left((\Delta w_i)^3\right) \;. \end{aligned} \tag{7.18}$$

That is a quadratic approximation of the function. See Fig. 7.8 for the quadratic approximation of the error function $R(\boldsymbol{w})$.

To minimize $R(w_i \,+\, \Delta w_i) \,-\, R(w_i)$ with respect to Δw_i we set the derivative of the right hand side to zero and obtain

$$\Delta w_i \;=\; -\,\frac{g_i}{g_i'}\;. \tag{7.19}$$

Now approximate $g_i' \;=\; g_i'(w_i)$ by $g_i'(w_i^{\text{old}})$ and use a difference quotient:

$$g_i' \;=\; \frac{g_i^{\text{new}} \,-\, g_i^{\text{old}}}{\Delta^{\text{old}} w_i}\;, \tag{7.20}$$

where $g_i^{\text{old}} \;=\; g_i(w_i \,-\, \Delta^{\text{old}} w_i)$. We now insert this approximation into eq. (7.19) which results in the quickprop update rule.

7.1.3.2 Line Search

Let us assume we found the update direction $\boldsymbol{d}$ either as the negative gradient or by also taking into account the curvature through the Hessian or its approximation.

The update rule is

$$\Delta \boldsymbol{w} \; = \eta \, \boldsymbol{d} \, . \tag{7.21}$$

We now want to find the value of η which minimizes

$$R(\boldsymbol{w} \; + \; \eta \, \boldsymbol{d}) \, . \tag{7.22}$$

For quadratic functions and $\boldsymbol{d} \; = \; - \, \boldsymbol{g}$ and very close to the minimum $\boldsymbol{w}^*$, η is given later in eq. (7.77) as

$$\eta \; = \; \frac{\boldsymbol{g}^T \, \boldsymbol{g}}{\boldsymbol{g}^T \, \boldsymbol{H}(\boldsymbol{w}^*) \, \boldsymbol{g}} \, . \tag{7.23}$$

However this is only valid near the minimum. Further we do not know $\boldsymbol{H}(\boldsymbol{w}^*)$.

Finding the best update step could be viewed as a separate task. If we assume that at the minimum the function $R(\boldsymbol{w})$ is convex then we can apply *line search*.

Line search fits first a parable though three points and determines its minimum. The point with largest value of R is discharged. The line search algorithm is given in Alg. 7.1.

Algorithm 7.1 Line Search

BEGIN initialization $a_0, b_0, c_0; R(a_0) > R(c_0); R(b_0) > R(c_0)$, Stop=false, $i = 0$
END initialization
BEGIN line search
  while STOP=false **do**
    fit quadratic function through a_i, b_i, c_i
    determine minimum d_i of quadratic function
    if stop criterion fulfilled, e.g. $|a_i - b_i| < \epsilon$ or $|R(b_0) - R(c_0)| < \epsilon$ **then**
      Stop=true
    else
      $c_{i+1} \; = \; d_i$
      $b_{i+1} \; = \; c_i$
      $a_{i+1} \; = \; \begin{cases} a_i & \text{if} \quad R(a_i) \; \leq \; R(b_i) \\ b_i & \text{if} \quad R(a_i) \; > \; R(b_i) \end{cases}$
    end if
    $i = i + 1$
  end while
END line search

Fig. 7.9 shows the line search procedure. At each step the length of the interval $[a, b]$ or $[b, a]$ is decreased.

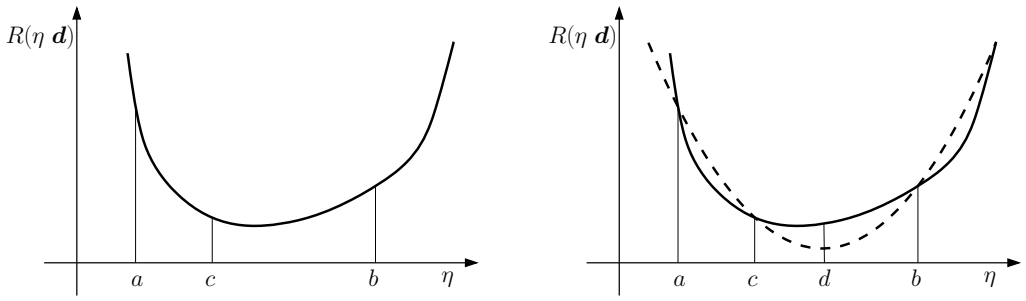

Figure 7.9: Line search. Left: the error function $R(\eta \boldsymbol{d})$ in search direction $\boldsymbol{d}$ is depicted as solid curve. The scalars a, b, c are given with $R(a\ \boldsymbol{d}) > R(c\boldsymbol{d})$ and $R(b\ \boldsymbol{d}) > R(c\ \boldsymbol{d})$. Right: The dashed curve is a quadratic approximation (parabola) of the error function by fitting the parabola through a, b, and c. The minimum of the parabola is at d. For the next step we set $a = b$, $b = c$, and $c = d$.

Line search starts with $R(\boldsymbol{w}), R(\boldsymbol{w} + \eta_0 \boldsymbol{d})$ and $R(\boldsymbol{w} + \eta_{\max}\ \boldsymbol{d})$, where $a_0, b_0, c_0 \in \{\eta_0, 0, \eta_{\max}\}$. If a large range of η values are possible, then the search can be on a logarithmic scale as preprocessing.

7.1.4 Optimization of the Update Direction

The default direction is the negative gradient $-\ \boldsymbol{g}$. However there are even better approaches.

7.1.4.1 Newton and Quasi-Newton Method

The gradient vanishes because it is a minimum $\boldsymbol{w}^*$: $\nabla_{\boldsymbol{w}} R(\boldsymbol{w}^*) = \boldsymbol{g}(\boldsymbol{w}^*) = \boldsymbol{0}$. Therefore we obtain for the Taylor series around $\boldsymbol{w}^*$:

$$R(\boldsymbol{w}) = R(\boldsymbol{w}^*) + \frac{1}{2} (\boldsymbol{w} - \boldsymbol{w}^*)^T \boldsymbol{H}(\boldsymbol{w}^*) (\boldsymbol{w} - \boldsymbol{w}^*) + O\left(\|\boldsymbol{w} - \boldsymbol{w}^*\|^3\right) . \tag{7.24}$$

The gradient $\boldsymbol{g} = \boldsymbol{g}(\boldsymbol{w})$ of the quadratic approximation of $\mathbb{R}(\boldsymbol{w})$ is

$$\boldsymbol{g} = \boldsymbol{H}(\boldsymbol{w}^*) (\boldsymbol{w} - \boldsymbol{w}^*) . \tag{7.25}$$

Solving this equation for $\boldsymbol{w}^*$ gives

$$\boldsymbol{w}^* = \boldsymbol{w} - \boldsymbol{H}^{-1} \boldsymbol{g} . \tag{7.26}$$

The update direction $\boldsymbol{H}^{-1}\boldsymbol{g}$ is the *Newton direction*. The Newton direction is depicted in Fig. 7.10 for a quadratic error surface.

Disadvantages of the Newton direction are

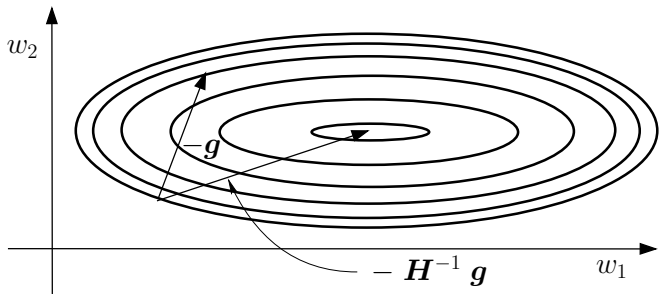

Figure 7.10: The Newton direction $-\boldsymbol{H}^{-1}\,\boldsymbol{g}$ for a quadratic error surface in contrast to the gradient direction $-\boldsymbol{g}$. The Newton direction points directly to the minimum and one Newton update step would find the minimum.

- that it is computationally expensive because computing the Hessian is expensive and its inversion needs $O(W^3)$ steps;
- that it only works near the minimum if the Hessian is positive definite.

A remedy for the later disadvantage is the *model trust region* approach, where the model is only trusted up to a certain value. A positive definite matrix is added to the Hessian:

$$\boldsymbol{H} \;+\; \lambda \boldsymbol{I}\;. \tag{7.27}$$

The update step is a compromise between gradient direction (large λ) and Newton direction ($lambda \;=\; 0$).

To address the problem of the expensive inversion of the Hessian, it can be approximated by a diagonal matrix. The diagonal matrix is simple to invert.

Quasi-Newton Method.

From the Newton equation eq. (7.26) two weight vectors $\boldsymbol{w}^{\mathrm{old}}$ and $\boldsymbol{w}^{\mathrm{new}}$ are related by

$$\boldsymbol{w}^{\mathrm{new}} \;-\; \boldsymbol{w}^{\mathrm{old}} \;=\; -\,\boldsymbol{H}^{-1}\left(\boldsymbol{g}^{\mathrm{new}} \;-\; \boldsymbol{g}^{\mathrm{old}}\right)\;. \tag{7.28}$$

This is the quasi-Newton condition.

The best known method is the *Broyden-Fletcher-Goldfarb-Shanno* (BFGS) method. The BFGS update is

$$\boldsymbol{w}^{\mathrm{new}} \;=\; \boldsymbol{w}^{\mathrm{old}} \;+\; \eta\,\boldsymbol{G}^{\mathrm{old}}\,\boldsymbol{g}^{\mathrm{old}}\;, \tag{7.29}$$

where η is found by line search.

The function $\boldsymbol{G}$ is an approximation of the inverse Hessian and is computed as follows:

$$\boldsymbol{G}^{\mathrm{new}} \;=\; \boldsymbol{G}^{\mathrm{old}} \;+\; \frac{\boldsymbol{p}\,\boldsymbol{p}^T}{\boldsymbol{p}^T\,\boldsymbol{v}} \;- \tag{7.30}$$
$$\frac{\left(\boldsymbol{G}^{\mathrm{old}}\,\boldsymbol{v}\right)\;\boldsymbol{v}^T\boldsymbol{G}^{\mathrm{old}}}{\boldsymbol{v}^T\boldsymbol{G}^{\mathrm{old}}\,\boldsymbol{v}} \;+\; \left(\boldsymbol{v}^T\boldsymbol{G}^{\mathrm{old}}\,\boldsymbol{v}\right)\;\boldsymbol{u}\,\boldsymbol{u}^T\;,$$

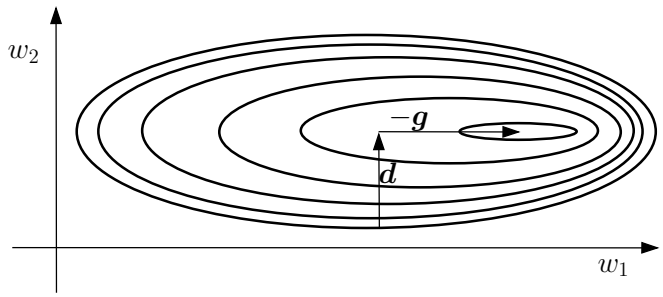

Figure 7.11: Conjugate gradient. After line search in search direction $\boldsymbol{d}$ the new gradient is orthogonal to the line search direction.

where

$$\boldsymbol{p} \;=\; \boldsymbol{w}^{\mathrm{new}} \;-\; \boldsymbol{w}^{\mathrm{old}} \tag{7.31}$$

$$\boldsymbol{v} \;=\; \boldsymbol{g}^{\mathrm{new}} \;-\; \boldsymbol{g}^{\mathrm{old}} \tag{7.32}$$

$$\boldsymbol{u} \;=\; \frac{\boldsymbol{p}}{\boldsymbol{p}^T\boldsymbol{v}} \;-\; \frac{G^{\mathrm{old}}\boldsymbol{v}}{\boldsymbol{v}^T\boldsymbol{G}^{\mathrm{old}}\,\boldsymbol{v}}\,. \tag{7.33}$$

Initialization of $\boldsymbol{G}$ can be done by $\boldsymbol{I}$, the identity matrix (only a gradient step as first step).

7.1.4.2 Conjugate Gradient

Note that for the line search algorithm we optimized η so that

$$R(\boldsymbol{w} \;+\; \eta\,\boldsymbol{d}) \tag{7.34}$$

is minimized.

The minimum condition is

$$\frac{\partial}{\partial\eta}\,R(\boldsymbol{w} \;+\; \eta\,\boldsymbol{d}) \;=\; 0\,, \tag{7.35}$$

which gives

$$(\boldsymbol{g}^{\mathrm{new}})^T\;\boldsymbol{d}^{\mathrm{old}} \;=\; 0\,. \tag{7.36}$$

The gradient of the new minimum $\boldsymbol{g}^{\mathrm{new}}$ is orthogonal to the previous search direction $\boldsymbol{d}^{\mathrm{old}}$. This fact is depicted in Fig. 7.11.

However still oscillations are possible. The oscillations appear especially in higher dimensions, where oscillations like in Fig. 7.3 can be present in different two-dimensional subspaces which alternate. Desirable to avoid oscillations would be that a new search directions are orthogonal to all previous search directions where orthogonal is defined via the Hessian. The later means that in the parameter space not all directions are equal important. In the following we construct such search directions.

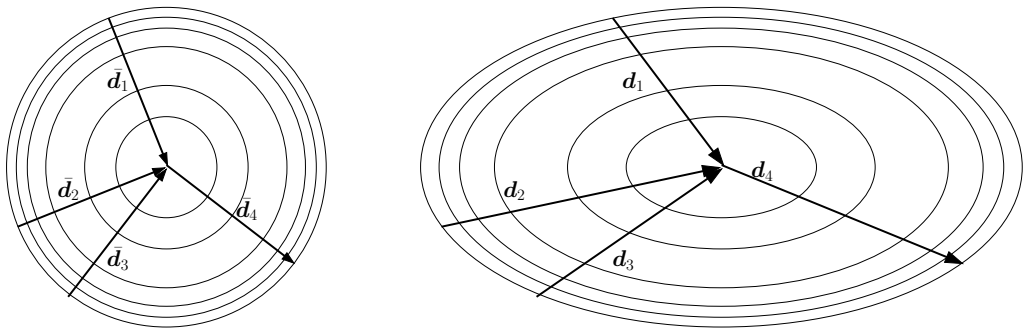

Figure 7.12: Conjugate gradient. Left: $\bar{\boldsymbol{d}}_1^T\ \bar{\boldsymbol{d}}_2\ =\ 0$ and $\bar{\boldsymbol{d}}_3^T\ \bar{\boldsymbol{d}}_4\ =\ 0$. Right: $\boldsymbol{d}_1^T\ \boldsymbol{H}\ \boldsymbol{d}_2\ =\ 0$ and $\boldsymbol{d}_3^T\ \boldsymbol{H}\ \boldsymbol{d}_4\ =\ 0$. Conjugate directions can be viewed as orthogonal directions which were transformed by a change of the coordinate system: $\boldsymbol{d}\ =\ \boldsymbol{H}^{-1/2}\bar{\boldsymbol{d}}$.

The oscillations can be avoided if we enforce not only

$$\boldsymbol{g}(\boldsymbol{w}^{\text{new}})^T\ \boldsymbol{d}^{\text{old}}\ =\ 0 \tag{7.37}$$

but also the same condition for the new gradient

$$\boldsymbol{g}(\boldsymbol{w}^{\text{new}}\ +\ \eta\boldsymbol{d}^{\text{new}})^T\ \boldsymbol{d}^{\text{old}}\ =\ 0\,. \tag{7.38}$$

Taylor expansion of $\boldsymbol{g}(\boldsymbol{w}^{\text{new}}\ +\ \eta\boldsymbol{d}^{\text{new}})$ with respect to η around 0 gives

$$\boldsymbol{g}(\boldsymbol{w}^{\text{new}}\ +\ \eta\boldsymbol{d}^{\text{new}})^T\ =\ \boldsymbol{g}(\boldsymbol{w}^{\text{new}})\ +\ \eta\boldsymbol{H}(\boldsymbol{w}^{\text{new}})\boldsymbol{d}^{\text{new}}\ +\ O(\eta^2)\,. \tag{7.39}$$

We insert that into eq. (7.38) and apply eq. (7.37) and divide through η and obtain

$$(\boldsymbol{d}^{\text{new}})^T\ \boldsymbol{H}(\boldsymbol{w}^{\text{new}})\ \boldsymbol{d}^{\text{old}}\ =\ 0\,. \tag{7.40}$$

Directions which satisfy eq. (7.40) are said to be *conjugate*. See Fig. 7.12 for conjugate directions.

Let us assume a quadratic problem

$$R(\boldsymbol{w})\ =\ \frac{1}{2}\,\boldsymbol{w}^T\boldsymbol{H}\boldsymbol{w}\ +\ \boldsymbol{c}^T\boldsymbol{w}\ +\ k \tag{7.41}$$

with

$$\boldsymbol{g}(\boldsymbol{w})\ =\ \boldsymbol{H}\boldsymbol{w}\ +\ \boldsymbol{c} \tag{7.42}$$

and

$$\boldsymbol{0}\ =\ \boldsymbol{H}\boldsymbol{w}^*\ +\ \boldsymbol{c}\,. \tag{7.43}$$

We want to have W conjugate directions, i.e.

$$\forall_{i \neq j} : \; \boldsymbol{d}_j^T \boldsymbol{H} \, \boldsymbol{d}_i \; = \; 0 \; , \tag{7.44}$$

which are linearly independent, therefore we can set

$$\boldsymbol{w}^* \; - \; \boldsymbol{w}_1 \; = \; \sum_{i=1}^{W} \eta_i \, \boldsymbol{d}_i \tag{7.45}$$

$$\tag{7.46}$$

and

$$\boldsymbol{w}_j \; - \; \boldsymbol{w}_1 \; = \; \sum_{i=1}^{j-1} \eta_i \, \boldsymbol{d}_i \; . \tag{7.47}$$

In the last but one equation $\boldsymbol{w}^* \; = \; - \, \boldsymbol{H}^{-1} \, \boldsymbol{c}$ is multiplied by $\boldsymbol{d}_j^T \boldsymbol{H}$ and we obtain

$$- \, \boldsymbol{d}_j^T \, (\boldsymbol{c} \; + \; \boldsymbol{H} \boldsymbol{w}_1) \; = \; \sum_{i=1}^{W} \eta_i \, \boldsymbol{d}_j^T \boldsymbol{H} \, \boldsymbol{d}_i \; = \; \eta_j \, \boldsymbol{d}_j^T \boldsymbol{H} \, \boldsymbol{d}_j \; . \tag{7.48}$$

From eq. (7.47) we obtain

$$\boldsymbol{d}_j^T \boldsymbol{H} \, \boldsymbol{w}_j \; = \; \boldsymbol{d}_j^T \boldsymbol{H} \, \boldsymbol{w}_1 \; . \tag{7.49}$$

Using $\boldsymbol{g}_j \; = \; \boldsymbol{c} \; + \; \boldsymbol{H} \boldsymbol{w}_j$, η_j can be determined as

$$\eta_j \; = \; - \, \frac{\boldsymbol{d}_j^T \boldsymbol{g}_j}{\boldsymbol{d}_j^T \boldsymbol{H} \, \boldsymbol{d}_j} \; . \tag{7.50}$$

Because of

$$\boldsymbol{w}_{j+1} \; = \; \boldsymbol{w}_j \; + \; \eta_j \, \boldsymbol{d}_j \; . \tag{7.51}$$

we have determined the learning rate η_j.

Now we have found the search directions $\boldsymbol{d}_j$. We set

$$\boldsymbol{d}_{j+1} \; = \; - \, \boldsymbol{g}_{j+1} \; + \; \beta_j \, \boldsymbol{d}_j \; . \tag{7.52}$$

Multiplying by $\boldsymbol{d}_j^T \boldsymbol{H}$ gives

$$\beta_j \; = \; \frac{\boldsymbol{g}_{j+1}^T \boldsymbol{H} \, \boldsymbol{d}_j}{\boldsymbol{d}_j^T \boldsymbol{H} \, \boldsymbol{d}_j} \; . \tag{7.53}$$

Since

$$\boldsymbol{g}_{j+1} \, - \, \boldsymbol{g}_j \, = \, \boldsymbol{H} \, \left(\boldsymbol{w}_{j+1} \, - \, \boldsymbol{w}_j \right) \, = \, \eta_j \, \boldsymbol{H} \, \boldsymbol{d}_j \; , \tag{7.54}$$

we can rewrite eq. (7.53) as

$$\mathbf{Hestenes - Stiefel} : \tag{7.55}$$

$$\beta_j \, = \, \frac{\boldsymbol{g}_{j+1}^T \left(\boldsymbol{g}_{j+1} \, - \, \boldsymbol{g}_j \right)}{\boldsymbol{d}_j^T \, \left(\boldsymbol{g}_{j+1} \, - \, \boldsymbol{g}_j \right)} \; . \tag{7.56}$$

Multiplying eq. (7.52) by $\boldsymbol{g}_{j+1}^T$ and using the conditions $\boldsymbol{d}_k^T \, \boldsymbol{g}_j \, = \, 0$ for $k < j$ gives

$$\boldsymbol{d}_j^T \, \boldsymbol{g}_j \, = \, - \, \boldsymbol{g}_j^T \, \boldsymbol{g}_j \; . \tag{7.57}$$

The equation for β_j can be rewritten as

$$\mathbf{Polak - Ribiere} : \tag{7.58}$$

$$\beta_j \, = \, \frac{\boldsymbol{g}_{j+1}^T \left(\boldsymbol{g}_{j+1} \, - \, \boldsymbol{g}_j \right)}{\boldsymbol{g}_j^T \, \boldsymbol{g}_j} \; . \tag{7.59}$$

Similar to the previous reformulation this expression can be simplified to

$$\mathbf{Fletcher - Reeves} : \tag{7.60}$$

$$\beta_j \, = \, \frac{\boldsymbol{g}_{j+1}^T \boldsymbol{g}_{j+1}}{\boldsymbol{g}_j^T \, \boldsymbol{g}_j} \; . \tag{7.61}$$

Even if the equations eq. (7.55), eq. (7.58), and eq. (7.60) are mathematically equivalent, there are numerical differences. The Polak-Ribiere equation eq. (7.58) has an edge over the other update rules.

The computation of the values η_j need the Hessian, therefore the η_j are in most implementations found by line search.

The disadvantage of conjugate gradient compared to the quasi-Newton methods is

- the line search must be done precisely in order to obtain the conjugate and orthogonal gradients

The advantage of conjugate gradient compared to the quasi-Newton methods is

- that the storage is $O(W)$ compared to $O(W^2)$ for quasi-Newton

Algorithm 7.2 Conjugate Gradient (Polak-Ribiere)

BEGIN initialization $\boldsymbol{g}_0 = \nabla_{\boldsymbol{w}} R(\boldsymbol{w}_0)$, $j = 0$, $d_0 = -\boldsymbol{g}_0$, Stop=false
END initialization
BEGIN Conjugate Gradient
  while STOP=false **do**
    determine η_j by line search
    $\boldsymbol{w}_{j+1} = \boldsymbol{w}_j + \eta_j \boldsymbol{d}_j$
    $\boldsymbol{g}_{j+1} = \nabla_{\boldsymbol{w}} R(\boldsymbol{w}_{j+1})$
    $\beta_j = \frac{\boldsymbol{g}_{j+1}^T (\boldsymbol{g}_{j+1} - \boldsymbol{g}_j)}{\boldsymbol{g}_j^T \boldsymbol{g}_j}$
    $\boldsymbol{d}_{j+1} = -\boldsymbol{g}_{j+1} + \beta_j \boldsymbol{d}_j$
    if stop criterion fulfilled, e.g. $\|g_{j+1}\|| < \epsilon$ or $|R(\boldsymbol{w}_{j+1}) - R(\boldsymbol{w}_j)| < \epsilon$ **then**
      STOP=true
    end if
    j = j + 1
  end while
END Conjugate Gradient

7.1.5 Levenberg-Marquardt Algorithm

This algorithm is designed for quadratic loss, i.e. for the mean squared error

$$R(\boldsymbol{w}) = \sum_{i=1}^{l} \left(e^i(\boldsymbol{w})\right)^2 . \tag{7.62}$$

We combine the errors e^i into a vector $\boldsymbol{e}$. The Jacobian of this vector is $\boldsymbol{J}$ defined as

$$J_{ij} = \frac{\partial e^i}{\partial w_j} . \tag{7.63}$$

The linear approximation of the error vector gives

$$\boldsymbol{e}(\boldsymbol{w}^{\text{new}}) = \boldsymbol{e}(\boldsymbol{w}^{\text{old}}) + \boldsymbol{J}\left(\boldsymbol{w}^{\text{new}} - \boldsymbol{w}^{\text{old}}\right) . \tag{7.64}$$

The Hessian of the loss function $R(\boldsymbol{w})$ is

$$H_{jk} = \frac{\partial^2 R}{\partial w_j \, \partial w_k} = 2\sum_{i=1}^{l}\left(\frac{\partial e^i}{\partial w_j}\frac{\partial e^i}{\partial w_k} + e^i \frac{\partial^2 e^i}{\partial w_j \, \partial w_k}\right) . \tag{7.65}$$

If we assume small e^i (if we are close to the targets) or if we assume that the term $e^i \frac{\partial^2 e^i}{\partial w_j \, \partial w_k}$ averages out, then we can approximate the Hessian by

$$\boldsymbol{H} = \boldsymbol{J}^T \boldsymbol{J} . \tag{7.66}$$

Note that this "outer product approximation" is only valid for quadratic loss functions.

We now can formulate an unconstrained minimization problem where

$$\frac{1}{2}\left\|\boldsymbol{e}(\boldsymbol{w}^{\text{old}}) \;+\; \boldsymbol{J}\left(\boldsymbol{w}^{\text{new}} \;-\; \boldsymbol{w}^{\text{old}}\right)\right\|^2 \;+\; \lambda\;\left\|\boldsymbol{w}^{\text{new}} \;-\; \boldsymbol{w}^{\text{old}}\right\|^2 \tag{7.67}$$

has to be minimized. The first term accounts for minimizing the error and the second term for minimizing the step size.

The solution is the *Levenberg-Marquardt* update rule

$$\boldsymbol{w}^{\text{new}} \;=\; \boldsymbol{w}^{\text{old}} \;-\; \left(\boldsymbol{J}^T\boldsymbol{J} \;+\; \lambda\,\boldsymbol{I}\right)^{-1}\;\boldsymbol{J}^T\,\boldsymbol{e}(\boldsymbol{w}^{\text{old}})\,. \tag{7.68}$$

Small λ gives the Newton formula while large λ gives gradient descent.

The Levenberg-Marquardt algorithm is a model trust region approach.

7.1.6 Predictor Corrector Methods

The problem to solve is $R(\boldsymbol{w}) \;=\; 0$. That problem is different from minimization problems.

The idea is to make a simple update (predictor step) by neglecting higher order terms. In a second step (corrector step) the value is corrected by involving the higher order terms. Here the higher order terms are evaluated with the solution obtained by the predictor step.

The Taylor series of $R(\boldsymbol{w}^{\text{new}})$ is

$$R(\boldsymbol{w}^{\text{new}}) \;=\; R(\boldsymbol{w}^{\text{old}}) \;+\; S(\boldsymbol{w}^{\text{old}}, \Delta\boldsymbol{w}) \;+\; T(\boldsymbol{w}^{\text{old}}, \Delta\boldsymbol{w})\,. \tag{7.69}$$

Here $S(\boldsymbol{w}^{\text{old}}, \boldsymbol{0}) \;=\; 0$ and $T(\boldsymbol{w}^{\text{old}}, \boldsymbol{0}) \;=\; 0$.

In the predictor step solve

$$R(\boldsymbol{w}^{\text{old}}) \;+\; S(\boldsymbol{w}^{\text{old}}, \Delta\boldsymbol{w}) \;=\; 0 \tag{7.70}$$

with respect to $\Delta\boldsymbol{w}$ which gives $\Delta_{\text{pred}}\boldsymbol{w}$. In the corrector step solve

$$R(\boldsymbol{w}^{\text{old}}) \;+\; S(\boldsymbol{w}^{\text{old}}, \Delta\boldsymbol{w}) \;+\; T(\boldsymbol{w}^{\text{old}}, \Delta_{\text{pred}}\boldsymbol{w}) \;=\; 0\,, \tag{7.71}$$

which gives the final update $\Delta\boldsymbol{w}$.

The predictor-corrector update can be formulated as an iterative algorithm.

7.1.7 Convergence Properties

Gradient Descent. We make a Taylor series expansion of the gradient function g locally at the minimum $\boldsymbol{w}^*$. The gradient vanishes because it is a minimum $\nabla_{\boldsymbol{w}} R(\boldsymbol{w}^*) = \boldsymbol{g}(\boldsymbol{w}^*) = \boldsymbol{0}$, therefore we obtain

$$R(\boldsymbol{w}) = R(\boldsymbol{w}^*) + \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}^*)^T \boldsymbol{H}(\boldsymbol{w}^*)(\boldsymbol{w} - \boldsymbol{w}^*) + O\left(\|\boldsymbol{w} - \boldsymbol{w}^*\|^3\right). \tag{7.72}$$

The improvement of the risk is

$$\begin{aligned} &R(\boldsymbol{w} - \eta\,\boldsymbol{g}) - R(\boldsymbol{w}^*) = \\ &\frac{1}{2}(\boldsymbol{w} - \eta\,\boldsymbol{g} - \boldsymbol{w}^*)^T \boldsymbol{H}(\boldsymbol{w}^*)(\boldsymbol{w} - \eta\,\boldsymbol{g} - \boldsymbol{w}^*) = \\ &\frac{1}{2}\eta^2\,\boldsymbol{g}^T \boldsymbol{H}(\boldsymbol{w}^*)\,\boldsymbol{g} - \eta\,\boldsymbol{g}^T\,\boldsymbol{g} + c\,, \end{aligned} \tag{7.73}$$

where c is independent of η.

First we note that

$$\frac{1}{2}\eta^2\,\boldsymbol{g}^T \boldsymbol{H}(\boldsymbol{w}^*)\,\boldsymbol{g} - \eta\,\boldsymbol{g}^T\,\boldsymbol{g} \leq 0\,, \tag{7.74}$$

to ensure improvement for arbitrary c. That is equivalent to

$$\eta \leq \frac{2\,\boldsymbol{g}^T\,\boldsymbol{g}}{\boldsymbol{g}^T \boldsymbol{H}(\boldsymbol{w}^*)\,\boldsymbol{g}}\,, \tag{7.75}$$

which gives the bound (the $\boldsymbol{g}$ which gives the smallest value)

$$\eta \leq \frac{2}{\lambda_{\max}}\,. \tag{7.76}$$

The optimal update is obtained if we minimize the left hand side of eq. (7.74) with respect to η. Setting the derivative with respect to η to zero and solving for η gives

$$\eta = \frac{\boldsymbol{g}^T\,\boldsymbol{g}}{\boldsymbol{g}^T \boldsymbol{H}(\boldsymbol{w}^*)\,\boldsymbol{g}}\,. \tag{7.77}$$

The update is

$$\boldsymbol{w}^{\text{new}} = \boldsymbol{w}^{\text{old}} - \frac{\boldsymbol{g}^T\,\boldsymbol{g}}{\boldsymbol{g}^T \boldsymbol{H}(\boldsymbol{w}^*)\,\boldsymbol{g}}\,\boldsymbol{g}\,. \tag{7.78}$$

The improvement is

$$
\begin{aligned}
& R\left(\boldsymbol{w}^{\text{old}}\right) \; - \; R\left(\boldsymbol{w}^{\text{new}}\right) \; = \\
& \left(\boldsymbol{w}^{\text{old}} \; - \; \boldsymbol{w}^{*}\right) \; \boldsymbol{H}(\boldsymbol{w}^{*}) \; \frac{\boldsymbol{g}^{T} \, \boldsymbol{g}}{\boldsymbol{g}^{T} \, \boldsymbol{H}(\boldsymbol{w}^{*}) \, \boldsymbol{g}} \boldsymbol{g} \; - \\
& \frac{1}{2} \; \left(\frac{\boldsymbol{g}^{T} \, \boldsymbol{g}}{\boldsymbol{g}^{T} \, \boldsymbol{H}(\boldsymbol{w}^{*}) \, \boldsymbol{g}}\right)^{2} \; \boldsymbol{g}^{T} \, \boldsymbol{H}(\boldsymbol{w}^{*}) \, \boldsymbol{g} \; = \\
& \frac{1}{2} \; \frac{\left(\boldsymbol{g}^{T} \, \boldsymbol{g}\right)^{2}}{\boldsymbol{g}^{T} \, \boldsymbol{H}(\boldsymbol{w}^{*}) \, \boldsymbol{g}} \; = \; R\left(\boldsymbol{w}^{\text{old}}\right) \left(\frac{\left(\boldsymbol{g}^{T} \, \boldsymbol{g}\right)^{2}}{\left(\boldsymbol{g}^{T} \, \boldsymbol{H}(\boldsymbol{w}^{*}) \, \boldsymbol{g}\right) \; \left(\boldsymbol{g}^{T} \, \boldsymbol{H}^{-1}(\boldsymbol{w}^{*}) \, \boldsymbol{g}\right)}\right) \; .
\end{aligned}
\tag{7.79}
$$

The Kantorovich inequality states that

$$
\frac{\left(\boldsymbol{g}^{T} \, \boldsymbol{g}\right)^{2}}{\left(\boldsymbol{g}^{T} \, \boldsymbol{H} \, \boldsymbol{g}\right) \; \left(\boldsymbol{g}^{T} \, \boldsymbol{H}^{-1} \, \boldsymbol{g}\right)} \; \geq \; \frac{4 \; \lambda_{\min} \; \lambda_{\max}}{\left(\lambda_{\min} \; + \; \lambda_{\max}\right)^{2}} \; \geq \; \frac{1}{\text{cond}(\boldsymbol{H})} \; , \tag{7.80}
$$

where $\lambda_{\max}$ and $\lambda_{\min}$ are the maximal and minimal eigenvalues of the Hessian, respectively, and $\text{cond}(\boldsymbol{H})$ is the condition of a matrix

$$
\text{cond}(\boldsymbol{H}) \; = \; \frac{\lambda_{\max}}{\lambda_{\min}} \; . \tag{7.81}
$$

The improvement depends strongly on the condition of the matrix $\boldsymbol{H}(\boldsymbol{w}^{*})$.

Newton Method.

The Newton method has the update rule:

$$
\boldsymbol{w}^{\text{new}} \; = \; \boldsymbol{w}^{\text{old}} \; - \; \nabla_{\boldsymbol{w}} R(\boldsymbol{w}^{\text{old}}) \boldsymbol{H}^{-1}(\boldsymbol{w}^{\text{old}}) \; . \tag{7.82}
$$

With previous definitions and Taylor expansion of R around $\boldsymbol{w}^{*}$ we have $\boldsymbol{g}(\boldsymbol{w}^{*}) \; = \; \boldsymbol{0}$.

The Taylor expansion of the i-th component of $\boldsymbol{g}$ around $\boldsymbol{w}^{\text{old}}$ is

$$
\begin{aligned}
& 0 \; = \; g_{i}(\boldsymbol{w}^{*}) \; = \; g_{i}(\boldsymbol{w}^{\text{old}}) \; + \; \nabla_{\boldsymbol{w}} g_{i}(\boldsymbol{w}^{\text{old}}) \left(\boldsymbol{w}^{*} \; - \; \boldsymbol{w}^{\text{old}}\right) \; + \\
& \boldsymbol{\xi}^{T} \boldsymbol{H}_{i}(\boldsymbol{w}^{\text{old}}) \, \boldsymbol{\xi} \; ,
\end{aligned}
\tag{7.83}
$$

where $\boldsymbol{H}_{i}$ is the Hessian of g_{i} and $\boldsymbol{\xi}$ is a vector $\boldsymbol{\xi}_{i} \; = \; \lambda \; \left(\boldsymbol{w}^{*} \; - \; \boldsymbol{w}^{\text{old}}\right)$ with $0 \; \leq \; \lambda \; \leq \; 1$, thus $\|\boldsymbol{\xi}_{i}\|^{2} \; \leq \; \left\|\boldsymbol{w}^{*} \; - \; \boldsymbol{w}^{\text{old}}\right\|$.

We obtain

$$
\begin{aligned}
& g_{i}(\boldsymbol{w}^{\text{old}}) \; = \; g_{i}(\boldsymbol{w}^{\text{old}}) \; - \; g_{i}(\boldsymbol{w}^{*}) \; = \; g_{i}(\boldsymbol{w}^{\text{old}}) \; - \\
& \left(g_{i}(\boldsymbol{w}^{\text{old}}) \; + \; \left(\boldsymbol{w}^{*} \; - \; \boldsymbol{w}^{\text{old}}\right)^{T} \nabla_{\boldsymbol{w}} g_{i}(\boldsymbol{w}^{\text{old}}) \; + \right. \\
& \left. \frac{1}{2} \boldsymbol{\xi}_{i}^{T} \boldsymbol{H}_{i}(\boldsymbol{w}^{\text{old}}) \, \boldsymbol{\xi}_{i}\right) \; .
\end{aligned}
\tag{7.84}
$$

Combining above equations we obtain

$$\begin{aligned} \boldsymbol{g}(\boldsymbol{w}^{\text{old}}) \;=\; -\,\boldsymbol{H}(\boldsymbol{w}^{\text{old}})\,\left(\boldsymbol{w}^{*} \;-\; \boldsymbol{w}^{\text{old}}\right) \;- \\ \frac{1}{2}\,\left(\boldsymbol{\xi}_1^T\boldsymbol{H}_1(\boldsymbol{w}^{\text{old}})\,\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_W^T\boldsymbol{H}_W(\boldsymbol{w}^{\text{old}})\,\boldsymbol{\xi}_W\right)^T \;. \end{aligned} \tag{7.85}$$

which gives

$$\begin{aligned} \boldsymbol{w}^{\text{old}} \;-\; \boldsymbol{H}^{-1}\,\boldsymbol{g}(\boldsymbol{w}^{\text{old}}) \;-\; \boldsymbol{w}^{*} \;= \\ \frac{1}{2}\boldsymbol{H}^{-1}\,\left(\boldsymbol{\xi}_1^T\boldsymbol{H}_1(\boldsymbol{w}^{\text{old}})\,\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_W^T\boldsymbol{H}_W(\boldsymbol{w}^{\text{old}})\,\boldsymbol{\xi}_W\right)^T \;. \end{aligned} \tag{7.86}$$

that means

$$\boldsymbol{w}^{\text{new}} \;-\; \boldsymbol{w}^{*} \;=\; \frac{1}{2}\,\boldsymbol{H}^{-1}\,\left(\boldsymbol{\xi}_1^T\boldsymbol{H}_1(\boldsymbol{w}^{\text{old}})\,\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_W^T\boldsymbol{H}_W(\boldsymbol{w}^{\text{old}})\,\boldsymbol{\xi}_W\right)^T \;. \tag{7.87}$$

Because

$$\rho \;=\; \max_i \|\boldsymbol{\xi}_i\| \;\leq\; \left\|\boldsymbol{w}^{\text{old}} \;-\; \boldsymbol{w}^{*}\right\| \tag{7.88}$$

and

$$\frac{1}{2}\,\boldsymbol{H}^{-1}\,\left(\boldsymbol{\xi}_1^T\boldsymbol{H}_1(\boldsymbol{w}^{\text{old}})\,\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_W^T\boldsymbol{H}_W(\boldsymbol{w}^{\text{old}})\,\boldsymbol{\xi}_W\right)^T \;=\; O(\rho^2) \tag{7.89}$$

the Newton method is quadratic convergent in $\left\|\boldsymbol{w}^{\text{old}} \;-\; \boldsymbol{w}^{*}\right\|$ assumed that

$$\left\|\frac{1}{2}\,\boldsymbol{H}^{-1}\,\left(\boldsymbol{\xi}_1^T\boldsymbol{H}_1(\boldsymbol{w}^{\text{old}})\,\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_W^T\boldsymbol{H}_W(\boldsymbol{w}^{\text{old}})\,\boldsymbol{\xi}_W\right)^T\right\| \;<\; 1\;. \tag{7.90}$$

7.2 On-line Optimization

Until now we considered optimization techniques where a fixed set of training examples was given.

However there are cases where we do not want to have a fixed set but update our solution incrementally. That means a single training example is used for update. The first method with fixed training size is called *batch* update whereas the incremental update is called *on-line*.

In a situation where the training examples are very cheap and a huge amount of them exist we would not like to restrict ourselves to a fixed training size. This might lead to overfitting which can be avoided if the training set size is large enough. On the other hand using all examples may be computationally too expensive. Here on-line methods are an alternative for learning because the danger of overfitting is reduced.

In a situation where the dynamics changes, i.e. for non-stationary problems, on-line methods are useful to track the dependencies in the data. If we have a solution, then as soon as the dynamics

changes, the error increases and learning starts again. Therefore on-line methods are a good choice for non-stationary problems.

The goal is to find $\boldsymbol{w}^*$ for which

$$g(\boldsymbol{w}^*) \;=\; 0\,. \tag{7.91}$$

We assume that g is a conditional expectation

$$g(\boldsymbol{w}) \;=\; \mathrm{E}\,(f(\boldsymbol{w}) \mid \boldsymbol{w})\;. \tag{7.92}$$

with finite variance

$$\mathrm{E}\left((g \;-\; f)^2 \mid \boldsymbol{w}\right) \;<\; \infty\,. \tag{7.93}$$

The *Robbins-Monro procedure* is

$$\boldsymbol{w}^{i+1} \;=\; \boldsymbol{w}^i \;-\; \eta_i\, f(\boldsymbol{w}^i)\,, \tag{7.94}$$

where $f(\boldsymbol{w}^i)$ is a random variable.

The learning rate sequence η_i satisfies

$$\lim_{i\to\infty} \eta_i \;=\; 0 \tag{7.95}$$

$$\sum_{i=1}^{\infty} \eta_i \;=\; \infty \tag{7.96}$$

$$\sum_{i=1}^{\infty} \eta_i^2 \;<\; \infty\,. \tag{7.97}$$

The first conditions ensures convergence. The second condition ensures that the changes are sufficiently large to find the root $\boldsymbol{w}^*$. The third condition ensures that the noise variance is limited.

The next theorem states that the Robbins-Monro procedure converges to the root $\boldsymbol{w}^*$.

Theorem 7.1 (Robbins-Monro) *Under the conditions eq. (7.95) the sequence eq. (7.94) converges to the root $\boldsymbol{w}^*$ of $\boldsymbol{g}$ with probability 1.*

If we apply the Robbins-Monro procedure to maximum likelihood we obtain

$$\frac{1}{l}\,\frac{\partial}{\partial \boldsymbol{w}} \sum_{i=1}^{l} \ln p(\boldsymbol{x}^i \mid \boldsymbol{w}) \;=\; 0\,. \tag{7.98}$$

The expectation is the limit

$$\lim_{l\to\infty}\; \frac{1}{l}\; \sum_{i=1}^{l} \frac{\partial}{\partial \boldsymbol{w}} \ln p(\boldsymbol{x}^i \mid \boldsymbol{w}) \;=\; \mathrm{E}\left(\frac{\partial}{\partial \boldsymbol{w}} \ln p(\boldsymbol{x}^i \mid \boldsymbol{w})\right)\;. \tag{7.99}$$

The maximum likelihood solution is asymptotically equivalent to

$$\mathrm{E}\left(\frac{\partial}{\partial \boldsymbol{w}} \ln p(\boldsymbol{x}^i \mid \boldsymbol{w})\right) \;=\; 0\,. \tag{7.100}$$

Therefore the Robbins-Monro procedure is applicable as

$$\boldsymbol{w}^{i+1} \;=\; \boldsymbol{w}^i \;+\; \eta_i \,\frac{\partial}{\partial \boldsymbol{w}} \ln p(\boldsymbol{x}^{i+1} \mid \boldsymbol{w}) \mid_{\boldsymbol{w}^i} \;\; . \tag{7.101}$$

This is an online update formula for maximum likelihood.

7.3 Convex Optimization

Convex Problems.

The optimization problem

$$\begin{aligned} &\min_{\boldsymbol{x}} \;\; f(\boldsymbol{x}) \\ &\text{s.t. } \forall_i : \;\; c_i(\boldsymbol{x}) \;\leq\; 0 \\ &\qquad \forall_j : \;\; e_j(\boldsymbol{x}) \;=\; 0\,, \end{aligned} \tag{7.102}$$

where f, c_i, and e_j are convex functions has as solution a convex set and if f is strictly convex then the solution is unique. This problem class is called "constraint convex minimization".

Note, that all SVM optimization problems we encountered so far are constraint convex minimization problems.

The Lagrangian is

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\alpha}, \boldsymbol{\mu}) \;=\; f(\boldsymbol{x}) \;+\; \sum_i \alpha_i \, c_i(\boldsymbol{x}) \;+\; \sum_j \mu_j \, e_j(\boldsymbol{x})\,, \tag{7.103}$$

where $\alpha_i \;\geq\; 0$. The variables $\boldsymbol{\alpha}$ and $\boldsymbol{\mu}$ are called "Lagrange multipliers". Note, that the Lagrangian can be build also for non-convex functions.

Assume that a feasible solution exists then the following statements are equivalent, where $\boldsymbol{x}$ denotes a feasible solution:

(a) an $\boldsymbol{x}$ exists with $c_i(\boldsymbol{x}) < 0$ for all i (Slater's condition),

(b) an $\boldsymbol{x}$ and $\alpha_i \geq 0$ exist such that $\sum_i \alpha_i \, c_i(\boldsymbol{x}) \;\leq\; 0$ (Karlin's condition).

Above statements (a) or (b) follow from the following statement

(c) there exist at least two feasible solutions and a feasible $\boldsymbol{x}$ such that all c_i are strictly convex at $\boldsymbol{x}$ w.r.t. the feasible set (strict constraint qualification).

The saddle point condition of Kuhn-Tucker:
If one of (a) - (c) holds then

$$\mathcal{L}\left(\hat{\boldsymbol{x}}, \boldsymbol{\alpha}, \boldsymbol{\mu}\right) \;\leq\; \mathcal{L}\left(\hat{\boldsymbol{x}}, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\mu}}\right) \;\leq\; \mathcal{L}\left(\boldsymbol{x}, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\mu}}\right) \;. \tag{7.104}$$

is necessary and sufficient for $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\mu}})$ being a solution to the optimization problem. Note, that "sufficient" also holds for non-convex functions.

The optimal Lagrange multipliers $\hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{\mu}}$ maximize $\mathcal{L}$ with $\boldsymbol{x}$ fixed to the optimal solution $\hat{\boldsymbol{x}}$. The optimal $\hat{\boldsymbol{x}}$ minimize $\mathcal{L}$ with Lagrange multipliers fixed to their optimal solution $\hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{\mu}}$.

All $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\mu}})$ which fulfill the saddle point condition eq. (7.104) are a solution to the optimization problem.

To see that assume that $(\hat{x}, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\mu}})$ satisfy the saddle point condition eq. (7.104). From $\mathcal{L}\left(\hat{\boldsymbol{x}}, \boldsymbol{\alpha}, \boldsymbol{\mu}\right) \leq \mathcal{L}\left(\hat{\boldsymbol{x}}, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\mu}}\right)$ it follows that

$$\sum_i \left(\alpha_i \;-\; \hat{\alpha}_i\right)\; c_i(\hat{\boldsymbol{x}}) \;+\; \sum_j \left(\mu_j \;-\; \hat{\mu}_j\right)\; e_j(\hat{\boldsymbol{x}}) \;\leq\; 0 \;. \tag{7.105}$$

If we set all $\mu_j = \hat{\mu}_j$ and $\alpha_i = \hat{\alpha}_i$ except $\alpha_k = \hat{\alpha}_k \;+\; 1$ then we obtain $c_k(\hat{\boldsymbol{x}}) \;\leq\; 0$ which shows the $\hat{\boldsymbol{x}}$ fulfills the constraints. The equality constraint $e_i(\boldsymbol{x}) \;=\; 0$ can be replaced by constraints $e_i(\boldsymbol{x}) \;\leq\; 0$ and $e_i(\boldsymbol{x}) \;\geq\; 0$. From both constraints follows $0 \;\leq\; e_k(\hat{\boldsymbol{x}}) \;\leq\; 0$, therefore, $e_k(\hat{\boldsymbol{x}}) \;=\; 0$ (here we can introduce μ^+ and μ^- and set $\mu_k = \mu_k^+ \;-\; \mu_k^-$).

If we set all $\mu_j = \hat{\mu}_j$ and $\alpha_i = \hat{\alpha}_i$ except $\alpha_k = 0$ then we obtain $\hat{\alpha}_k \; c_k(\hat{\boldsymbol{x}}) \;\geq\; 0$. Because $\hat{\alpha}_k \;\geq\; 0$ and from above $c_k(\hat{\boldsymbol{x}}) \;\leq\; 0$ we have $\hat{\alpha}_k \; c_k(\hat{\boldsymbol{x}}) \;\leq\; 0$. It follows that

$$\hat{\alpha}_i \; c_i(\hat{\boldsymbol{x}}) \;=\; 0 \tag{7.106}$$

and analog

$$\hat{\mu}_j \; e_j(\hat{\boldsymbol{x}}) \;=\; 0 \;. \tag{7.107}$$

These conditions are called "Karush-Kuhn-Tucker" conditions or KKT conditions.

For differentiable problems the minima and maxima can be determined.

Theorem 7.2 (KKT and Differentiable Convex Problems)
A solution to the problem eq. (7.102) with convex, differentiable f, c_i, and e_j is given by $\hat{\boldsymbol{x}}$ if $\hat{\alpha}_i \;\geq\; 0$ and $\hat{\mu}_j$ exist which satisfy:

$$\frac{\partial \mathcal{L}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\mu}})}{\partial \boldsymbol{x}} \;=\; \frac{\partial f(\hat{\boldsymbol{x}})}{\partial \boldsymbol{x}} + \sum_i \hat{\alpha}_i \; \frac{\partial c_i(\hat{\boldsymbol{x}})}{\partial \boldsymbol{x}} \;+\; \sum_j \hat{\mu}_j \; \frac{\partial e_j(\hat{\boldsymbol{x}})}{\partial \boldsymbol{x}} \;=\; 0 \tag{7.108}$$

$$\frac{\partial \mathcal{L}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\mu}})}{\partial \alpha_i} \;=\; c_i(\hat{\boldsymbol{x}}) \;\leq\; 0 \tag{7.109}$$

$$\frac{\partial \mathcal{L}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\mu}})}{\partial \mu_j} \;=\; e_j(\hat{\boldsymbol{x}}) \;=\; 0 \tag{7.110}$$

$$\forall_i : \;\; \hat{\alpha}_i \; c_i(\hat{\boldsymbol{x}}) \;=\; 0 \tag{7.111}$$

$$\forall_j : \;\; \hat{\mu}_j \; e_j(\hat{\boldsymbol{x}}) \;=\; 0 \tag{7.112}$$

For all $\boldsymbol{x}$,$\boldsymbol{\alpha}$ and μ for which eq. (7.108) to eq. (7.110) are fulfilled we have

$$f(\boldsymbol{x}) \;\geq\; f(\hat{\boldsymbol{x}}) \;\geq\; f(\boldsymbol{x}) \;+\; \sum_i \alpha_i \, c_i(\boldsymbol{x}) \;, \tag{7.113}$$

note that $e_j(\boldsymbol{x}) = 0$.

The dual optimization problem (Wolfe's dual) to the optimization problem eq. (7.102) is

$$\begin{aligned} \max_{\boldsymbol{x},\boldsymbol{\alpha},\boldsymbol{\mu}} \; & f(\boldsymbol{x}) \;+\; \sum_i \alpha_i \, c_i(\boldsymbol{x}) \;+\; \sum_j \mu_j \, e_j(\boldsymbol{x}) \\ \text{s.t. } \forall_i : \; & \alpha_i \;\geq\; 0 \\ & \frac{\partial \mathcal{L}(\boldsymbol{x},\boldsymbol{\alpha},\boldsymbol{\mu})}{\partial \boldsymbol{x}} \;=\; 0 \;. \end{aligned} \tag{7.114}$$

The solutions of the dual eq. (7.114) are the solutions of the primal eq. (7.102). If $\frac{\partial \mathcal{L}(\boldsymbol{x},\boldsymbol{\alpha},\boldsymbol{\mu})}{\partial \boldsymbol{x}} = 0$ can be solved for $\boldsymbol{x}$ and inserted into the dual, then we obtain a maximization problem in $\boldsymbol{\alpha}$ and $\boldsymbol{\mu}$.

Linear Programs.

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \boldsymbol{c}^T \boldsymbol{x} \\ \text{s.t. } & \boldsymbol{A}\,\boldsymbol{x} \;+\; \boldsymbol{d} \;\leq\; \boldsymbol{0} \;, \end{aligned} \tag{7.115}$$

where $\boldsymbol{A}\,\boldsymbol{x} \;+\; \boldsymbol{d} \;\leq\; \boldsymbol{0}$ means that for all i: $\sum_{j=1}^{l} A_{ij}\, x^j \;+\; d_j \;\leq\; 0$.

The Lagrangian is

$$\mathcal{L} \;=\; \boldsymbol{c}^T \boldsymbol{x} \;+\; \boldsymbol{\alpha}^T \left(\boldsymbol{A}\,\boldsymbol{x} \;+\; \boldsymbol{d}\right) \;. \tag{7.116}$$

The optimality conditions are

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{x}} \;=\; \boldsymbol{A}^T \boldsymbol{\alpha} \;+\; \boldsymbol{c} \;=\; 0 \tag{7.117}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\alpha}} \;=\; \boldsymbol{A}\,\boldsymbol{x} \;+\; \boldsymbol{d} \;\leq\; \boldsymbol{0} \tag{7.118}$$

$$\boldsymbol{\alpha}^T \left(\boldsymbol{A}\,\boldsymbol{x} \;+\; \boldsymbol{d}\right) \;=\; \boldsymbol{0} \tag{7.119}$$

$$\boldsymbol{\alpha} \;\geq\; \boldsymbol{0} \;. \tag{7.120}$$

The dual formulation after inserting $\boldsymbol{A}^T \boldsymbol{\alpha} \;+\; \boldsymbol{c} \;=\; \boldsymbol{0}$ into the Lagrangian is:

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \; & \boldsymbol{d}^T \boldsymbol{\alpha} \\ \text{s.t. } & \boldsymbol{A}^T \boldsymbol{\alpha} \;+\; \boldsymbol{c} \;=\; \boldsymbol{0} \\ & \boldsymbol{\alpha} \;\geq\; \boldsymbol{0} \;. \end{aligned} \tag{7.121}$$

We compute the dual of the dual. We first make a minimization problem by using $-\,\boldsymbol{d}^T \boldsymbol{\alpha}$ as objective and also use $-\,\boldsymbol{\alpha} \;\leq\; 0$ as well as $-\,\boldsymbol{A}^T \boldsymbol{\alpha} \;-\; \boldsymbol{c} \;=\; \boldsymbol{0}$ with Lagrange multiplier $\boldsymbol{x}'$ for

the equality constraints $-\boldsymbol{A}^T\,\boldsymbol{\alpha}\,-\,\boldsymbol{c}\,=\,\mathbf{0}$ and $\boldsymbol{\mu}$ for $-\boldsymbol{\alpha}\,\leq\,0$. The dual of the dual is after again transforming it into a minimization problem:

$$\begin{aligned}&\min_{\boldsymbol{\alpha},\boldsymbol{\mu}}\ \boldsymbol{c}^T\boldsymbol{x}'\\&\text{s.t. }\boldsymbol{A}\,\boldsymbol{x}'\,+\,\boldsymbol{d}\,+\,\boldsymbol{\mu}\,=\,\mathbf{0}\\&\qquad\boldsymbol{\mu}\,\geq\,\mathbf{0}\,.\end{aligned}\tag{7.122}$$

Because $\boldsymbol{\mu}$ do not influence the objective, we can chose them freely. Therefore we obtain again the primal eq. (7.102) because we only have to ensure $\boldsymbol{A}\,\boldsymbol{x}'\,+\,\boldsymbol{d}\,\leq\,\mathbf{0}$.

Quadratic Programs.

The primal quadratic problem is

$$\begin{aligned}&\min_{\boldsymbol{x}}\ \frac{1}{2}\boldsymbol{x}^T\boldsymbol{K}\boldsymbol{x}\,+\,\boldsymbol{c}^T\boldsymbol{x}\\&\text{s.t. }\boldsymbol{A}\,\boldsymbol{x}\,+\,\boldsymbol{d}\,\leq\,\mathbf{0}\,,\end{aligned}\tag{7.123}$$

where $\boldsymbol{K}$ is strictly positive definite (implying that $\boldsymbol{K}^{-1}$ exists).

The Lagrangian is

$$\mathcal{L}(\boldsymbol{x},\boldsymbol{\alpha})\,=\,\frac{1}{2}\boldsymbol{x}^T\boldsymbol{K}\boldsymbol{x}\,+\,\boldsymbol{c}^T\boldsymbol{x}\,+\,\boldsymbol{\alpha}^T\left(\boldsymbol{A}\,\boldsymbol{x}\,+\,\boldsymbol{d}\right)\,.\tag{7.124}$$

The optimality conditions are

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{x}}\,=\,\boldsymbol{K}\,\boldsymbol{x}\,+\,\boldsymbol{A}^T\boldsymbol{\alpha}\,+\,\boldsymbol{c}\,=\,0\tag{7.125}$$

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{\alpha}}\,=\,\boldsymbol{A}\,\boldsymbol{x}\,+\,\boldsymbol{d}\,\leq\,\mathbf{0}\tag{7.126}$$

$$\boldsymbol{\alpha}^T\left(\boldsymbol{A}\,\boldsymbol{x}\,+\,\boldsymbol{d}\right)\,=\,\mathbf{0}\tag{7.127}$$

$$\boldsymbol{\alpha}\,\geq\,\mathbf{0}\,.\tag{7.128}$$

The first equation is used to substitute $\boldsymbol{x}$ in the Lagrangian:

$$\begin{aligned}&\mathcal{L}(\boldsymbol{x},\boldsymbol{\alpha})\,=\,\frac{1}{2}\boldsymbol{x}^T\boldsymbol{K}\boldsymbol{x}\,+\,\boldsymbol{c}^T\boldsymbol{x}\,+\,\boldsymbol{\alpha}^T\left(\boldsymbol{A}\,\boldsymbol{x}\,+\,\boldsymbol{d}\right)\,=\\&-\,\frac{1}{2}\boldsymbol{x}^T\boldsymbol{K}\boldsymbol{x}\,+\,\left(\boldsymbol{x}^T\boldsymbol{K}\,+\,\boldsymbol{c}^T\,+\,\boldsymbol{\alpha}^T\boldsymbol{A}\right)\,\boldsymbol{x}\,+\,\boldsymbol{\alpha}^T\boldsymbol{d}\,=\\&-\,\frac{1}{2}\boldsymbol{x}^T\boldsymbol{K}\boldsymbol{x}\,+\,\boldsymbol{\alpha}^T\boldsymbol{d}\,=\\&-\,\frac{1}{2}\left(-\boldsymbol{K}^{-1}\left(\boldsymbol{c}\,+\,\boldsymbol{A}^T\boldsymbol{\alpha}\right)\right)^T\boldsymbol{K}\,\left(-\boldsymbol{K}^{-1}\left(\boldsymbol{c}\,+\,\boldsymbol{A}^T\boldsymbol{\alpha}\right)\right)\,+\,\boldsymbol{\alpha}^T\boldsymbol{d}\,=\\&-\,\frac{1}{2}\boldsymbol{\alpha}^T\boldsymbol{A}\,\boldsymbol{K}^{-1}\boldsymbol{A}^T\boldsymbol{\alpha}\,+\,\left(\boldsymbol{d}^T\,-\,\boldsymbol{c}^T\boldsymbol{K}^{-1}\boldsymbol{A}^T\right)\,\boldsymbol{\alpha}\,-\,\frac{1}{2}\boldsymbol{c}^T\boldsymbol{K}^{-1}\boldsymbol{c}\,.\end{aligned}\tag{7.129}$$

Note that $\frac{1}{2}\boldsymbol{c}^T\boldsymbol{K}^{-1}\boldsymbol{c}$ is constant in $\boldsymbol{\alpha}$ and $\boldsymbol{x}$. We transform the maximization again into minimization by using the negative objective.

The dual is

$$\begin{aligned} &\min_{\boldsymbol{x}} \ \frac{1}{2}\boldsymbol{\alpha}^T \boldsymbol{A}\boldsymbol{K}^{-1}\boldsymbol{A}^T\boldsymbol{\alpha} \ - \ \left(\boldsymbol{d}^T \ - \ \boldsymbol{c}^T\boldsymbol{K}^{-1}\boldsymbol{A}^T\right) \ \boldsymbol{\alpha} \\ &\text{s.t. } \mathbf{0} \ \leq \ \boldsymbol{\alpha} \, . \end{aligned} \tag{7.130}$$

Note that the dual of the dual is in general not the primal but a similar problem. Dualizing twice, however, gives again the dual.

Optimization of Convex Problems.

The convex optimization can be solved by gradient descent or constraint gradient descent methods. See next chapter for such methods.

Efficient methods are *interior point methods*. An interior point is a pair $(\boldsymbol{x}, \boldsymbol{\alpha})$ which satisfies both the primal and dual constraints.

We can rewrite the optimality conditions of eq. (7.125) as

$$\boldsymbol{K}\,\boldsymbol{x} \ + \ \boldsymbol{A}^T\boldsymbol{\alpha} \ + \ \boldsymbol{c} \ = \ 0 \tag{7.131}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\alpha}} \ = \ \boldsymbol{A}\,\boldsymbol{x} \ + \ \boldsymbol{d} \ + \ \boldsymbol{\xi} \ = \ \mathbf{0} \tag{7.132}$$

$$\boldsymbol{\alpha}^T\boldsymbol{\xi} \ = \ \mathbf{0} \tag{7.133}$$

$$\boldsymbol{\alpha}, \boldsymbol{\xi} \ \geq \ \mathbf{0} \, , \tag{7.134}$$

where we set $\mathbf{0} \ \leq \ \boldsymbol{\xi} \ = \ - \ (\boldsymbol{A}\,\boldsymbol{x} \ + \ \boldsymbol{d})$.

The first two equations are linear in the variables $\boldsymbol{\alpha}$ and $\boldsymbol{\xi}$, however the third equations is quadratic.

Interior point algorithms solve these equations by an iterative method called "predictor-corrector" and set $\alpha_i \, \xi_i \ = \ \eta \ > \ 0$ which is decreased (annealed) to zero.

Chapter 8

Bayes Techniques

In this chapter we introduce a probabilistic framework, the Bayes framework, for the empirical error and regularization. In particular the framework will be applied to neural networks but it can also be applied to other models. The Bayes framework gives tools for dealing with the hyper-parameters which often trade-off the empirical error with the complexity term. However an optimal value for these parameters can so far only be found by cross-validation on the training set. The Bayes framework helps to formally treat these parameters. Especially in the case when many hyper-parameters are needed then their combination cannot be tested by cross-validation and a formal treatment is necessary.

Another important issue is that Bayes methods allow to introduce error bars and confidence intervals for the model outputs. Bayes approaches also help to compare quite different models like different neural networks architectures. Bayes techniques can be used to select relevant features. Bayes methods can be used to build averages and committees of models.

Summarizing, Bayes techniques allow

- to introduce a probabilistic framework
- to deal with hyper-parameters
- to supply error bars and confidence intervals for the model output
- to compare different models
- to select relevant features
- to make averages and committees.

8.1 Likelihood, Prior, Posterior, Evidence

As in Section 2.2.1 we have the training data $\left\{\boldsymbol{z}^1, \ldots, \boldsymbol{z}^l\right\}$ ($\boldsymbol{z}^i = (\boldsymbol{x}^i, y^i)$), the matrix of feature vectors $\boldsymbol{X} = \left(\boldsymbol{x}^1, \ldots, \boldsymbol{x}^l\right)^T$, the vector of labels $\boldsymbol{y} = \left(y^1, \ldots, y^l\right)^T$, and the training data matrix $\boldsymbol{Z} = \left(\boldsymbol{z}^1, \ldots, \boldsymbol{z}^l\right)$. Further we define the training data as

$$\{\boldsymbol{z}\} = \left\{\boldsymbol{z}^1, \ldots, \boldsymbol{z}^l\right\}. \tag{8.1}$$

In Section 3.4 the likelihood $\mathcal{L}$ was defined as

$$\mathcal{L}(\{\boldsymbol{z}\}; \boldsymbol{w}) \;=\; p(\{\boldsymbol{z}\}; \boldsymbol{w}) \;, \tag{8.2}$$

i.e. the probability of the model $p(\boldsymbol{z}; \boldsymbol{w})$ to produce the data set. We found that for iid data sampling the likelihood is

$$\mathcal{L}(\{\boldsymbol{z}\}; \boldsymbol{w}) \;=\; p(\{\boldsymbol{z}\}; \boldsymbol{w}) \;=\; \prod_{i=1}^{l} p(\boldsymbol{z}^i; \boldsymbol{w}) \;. \tag{8.3}$$

In supervised learning we can write

$$p\left(\boldsymbol{z}; \boldsymbol{w}\right) \;=\; p\left(\boldsymbol{x}\right)\; p(y \mid \boldsymbol{x}; \boldsymbol{w}) \tag{8.4}$$

and

$$\mathcal{L}(\{\boldsymbol{z}\}; \boldsymbol{w}) \;=\; \prod_{i=1}^{l} p(\boldsymbol{x}^i) \prod_{i=1}^{l} p(y^i \mid \boldsymbol{x}^i; \boldsymbol{w}) \;. \tag{8.5}$$

Because $\prod_{i=1}^{l} p(\boldsymbol{x}^i)$ is independent of the parameters, it is sufficient to maximize the conditional likelihood

$$\mathcal{L}(\{y\} \mid \{\boldsymbol{x}\}; \boldsymbol{w}) \;=\; \prod_{i=1}^{l} p(y^i \mid \boldsymbol{x}^i; \boldsymbol{w}) \;. \tag{8.6}$$

The likelihood or the negative $\log$-likelihood can be treated as any error term.

For the likelihood in this chapter the parameter vector $\boldsymbol{w}$ is not used to parameterize the likelihood but the likelihood is conditioned on $\boldsymbol{w}$.

The likelihood is

$$p(\{\boldsymbol{z}\} \mid \boldsymbol{w}) \;. \tag{8.7}$$

However we found that only maximizing the likelihood would lead to overfitting if the model is complex enough. In the most extreme case the model would only produce the training examples with equal probability and other data with probability zero. That means $p\left(\boldsymbol{z}; \boldsymbol{w}\right)$ is the sum of Dirac delta-distributions.

To avoid overfitting we can assume that certain $\boldsymbol{w}$ are more probable to be observed in the real world than others. That means some models are more likely in the world.

The fact that some models are more likely can be expressed by a distribution $p(\boldsymbol{w})$, the *prior distribution*. The information in $p(\boldsymbol{w})$ stems from prior knowledge about the problem. This is knowledge without seeing the data, that means we would choose a model according to $p(\boldsymbol{w})$ if we do not have data available.

Now we can use the *Bayes formula*:

$$p(\boldsymbol{w} \mid \{\boldsymbol{z}\}) \;=\; \frac{p(\{\boldsymbol{z}\} \mid \boldsymbol{w})\; p(\boldsymbol{w})}{p_{\boldsymbol{w}}(\{\boldsymbol{z}\})} \tag{8.8}$$

where

$$p(\boldsymbol{w} \mid \{\boldsymbol{z}\}) \tag{8.9}$$

is called the *posterior distribution* and the normalization constant

$$p_{\boldsymbol{w}}(\{\boldsymbol{z}\}) \;=\; \int_W p(\{\boldsymbol{z}\} \mid \boldsymbol{w})\, p(\boldsymbol{w})\, d\boldsymbol{w} \tag{8.10}$$

is called the *evidence* for a class of models parameterized by $\boldsymbol{w}$, however it is also called *accessible volume of the configuration space* (from statistical mechanics), *partition function* (from statistical mechanics), or *error moment generating function*.

Bayes formula is

$$\text{posterior} \;=\; \frac{\text{likelihood} \;\times\; \text{prior}}{\text{evidence}}\,. \tag{8.11}$$

Note that only if $p(\boldsymbol{w})$ is indeed the distribution of model parameters in the real world then

$$p_{\boldsymbol{w}}(\{\boldsymbol{z}\}) \;=\; p(\{\boldsymbol{z}\})\,. \tag{8.12}$$

That means if the real data is indeed produced by first choosing $\boldsymbol{w}$ according to $p(\boldsymbol{w})$ and then generating $\{\boldsymbol{z}\}$ through $p(\{\boldsymbol{z}\} \mid \boldsymbol{w})$ then $p_{\boldsymbol{w}}(\{\boldsymbol{z}\})$ is the probability of observing data $\{\boldsymbol{z}\}$.

However in general the data in real world is not produced according to some mathematical models and therefore $p_{\boldsymbol{w}}(\{\boldsymbol{z}\})$ is not the distribution of occurrence of data $\{\boldsymbol{z}\}$ in the real world.

However $p_{\boldsymbol{w}}(\{\boldsymbol{z}\})$ gives the probability of observing data $\{\boldsymbol{z}\}$ with the model class which is parameterized by $\boldsymbol{w}$.

8.2 Maximum A Posteriori Approach

The Maximum A Posteriori Approach (MAP) searches for the maximal posterior $p(\boldsymbol{w} \mid \{\boldsymbol{z}\})$. Fig. 8.1 shows the maximum a posteriori estimator $\boldsymbol{w}_{\mathrm{MAP}}$ which maximizes the posterior.

For applying the MAP approach the prior $p(\boldsymbol{w})$ must be defined. For neural networks the weight decay method leads to the simple term

$$\Omega(\boldsymbol{w}) \;=\; \|\boldsymbol{w}\|^2 \;=\; \boldsymbol{w}^T\boldsymbol{w} \;=\; \sum_{ij} w_{ij}^2\,. \tag{8.13}$$

This can be expressed through a Gaussian weight prior

$$\begin{aligned} p(\boldsymbol{w}) \;&=\; \frac{1}{Z_w(\alpha)}\, \exp\left(-\frac{1}{2}\,\alpha\,\|\boldsymbol{w}\|^2\right) \\ Z_w(\alpha) \;&=\; \int_W \exp\left(-\frac{1}{2}\,\alpha\,\|\boldsymbol{w}\|^2\right)\, d\boldsymbol{w} \;=\; \left(\frac{2\,\pi}{\alpha}\right)^{W/2}\,. \end{aligned} \tag{8.14}$$

The parameter α is a hyper-parameter which trades in the $\log$-posterior in the error term against the complexity term and is here correlated with the allowed variance of the weights.

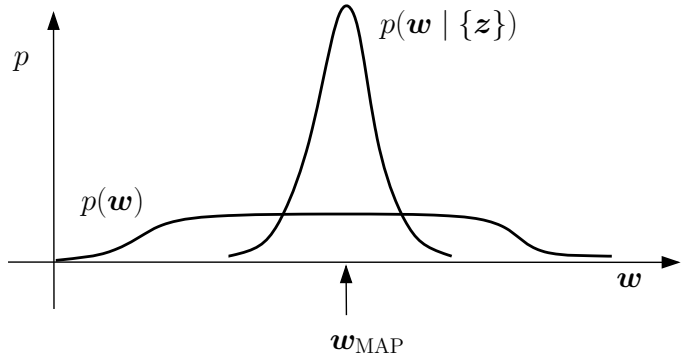

Figure 8.1: The maximum a posteriori estimator $\boldsymbol{w}_{\mathrm{MAP}}$ is the weight vector which maximizes the posterior $p(\boldsymbol{w} \mid \{\boldsymbol{z}\})$. The prior distribution $p(\boldsymbol{w})$ is also shown.

Other weight decay terms give either a Laplace distribution ($\Omega(\boldsymbol{w}) \;=\; \|\boldsymbol{w}\|_1$)

$$\begin{aligned} p(\boldsymbol{w}) \;&=\; \frac{1}{Z_w(\alpha)} \;\exp\left(-\frac{1}{2}\,\alpha\,\|\boldsymbol{w}\|_1\right) \\ Z_w(\alpha) \;&=\; \int_W \exp\left(-\frac{1}{2}\,\alpha\,\|\boldsymbol{w}\|_1\right)\,d\boldsymbol{w} \end{aligned} \tag{8.15}$$

or for compact weight sets a Cauchy distribution ($\Omega(\boldsymbol{w}) \;=\; \log\left(1 \;+\; \|\boldsymbol{w}\|^2\right)$):

$$\begin{aligned} p(\boldsymbol{w}) \;&=\; \frac{1}{Z_w(\alpha)} \;\exp\left(-\frac{1}{2}\,\alpha\right)\;\left(1 \;+\; \|\boldsymbol{w}\|^2\right) \\ Z_w(\alpha) \;&=\; \int_W \exp\left(-\frac{1}{2}\,\alpha\right)\;\left(1 \;+\; \|\boldsymbol{w}\|^2\right)\,d\boldsymbol{w}\;. \end{aligned} \tag{8.16}$$

For Gaussian noise models from Section 4.1 we have

$$\begin{aligned} &p(\{\boldsymbol{z}\} \mid \boldsymbol{w}) \;= \\ &\frac{1}{(2\,\pi)^{d/2}\,|\boldsymbol{\Sigma}|^{1/2}} \;\exp\left(-\frac{1}{2}\,(\boldsymbol{y} \;-\; \boldsymbol{X}\,\boldsymbol{w})^T\,\boldsymbol{\Sigma}^{-1}(\boldsymbol{y} \;-\; \boldsymbol{X}\,\boldsymbol{w})\right)\;p(\{\boldsymbol{x}\}) \end{aligned} \tag{8.17}$$

and for $\boldsymbol{\Sigma} \;=\; \sigma^2\,\boldsymbol{I}$

$$\begin{aligned} &p(\{\boldsymbol{z}\} \mid \boldsymbol{w}) \;= \\ &\frac{1}{(2\,\pi)^{d/2}\,\sigma^d} \;\exp\left(-\frac{1}{2\,\sigma^2}\,(\boldsymbol{y} \;-\; \boldsymbol{X}\,\boldsymbol{w})^T\,(\boldsymbol{y} \;-\; \boldsymbol{X}\,\boldsymbol{w})\right)\;p(\{\boldsymbol{x}\})\;. \end{aligned} \tag{8.18}$$

The term $R_{\mathrm{emp}} \;=\; (\boldsymbol{y} \;-\; \boldsymbol{X}\,\boldsymbol{w})^T\,(\boldsymbol{y} \;-\; \boldsymbol{X}\,\boldsymbol{w})$ is only the mean squared error.

The negative log-posterior is

$$-\;\log p(\boldsymbol{w} \mid \{\boldsymbol{z}\}) \;=\; -\;\log p(\{\boldsymbol{z}\} \mid \boldsymbol{w}) \;-\; \log p(\boldsymbol{w}) \;+\; \log p_{\boldsymbol{w}}(\{\boldsymbol{z}\})\;, \tag{8.19}$$

where $p_{\boldsymbol{w}}(\{\boldsymbol{z}\})$ does not depend on $\boldsymbol{w}$.

For maximum a posteriori estimation only $-\log p(\{\boldsymbol{z}\} \mid \boldsymbol{w}) - \log p(\boldsymbol{w})$ must be minimized which results in the terms

$$\tilde{R}(\boldsymbol{w}) = \frac{1}{2\,\sigma^2} R_{\text{emp}} + \frac{1}{2}\alpha\,\Omega(\boldsymbol{w}) = \frac{1}{2}\,\beta\,R_{\text{emp}} + \frac{1}{2}\,\alpha\,\Omega(\boldsymbol{w})\,, \tag{8.20}$$

where $\beta^{-1} = \sigma^2$. If we set $R(\boldsymbol{w}) = \tilde{R}(\boldsymbol{w})\,2\,\sigma^2$ and setting $\lambda = \sigma^2\,\alpha$ we have to minimize

$$R(\boldsymbol{w}) = R_{\text{emp}} + \lambda\,\Omega(\boldsymbol{w})\,. \tag{8.21}$$

Therefore minimizing error terms consisting of the empirical error plus a complexity term can be viewed in most cases as maximum a posteriori estimation.

Note that the likelihood is the exponential function with empirical error as argument

$$p(\{\boldsymbol{z}\} \mid \boldsymbol{w}) = \frac{1}{Z_R(\beta)} \exp\left(-\frac{1}{2}\,\beta\,R_{\text{emp}}\right) \tag{8.22}$$

and the prior is an exponential function of the complexity

$$p(\boldsymbol{w}) = \frac{1}{Z_w(\alpha)} \exp\left(-\frac{1}{2}\,\alpha\,\Omega(\boldsymbol{w})\right) \tag{8.23}$$

and the posterior is

$$p(\boldsymbol{w} \mid \{\boldsymbol{z}\}) = \frac{1}{Z(\alpha,\beta)} \exp\left(-\frac{1}{2}\,(\alpha\,\Omega(\boldsymbol{w}) + \beta\,R_{\text{emp}})\right)\,, \tag{8.24}$$

where

$$Z(\alpha,\beta) = \int_W \exp\left(-\frac{1}{2}\,(\alpha\,\Omega(\boldsymbol{w}) + \beta\,R_{\text{emp}})\right)\,d\boldsymbol{w}\,. \tag{8.25}$$

8.3 Posterior Approximation

In order to approximate the posterior a Gaussian assumption is made.

First we make a Taylor expansion of $\tilde{R}(\boldsymbol{w})$ around its minimum $\boldsymbol{w}_{\text{MAP}}$:

$$\tilde{R}(\boldsymbol{w}) = \tilde{R}(\boldsymbol{w}_{\text{MAP}}) + \frac{1}{2}\,(\boldsymbol{w} - \boldsymbol{w}_{\text{MAP}})^T\,\boldsymbol{H}\,(\boldsymbol{w} - \boldsymbol{w}_{\text{MAP}})\,, \tag{8.26}$$

where the first order derivatives vanish at the minimum and $\boldsymbol{H}$ is the Hessian of $\tilde{R}(\boldsymbol{w})$ at $\boldsymbol{w}_{\text{MAP}}$.

The posterior is now a Gaussian

$$\begin{aligned} &p(\boldsymbol{w} \mid \{\boldsymbol{z}\}) = \frac{1}{Z}\,\exp(-\tilde{R}(\boldsymbol{w})) = \\ &\frac{1}{Z}\,\exp\left(-\tilde{R}(\boldsymbol{w}_{\text{MAP}}) - \frac{1}{2}\,(\boldsymbol{w} - \boldsymbol{w}_{\text{MAP}})^T\,\boldsymbol{H}\,(\boldsymbol{w} - \boldsymbol{w}_{\text{MAP}})\right)\,, \end{aligned} \tag{8.27}$$

where Z is a normalization constant.

The Hessian for weight decay is given by

$$\boldsymbol{H} \;=\; \frac{1}{2\,\sigma^2}\,\boldsymbol{H}_{\mathrm{emp}} \;+\; \alpha\,\boldsymbol{I} \;=\; \frac{\beta}{2}\,\boldsymbol{H}_{\mathrm{emp}} \;+\; \alpha\,\boldsymbol{I}\;, \tag{8.28}$$

where $\boldsymbol{H}_{\mathrm{emp}}$ is the Hessian of the empirical error.

The normalization constant is

$$Z(\alpha,\beta) \;=\; \exp\left(-\tilde{R}(\boldsymbol{w}_{\mathrm{MAP}})\right)\;(2\,\pi)^{W/2}\;|\boldsymbol{H}|^{-1/2}\;. \tag{8.29}$$

8.4 Error Bars and Confidence Intervals

We now want to derive confidence intervals for model outputs. A user is not only interested in the best prediction but also wants to know how reliable the prediction is.

The distribution for the outputs is

$$p(y \mid \boldsymbol{x},\{\boldsymbol{z}\}) \;=\; \int_W p(y \mid \boldsymbol{x},\boldsymbol{w})\;p(\boldsymbol{w} \mid \{\boldsymbol{z}\})\;d\boldsymbol{w}\;, \tag{8.30}$$

where we used the posterior distribution $p(\boldsymbol{w} \mid \{\boldsymbol{z}\})$ and a noise model $p(y \mid \boldsymbol{x},\boldsymbol{w})$.

The Gaussian noise model for one dimension is

$$p(y \mid \boldsymbol{x},\boldsymbol{w}) \;=\; \frac{1}{Z_R(\beta)}\;\exp\left(-\frac{\beta}{2}\;(y \;-\; g(\boldsymbol{x};\boldsymbol{w})\;)^2\right)\;, \tag{8.31}$$

where we again used $\beta \;=\; \frac{1}{\sigma^2}$ and

$$Z_R(\beta) \;=\; \left(\frac{2\,\pi}{\beta}\right)^{l/2}\;. \tag{8.32}$$

We now approximate $g(\boldsymbol{x};\boldsymbol{w})$ linearly around $\boldsymbol{w}_{\mathrm{MAP}}$:

$$g(\boldsymbol{x};\boldsymbol{w}) \;=\; g(\boldsymbol{x};\boldsymbol{w}_{\mathrm{MAP}}) \;+\; \boldsymbol{g}^T\;(\boldsymbol{w} \;-\; \boldsymbol{w}_{\mathrm{MAP}})\;, \tag{8.33}$$

where $\boldsymbol{g}$ is the gradient of $g(\boldsymbol{x};\boldsymbol{w})$ evaluated at $\boldsymbol{w}_{\mathrm{MAP}}$. This approximation together with the approximation for the posterior gives us

$$p(y \mid \boldsymbol{x},\{\boldsymbol{z}\}) \;\propto\; \int_W \exp\left(-\frac{\beta}{2}\left(y \;-\; g(\boldsymbol{x};\boldsymbol{w}_{\mathrm{MAP}}) \;-\; \boldsymbol{g}^T\,(\boldsymbol{w} \;-\; \boldsymbol{w}_{\mathrm{MAP}})\right)^2 \;-\; \frac{1}{2}\;(\boldsymbol{w} \;-\; \boldsymbol{w}_{\mathrm{MAP}})^T\,\boldsymbol{H}\;(\boldsymbol{w} \;-\; \boldsymbol{w}_{\mathrm{MAP}})\right)\,d\boldsymbol{w}\;. \tag{8.34}$$

This integral can be computed and results in

$$p(y \mid \boldsymbol{x},\{\boldsymbol{z}\}) \;=\; \frac{1}{\sqrt{2\,\pi}\,\sigma_y}\;\exp\left(-\frac{1}{2\,\sigma_y^2}\;(y \;-\; g(\boldsymbol{x};\boldsymbol{w}_{\mathrm{MAP}}))^2\right)\;, \tag{8.35}$$

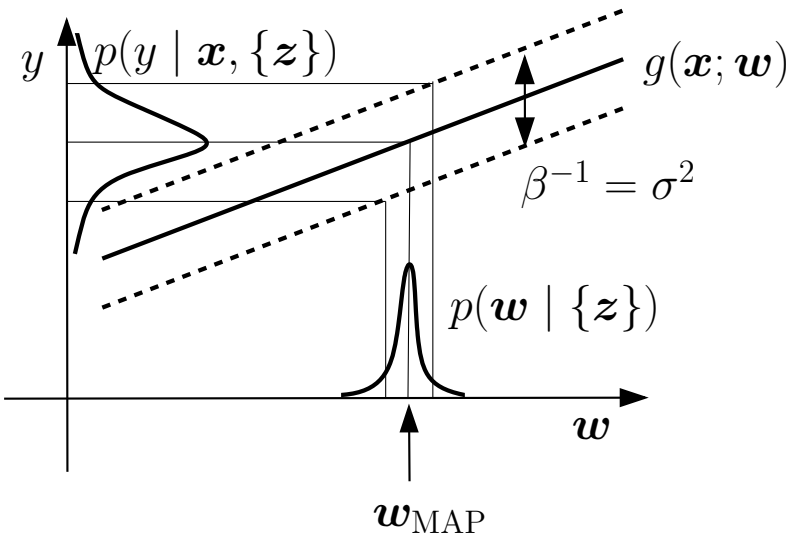

Figure 8.2: Error bars obtained by Bayes technique. Depicted is the double error line corresponding to 2σ. On the y-axis the error bars are given as quantiles of the distribution $p(y \mid \boldsymbol{x}; \{\boldsymbol{z}\})$. The large error bars result from the high inherent error $\beta^{-1} = \sigma^2$ of the data. The parameter $\boldsymbol{w}_{\mathrm{MAP}}$ has been chosen very precisely (e.g. if many training data points were available).

where

$$\sigma_y^2 \;=\; \frac{1}{\beta} \;+\; \boldsymbol{g}^T \boldsymbol{H}^{-1}\, \boldsymbol{g} \;=\; \sigma^2 \;+\; \boldsymbol{g}^T \boldsymbol{H}^{-1}\, \boldsymbol{g}\,. \tag{8.36}$$

The output variance is the inherent data noise variance σ^2 plus the approximation uncertainty $\boldsymbol{g}^T \boldsymbol{H}^{-1}\, \boldsymbol{g}$.

Fig. 8.2 shows error bars which are large because of a high inherent error $\beta^{-1} = \sigma^2$. Fig. 8.3 shows error bars which are large because of a not very precisely chosen $\boldsymbol{w}_{\mathrm{MAP}}$. The later can be if few training data are available or if the data contradicts the prior.

To derive the solution Eq. (8.35) of the integral Eq. (8.34), we require the identity:

$$\int_W \exp\left(-\frac{1}{2}\boldsymbol{w}^T \boldsymbol{A}\boldsymbol{w} \;+\; \boldsymbol{v}^T \boldsymbol{w}\right)\, d\boldsymbol{w} \;= \tag{8.37}$$
$$(2\pi)^{W/2}\, |\boldsymbol{A}|^{-1/2}\, \exp\left(\frac{1}{2}\boldsymbol{v}^T \boldsymbol{A}^{-1} \boldsymbol{v}\right)\,.$$

In the exponent of the integrand of the integral Eq. (8.34), we first multiply out the polynomials and then collect terms in $(\boldsymbol{w} \;-\; \boldsymbol{w}_{\mathrm{MAP}})$. Thereafter, we obtain

- the quadratic part:

$$\boldsymbol{A} \;=\; \boldsymbol{H} \;+\; \beta \boldsymbol{g}\boldsymbol{g}^T\,,$$

- the linear part:

$$\boldsymbol{v} \;=\; \beta\; (y \;-\; g(\boldsymbol{x}; \boldsymbol{w}_{\mathrm{MAP}}))\; \boldsymbol{g}\,,$$

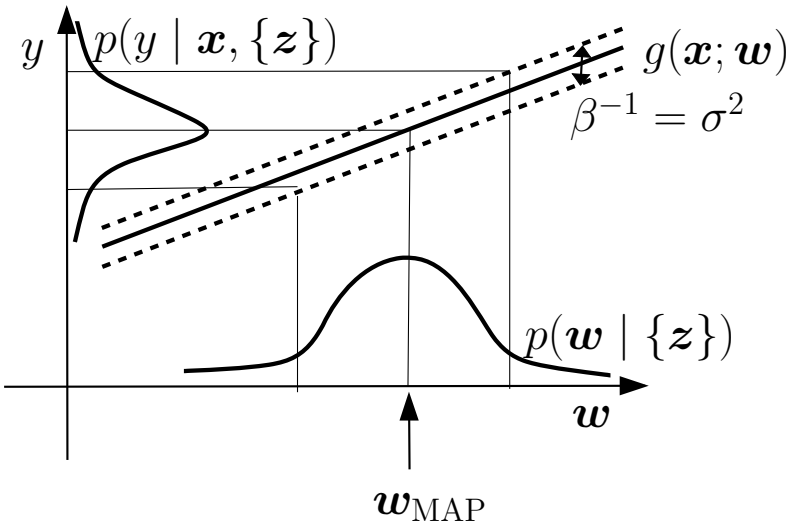

Figure 8.3: Error bars obtained by Bayes technique. As in Fig. 8.2 the double error line corresponds to 2σ but with much smaller noise variance in the data. On the y-axis the error bars are given as quantiles of the distribution $p(y \mid \boldsymbol{x}; \{\boldsymbol{z}\})$. The large error bars result from the broad posterior, that means the parameter $\boldsymbol{w}_{\mathrm{MAP}}$ has not been chosen very precisely (few data points or prior and data were not compatible).

- the constant part

$$\boldsymbol{v} \;=\; \beta \;(y \;-\; g(\boldsymbol{x}; \boldsymbol{w}_{\mathrm{MAP}}))^2 \;.$$

Using the identity Eq. (8.37), the exponent of the result of the integral Eq. (8.34) is

$$-\frac{1}{2}\boldsymbol{v}^T \boldsymbol{A}^{-1}\boldsymbol{v} \;+\; \frac{1}{2}c \;=$$
$$\frac{1}{2}\;(y \;-\; g(\boldsymbol{x}; \boldsymbol{w}_{\mathrm{MAP}}))^2 \;\left(\beta \;-\; \beta^2 \boldsymbol{g}^T \left(\boldsymbol{H} \;+\; \beta \boldsymbol{g}\boldsymbol{g}^T\right)^{-1} \boldsymbol{g}\right) \;.$$

The variance σ_y^2 can be reformulated

$$\sigma_y^2 \;=\; \left(\beta \;-\; \beta^2\, \boldsymbol{g}^T \left(\boldsymbol{H} \;+\; \beta \boldsymbol{g}\, \boldsymbol{g}^T\right)^{-1} \boldsymbol{g}\right)^{-1} \;=\; \frac{1}{\beta} \;+\; \boldsymbol{g}^T \boldsymbol{H}^{-1} \boldsymbol{g} \;.$$

This last equality is obtained by the matrix inversion lemma:

$$(\boldsymbol{A} \;+\; \boldsymbol{U}\, \boldsymbol{C}\, \boldsymbol{V})^{-1} \;=\; \boldsymbol{A}^{-1} \;-\; \boldsymbol{A}^{-1}\boldsymbol{U} \;\left(\boldsymbol{C}^{-1} \;+\; \boldsymbol{V}\, \boldsymbol{A}^{-1}\boldsymbol{U}\right)^{-1} \boldsymbol{V}\, \boldsymbol{A}^{-1} \;, \tag{8.38}$$

where we set $\boldsymbol{A} = \frac{1}{\beta}$, $\boldsymbol{U} = \boldsymbol{g}^T$, $\boldsymbol{V} = \boldsymbol{g}$, and $\boldsymbol{C} = \boldsymbol{H}^{-1}$.

8.5 Hyper-parameter Selection: Evidence Framework

We are focusing on the hyper-parameters α and β from the objective eq. (8.20). β is an assumption on the noise in the data. α is an assumption on the optimal network complexity relative to the empirical error.

The posterior can be expressed by integrating out α and β which is called *marginalization*:

$$p(\boldsymbol{w} \mid \{\boldsymbol{z}\}) = \int_{S_\alpha} \int_{S_\beta} p(\boldsymbol{w}, \alpha, \beta \mid \{\boldsymbol{z}\})\, d\alpha\, d\beta = \int_{S_\alpha} \int_{S_\beta} p(\boldsymbol{w} \mid \alpha, \beta, \{\boldsymbol{z}\})\, p(\alpha, \beta \mid \{\boldsymbol{z}\})\, d\alpha\, d\beta\,. \tag{8.39}$$

To compute the integrals will be considered in Section 8.6.

Here we first consider to approximate the posterior. We assume that the posterior $p(\alpha, \beta \mid \{\boldsymbol{z}\})$ is sharply peaked around the maximal values α_{MAP} and β_{MAP}. That means around high values of $p(\alpha, \beta \mid \{\boldsymbol{z}\})$ the $p(\boldsymbol{w} \mid \alpha, \beta, \{\boldsymbol{z}\})$ is constant $p(\boldsymbol{w} \mid \alpha_{\mathrm{MAP}}, \beta_{\mathrm{MAP}}, \{\boldsymbol{z}\})$. We obtain

$$p(\boldsymbol{w} \mid \{\boldsymbol{z}\}) = p(\boldsymbol{w} \mid \alpha_{\mathrm{MAP}}, \beta_{\mathrm{MAP}}, \{\boldsymbol{z}\}) \int_{S_\alpha} \int_{S_\beta} p(\alpha, \beta \mid \{\boldsymbol{z}\})\, d\alpha\, d\beta = p(\boldsymbol{w} \mid \alpha_{\mathrm{MAP}}, \beta_{\mathrm{MAP}}, \{\boldsymbol{z}\})\,. \tag{8.40}$$

Using this approximation we are searching for the hyper-parameters which maximize the posterior. We will try to express the posterior with the variables α and β and then to search for the variables which maximize the posterior.

The posterior of α and β is

$$p(\alpha, \beta \mid \{\boldsymbol{z}\}) = \frac{p(\{\boldsymbol{z}\} \mid \alpha, \beta)\, p(\alpha, \beta)}{p_{\alpha,\beta}(\{\boldsymbol{z}\})}\,. \tag{8.41}$$

Here the prior for α and β, $p(\alpha, \beta)$ must be chosen. For example *non-informative priors* which give equal probability to all values are a popular choice.

Note that from objective eq. (8.20). we see that $\beta = \frac{1}{\sigma^2}$ is only present in the $\boldsymbol{w}$-likelihood because it determines the noise in the data. In contrast α is only present in $\boldsymbol{w}$-prior as a weighting factor for the $\boldsymbol{w}$-prior which scales the complexity against the error. We express the (α, β)-likelihood through marginalization over $\boldsymbol{w}$:

$$p(\{\boldsymbol{z}\} \mid \alpha, \beta) = \int_W p(\{\boldsymbol{z}\} \mid \boldsymbol{w}, \alpha, \beta)\, p(\boldsymbol{w} \mid \alpha, \beta)\, d\boldsymbol{w} = \int_W p(\{\boldsymbol{z}\} \mid \boldsymbol{w}, \beta)\, p(\boldsymbol{w} \mid \alpha)\, d\boldsymbol{w}\,. \tag{8.42}$$

Using eq. (8.22), eq. (8.23), eq. (8.24), and eq. (8.25) we obtain

$$p(\{\boldsymbol{z}\} \mid \alpha, \beta) = \frac{Z(\alpha, \beta)}{Z_w(\alpha)\, Z_R(\beta)}\,. \tag{8.43}$$

Especially,

$$p(\boldsymbol{w} \mid \{\boldsymbol{z}\}) = \frac{Z(\alpha, \beta)}{Z_R(\beta)\, Z_w(\alpha)}\, p(\{\boldsymbol{z}\} \mid \boldsymbol{w})\, p(\boldsymbol{w})\,. \tag{8.44}$$

Example.

We will show this on an example with concrete empirical error and prior term.

For example if we use the mean squared error and as regularization a weight decay term we already computed $Z(\alpha, \beta)$ in eq. (8.29) as

$$Z(\alpha, \beta) \;=\; \exp\Big(-\;\tilde{R}(\boldsymbol{w}_{\mathrm{MAP}})\Big)\;(2\,\pi)^{W/2}\;|\boldsymbol{H}|^{-1/2}\;, \tag{8.45}$$

where

$$\tilde{R}(\boldsymbol{w}_{\mathrm{MAP}}) \;=\; \frac{1}{2}\,\beta R_{\mathrm{emp}} \;+\; \frac{1}{2}\alpha\,\Omega(\boldsymbol{w})\;. \tag{8.46}$$

According to eq. (8.32)

$$Z_R(\beta) \;=\; \left(\frac{2\,\pi}{\beta}\right)^{l/2} \tag{8.47}$$

and according to eq. (8.14)

$$Z_w(\alpha) \;=\; \left(\frac{2\,\pi}{\alpha}\right)^{W/2}\;. \tag{8.48}$$

$$\begin{aligned}\ln p(\{\boldsymbol{z}\} \mid \alpha, \beta) \;&=\; -\,\frac{1}{2}\,\alpha\,\Omega(\boldsymbol{w}_{\mathrm{MAP}}) \;-\; \frac{1}{2}\,\beta\,R_{\mathrm{emp}} \;-\; \frac{1}{2}\,\ln|\boldsymbol{H}| \;+ \\ &\frac{W}{2}\,\ln\alpha \;+\; \frac{l}{2}\,\ln\beta \;-\; \frac{l}{2}\ln(2\pi)\;,\end{aligned} \tag{8.49}$$

where according to eq. (8.28)

$$\boldsymbol{H} \;=\; \frac{\beta}{2}\,\boldsymbol{H}_{\mathrm{emp}} \;+\; \alpha\boldsymbol{I}\;. \tag{8.50}$$

Assume we already computed the eigenvalues λ_j of $1/2\;\boldsymbol{H}_{\mathrm{emp}}$ then

$$\begin{aligned}\frac{\partial}{\partial\alpha}\ln|\boldsymbol{H}| \;&=\; \frac{\partial}{\partial\alpha}\ln\prod_{j=1}^{W}(\beta\,\lambda_j \;+\; \alpha) \;= \\ \frac{\partial}{\partial\alpha}\sum_{j=1}^{W}\ln(\beta\,\lambda_j \;+\; \alpha) \;&=\; \sum_{j=1}^{W}\frac{1}{\beta\,\lambda_j \;+\; \alpha} \;=\; \mathrm{Tr}\boldsymbol{H}^{-1}\;,\end{aligned} \tag{8.51}$$

where we assumed that λ_j do not depend on α. However the Hessian $\boldsymbol{H}$ was evaluated at $\boldsymbol{w}_{\mathrm{MAP}}$ which depends on α, therefore terms in $\frac{\partial\lambda_j}{\partial\alpha}$ were neglected.

Setting the derivative of the negative log-posterior (for α and β) with respect to α to zero gives

$$\begin{aligned}\frac{\partial}{\partial\alpha}\;(-\;\ln p(\{\boldsymbol{z}\} \mid \alpha, \beta)) \;&=\; \\ \frac{1}{2}\,\Omega(\boldsymbol{w}_{\mathrm{MAP}}) \;+\; \frac{1}{2}\sum_{j=1}^{W}\frac{1}{\beta\,\lambda_j \;+\; \alpha} \;-\; \frac{1}{2}\,W\,\frac{1}{\alpha} \;&=\; 0\;,\end{aligned} \tag{8.52}$$

which gives

$$\alpha\,\Omega(\boldsymbol{w}_{\mathrm{MAP}}) \;= \tag{8.53}$$
$$-\sum_{j=1}^{W}\frac{\alpha}{\beta\,\lambda_j\;+\;\alpha}\;+\;W\;=\sum_{j=1}^{W}\frac{\beta\,\lambda_j}{\beta\,\lambda_j\;+\;\alpha}\;=\;\gamma\,.$$

If $\Omega(\boldsymbol{w}_{\mathrm{MAP}})\;=\;0$ then the weight vector is zero, so $\Omega(\boldsymbol{w}_{\mathrm{MAP}})$ shows how far the weights are pushed away from their prior value of zero by the data.

The term $\frac{\beta\,\lambda_j}{\beta\,\lambda_j\,+\,\alpha}$ is in $[0;1]$ and if it is close to one then the data governs this term and terms close to zero are driven by the prior.

The term γ measures the effective number of weights which are driven by the data.

Note, however that the Hessian is not evaluated at the minimum of R_{emp} but at $\boldsymbol{w}_{\mathrm{MAP}}$, therefore the eigenvalues λ_j of $1/2\;\boldsymbol{H}_{\mathrm{emp}}$ are not guaranteed to be positive. Therefore terms $\frac{\beta\,\lambda_j}{\beta\,\lambda_j\,+\,\alpha}$ may be negative because $(\beta\,\lambda_j\;+\;\alpha)$ is positive.

Now we compute the derivative of the negative log-posterior (for α and β) with respect to β.

The derivative of the log of the absolute Hessian determinant with respect to β is

$$\frac{\partial}{\partial\beta}\ln|\boldsymbol{H}|\;=\;\frac{\partial}{\partial\beta}\sum_{j=1}^{W}\ln\left(\beta\,\lambda_j\;+\;\alpha\right)\;= \tag{8.54}$$
$$\sum_{j=1}^{W}\frac{\lambda_j}{\beta\,\lambda_j\;+\;\alpha}\,.$$

Setting the derivative of the negative log-posterior with respect to β to zero gives

$$\beta\,R_{\mathrm{emp}}\;=\;l\;-\;\sum_{j=1}^{W}\frac{\beta\,\lambda_j}{\beta\,\lambda_j\;+\;\alpha}\;=\;l\;-\;\gamma\,. \tag{8.55}$$

The updates for the hyper-parameters are

$$\alpha^{\mathrm{new}}\;=\;\frac{\gamma}{\Omega(\boldsymbol{w}_{\mathrm{MAP}})} \tag{8.56}$$
$$\beta^{\mathrm{new}}\;=\;\frac{l\;-\;\gamma}{R_{\mathrm{emp}}(\boldsymbol{w}_{\mathrm{MAP}})}\,.$$

Now with these new hyper-parameters the new values of $\boldsymbol{w}_{\mathrm{MAP}}$ can be estimated through gradient based methods. Then again the hyper-parameters α and β can be updated and so forth.

If all parameters are well defined $\gamma\;=\;W$ and if much more training examples than weights are present $l\;>>\;W$ then one can use as an approximation for the update formulae

$$\alpha^{\mathrm{new}}\;=\;\frac{W}{\Omega(\boldsymbol{w}_{\mathrm{MAP}})} \tag{8.57}$$
$$\beta^{\mathrm{new}}\;=\;\frac{l}{R_{\mathrm{emp}}(\boldsymbol{w}_{\mathrm{MAP}})}\,.$$

In some textbooks the weight decay term is defined as

$$\Omega(\boldsymbol{w}) \;=\; \frac{1}{2}\,\|\boldsymbol{w}\|^2 \;=\; \frac{1}{2}\,\boldsymbol{w}^T\boldsymbol{w} \;=\; \frac{1}{2}\,\sum_{ij} w_{ij}^2\;, \tag{8.58}$$

which contains the factor $1/2$. In these cases the empirical error contains the factor $1/2$, too. Consequently, the update rules eq. (8.57) for the hyperparameters contain the factor 2 in their denominators.

8.6 Hyper-parameter Selection: Integrate Out

In previous section we started with the posterior which was obtained by integrating out α and β.

$$\begin{aligned} p(\boldsymbol{w} \mid \{\boldsymbol{z}\}) \;&=\; \int_{S_\alpha}\int_{S_\beta} p(\boldsymbol{w},\alpha,\beta \mid \{\boldsymbol{z}\})\; d\alpha\; d\beta \;= \\ &\int_{S_\alpha}\int_{S_\beta} p(\boldsymbol{w} \mid \alpha,\beta,\{\boldsymbol{z}\})\; p(\alpha,\beta \mid \{\boldsymbol{z}\})\; d\alpha\; d\beta \;= \\ &\frac{1}{p_w(\{\boldsymbol{z}\})}\;\int_{S_\alpha}\int_{S_\beta} p(\{\boldsymbol{z}\} \mid \boldsymbol{w},\beta)\; p(\boldsymbol{w} \mid \alpha)\; p(\alpha)\; p(\beta)\; d\alpha\; d\beta\;. \end{aligned} \tag{8.59}$$

Here we used that the hyper-parameters α and β are independent from one another and do not depend on the data: $p(\alpha,\beta \mid \{\boldsymbol{z}\}) \;=\; p(\alpha)\; p(\beta)$. The $\boldsymbol{w}$-posterior $p(\boldsymbol{w} \mid \alpha,\beta,\{\boldsymbol{z}\})$ was expressed through the Bayes formula

$$p(\boldsymbol{w} \mid \{\boldsymbol{z}\}) \;=\; \frac{p(\{\boldsymbol{z}\} \mid \boldsymbol{w})\; p(\boldsymbol{w})}{p_{\boldsymbol{w}}(\{\boldsymbol{z}\})} \tag{8.60}$$

and then the hyper-parameters are removed from densities where the variable is independent from the hyper-parameter . That is $p(\{\boldsymbol{z}\} \mid \boldsymbol{w},\alpha,\beta) \;=\; p(\{\boldsymbol{z}\} \mid \boldsymbol{w},\beta)$ and $p(\boldsymbol{w} \mid \alpha,\beta) \;=\; p(\boldsymbol{w} \mid \alpha)$.

The parameters α and β are scaling parameters. If target and output range is increased then β should re-scale the empirical error.

Similar hold for weight scaling. If the activation functions say $\frac{1}{1+\exp(-\rho\;\mathrm{net})}$ change their slopes ρ then the same network functions are obtained by re-scaling the weights and, therefore, net. That means different weight ranges may implement the same function.

Such parameters as the standard deviation σ for the Gaussians are scale parameters. For scale parameters the prior is often chosen to be non-informative (uniformly) on a logarithmic scale, that is $p(\ln(\alpha))$ and $p(\ln(\beta))$ are constant.

From this follows (note, $p_x(\boldsymbol{x}) \;=\; p_g(g(\boldsymbol{x}))\;\left|\frac{\partial g}{\partial x}\right|$):

$$p(\alpha) \;=\; \frac{1}{\alpha} \tag{8.61}$$

$$p(\beta) \;=\; \frac{1}{\beta}\;. \tag{8.62}$$

We first consider the prior over the weights

$$\begin{aligned} p(\boldsymbol{w}) \;&=\; \int_0^\infty p(\boldsymbol{w} \mid \alpha)\, p(\alpha)\, d\alpha \;= \\ &\int_0^\infty \frac{1}{Z_w(\alpha)}\, \exp(-\frac{1}{2}\,\alpha\,\Omega(\boldsymbol{w}))\, \frac{1}{\alpha}\, d\alpha \;= \\ &(2\,\pi)^{-W/2} \int_0^\infty \exp(-\frac{1}{2}\,\alpha\,\Omega(\boldsymbol{w}))\, \alpha^{W/2-1}\, d\alpha \;=\; \frac{\Gamma(W/2)}{(\pi\,\Omega(\boldsymbol{w}))^{W/2}}\,, \end{aligned} \tag{8.63}$$

where Γ is the gamma function.

Analog we obtain

$$p(\{\boldsymbol{z}\} \mid \boldsymbol{w}) \;=\; \frac{\Gamma(l/2)}{(\pi\, R_{\mathrm{emp}})^{l/2}}\,. \tag{8.64}$$

From these two values and the Bayes formula we can compute the negative log-posterior as

$$-\,\ln p(\boldsymbol{w} \mid \{\boldsymbol{z}\}) \;=\; \frac{l}{2}\,\ln R_{\mathrm{emp}} \;+\; \frac{W}{2}\,\ln\Omega(\boldsymbol{w}) \;+\; \mathrm{const}\,. \tag{8.65}$$

The logarithm of eq. (8.24) gives for negative log-posterior

$$-\,\ln p(\boldsymbol{w} \mid \{\boldsymbol{z}\}) \;=\; \frac{1}{2}\,\beta\, R_{\mathrm{emp}} \;+\; \frac{1}{2}\,\alpha\,\Omega(\boldsymbol{w})\,. \tag{8.66}$$

We compute the gradient of eq. (8.65)

$$-\,\nabla_{Bw} \ln p(\boldsymbol{w} \mid \{\boldsymbol{z}\}) \;=\; \frac{l}{2\, R_{\mathrm{emp}}}\,\nabla_{Bw} R_{\mathrm{emp}} \;+\; \frac{W}{2\,\Omega(\boldsymbol{w})}\,\nabla_{Bw}\Omega(\boldsymbol{w})\,. \tag{8.67}$$

and the gradient of eq. (8.66)

$$-\,\nabla_{Bw} \ln p(\boldsymbol{w} \mid \{\boldsymbol{z}\}) \;=\; \frac{1}{2}\,\beta\,\nabla_{Bw} R_{\mathrm{emp}} \;+\; \frac{1}{2}\,\alpha\,\nabla_{Bw}\Omega(\boldsymbol{w})\,. \tag{8.68}$$

In the last two equations we set the factors in front of $\nabla_{Bw} R_{\mathrm{emp}}$ and in front of $\nabla_{Bw}\Omega(\boldsymbol{w})$ equal. Solving for α and for β gives the update rules from eq. (8.57):

$$\begin{aligned} \alpha^{\mathrm{new}} \;&=\; \frac{W}{\Omega(\boldsymbol{w}_{\mathrm{MAP}})} \\ \beta^{\mathrm{new}} \;&=\; \frac{l}{R_{\mathrm{emp}}(\boldsymbol{w}_{\mathrm{MAP}})}\,. \end{aligned} \tag{8.69}$$

These values of α and β are sometimes called *effective* values because they result from averaging over all values of α or β.

Again an iterative methods first uses the actual α and β to find $\boldsymbol{w}_{\mathrm{MAP}}$ through gradient descent. And then α and β are updated whereafter again the new $\boldsymbol{w}_{\mathrm{MAP}}$ is estimated and so on.

8.7 Model Comparison

Using the Bayes formula we can compare model classes $\mathcal{M}$.

Bayes formula gives

$$p(\mathcal{M} \mid \{\boldsymbol{z}\}) \;=\; \frac{p(\{\boldsymbol{z}\} \mid \mathcal{M})\, p(\mathcal{M})}{p_{\mathcal{M}}(\{\boldsymbol{z}\})}\;, \tag{8.70}$$

where $p(\{\boldsymbol{z}\} \mid \mathcal{M})$ is here the likelihood of the data given a model class but is at the same time the evidence introduced in Section 8.1 for model selection.

The evidence for model selection was defined in eq. (8.10) as

$$p(\{\boldsymbol{z}\} \mid \mathcal{M}) \;=\; \int_W p(\{\boldsymbol{z}\} \mid \boldsymbol{w}, \mathcal{M})\, p(\boldsymbol{w} \mid \mathcal{M})\, d\boldsymbol{w}\;, \tag{8.71}$$

where we only made all probabilities conditioned on the model class $\mathcal{M}$.

If the posterior $p(\boldsymbol{w} \mid \{\boldsymbol{z}\}, \mathcal{M})$ (or according to the Bayes formula equivalently $p(\{\boldsymbol{z}\} \mid \boldsymbol{w}, \mathcal{M})\; p(\boldsymbol{w} \mid \mathcal{M})$) is peaked in weight space then we can approximate the posterior by a box around the maximum a posteriori value $\boldsymbol{w}_{\mathrm{MAP}}$:

$$p(\{\boldsymbol{z}\} \mid \mathcal{M}) \;\approx\; p(\{\boldsymbol{z}\} \mid \boldsymbol{w}_{\mathrm{MAP}}, \mathcal{M})\, p(\boldsymbol{w}_{\mathrm{MAP}} \mid \mathcal{M})\, \Delta\boldsymbol{w}_{\mathrm{MAP}}\;. \tag{8.72}$$

If we assume a Gaussian distribution of the posterior then $\Delta\boldsymbol{w}_{\mathrm{MAP}}$ can be estimated from the Hessian $\boldsymbol{H}$.

We can also use the eq. (8.49)

$$\begin{aligned}\ln p(\{\boldsymbol{z}\} \mid \alpha, \beta) \;=\; &-\frac{1}{2}\,\alpha\,\Omega(\boldsymbol{w}_{\mathrm{MAP}}) \;-\; \frac{1}{2}\,\beta\,R_{\mathrm{emp}} \;-\; \frac{1}{2}\,\ln|\boldsymbol{H}| \;+ \\ &\frac{W}{2}\,\ln\alpha \;+\; \frac{l}{2}\,\ln\beta \;-\; \frac{l}{2}\ln(2\pi)\;,\end{aligned} \tag{8.73}$$

where we insert α_{MAP} and β_{MAP} for α and β.

It can be shown (e.g. [Bishop, 1995] page pp 420/421) that the more exact term is

$$\begin{aligned}\ln p(\{\boldsymbol{z}\} \mid \mathcal{M}) \;=\; &-\frac{\alpha_{\mathrm{MAP}}}{2}\,\Omega(\boldsymbol{w}_{\mathrm{MAP}}) \;-\; \frac{\beta_{\mathrm{MAP}}}{2}\,R_{\mathrm{emp}} \;-\; \frac{1}{2}\,\ln|\boldsymbol{H}| \;+ \\ &\frac{W}{2}\,\ln\alpha_{\mathrm{MAP}} \;+\; \frac{l}{2}\,\ln\beta_{\mathrm{MAP}} \;+\; \ln M! \;+\; 2\,\ln M \;+\; \frac{1}{2}\,\ln\left(\frac{2}{\gamma}\right) \;+ \\ &\frac{1}{2}\,\ln\left(\frac{2}{l\;-\;\gamma}\right)\;.\end{aligned} \tag{8.74}$$

Here the terms in M, the number of hidden units in the network, appear because the posterior is locally approximated but there are equivalent regions in weights space. For each local optimum, equivalent optima exist which are obtained by permutations of the hidden units (thus the term $M!$).

The networks are symmetric so that the signs of incoming and outgoing weights to a hidden unit can be flipped. This gives 2^M weight vectors representing the same function. The hidden units can also be reordered which gives $M!$ orderings. Together we obtain a factor of $(M!\; 2^M)$ equivalent representations through weight vectors of the same function.

8.8 Posterior Sampling

In order to compute the integrals like

$$A(f) \;=\; \int_W f(\boldsymbol{w})\, p(\boldsymbol{w} \mid \{\boldsymbol{z}\})\, d\boldsymbol{w} \tag{8.75}$$

we can sample weight vectors $\boldsymbol{w}_i$ to estimate

$$A(f) \;\approx\; \frac{1}{L} \sum_{i=1}^{L} f(\boldsymbol{w}_i)\;, \tag{8.76}$$

where the $\boldsymbol{w}_i$ are sampled according to $p(\boldsymbol{w} \mid \{\boldsymbol{z}\})$.

Because we cannot easily sample from $p(\boldsymbol{w} \mid \{\boldsymbol{z}\})$ we use a simpler distribution $q(\boldsymbol{w})$ where we can sample from. We obtain

$$A(f) \;=\; \int_W f(\boldsymbol{w})\, \frac{p(\boldsymbol{w} \mid \{\boldsymbol{z}\})}{q(\boldsymbol{w})}\, q(\boldsymbol{w})\, d\boldsymbol{w} \tag{8.77}$$

which is an expectation in $q(\boldsymbol{w})$. This expectation can be approximated by

$$A(f) \;\approx\; \frac{1}{L} \sum_{i=1}^{L} f(\boldsymbol{w}_i) \frac{p(\boldsymbol{w}_i \mid \{\boldsymbol{z}\})}{q(\boldsymbol{w}_i)}\;, \tag{8.78}$$

where now the $\boldsymbol{w}_i$ are sampled according to $q(\boldsymbol{w})$.

To avoid the normalization of $p(\boldsymbol{w} \mid \{\boldsymbol{z}\})$ which also includes integrations which are difficult to perform, the following term can be used

$$A(f) \;\approx\; \frac{\sum_{i=1}^{L} f(\boldsymbol{w}_i)\, \tilde{p}(\boldsymbol{w}_i \mid \{\boldsymbol{z}\}) \;/\; q(\boldsymbol{w}_i)}{\sum_{i=1}^{L} \tilde{p}(\boldsymbol{w}_i \mid \{\boldsymbol{z}\}) \;/\; q(\boldsymbol{w}_i)}\;, \tag{8.79}$$

where $\tilde{p}(\boldsymbol{w} \mid \{\boldsymbol{z}\})$ is the unnormalized posterior, i.e. the product of the likelihood and the prior.

This approach is called *importance sampling*.

Because $p(\boldsymbol{w} \mid \{\boldsymbol{z}\})$ is in general very small we must guarantee to sample in regions with large probability mass. This can be done by using *Markov Chain Monte Carlo* methods where regions with large mass are only left with low probability.

One method which improves random walk in a way that regions with large $p(\boldsymbol{w} \mid \{\boldsymbol{z}\})$ are sampled is called *Metropolis* algorithm. The Metropolis algorithm can be characterized as follows

$$\begin{aligned} &\boldsymbol{w}^{\text{new}} \;=\; \boldsymbol{w}^{\text{candidate}} \text{ with probability} \\ &\begin{cases} 1 & \text{if} \quad p(\boldsymbol{w}^{\text{candidate}} \mid \{\boldsymbol{z}\}) \;>\; p(\boldsymbol{w}^{\text{old}} \mid \{\boldsymbol{z}\}) \\ \frac{p(\boldsymbol{w}^{\text{candidate}} \mid \{\boldsymbol{z}\})}{p(\boldsymbol{w}^{\text{old}} \mid \{\boldsymbol{z}\})} & \text{if} \quad p(\boldsymbol{w}^{\text{candidate}} \mid \{\boldsymbol{z}\}) \;<\; p(\boldsymbol{w}^{\text{old}} \mid \{\boldsymbol{z}\}) \end{cases}. \end{aligned} \tag{8.80}$$

Also *simulated annealing* can be used to estimate the expectation under the posterior which is similar to the Metropolis algorithm.

These sampling methods are discussed in [Neal, 1996] which is recommended for further reading.

Chapter 9

Linear Models

For linear regression we express one variable y as a linear function of the other variable x:

$$y \;=\; a \;+\; b\,x\,. \tag{9.1}$$

If fitting a linear function (a line), that is, to find optimal parameters a and b, the objective was the *sum of the squared deviations* between the y values and the regression line. The line that optimized this criterion is the *least squares line*. We now generalize this approach to the multivariate case. We already noticed in the simple bivariate case that interchanging the role of x and y may result in a different functional dependency between x and y.

The scalar variable y is called the *dependent variable*. We now generalize x to a vector of features $\boldsymbol{x}$ with components x_j which are called *explanatory variables*, *independent variables*, *regressors*, or *features*.

The estimation of y from a vector of explanatory variables $\boldsymbol{x}$ is called *multiple linear regression*. If y is generalized to a vector $\boldsymbol{y}$, then this is called *multivariate linear regression*. We focus on multiple linear regression, that is, the case where multiple features are summarized in the vector $\boldsymbol{x}$.

9.1 Linear Regression

9.1.1 The Linear Model

We assume to have m features $x_1, \ldots, x_m$ which are summarized by the vector $\boldsymbol{x} = (x_1, \ldots, x_m)$. The general form of a linear model is

$$y \;=\; \beta_0 \;+\; \sum_{j=1}^{m} x_j\,\beta_j \;+\; \epsilon\,. \tag{9.2}$$

This model has $(m+1)$ *parameters* $\beta_0, \beta_1, \ldots, \beta_m$ which are unknown and have to be estimated. ϵ is an additive *noise* or *error* term which accounts for the difference between the predicted value and the observed outcome y.

To simplify the notation, we extend the vector of features by a one: $\boldsymbol{x} = (1, x_1, \ldots, x_m)$. Consequently, we use the parameter vector $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_m)$ to denote the linear model in vector notation by:

$$y \;=\; \boldsymbol{x}^T\boldsymbol{\beta} \;+\; \epsilon\,. \tag{9.3}$$

If the constant 1 is counted as independent variable, then $(m+1)$ is both the number of parameters and the number of the independent variables. In some textbooks this might be confusing because m and $(m+1)$ may appear in the formulas.

We assume to have n observations $\{(y_i, \boldsymbol{x}_i) \mid 1 \leq i \leq n\}$. The y_i are summarized by a vector $\boldsymbol{y}$, the $\boldsymbol{x}_i$ in a matrix $\boldsymbol{X} \in \mathbb{R}^{n \times (m+1)}$ ($\boldsymbol{x}_i$ is the i-th row of $\boldsymbol{X}$), and the ϵ_i in a vector $\boldsymbol{\epsilon} \in \mathbb{R}^n$. For n observations we obtain the matrix equation:

$$\boldsymbol{y} \;=\; \boldsymbol{X}\,\boldsymbol{\beta} \;+\; \boldsymbol{\epsilon}\,. \tag{9.4}$$

9.1.2 Interpretations and Assumptions

The linear model can be applied in different frameworks, where the independent variables have different interpretations and assumptions. The parameter estimation depends only on the noise assumption. The task which must be solved or the study design, from which the data comes, determines interpretations, assumptions, and design of the dependent variables.

9.1.2.1 Interpretations

One of the main differences is whether the explanatory / independent variables are random variables sampled together with the dependent variable or constants which are fixed according to the task to solve.

Our model for bivariate data is a model with one independent variable:

$$y_i \;=\; \beta_0 \;+\; \beta_1 x_i \;+\; \epsilon_i\,, \tag{9.5}$$

where β_0 is the y-intercept and β_1 the slope.

An example with 7 observations in matrix notation is

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{pmatrix} = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \\ 1 & x_4 \\ 1 & x_5 \\ 1 & x_6 \\ 1 & x_7 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \\ \epsilon_6 \\ \epsilon_7 \end{pmatrix}. \tag{9.6}$$

An example for a model with two regressors is

$$y_i \;=\; \beta_0 \;+\; \beta_1 x_{i1} \;+\; \beta_2 x_{i2} \;+\; \epsilon_i\,. \tag{9.7}$$

For 7 observations this model leads to following matrix equation:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & x_{12} \\ 1 & x_{21} & x_{22} \\ 1 & x_{31} & x_{32} \\ 1 & x_{41} & x_{42} \\ 1 & x_{51} & x_{52} \\ 1 & x_{61} & x_{62} \\ 1 & x_{71} & x_{72} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix} + \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \\ \epsilon_6 \\ \epsilon_7 \end{pmatrix}. \tag{9.8}$$

We show an example for a *cell means model* or a *one-way ANOVA* model. We assume that from the study design we know 3 groups and want to find the mean for each group. The model is

$$y_{gi} = \beta_g + \epsilon_{gi} , \tag{9.9}$$

where β_g is the mean of group g. For example we have three groups and 3 examples for the first group, and two examples for the second and third group. In matrix notation this example with 7 observations and three groups is

$$\begin{pmatrix} y_{11} \\ y_{12} \\ y_{13} \\ y_{21} \\ y_{22} \\ y_{31} \\ y_{32} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{pmatrix} + \begin{pmatrix} \epsilon_{11} \\ \epsilon_{12} \\ \epsilon_{13} \\ \epsilon_{21} \\ \epsilon_{22} \\ \epsilon_{31} \\ \epsilon_{32} \end{pmatrix} . \tag{9.10}$$

We present another example of an ANOVA model which is again a one-way ANOVA model. We are interested in the offset from a reference group. This model is typically for a study design with one *control group* or *reference group* and multiple *treatment groups*. The offset of group g from group 1 is denoted by β_g, thus $\beta_1 = 0$.

$$y_{gi} = \beta_0 + \beta_g + \epsilon_{gi} . \tag{9.11}$$

For three groups and 7 observations (3 in group $g = 1$, 2 in group $g = 2$, and 2 in group $g = 3$), the matrix equation is

$$\begin{pmatrix} y_{11} \\ y_{12} \\ y_{13} \\ y_{21} \\ y_{22} \\ y_{31} \\ y_{32} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_2 \\ \beta_3 \end{pmatrix} + \begin{pmatrix} \epsilon_{11} \\ \epsilon_{12} \\ \epsilon_{13} \\ \epsilon_{21} \\ \epsilon_{22} \\ \epsilon_{31} \\ \epsilon_{32} \end{pmatrix} . \tag{9.12}$$

The mean of the reference group is β_0 and β_g is the difference to the reference group. In this design we know that $\beta_1 = 0$, therefore we did not include it.

A more complicated model is the *two-way ANOVA* model which has two known groupings or two known *factors*. Each observation belongs to a group of the first grouping and at the same time to a group of the second grouping, that is, each observation is characterized by two factors.

The model is

$$y_{ghi} = \beta_0 + \beta_g + \alpha_h + (\beta\alpha)_{gh} + \epsilon_{ghi} , \tag{9.13}$$

where g denotes the first factor (grouping) and h the second factor (grouping), and i indicates the replicate for this combination of factors. The term $(\beta\alpha)_{gh}$ accounts for *interaction effects* between the factors, while β_g and α_h are the *main effects* of the factors.

This model has too many parameters to possess a unique solution for the parameters if each combination of groups is observed exactly once. One observation per combination of groups is the minimal data set. Consequently, noise free observations can be modeled by more than one set of parameters. Even for a large number of noise free observations the situation does not change: there is more that one set of parameters which gives the optimal solution. The solution to this over-parametrization is to include additional constraints which use up some degrees of freedom. These constraints are that either

- the main and interaction effect parameters sum to zero for each index (*sum-to-zero constraint*) or
- all parameters that contain the index 1 are zero (*corner point parametrization*).

With the corner point parametrization we have

$$\begin{aligned} \alpha_1 &= 0 && (9.14)\\ \beta_1 &= 0 && (9.15)\\ (\beta\alpha)_{1h} &= 0 && (9.16)\\ (\beta\alpha)_{g1} &= 0\,. && (9.17) \end{aligned}$$

We present an example, where the first factor has 3 levels $1 \leq g \leq 3$, the second factor has 2 levels $1 \leq h \leq 2$, and for each combination of factors there are two replicates $1 \leq i \leq 2$. In matrix notation we have

$$\begin{pmatrix} y_{111}\\ y_{112}\\ y_{211}\\ y_{212}\\ y_{311}\\ y_{312}\\ y_{121}\\ y_{122}\\ y_{221}\\ y_{222}\\ y_{321}\\ y_{322} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0\\ 1 & 0 & 0 & 0 & 0 & 0\\ 1 & 0 & 0 & 1 & 0 & 0\\ 1 & 0 & 0 & 1 & 0 & 0\\ 1 & 1 & 0 & 0 & 0 & 0\\ 1 & 1 & 0 & 0 & 0 & 0\\ 1 & 1 & 0 & 1 & 1 & 0\\ 1 & 1 & 0 & 1 & 1 & 0\\ 1 & 0 & 1 & 0 & 0 & 0\\ 1 & 0 & 1 & 0 & 0 & 0\\ 1 & 0 & 1 & 1 & 0 & 1\\ 1 & 0 & 1 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} \beta_0\\ \beta_2\\ \beta_3\\ \alpha_2\\ (\beta\alpha)_{22}\\ (\beta\alpha)_{32} \end{pmatrix} + \begin{pmatrix} \epsilon_{111}\\ \epsilon_{112}\\ \epsilon_{211}\\ \epsilon_{212}\\ \epsilon_{311}\\ \epsilon_{312}\\ \epsilon_{121}\\ \epsilon_{122}\\ \epsilon_{221}\\ \epsilon_{222}\\ \epsilon_{321}\\ \epsilon_{322} \end{pmatrix}. \qquad (9.18)$$

9.1.2.2 Assumptions

The standard linear regression model has the following assumptions:

- **Strict exogeneity.** The errors have zero mean conditioned on the regressors:

$$\mathrm{E}(\boldsymbol{\epsilon} \mid \boldsymbol{X}) = \mathbf{0}\,. \qquad (9.19)$$

  Therefore the errors have zero mean $\mathrm{E}(\boldsymbol{\epsilon}) = \mathbf{0}$ and they are independent of the regressors $\mathrm{E}(\boldsymbol{X}^T\boldsymbol{\epsilon}) = \mathbf{0}$.

- **Linear independence.** The regressors must be linearly independent almost surely.

$$\Pr(\operatorname{rank}(\boldsymbol{X}) = m + 1) \;=\; 1\,. \tag{9.20}$$

  If $\boldsymbol{X}$ does not have full rank, then estimation is only possible in the subspace spanned by the $\boldsymbol{x}_i$. To obtain theoretical properties of the estimator, the second moments should be finite to ensure $\mathrm{E}(\frac{1}{n}\boldsymbol{X}^T\boldsymbol{X})$ to be finite and positive definite.

- **Spherical errors.**

$$\operatorname{Var}(\boldsymbol{\epsilon} \mid \boldsymbol{X}) \;=\; \sigma^2\, \boldsymbol{I}_n\,. \tag{9.21}$$

  Therefore the error has the same variance in each observation $\mathrm{E}(\epsilon_i^2 \mid \boldsymbol{X}) = \sigma^2$ (*homoscedasticity*). If this is violated, then a weighted least squared estimate should be used. Further the errors of different observations are not correlated $\mathrm{E}(\epsilon_i\epsilon_k \mid \boldsymbol{X}) = 0$ for $i \neq k$ (no autocorrelation).

Normality of the Errors. For further theoretical properties often the errors are assumed to be normally distributed given the regressors:

$$\boldsymbol{\epsilon} \mid \boldsymbol{X} \;\sim\; \mathcal{N}\big(\mathbf{0}\,,\; \sigma^2\, \boldsymbol{I}_n\big) \tag{9.22}$$

In this case the estimator is the maximum likelihood estimator, which is asymptotically efficient, that is, it is asymptotically the best possible estimator. Further it is possible to test hypotheses based on the normality assumption because the distribution of the estimator is known.

In many applications the samples $\{(y_i, \boldsymbol{x}_i)\}$ are assumed to be *independent and identically distributed* (iid). The samples are independent,

$$\Pr\left((y_i, \boldsymbol{x}_i) \mid (y_1, \boldsymbol{x}_1), \ldots, (y_{i-1}, \boldsymbol{x}_{i-1}), (y_{i+1}, \boldsymbol{x}_{i+1}), \ldots, (y_n, \boldsymbol{x}_n)\right) \;=\; \Pr\left((y_i, \boldsymbol{x}_i)\right)\,, \tag{9.23}$$

and are identically distributed,

$$\Pr\left((y_i, \boldsymbol{x}_i)\right) \;=\; \Pr\left((y_k, \boldsymbol{x}_k)\right)\,. \tag{9.24}$$

For iid samples the assumptions simplify to

- **Exogeneity.** Each error has zero mean conditioned on the regressor:

$$\mathrm{E}(\epsilon_i \mid \boldsymbol{x}_i) \;=\; 0\,. \tag{9.25}$$

- **Linear independence.** The covariance matrix

$$\operatorname{Var}(\boldsymbol{x}) \;=\; \mathrm{E}(\boldsymbol{x}\boldsymbol{x}^T) \;=\; \frac{1}{n}\sum_{i=1}^{n} \mathrm{E}(\boldsymbol{x}_i\boldsymbol{x}_i^T) \;=\; \mathrm{E}(\frac{1}{n}\boldsymbol{X}^T\boldsymbol{X})\,. \tag{9.26}$$

  must have full rank.

- **Homoscedasticity.**

$$\operatorname{Var}(\epsilon_i \mid \boldsymbol{x}_i) \;=\; \sigma^2\,. \tag{9.27}$$

For *time series models* the iid assumption does not hold. In this case the assumptions are

- the stochastic process $\{(y_i, \boldsymbol{x}_i)\}$ is stationary (probability distribution is the same when shifted in time) and ergodic (time average is the population average);
- the regressors are predetermined: $\mathrm{E}(\boldsymbol{x}_i \epsilon_i) = 0$ for all $i = 1, \ldots, n$;
- the $(m+1) \times (m+1)$ matrix $\mathrm{E}(\boldsymbol{x}_i \boldsymbol{x}_i^T)$ is of full rank;
- the sequence $\{\boldsymbol{x}_i \epsilon_i\}$ is a martingale difference sequence (zero mean given the past) with existing second moments $\mathrm{E}(\epsilon_i^2 \boldsymbol{x}_i \boldsymbol{x}_i^T)$.

Linear models for time series are called *autoregressive models*.

9.1.3 Least Squares Parameter Estimation

The *residual* for the i-th observation is

$$r_i \;=\; y_i \;-\; \boldsymbol{x}_i^T \, \tilde{\boldsymbol{\beta}} \,, \tag{9.28}$$

where $\tilde{\boldsymbol{\beta}}$ is a candidate for the parameter vector $\boldsymbol{\beta}$. The residual r_i measures how well y_i is predicted by the linear model with parameters $\tilde{\boldsymbol{\beta}}$.

To assess how well all observations are fitted simultaneously by a linear model, the squared residuals of all observation are summed up to S, which is called the *sum of squared residuals* (SSR), the *error sum of squares* (ESS), or *residual sum of squares* (RSS):

$$S(\tilde{\boldsymbol{\beta}}) \;=\; \sum_{i=1}^{n} r_i^2 \;=\; \sum_{i=1}^{n} \left(y_i \;-\; \boldsymbol{x}_i^T \, \tilde{\boldsymbol{\beta}} \right)^2 \;=\; \left(\boldsymbol{y} \;-\; \boldsymbol{X} \tilde{\boldsymbol{\beta}} \right)^T \left(\boldsymbol{y} \;-\; \boldsymbol{X} \, \tilde{\boldsymbol{\beta}} \right) \,. \tag{9.29}$$

The *least squares estimator* $\hat{\boldsymbol{\beta}}$ for $\boldsymbol{\beta}$ minimizes $S(\tilde{\boldsymbol{\beta}})$:

$$\hat{\boldsymbol{\beta}} \;= \arg\min_{\tilde{\boldsymbol{\beta}}} S(\tilde{\boldsymbol{\beta}}) \;=\; \left(\boldsymbol{X}^T \boldsymbol{X} \right)^{-1} \boldsymbol{X}^T \boldsymbol{y} \,. \tag{9.30}$$

The solution is obtained by setting the derivative of $S(\tilde{\boldsymbol{\beta}})$ with respect to the parameter vector $\tilde{\boldsymbol{\beta}}$ to zero:

$$\frac{\partial S(\tilde{\boldsymbol{\beta}})}{\partial \tilde{\boldsymbol{\beta}}} \;=\; 2 \, \boldsymbol{X}^T \left(\boldsymbol{y} \;-\; \boldsymbol{X} \, \tilde{\boldsymbol{\beta}} \right) \;=\; \boldsymbol{0} \,. \tag{9.31}$$

The matrix $\boldsymbol{X}^+ = \left(\boldsymbol{X}^T \boldsymbol{X} \right)^{-1} \boldsymbol{X}^T$ is called the *pseudo inverse* of the matrix $\boldsymbol{X}$ because $\boldsymbol{X}^+ \boldsymbol{X} = \boldsymbol{I}_m$.

The least squares estimator is the minimal variance linear unbiased estimator (MVLUE), that is, it is the best linear unbiased estimator. Under the normality assumption for the errors, the least squares estimator is the maximum likelihood estimator (MLE).

Concerning notation and the parameter vector, we have the true parameter vector $\boldsymbol{\beta}$, a candidate parameter vector or a variable $\tilde{\boldsymbol{\beta}}$, and an estimator $\hat{\boldsymbol{\beta}}$, which is in our case the least squares estimator.

9.1.4 Evaluation and Interpretation of the Estimation

9.1.4.1 Residuals and Error Variance

The estimated values for y are

$$\hat{\boldsymbol{y}} \;=\; \boldsymbol{X}\hat{\boldsymbol{\beta}} \;=\; \boldsymbol{X}\;\boldsymbol{X}^{+}\boldsymbol{y} \;=\; \boldsymbol{X}\;\left(\boldsymbol{X}^{T}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{T}\boldsymbol{y} \;=\; \boldsymbol{P}\boldsymbol{y}\,, \tag{9.32}$$

where

$$\boldsymbol{P} \;=\; \boldsymbol{X}\;\left(\boldsymbol{X}^{T}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{T} \tag{9.33}$$

is a projection matrix, the *hat matrix* as it puts a hat on $\boldsymbol{y}$. We have $\boldsymbol{P}\boldsymbol{X} = \boldsymbol{X}$ and $\boldsymbol{P}^2 = \boldsymbol{P}$.

The minimal residuals or the least squares residuals are

$$\hat{\boldsymbol{\epsilon}} \;=\; \boldsymbol{y} \;-\; \boldsymbol{X}\hat{\boldsymbol{\beta}} \;=\; \left(\boldsymbol{I}_n \;-\; \boldsymbol{P}\right)\;\boldsymbol{y} \;=\; \left(\boldsymbol{I}_n \;-\; \boldsymbol{P}\right)\;\boldsymbol{\epsilon}\,. \tag{9.34}$$

Both $\boldsymbol{P}$ and $(\boldsymbol{I}_n \;-\; \boldsymbol{P})$ are symmetric and idempotent ($\boldsymbol{P} = \boldsymbol{P}^2$).

$S(\hat{\boldsymbol{\beta}})$ is the sum of squared residuals for the least squares estimator $\hat{\boldsymbol{\beta}}$, which can be used to estimate σ^2.

$$\begin{aligned} S(\hat{\boldsymbol{\beta}}) \;&=\; \left(\boldsymbol{y} \;-\; \boldsymbol{X}\hat{\boldsymbol{\beta}}\right)^{T}\left(\boldsymbol{y} \;-\; \boldsymbol{X}\hat{\boldsymbol{\beta}}\right) \\ &=\; \boldsymbol{y}^{T}\boldsymbol{y} \;-\; 2\,\hat{\boldsymbol{\beta}}^{T}\boldsymbol{X}^{T}\boldsymbol{y} \;+\; \hat{\boldsymbol{\beta}}^{T}\boldsymbol{X}^{T}\boldsymbol{X}\hat{\boldsymbol{\beta}} \\ &=\; \boldsymbol{y}^{T}\boldsymbol{y} \;-\; \hat{\boldsymbol{\beta}}^{T}\boldsymbol{X}^{T}\boldsymbol{y} \;=\; \hat{\boldsymbol{\epsilon}}^{T}\boldsymbol{y}\,, \end{aligned} \tag{9.35}$$

where we used $\boldsymbol{X}^{T}\boldsymbol{X}\hat{\boldsymbol{\beta}} = \boldsymbol{X}^{T}\boldsymbol{y}$.

The least squares estimate for σ^2 is

$$s^2 \;=\; \frac{1}{n-m-1}S(\hat{\boldsymbol{\beta}}) \tag{9.36}$$

and the maximum likelihood estimate for σ^2 is

$$\hat{\sigma}^2 \;=\; \frac{1}{n}S(\hat{\boldsymbol{\beta}})\,. \tag{9.37}$$

The estimate s^2 is an unbiased estimator for σ^2 while the ML estimate $\hat{\sigma}^2$ is biased. Both are asymptotically optimal, that is, unbiased and efficient. The estimator with minimal mean squared error is

$$\tilde{\sigma}^2 \;=\; \frac{1}{n-m+1}S(\hat{\boldsymbol{\beta}})\,. \tag{9.38}$$

The covariance of the vector of residuals is

$$\begin{aligned} \mathrm{E}(\hat{\boldsymbol{\epsilon}}\hat{\boldsymbol{\epsilon}}^{T}) \;&=\; \left(\boldsymbol{I}_n \;-\; \boldsymbol{P}\right)\;\mathrm{E}(\boldsymbol{\epsilon}\,\boldsymbol{\epsilon}^{T})\;\left(\boldsymbol{I}_n \;-\; \boldsymbol{P}\right) \\ &=\; \sigma^2\left(\boldsymbol{I}_n \;-\; \boldsymbol{P}\right)^2 \;=\; \sigma^2\left(\boldsymbol{I}_n \;-\; \boldsymbol{P}\right)\,, \end{aligned} \tag{9.39}$$

where we used Eq. (9.34). Further we assumed that the residuals have the covariance structure $\sigma^2\boldsymbol{I}_n$ as the assumptions state.

9.1.4.2 Coefficient of determination

The *coefficient of determination* R^2 is the ratio of the variance "explained" by the model to the "total" variance of the dependent variable y:

$$R^2 = \frac{\sum_{i=1}^{n} \left(\hat{y}_i - \overline{\hat{y}}\right)^2}{\sum_{i=1}^{n} \left(y_i - \overline{y}\right)^2} = \frac{\boldsymbol{y}^T \boldsymbol{P}^T \boldsymbol{L}\, \boldsymbol{P}\, \boldsymbol{y}}{\boldsymbol{y}^T \boldsymbol{L}\, \boldsymbol{y}} = 1 - \frac{\boldsymbol{y}^T (\boldsymbol{I} - \boldsymbol{P})\, \boldsymbol{y}}{\boldsymbol{y}^T \boldsymbol{L}\, \boldsymbol{y}} = 1 - \frac{\mathrm{SSR}}{\mathrm{TSS}}, \tag{9.40}$$

where $\boldsymbol{L} = \boldsymbol{I}_n - (1/n)\mathbf{1}\,\mathbf{1}^T$, with $\mathbf{1}$ as the n-dimensional vector of ones. $\boldsymbol{L}$ is the centering matrix which subtracts the mean from each variable. "TSS" is the total sum of squares for the dependent variable and "SSR" the sum of squared residuals denoted by S. To account for a constant offset, that is, the regression intercept, the data matrix $\boldsymbol{X}$ should contain a column vector of ones. In that case R^2 is between 0 and 1, the closer R^2 is to 1, the better the fit.

9.1.4.3 Outliers and Influential Observations

An *outlier* is an observation which is worse fitted by the model than other observations, that is, it has large error compared to other errors. An *influential observation* is an observation which has large effect on the model fitting or has large effect on the inferences based on the model. Outliers can be influential observations but need not be. Analogously, influential observations can be outliers but need not be.

9.1.4.3.1 Outliers. We define the *standardized residuals* or *studentized residuals* ρ_i as

$$\rho_i = \frac{\hat{\epsilon}_i}{\hat{\sigma}\sqrt{1 - P_{ii}}}. \tag{9.41}$$

P_{ii} are the diagonal elements of the hat matrix $\boldsymbol{P}$ defined in Eq. (9.33) and $\hat{\sigma}^2$ is an estimate of σ^2. The standardized residuals can be used to check the fitted model and whether the model assumptions are met or not. Such assumptions are linearity, normality, and independence. In particular, an outlier may be detected via the standardized residuals ρ_i because they have the same variance.

Another way is to do leave-one-out regression, where observation $(y_i, \boldsymbol{x}_i)$ is removed from the data set and a least squares estimate performed on the remaining $(n-1)$ observations. The least squares estimator $\hat{\boldsymbol{\beta}}_{(i)}$ on the data set where $(y_i, \boldsymbol{x}_i)$ is left out is:

$$\hat{\boldsymbol{\beta}}_{(i)} = \hat{\boldsymbol{\beta}} - \frac{\hat{\epsilon}_i}{1 - P_{ii}} (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{x}_i. \tag{9.42}$$

Therefore the residual of the left-out observation is

$$\hat{\epsilon}_{(i)} = \frac{\hat{\epsilon}_i}{1 - P_{ii}}. \tag{9.43}$$

Plotting the leave-one-out residuals against the standard residuals may reveal outliers. However outliers can already be detected by $(1 - P_{ii})$: the closer P_{ii} to one, the more likely is the i-th observation an outlier.

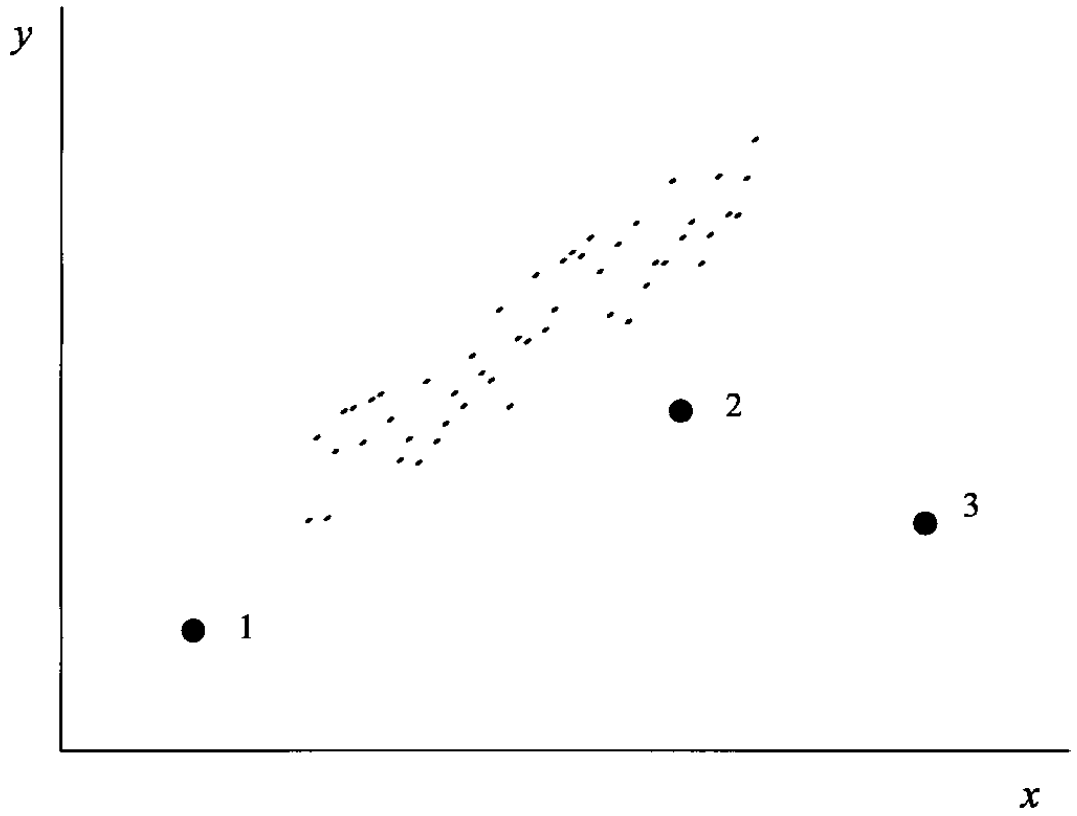

Figure 9.1: Simple linear regression with three marked outliers. Observations 1 and 3 deviate in the x-direction. Observations 2 and 3 appear as outliers in the y-direction. Observation 1 is not influential but observation 3 is. Figure from Rencher and Schaalje [2008].

9.1.4.3.2 Influential Observations. An influential observation $(y_i, \boldsymbol{x}_i)$ has large effect on the estimates $\hat{\boldsymbol{\beta}}$ or $\boldsymbol{X}\hat{\boldsymbol{\beta}}$. This means that the estimates are considerably different if observation $(y_i, \boldsymbol{x}_i)$ is removed. Fig. 9.1 shows a simple linear regression with three marked outliers. Observations 1 and 3 deviate in the x-direction. Observations 2 and 3 appear as outliers in the y-direction. Observation 1 is located close to the regression line which would be obtained without it. Thus, it is not influential. However, observation 3 has a large effect on the regression line compared to regression if it is removed. Thus, observation 3 is influential. Observation 2 is influential to some degree but much less than observation 3.

With the hat matrix $\boldsymbol{P}$ we can express $\hat{\boldsymbol{y}}$ as $\hat{\boldsymbol{y}} = \boldsymbol{P}\boldsymbol{y}$, therefore

$$\hat{y}_i \;=\; \sum_{j=1}^{n} P_{ij} y_j \;=\; P_{ii} y_i \;+\; \sum_{j,j\neq i} P_{ij} y_j \;. \tag{9.44}$$

If P_{ii} is large, then P_{ij} for $j \neq i$ is small because $\boldsymbol{P}$ is idempotent. Therefore P_{ii} is called the *leverage* of y_i, that is, how much y_i contributes to its estimate.

The influence of the i-th observation can be measured by *Cook's distance*

$$D_i \;=\; \frac{\rho_i^2}{m+1} \, \frac{P_{ii}}{1 \;-\; P_{ii}} \;. \tag{9.45}$$

If D_i is large, the observation $(y_i, \boldsymbol{x}_i)$ has considerable influence on the estimates.

This distance can be written as

$$\begin{aligned} D_i &= \frac{(\hat{\boldsymbol{\beta}}_{(i)} - \hat{\boldsymbol{\beta}})^T \boldsymbol{X}^T \boldsymbol{X} (\hat{\boldsymbol{\beta}}_{(i)} - \hat{\boldsymbol{\beta}})}{(m+1)\, s^2} \\ &= \frac{(\boldsymbol{X}\, \hat{\boldsymbol{\beta}}_{(i)} - \boldsymbol{X}\, \hat{\boldsymbol{\beta}})^T (\boldsymbol{X}\, \hat{\boldsymbol{\beta}}_{(i)} - \boldsymbol{X}\, \hat{\boldsymbol{\beta}})}{(m+1)\, s^2} \\ &= \frac{(\hat{\boldsymbol{y}}_{(i)} - \hat{\boldsymbol{y}})^T (\hat{\boldsymbol{y}}_{(i)} - \hat{\boldsymbol{y}})}{(m+1)\, s^2} \,. \end{aligned} \tag{9.46}$$

Thus, D_i is proportional to the Euclidean distance between the estimate $\hat{\boldsymbol{y}}$ using all data and the estimate $\hat{\boldsymbol{y}}_{(i)}$ where observation $(y_i, \boldsymbol{x}_i)$ is removed.

More complicated but giving the same result is first to perform a leave-one-out estimate, where each observation $(y_i, \boldsymbol{x}_i)$ is left out, and, subsequently, compare the estimated values to the estimated values with all data.

9.1.5 Confidence Intervals for Parameters and Prediction

9.1.5.1 Normally Distributed Error Terms

If the error terms are normally distributed then the least squares estimator is a maximum likelihood estimator which is asymptotically normally distributed:

$$\hat{\boldsymbol{\beta}} \xrightarrow{d} \mathcal{N}\big(\boldsymbol{\beta},\, \sigma^2\, (\boldsymbol{X}^T\, \boldsymbol{X})^{-1}\big)\,, \tag{9.47}$$

where $\xrightarrow{d}$ means convergence in distribution. This means that the distribution of $\hat{\boldsymbol{\beta}}$ is increasingly (with number of samples) better modeled by the normal distribution. This maximum likelihood estimator is efficient and unbiased, that is, it reaches the Cramer-Rao lower bound, and therefore is optimal for unbiased estimators.

This asymptotic distribution gives an approximated two-sided confidence interval for the j-th component of the vector $\hat{\boldsymbol{\beta}}$:

$$\beta_j \in \left[\, \hat{\beta}_j \pm t_{\alpha/2,n-m-1}\, s\, \sqrt{\left[(\boldsymbol{X}^T \boldsymbol{X})^{-1}\right]_{jj}} \,\right] \tag{9.48}$$

where $t_{\alpha/2,n-m-1}$ is the upper $\alpha/2$ percentage point of the central t-distribution and α is the desired significance level of the test (probability of rejecting H_0 when it is true). This means we are $100(1-\alpha)\%$ confident that the interval contains the true β_j. It is important to known that the confidence intervals do not hold simultaneously for all β_j.

The confidence interval for the noise free prediction is

$$\boldsymbol{x}^T \boldsymbol{\beta} \in \left[\, \boldsymbol{x}^T \hat{\boldsymbol{\beta}} \pm t_{\alpha/2,n-m-1}\, s\, \sqrt{\boldsymbol{x}^T\, (\boldsymbol{X}^T \boldsymbol{X})^{-1}\, \boldsymbol{x}} \,\right]. \tag{9.49}$$

Again this holds only for a single prediction but not for multiple simultaneous predictions.

If noise ϵ is added then we have a confidence interval for the prediction:

$$y \in \left[\hat{y} \pm t_{\alpha/2,n-m-1}\, s \sqrt{1 + \boldsymbol{x}^T (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{x}} \right], \tag{9.50}$$

where $\hat{y} = \boldsymbol{x}^T \hat{\boldsymbol{\beta}}$. Of course, this holds only for a single prediction but not for multiple simultaneous predictions.

The estimator s^2 is distributed according to a chi-squared distribution:

$$s^2 \sim \frac{\sigma^2}{n-m-1} \chi^2_{n-m-1} \tag{9.51}$$

The variance is $2\sigma^4/(n-m-1)$ and does not attain the Cramer-Rao lower bound $2\sigma^4/n$. There is no unbiased estimator with lower variance, this means that the estimator is the minimal variance unbiased estimator (MVUE). The estimator $\tilde{\sigma}^2$ from above has the minimal mean squared error. An advantage of s^2 is that it is independent of $\hat{\boldsymbol{\beta}}$ which helps for tests based on these estimators.

A confidence interval for σ^2 is given by

$$\frac{(n-m-1)\, s^2}{\chi^2_{\alpha/2,n-m-1}} \leq \sigma^2 \leq \frac{(n-m-1)\, s^2}{\chi^2_{1-\alpha/2,n-m-1}} \tag{9.52}$$

at $100(1-\alpha)\%$ confidence.

9.1.5.2 Error Term Distribution Unknown

For unknown error distributions we still known that the least squares estimator for $\boldsymbol{\beta}$ is consistent, that is, $\hat{\boldsymbol{\beta}}$ converges in probability to the true value $\boldsymbol{\beta}$. The following results are obtained by the law of large number and the central limit theorem. The estimator $\hat{\boldsymbol{\beta}}$ is asymptotically normally distributed:

$$\sqrt{n}\,(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) \xrightarrow{d} \mathcal{N}\big(\mathbf{0},\, \sigma^2\, (\boldsymbol{X}^T \boldsymbol{X})^{-1}\big), \tag{9.53}$$

which gives

$$\hat{\boldsymbol{\beta}} \sim_a \mathcal{N}\big(\boldsymbol{\beta},\, \frac{\sigma^2}{n}\, (\boldsymbol{X}^T \boldsymbol{X})^{-1}\big), \tag{9.54}$$

where $\sim_a$ means asymptotically distributed. This asymptotic distribution gives an approximated two-sided confidence interval for the j-th component of the vector $\hat{\boldsymbol{\beta}}$:

$$\beta_j \in \left[\hat{\beta}_j \pm q^{\mathcal{N}(0,1)}_{1-\alpha/2} \sqrt{\tfrac{1}{n}\hat{\sigma}^2 \big[(\boldsymbol{X}^T \boldsymbol{X})^{-1}\big]_{jj}} \right] \tag{9.55}$$

at a $(1-\alpha)$ confidence level.

If the fourth moment of the error ϵ exists, the least squares estimator for σ^2 is consistent and asymptotically normal, too. The asymptotic normal distribution is

$$\sqrt{n}(\hat{\sigma}^2 - \sigma^2) \xrightarrow{d} \mathcal{N}\big(0,\, \mathrm{E}(\epsilon^4) - \sigma^4\big), \tag{9.56}$$

which gives

$$\hat{\sigma}^2 \sim_a \mathcal{N}\big(\sigma^2 \,,\, (\mathrm{E}(\epsilon^4) - \sigma^4) \,/\, n\big) \,. \tag{9.57}$$

Also the predicted response $\hat{y}$ is a random variable given $\boldsymbol{x}$, the distribution of which is determined by that of $\hat{\boldsymbol{\beta}}$:

$$\sqrt{n}\,(\hat{y} - y) \xrightarrow{d} \mathcal{N}\big(0,\, \sigma^2\, \boldsymbol{x}^T\, (\boldsymbol{X}^T \boldsymbol{X})^{-1}\, \boldsymbol{x}\big) \,, \tag{9.58}$$

which gives

$$\hat{y} \sim_a \mathcal{N}\big(y,\, \frac{\sigma^2}{n}\, \boldsymbol{x}^T\, (\boldsymbol{X}^T \boldsymbol{X})^{-1}\, \boldsymbol{x}\big) \,. \tag{9.59}$$

This distribution gives a confidence interval for mean response y, that is, an error bar on the prediction:

$$y \in \left[\, \boldsymbol{x}^T \hat{\boldsymbol{\beta}} \pm q^{\mathcal{N}(0,1)}_{1-\alpha/2} \sqrt{\tfrac{1}{n}\, \hat{\sigma}^2\, \boldsymbol{x}^T\, (\boldsymbol{X}^T \boldsymbol{X})^{-1}\, \boldsymbol{x}} \,\right] \tag{9.60}$$

at a $(1 - \alpha)$ confidence level.

9.1.6 Tests of Hypotheses

We want to test whether some independent variables (regressors) are relevant for the regression. These tests assume the null hypothesis that models without some variables have the same fitting quality as models with these variables. If the null hypothesis is rejected, then the variables are relevant for fitting.

9.1.6.1 Test for a Set of Variables Equal to Zero

We remove h variables from the original data and fit a reduced model. The error is assumed to be normally distributed. We divide the data in $m - h + 1$ variables (including the constant variable) and h variables which will be removed:

$$\boldsymbol{X} = (\boldsymbol{X}_1, \boldsymbol{X}_2) \tag{9.61}$$

with $\boldsymbol{X}_1 \in \mathbb{R}^{n \times (m-h+1)}$ and $\boldsymbol{X}_2 \in \mathbb{R}^{n \times h}$. Also the parameters are accordingly partitioned:

$$\boldsymbol{\beta} = (\boldsymbol{\beta}_1, \boldsymbol{\beta}_2) \tag{9.62}$$

with $\boldsymbol{\beta}_1 \in \mathbb{R}^{m-h+1}$ and $\boldsymbol{\beta}_2 \in \mathbb{R}^h$. We want to test the null hypothesis H_0

$$\boldsymbol{\beta}_2 = \boldsymbol{0} \,. \tag{9.63}$$

We denote the least squares estimator for the reduced model that uses only $\boldsymbol{X}_1$ by $\hat{\boldsymbol{\beta}}_r \in \mathbb{R}^{m-h+1}$. In contrast, $\hat{\boldsymbol{\beta}}_1 \in \mathbb{R}^{m-h+1}$ are the first $(m - h + 1)$ components of the least squares estimator $\hat{\boldsymbol{\beta}}$ of the full model. We define an F statistic as follows:

$$F = \frac{\boldsymbol{y}^T\, (\boldsymbol{P} - \boldsymbol{P}_1)\, \boldsymbol{y} \,/\, h}{\boldsymbol{y}^T\, (\boldsymbol{I} - \boldsymbol{P})\, \boldsymbol{y} \,/\, (n - m - 1)} = \frac{\left(\hat{\boldsymbol{\beta}}^T \boldsymbol{X}^T \boldsymbol{y} - \hat{\boldsymbol{\beta}}_r^T \boldsymbol{X}_1^T \boldsymbol{y}\right) \,/\, h}{\left(\boldsymbol{y}^T \boldsymbol{y} - \hat{\boldsymbol{\beta}}^T \boldsymbol{X}^T \boldsymbol{y}\right) \,/\, (n - m - 1)} \,, \tag{9.64}$$

Source of Variation	Degrees of freedom	Sum of squares	Mean square
reduced $\boldsymbol{\beta}_r$	$\mathrm{df} = m - h + 1$	$S = \hat{\boldsymbol{\beta}}_r^T \boldsymbol{X}_1^T \boldsymbol{y}$	S / df
improved $\boldsymbol{\beta}$	$\mathrm{df} = h$	$S = \hat{\boldsymbol{\beta}}^T \boldsymbol{X}^T \boldsymbol{y} - \hat{\boldsymbol{\beta}}_r^T \boldsymbol{X}_1^T \boldsymbol{y}$	S / df
residual	$\mathrm{df} = n - m - 1$	$S = \boldsymbol{y}^T \boldsymbol{y} - \hat{\boldsymbol{\beta}}^T \boldsymbol{X}^T \boldsymbol{y}$	S / df
total center	$\mathrm{df} = n - 1$	$S = \boldsymbol{y}^T \boldsymbol{y} - n\,\bar{y}$	S / df
total	$\mathrm{df} = n$	$S = \boldsymbol{y}^T \boldsymbol{y}$	S / df

Table 9.1: ANOVA table for F test of H_0: $\boldsymbol{\beta}_2 = \mathbf{0}$.

where $\hat{\boldsymbol{\beta}}$ is the least squares estimator of the full model and $\hat{\boldsymbol{\beta}}_r$ the least squares estimator of the reduced model. The distribution of the F statistic is the following:

(i) If H_0: $\boldsymbol{\beta}_2 = \mathbf{0}$ is **false**, then F is distributed according to $F(h, n - m - 1, \lambda)$, where

$$\lambda = \boldsymbol{\beta}_2^T \left(\boldsymbol{X}_2^T \boldsymbol{X}_2 - \boldsymbol{X}_2^T \boldsymbol{X}_1 \left(\boldsymbol{X}_1^T \boldsymbol{X}_1 \right)^{-1} \boldsymbol{X}_1^T \boldsymbol{X}_2 \right) \boldsymbol{\beta}_2 \,/\, (2\sigma^2) \,. \tag{9.65}$$

(ii) If H_0: $\boldsymbol{\beta}_2 = \mathbf{0}$ is **true**, then $\lambda = 0$ and F is distributed according to $F(h, n - m - 1)$.

H_0 is rejected if $F \geq F_{\alpha,h,n-m-1}$, where $F_{\alpha,h,n-m-1}$ is the upper α percentage of the central F distribution. That is, H_0 is rejected if the p-value is smaller than α.

The statistic F can also be expressed by R^2:

$$F = \frac{\left(R^2 - R_1^2\right) \,/\, h}{\left(1 - R^2\right) \,/\, (n - m - 1)} \,, \tag{9.66}$$

where R^2 is the coefficient of determination for the full model and R_1^2 is the coefficient of determination for the reduced model using only $\boldsymbol{X}_1$. It can be shown that this test is equivalently to a likelihood ratio test.

These hypotheses tests are often summarized by the Analysis-of-Variance (ANOVA) table as shown in Tab. 9.1.

9.1.6.2 Test for a Single Variable Equal to Zero

To test the null hypothesis H_0: $\beta_j = 0$, the F statistic

$$F = \frac{\hat{\beta}_j^2}{s^2 \left[(\boldsymbol{X}^T \boldsymbol{X})^{-1} \right]_{jj}} \tag{9.67}$$

can be used. If H_0: $\beta_j = 0$ is **true**, then F is distributed according to $F(1, n - m - 1)$. We reject H_0: $\beta_j = 0$ if $F \geq F_{\alpha,1,(n-m-1)}$ or, equivalently, if the p-value is smaller than α.

Alternatively, the t-statistic

$$t_j = \frac{\hat{\beta}_j}{s \sqrt{\left[(\boldsymbol{X}^T \boldsymbol{X})^{-1} \right]_{jj}}} \tag{9.68}$$

can be used. We reject H_0: $\beta_j = 0$ if $|t_j| \geq t_{\alpha/2,(n-m-1)}$ or, equivalently, if the p-value is smaller than α.

If several β_j are tested for being zero, then we have to correct for multiple testing. The false discovery rate (FDR) can be controlled by the Benjamini-Hochberg procedure Benjamini and Hochberg [1995], Benjamini and Yekutieli [2001]. Alternatively, the familywise α level can be adjusted by the Bonferroni approach Bonferroni [1936].

9.1.7 Examples

9.1.7.1 Hematology Data

This data set is from Rencher and Schaalje [2008] page 252, Ex. 10.3, Table 10.1 and stems from Royston (1983). The following six hematology variables were measured on 51 workers:

1. y: lymphocyte count,
2. x_1: hemoglobin concentration,
3. x_2: packed-cell volume,
4. x_3: white blood cell count ($\times .01$),
5. x_4: neutrophil count,
6. x_5: serum lead concentration.

The data are given in Tab. 9.2.

If we look at the correlation matrix

```
           [,1]        [,2]      [,3]       [,4]       [,5]        [,6]
[1,] 1.00000000  0.23330745 0.2516182 0.79073232 0.02264257  0.08290783
[2,] 0.23330745  1.00000000 0.7737330 0.27650957 0.05537581 -0.08376682
[3,] 0.25161817  0.77373300 1.0000000 0.30847841 0.07642710  0.12970593
[4,] 0.79073232  0.27650957 0.3084784 1.00000000 0.60420947  0.07147757
[5,] 0.02264257  0.05537581 0.0764271 0.60420947 1.00000000  0.03169314
[6,] 0.08290783 -0.08376682 0.1297059 0.07147757 0.03169314  1.00000000
```

we see that the largest correlation between the response y and an explanatory variable is 0.79 between y and x_3.

9.1.7.1.1 Computing Estimates, Confidence Intervals, Tests. The mean $\bar{y}$ of the response variable y is 22.98039.

The means of the explanatory variables x_1 to x_5 are:

```
15.10784 45.19608 53.82353 25.62745 21.07843
```

#	y	x_1	x_2	x_3	x_4	x_5	#	y	x_1	x_2	x_3	x_4	x_5
1	14	13.4	39	41	25	17	27	16	15.5	45	52	30	20
2	15	14.6	46	50	30	20	28	18	14.5	43	39	18	25
3	19	13.5	42	45	21	18	29	17	14.4	45	60	37	23
4	23	15.0	46	46	16	18	30	23	14.6	44	47	21	27
5	17	14.6	44	51	31	19	31	43	15.3	45	79	23	23
6	20	14.0	44	49	24	19	32	17	14.9	45	34	15	24
7	21	16.4	49	43	17	18	33	23	15.8	47	60	32	21
8	16	14.8	44	44	26	29	34	31	14.4	44	77	39	23
9	27	15.2	46	41	13	27	35	11	14.7	46	37	23	23
10	34	15.5	48	84	42	36	36	25	14.8	43	52	19	22
11	26	15.2	47	56	27	22	37	30	15.4	45	60	25	18
12	28	16.9	50	51	17	23	38	32	16.2	50	81	38	18
13	24	14.8	44	47	20	23	39	17	15.0	45	49	26	24
14	26	16.2	45	56	25	19	40	22	15.1	47	60	33	16
15	23	14.7	43	40	13	17	41	20	16.0	46	46	22	22
16	9	14.7	42	34	22	13	42	20	15.3	48	55	23	23
17	18	16.5	45	54	32	17	43	20	14.5	41	62	36	21
18	28	15.4	45	69	36	24	44	26	14.2	41	49	20	20
19	17	15.1	45	46	29	17	45	40	15.0	45	72	25	25
20	14	14.2	46	42	25	28	46	22	14.2	46	58	31	22
21	8	15.9	46	52	34	16	47	61	14.9	45	84	17	17
22	25	16.0	47	47	14	18	48	12	16.2	48	31	15	18
23	37	17.4	50	86	39	17	49	20	14.5	45	40	18	20
24	20	14.3	43	55	31	19	50	35	16.4	49	69	22	24
25	15	14.8	44	42	24	29	51	38	14.7	44	78	34	16
26	9	14.9	43	43	32	17							

Table 9.2: Rencher's hematology data Rencher and Schaalje [2008] page 252, Ex. 10.3, Table 10.1 — originally from Royston (1983). The variables are y: lymphocyte count, x_1: hemoglobin concentration, x_2: packed-cell volume, x_3: white blood cell count ($\times.01$), x_4: neutrophil count, and x_5: serum lead concentration.

We assume centered data and estimate the coefficients without β_0 which is estimated separately. The covariance matrix $\mathrm{Cov}(\boldsymbol{X})$ of the explanatory variables is

```
           [,1]      [,2]       [,3]       [,4]       [,5]
[1,]  0.6907373  1.494431   3.255412  0.3509804 -0.2966275
[2,]  1.4944314  5.400784  10.155294  1.3545098  1.2843137
[3,]  3.2554118 10.155294 200.668235 65.2729412  4.3141176
[4,]  0.3509804  1.354510  65.272941 58.1584314  1.0298039
[5,] -0.2966275  1.284314   4.314118  1.0298039 18.1537255
```

The covariance $\mathrm{Cov}(\boldsymbol{y}, \boldsymbol{X})$ between the response and the explanatory variables is

```
[1]   1.878157   5.663922 108.496471   1.672549   3.421569
```

We now compute

$$\hat{\boldsymbol{\beta}} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}\,. \tag{9.69}$$

for the centered data, i.e. we assume that $\boldsymbol{X}$ is centered. In this case $\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1} = 1/n\left(\mathrm{Cov}(\boldsymbol{X})\right)^{-1}$ is the inverse of the covariance matrix divided by the number of samples n. $\boldsymbol{X}^T\boldsymbol{y} = n\mathrm{Cov}(\boldsymbol{y}, \boldsymbol{X})$ is the covariance between the response and the explanatory variables multiplied by the number of samples n. Since the number of samples n cancel, we have

$$\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y} \;=\; \left(\mathrm{Cov}(\boldsymbol{X})\right)^{-1}\mathrm{Cov}(\boldsymbol{y}, \boldsymbol{X})\,. \tag{9.70}$$

Therefore the least squares estimate can be computed as:

```
            [,1]
[1,] -0.21318219
[2,] -0.28884109
[3,]  0.85984756
[4,] -0.92921309
[5,]  0.05380269
```

We assumed centered data. Now we estimate β_0 using the mean of the response $\bar{y}$ and the mean of the explanatory variables: 15.65486.

In our derivation of the least squares estimator, we used the formula

$$\hat{\boldsymbol{\beta}} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}\,, \tag{9.71}$$

where the first column of $\boldsymbol{X}$ contains 1's to account for the intercept. Therefore the least squares estimate is:

```
            [,1]
[1,] 15.65485611
[2,] -0.21318219
[3,] -0.28884109
[4,]  0.85984756
[5,] -0.92921309
[6,]  0.05380269
```

This is the same result as previously, where we first estimated the parameter for the centered data and then adjusted β_0.

$$S(\hat{\boldsymbol{\beta}}) \;=\; \boldsymbol{y}^T\boldsymbol{y} \;-\; \hat{\boldsymbol{\beta}}^T\boldsymbol{X}^T\boldsymbol{y}\,. \tag{9.72}$$

The estimate for the error variance s^2 is

$$s^2 \;=\; \frac{1}{n-m-1}S(\hat{\boldsymbol{\beta}}) \tag{9.73}$$

where $n = 51$ and $m = 5$ in our example. We compute s^2 and the standard error s:

```
s2:        4.3729
sqrt(s2): 2.091148
```

The coefficient of determination R^2 is

$$R^2 \;=\; \frac{\sum_{i=1}^{n}(\hat{y}_i \;-\; \overline{y})^2}{\sum_{i=1}^{n}(y_i \;-\; \overline{y})^2}\,, \tag{9.74}$$

which is $R^2 = 0.9580513$. R^2 is the variance of the estimated response divided by the variance of the response.

The approximate two-sided confidence intervals for components of the vector $\hat{\boldsymbol{\beta}}$ are:

$$\beta_j \;\in\; \left[\,\hat{\beta}_j \;\pm\; t_{\alpha/2,n-m-1}\; s\,\sqrt{\left[(\boldsymbol{X}^T\boldsymbol{X})^{-1}\right]_{jj}}\,\right] \tag{9.75}$$

where $t_{\alpha/2,n-m-1}$ is the upper $\alpha/2$ percentage point of the central t-distribution and α is the desired significance level of the test. These confidence intervals are:

```
             [,1]       [,2]
[1,]  3.03587336 28.2738389
[2,] -1.40187932  0.9755149
[3,] -0.71833021  0.1406480
[4,]  0.80366905  0.9160261
[5,] -1.02844916 -0.8299770
[6,] -0.09389755  0.2015029
```

Only for the intercept (component 1), x_3 (component 4), and x_4 (component 5), the confidence intervals do not include zero.

For testing whether the components of the estimated parameter vector are significantly different from zero, we compute the t-statistics:

$$t_j \;=\; \frac{\hat{\beta}_j}{s\,\sqrt{\left[(\boldsymbol{X}^T\boldsymbol{X})^{-1}\right]_{jj}}} \tag{9.76}$$

the t-statistics are

```
            [,1]
[1,]   2.4986561
[2,]  -0.3612114
[3,]  -1.3545299
[4,]  30.8271243
[5,] -18.8593854
[6,]   0.7336764
```

These t-statistics together with $n - m - 1 = 51 - 5 - 1 = 45$ degrees of freedom allow to compute the p-values:

```
             [,1]
[1,] 1.618559e-02
[2,] 7.196318e-01
[3,] 1.823298e-01
[4,] 6.694743e-32
[5,] 5.395732e-23
[6,] 4.669514e-01
```

Only the intercept, x_3, and x_4 are significant, where the latter two are highly significant.

9.1.7.2 Carbohydrate Diet Data

This example is from Dobson [2002], page 96, data of Table 6.3. The data are shown in Tab. 9.3 and contain for twenty male insulin-dependent diabetics: responses, age, weight, and percentages of total calories obtained from complex carbohydrates. The individuals had been on a high-carbohydrate diet for six months. Compliance with the regime was thought to be related to age (in years), body weight (relative to "ideal" weight for height) and other components of the diet, such as the percentage of calories as protein. These other variables are treated as explanatory variables.

We fitted a normal linear model by least squares:

```
Residuals:
     Min       1Q   Median       3Q      Max
-10.3424  -4.8203   0.9897   3.8553   7.9087

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 36.96006   13.07128   2.828  0.01213 *
age         -0.11368    0.10933  -1.040  0.31389
wgt         -0.22802    0.08329  -2.738  0.01460 *
prot         1.95771    0.63489   3.084  0.00712 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.956 on 16 degrees of freedom
Multiple R-squared:  0.4805,    Adjusted R-squared:  0.3831
F-statistic: 4.934 on 3 and 16 DF,  p-value: 0.01297
```

Carbohydrate	Age	Weight	Protein
y	x_1	x_2	x_3
33	33	100	14
40	47	92	15
37	49	135	18
27	35	144	12
30	46	140	15
43	52	101	15
34	62	95	14
48	23	101	17
30	32	98	15
38	42	105	14
50	31	108	17
51	61	85	19
30	63	130	19
36	40	127	20
41	50	109	15
42	64	107	16
46	56	117	18
24	61	100	13
35	48	118	18
37	28	102	14

Table 9.3: Dobson's carbohydrate diet data Dobson [2002], page 96, data of Table 6.3. Carbohydrate, age, relative weight, and protein for twenty male insulin-dependent diabetics.

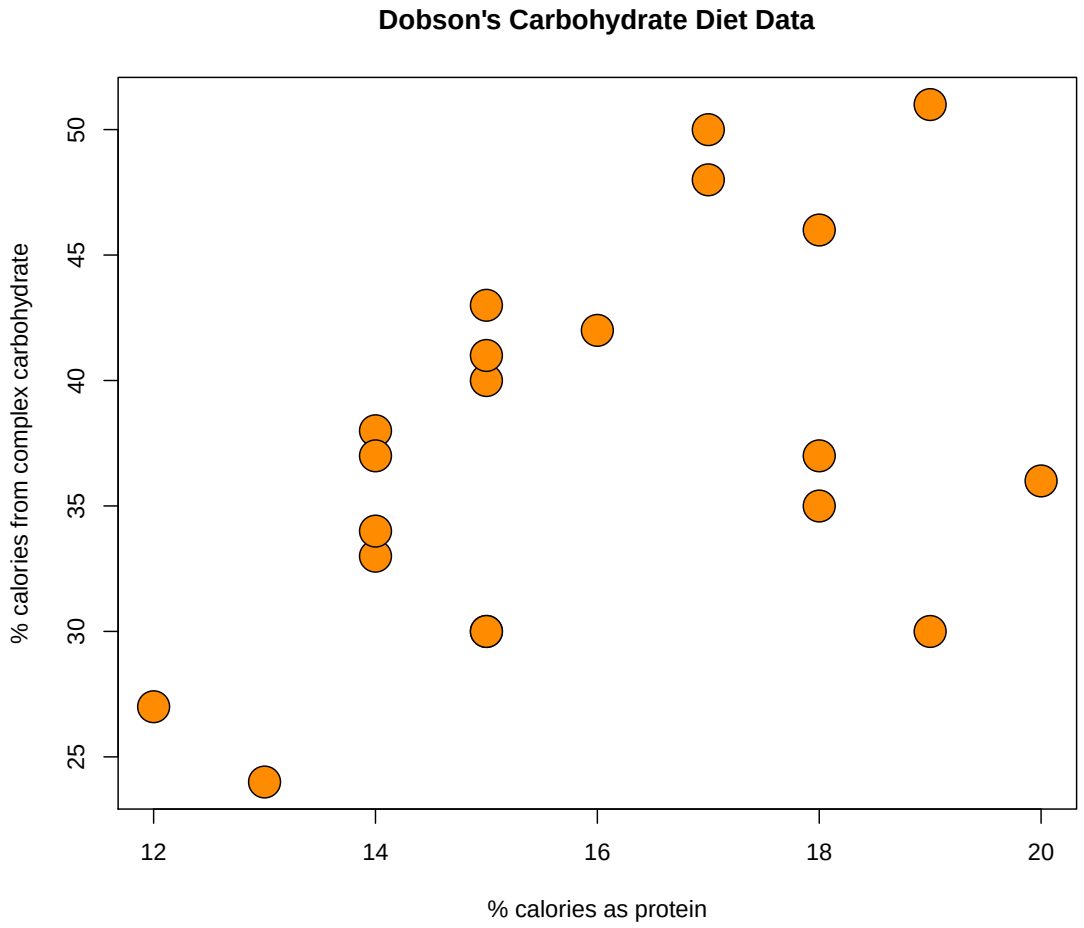

Figure 9.2: Dobson's carbohydrate diet data Dobson [2002] with percentages of total calories obtained from complex carbohydrates plotted against percentage of calories as protein.

The feature "Protein" seems to be the feature that is most related to carbohydrates. We verify this by a scatter plot. Fig. 9.2 shows percentages of total calories obtained from complex carbohydrates plotted against percentage of calories as protein. A linear dependence is visible which supports the finding that protein is significantly related to carbohydrate.

9.2 Analysis of Variance

Analysis-of-variance (ANOVA) models apply linear models to compare means of responses to different treatments. The treatments are the levels of one *factor*. Thus, they compare means of different groups which are known a priori. Typically, the results of fitting linear models are analyzed by the variance explained as shown previously. $\boldsymbol{x}$ is neither measured nor a sample but constructed and contains dummy variables, therefore, the matrix $\boldsymbol{X}$ is called *design matrix*. Typically, ANOVA models use more parameters than can be estimated, therefore $\boldsymbol{X}$ may not have full rank. We first consider the case where observations are divided into different groups corresponding to a factor. Then we consider the case where observations can be divided by two ways into different groups, that is, two factors. In this case, besides the treatment, a second factor influences the outcome of a study.

9.2.1 One Factor

The response variable, that is, the dependent variable, has now two indices: the first index gives the group to which the observation belongs and the second index gives the replicate number for this group. The standard case is a treatment-control study, where one group are controls and the other group are the treatments. It is possible to analyze different treatments if they are mutually exclusive.

The response variable is y_{gi} with $y_{11}, y_{12}, \ldots, y_{1n_1}, y_{21}, y_{22}, \ldots, y_{2n_2}, y_{31}, \ldots, y_{Gn_G}$, where the j-th group has n_j replicates and G denotes the number of groups. The model is

$$y_{gi} \;=\; \beta_0 \;+\; \beta_g \;+\; \epsilon_{gi}\,. \tag{9.77}$$

The value β_0 is a constant offset or the mean of group 1 if we force $\beta_1 = 0$. The value β_g is the mean difference to the offset (or group 1). As previously ϵ_{gi} is an additive error term with previously introduced assumptions.

For each group the model uses different parameters, therefore the model equation depends on the group to which the observation belongs. The model equations are written down as a matrix equation. For example, in a case-control study with 3 controls and 3 cases, we write:

$$\begin{pmatrix} y_{11} \\ y_{12} \\ y_{13} \\ y_{21} \\ y_{22} \\ y_{23} \end{pmatrix} \;=\; \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix} \;+\; \begin{pmatrix} \epsilon_{11} \\ \epsilon_{12} \\ \epsilon_{13} \\ \epsilon_{21} \\ \epsilon_{22} \\ \epsilon_{23} \end{pmatrix}\,. \tag{9.78}$$

In matrix notation we have the linear model:

$$\boldsymbol{y} \;=\; \boldsymbol{X}\,\boldsymbol{\beta} \;+\; \boldsymbol{\epsilon}\,, \tag{9.79}$$

where $\boldsymbol{X}$ is designed depending on the groups to which the observations $\boldsymbol{y}$ belong.

However $\boldsymbol{X}$ does not have full rank. In the example above, the 6×3 matrix has rank 2 because the first column is the sum of the other two. The least squares estimator cannot be computed because $(\boldsymbol{X}^T\boldsymbol{X})^{-1}$ does not exist. The model is not identifiable, that is, for every data set there exists more than one solution. For example, we can subtract δ from β_0 and, at the same time, add δ to β_1 and β_2. The solution will not change, only the parameters.

There are different ways to ensure that $\boldsymbol{X}$ has full rank and the least squares estimate can be applied:

(i) *re-parametrization* using fewer parameters, e.g., corner point parametrization,

(ii) *side conditions* as constraints on the parameters, e.g., sum-to-zero constraints,

(iii) *linear projections* $\boldsymbol{a}^T\boldsymbol{\beta}$ of parameter vector $\boldsymbol{\beta}$ which is estimable.

ad (i) re-parametrization:
We assume that β_0 is the mean response of the controls and β_g is the offset of group g to the controls. Therefore we set $\beta_1 = 0$ because controls have zero offset to themselves. We obtain:

$$\begin{pmatrix} y_{11} \\ y_{12} \\ y_{13} \\ y_{21} \\ y_{22} \\ y_{23} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_2 \end{pmatrix} + \begin{pmatrix} \epsilon_{11} \\ \epsilon_{12} \\ \epsilon_{13} \\ \epsilon_{21} \\ \epsilon_{22} \\ \epsilon_{23} \end{pmatrix} . \tag{9.80}$$

Setting $\beta_1 = 0$ is called *corner point parametrization* which removes β_1 from the equations. In general corner point parametrization removes all variables that contain the index one. This means that variables that contain the index one are considered as reference groups or as reference group combinations.

In general, the re-parametrization is

$$\boldsymbol{\gamma} = \boldsymbol{U}\,\boldsymbol{\beta} \tag{9.81}$$

which gives with

$$\boldsymbol{X} = \boldsymbol{Z}\,\boldsymbol{U} \tag{9.82}$$

$$\boldsymbol{y} = \boldsymbol{Z}\,\boldsymbol{\gamma} + \boldsymbol{\epsilon}\,. \tag{9.83}$$

The matrix $\boldsymbol{Z}$ has full rank and $\boldsymbol{U}$ blows $\boldsymbol{Z}$ up to $\boldsymbol{X}$, therefore, $\boldsymbol{Z}$ and $\boldsymbol{X}$ have the same rank.

ad (ii) side conditions:
We can assume that $\beta_1 + \beta_2 = 0$. If group 1 and group 2 have the same number of replicates, then β_0 is the mean over all groups. From the condition $\beta_1 + \beta_2 = 0$ we immediately obtain $\beta_2 = -\beta_1$. This gives the matrix equation

$$\begin{pmatrix} y_{11} \\ y_{12} \\ y_{13} \\ y_{21} \\ y_{22} \\ y_{23} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & -1 \\ 1 & -1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} \epsilon_{11} \\ \epsilon_{12} \\ \epsilon_{13} \\ \epsilon_{21} \\ \epsilon_{22} \\ \epsilon_{23} \end{pmatrix} . \tag{9.84}$$

Variable β_2 is removed from these equations.

The constraint $\sum_{g=1}^{G} \beta_g = 0$ is the *sum-to-zero constraint*. This ensures that β_0 is the overall mean. The β_g estimate the deviation of the mean of group g from the overall mean. In general sum-to-zero constraints set sums over an index to zero and, thereby, define the constant offset as the overall mean.

ad (iii) linear projection:
$\boldsymbol{a} = (0, 1, -1))$ gives $\beta_1' = \boldsymbol{a}^T\boldsymbol{\beta} = \beta_1 - \beta_2$, which is estimable. This approach is of interest, if specific questions have to be answered. In our example, the difference of the means of group 1 and group 2 may be relevant but not the means themselves. We obtain the matrix equation:

$$\begin{pmatrix} y_{11} \\ y_{12} \\ y_{13} \\ y_{21} \\ y_{22} \\ y_{23} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ -1 \\ -1 \\ -1 \end{pmatrix} \beta_1' + \begin{pmatrix} \epsilon_{11} \\ \epsilon_{12} \\ \epsilon_{13} \\ \epsilon_{21} \\ \epsilon_{22} \\ \epsilon_{23} \end{pmatrix} . \tag{9.85}$$

The models can be used to test hypotheses. A common null hypothesis is H_0: $\beta_1 = \beta_2 = \ldots = \beta_G$, where the null hypothesis states that the means of all groups are equal. This can be expressed by the new variables $\beta_1^* = \beta_1 - \beta_2, \beta_2^* = \beta_1 - \beta_3, \ldots, \beta_{G-1}^* \beta_1 - \beta_G$, which are tested for $\beta_1^* = \beta_2^* = \ldots = \beta_{G-1}^* = 0$. Or we introduce the constraint $\sum_{g=1}^{G} \beta_g = 0$ while keeping β_0. We then can test for $\beta_1 = \beta_2 = \ldots = \beta_G = 0$, that is, deviation from the overall mean β_0. Tests for these hypotheses have been presented earlier. The reduced model has only the overall mean β_0 as parameter.

9.2.2 Two Factors

We now consider the case where two factors influence the response. Consequently, the response variable has now three indices: the first index gives the group for the first factor, the second index the group for the second factor, and the third index gives the replicate number for this combination of groups.

The response variable is y_{ghi} with the model

$$y_{ghi} \;=\; \beta_0 \;+\; \beta_g \;+\; \alpha_h \;+\; (\alpha\beta)_{gh} \;+\; \epsilon_{ghi}\,. \tag{9.86}$$

The value β_0 is a constant offset. The values β_g are the mean difference for the first factor and α_h the mean differences for the second factor. The new term $(\alpha\beta)_{gh}$ models the *interaction effects* between the two factors. As always, ϵ_{ghi} is the additive error with previously introduced assumptions.

The following hypotheses are often tested and correspond to different reduced models:

(i) the *additive model* with the hypothesis H_0: $(\alpha\beta)_{gh} = 0$ for all g and all h:

$$y_{ghi} \;=\; \beta_0 \;+\; \beta_g \;+\; \alpha_h \;+\; \epsilon_{ghi}\,. \tag{9.87}$$

This model should be compared to the full model.

(ii) factor corresponding to α has no effect:

$$y_{ghi} \;=\; \beta_0 \;+\; \beta_g \;+\; \epsilon_{ghi}\,. \tag{9.88}$$

This model should be compared to the additive model in (i).

(iii) factor corresponding to β has no effect:

$$y_{ghi} \;=\; \beta_0 \;+\; \alpha_h \;+\; \epsilon_{ghi}\,. \tag{9.89}$$

As for the model in (ii), also this model should be compared to the additive model in (i).

These models should be either tested with *sum-zero constraints*

(i) $\sum_{g=1}^{G} \beta_g = 0$,

(ii) $\sum_{h=1}^{H} \alpha_h = 0$,

(iii) $\forall_g : \;\; \sum_{h=1}^{H} (\alpha\beta)_{gh} = 0$,

(iv) $\forall_h : \;\; \sum_{g=1}^{G} (\alpha\beta)_{gh} = 0$,

or with *corner point constraints*

(i) $\beta_1 = 0$,

(ii) $\alpha_1 = 0$,

(iii) $\forall_g : \;\; (\alpha\beta)_{g1} = 0$,

(iv) $\forall_h : \;\; (\alpha\beta)_{1h} = 0$.

We have one offset parameter β_0, G factor parameters β_g, H factor parameters α_h, and GH interaction parameters $(\alpha\beta)_{gh}$, which sums up to $GH+G+H+1 = (G+1)(H+1)$ parameters. The minimal data set has only GH observations, one observation for each combination of factors. For both sets of constraints the β_g equations use up one degree of freedom, the α_h use also up one degree of freedom, the $(\alpha\beta)_{gh}$ equations for all g use up G degrees of freedom, and the $(\alpha\beta)_{gh}$ equations for all h use up H degrees of freedom. We have to add one degree of freedom because for corner point constraints $(\alpha\beta)_{11}$ is counted twice and for sum-zero constraints the last equation follows from the other equations. We have $1 + 1 + G + H - 1 = G + H + 1$ degrees of freedom used up. Therefore we have $(G+1)(H+1) - (G+H+1) = GH$ free parameters. For sum-zero constraints we show that the last equation follows from the others. From $\forall_g : \;\; \sum_{h=1}^{H} (\alpha\beta)_{gh} = 0$ follows that $\sum_{g=1}^{G} \sum_{h=1}^{H} (\alpha\beta)_{gh} = 0$. We have $\sum_{h=1}^{H} (\sum_{g=1}^{G} (\alpha\beta)_{gh}) = 0$ and $\sum_{g=1}^{G} (\alpha\beta)_{gh} = 0$ for $h < H$ since the last equation is not used. Thus, $\sum_{g=1}^{G} (\alpha\beta)_{gH} = 0$, which is the last equation. We showed that the last equation can be deduced from the others. Therefore for both constraint sets we have GH free parameters, as desired.

The design matrix $\boldsymbol{X}$ should have at least rank GH to distinguish all interaction effects $(\alpha\beta)_{gh}$. Thus, the least squares estimator can be computed and the according tests performed.

To simplify notations, means are denoted by

(i) mean of group combination gh:

$$\bar{y}_{gh} \;=\; \frac{1}{n_{gh}} \sum_{i=1}^{n_{gh}} y_{ghi} \;, \tag{9.90}$$

where n_{gh} are the number of replicates of group combination gh.

(ii) mean of group g:

$$\bar{y}_{g.} \;=\; \frac{1}{\sum_{h=1}^{H} n_{gh}} \sum_{h=1}^{H} \sum_{i=1}^{n_{gh}} y_{ghi} \;, \tag{9.91}$$

(iii) mean of group h:

$$\bar{y}_{.h} \;=\; \frac{1}{\sum_{g=1}^{G} n_{gh}} \sum_{g=1}^{G} \sum_{i=1}^{n_{gh}} y_{ghi} \;, \tag{9.92}$$

(iv) overall mean:

$$\bar{y}_{..} \;=\; \frac{1}{\sum_{g,h=1,1}^{G,H} n_{gh}} \sum_{g=1}^{G} \sum_{h=1}^{H} \sum_{i=1}^{n_{gh}} y_{ghi} \;. \tag{9.93}$$

If we use the full design matrix $\boldsymbol{X}$ then the *normal equations* are

$$\boldsymbol{X}^T \boldsymbol{X} \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\alpha} \\ (\boldsymbol{\alpha}\boldsymbol{\beta}) \end{pmatrix} \;=\; \boldsymbol{X}^T \, \boldsymbol{y} \;, \tag{9.94}$$

where β_0 is the first component of $\boldsymbol{\beta}$. The matrix $\boldsymbol{X}^T\boldsymbol{X}$ is not invertible. However for the optimal solution $(\hat{\boldsymbol{\beta}}^T, \hat{\boldsymbol{\alpha}}^T, (\hat{\boldsymbol{\alpha}}\hat{\boldsymbol{\beta}})^T)^T$ with sum-zero constraints or with corner point constraints the normal equations must hold.

The normal equations can be written as:

$$\begin{aligned}
&\left(\sum_{g,h=1,1}^{G,H} n_{gh}\right) \hat{\beta}_0 \;+\; \sum_{g=1}^{G} \left(\sum_{h=1}^{H} n_{gh}\right) \hat{\beta}_g \;+\; \sum_{h=1}^{H} \left(\sum_{g=1}^{G} n_{gh}\right) \hat{\alpha}_h \;+\; \sum_{g=1}^{G} \sum_{h=1}^{H} n_{gh} (\hat{\alpha}\hat{\beta})_{gh} \\
&= \sum_{g,h=1,1}^{G,H} n_{gh} \, \bar{y}_{..} \\
&\left(\sum_{h=1}^{H} n_{gh}\right) \hat{\beta}_0 \;+\; \left(\sum_{h=1}^{H} n_{gh}\right) \hat{\beta}_g \;+\; \sum_{h=1}^{H} n_{gh} \, \hat{\alpha}_h \;+\; \sum_{h=1}^{H} n_{gh} (\hat{\alpha}\hat{\beta})_{gh} \;=\; \sum_{h=1}^{H} n_{gh} \, \bar{y}_{g.}, \; 1 \le g \le G \\
&\left(\sum_{g=1}^{G} n_{gh}\right) \hat{\beta}_0 \;+\; \sum_{g=1}^{G} n_{gh} \, \hat{\beta}_g \;+\; \left(\sum_{g=1}^{G} n_{gh}\right) \hat{\alpha}_h \;+\; \sum_{g=1}^{G} n_{gh} (\hat{\alpha}\hat{\beta})_{gh} \;=\; \sum_{g=1}^{G} n_{gh} \, \bar{y}_{.h}, \; 1 \le h \le H \\
&n_{gh} \, \hat{\beta}_0 \;+\; n_{gh} \, \hat{\beta}_g \;+\; n_{gh} \, \hat{\alpha}_h \;+\; n_{gh} (\hat{\alpha}\hat{\beta})_{gh} \;=\; n_{gh} \bar{y}_{gh}, \;\; 1 \le g \le G, \;\; 1 \le h \le H \;\; .
\end{aligned} \tag{9.95}$$

These are $1 + G + H + GH = (G + 1)(H + 1)$ equations but in the worst case we have only GH observations. The constraints use up $G + H + 1$ degrees of freedom, e.g. via the zero sum conditions

$$\sum_{g=1}^{G} \hat{\beta}_g \;=\; 0\,, \tag{9.96}$$

$$\sum_{h=1}^{H} \hat{\alpha}_h \;=\; 0\,, \tag{9.97}$$

$$\sum_{g=1}^{G} (\hat{\alpha}\hat{\beta})_{gh} \;=\; 0\,, \tag{9.98}$$

$$\sum_{h=1}^{H} (\hat{\alpha}\hat{\beta})_{gh} \;=\; 0\,. \tag{9.99}$$

We then have at least GH observations and GH free parameters and the normal equations can be solved.

For the *balanced case* the number of replicates is the same for each combination of conditions. That means

$$n_{gh} \;=\; \tilde{n}\,. \tag{9.100}$$

In this case the means simplify to:

(i) mean of group combination gh:

$$\bar{y}_{gh} \;=\; \frac{1}{\tilde{n}} \sum_{i=1}^{\tilde{n}} y_{ghi} \tag{9.101}$$

(ii) mean of group g:

$$\bar{y}_{g\cdot} \;=\; \frac{1}{H\,\tilde{n}} \sum_{h=1}^{H} \sum_{i=1}^{\tilde{n}} y_{ghi}\,, \tag{9.102}$$

(iii) mean of group h:

$$\bar{y}_{\cdot h} \;=\; \frac{1}{G\,\tilde{n}} \sum_{g=1}^{G} \sum_{i=1}^{\tilde{n}} y_{ghi}\,, \tag{9.103}$$

(iv) overall mean:

$$\bar{y}_{\cdot\cdot} \;=\; \frac{1}{G\,H\,\tilde{n}} \sum_{g=1}^{G} \sum_{h=1}^{H} \sum_{i=1}^{\tilde{n}} y_{ghi}\,. \tag{9.104}$$

The normal equations become:

$$
\begin{aligned}
G\,H\,\tilde{n}\,\hat{\beta}_0 \;+\; H\,\tilde{n}\sum_{g=1}^{G}\hat{\beta}_g \;+\; G\,\tilde{n}\sum_{h=1}^{H}\hat{\alpha}_h \;+\; \tilde{n}\sum_{g=1}^{G}\sum_{h=1}^{H}(\hat{\alpha}\hat{\beta})_{gh} \;&=\; G\,H\,\tilde{n}\,\bar{y}_{..} \\
H\,\tilde{n}\,\hat{\beta}_0 \;+\; H\,\tilde{n}\,\hat{\beta}_g \;+\; \tilde{n}\sum_{h=1}^{H}\hat{\alpha}_h \;+\; \tilde{n}\sum_{h=1}^{H}(\hat{\alpha}\hat{\beta})_{gh} \;&=\; H\,\tilde{n}\,\bar{y}_{g.},\;\; 1\le g\le G \\
G\,\tilde{n}\,\hat{\beta}_0 \;+\; \tilde{n}\sum_{g=1}^{G}\hat{\beta}_g \;+\; G\,\tilde{n}\,\hat{\alpha}_h \;+\; \tilde{n}\sum_{g=1}^{G}(\hat{\alpha}\hat{\beta})_{gh} \;&=\; G\,\tilde{n}\,\bar{y}_{.h},\;\; 1\le h\le H \\
\tilde{n}\,\hat{\beta}_0 \;+\; \tilde{n}\,\hat{\beta}_g \;+\; \tilde{n}\,\hat{\alpha}_h \;+\; \tilde{n}\,(\hat{\alpha}\hat{\beta})_{gh} \;&=\; \tilde{n}\,\bar{y}_{gh},\;\; 1\le g\le G,\;\; 1\le h\le H\;\;. \qquad (9.105)
\end{aligned}
$$

Using the zero sum conditions

$$\sum_{g=1}^{G}\hat{\beta}_g \;=\; 0 \qquad (9.106)$$

$$\sum_{h=1}^{H}\hat{\alpha}_h \;=\; 0 \qquad (9.107)$$

$$\sum_{g=1}^{G}(\hat{\alpha}\hat{\beta})_{gh} \;=\; 0 \qquad (9.108)$$

$$\sum_{h=1}^{H}(\hat{\alpha}\hat{\beta})_{gh} \;=\; 0 \qquad (9.109)$$

the normal equations further simplify to

$$
\begin{aligned}
G\,H\,\tilde{n}\,\hat{\beta}_0 \;&=\; G\,H\,\tilde{n}\,\bar{y}_{..} \\
H\,\tilde{n}\,\hat{\beta}_0 \;+\; H\,\tilde{n}\,\hat{\beta}_g \;&=\; H\,\tilde{n}\,\bar{y}_{g.},\;\; 1\le g\le G \\
G\,\tilde{n}\,\hat{\beta}_0 \;+\; G\,\tilde{n}\,\hat{\alpha}_h \;&=\; G\,\tilde{n}\,\bar{y}_{.h},\;\; 1\le h\le H \\
\tilde{n}\,\hat{\beta}_0 \;+\; \tilde{n}\,\hat{\beta}_g \;+\; \tilde{n}\,\hat{\alpha}_h \;+\; \tilde{n}\,(\hat{\alpha}\hat{\beta})_{gh} \;&=\; \tilde{n}\,\bar{y}_{gh},\;\; 1\le g\le G,\;\; 1\le h\le H\;\;, \qquad (9.110)
\end{aligned}
$$

which gives

$$
\begin{aligned}
\hat{\beta}_0 \;&=\; \bar{y}_{..} \\
\hat{\beta}_g \;&=\; \bar{y}_{g.} \;-\; \hat{\beta}_0 \;=\; \bar{y}_{g.} \;-\; \bar{y}_{..},\;\; 1\le g\le G \\
\hat{\alpha}_h \;&=\; \bar{y}_{.h} \;-\; \hat{\beta}_0 \;=\; \bar{y}_{.h} \;-\; \bar{y}_{..},\;\; 1\le h\le H \\
(\hat{\alpha}\hat{\beta})_{gh} \;&=\; \bar{y}_{gh} \;-\; \hat{\beta}_0 \;-\; \hat{\beta}_g \;-\; \hat{\alpha}_h \\
&=\; \bar{y}_{gh} \;-\; \bar{y}_{g.} \;-\; \bar{y}_{.h} \;+\; \bar{y}_{..}\;. \qquad (9.111)
\end{aligned}
$$

These are the estimators for the means which one would use intuitively. Actually these are unbiased estimators for the according means.

Treatment group		Control group	
4.81	5.36	4.17	4.66
4.17	3.48	3.05	5.58
4.41	4.69	5.18	3.66
3.59	4.44	4.01	4.50
5.87	4.89	6.11	3.90
3.83	4.71	4.10	4.61
6.03	5.48	5.17	5.62
4.98	4.32	3.57	4.53
4.90	5.15	5.33	6.05
5.75	6.34	5.59	5.14

Table 9.4: Weights of dried plants which were grown under two conditions. The data are from Dobson [2002], page 46, data of Table 2.7.

9.2.3 Examples

9.2.3.1 Dried Plant Weights

The first example is from Dobson [2002], page 46, data from Table 2.7. Genetically similar seeds are randomly assigned to be raised in either a nutritionally enriched environment (treatment group) or standard conditions (control group) using a completely randomized experimental design. After a predetermined time, all plants are harvested, dried and weighed. The results, expressed in grams, for 20 plants in each group are shown in Tab. 9.4 and in Fig. 9.3. The goal is to test whether there is a difference in yield between the treatment and the control group.

To obtain an overview of the data, we do a simple summary:

```
ctl:
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  3.050   4.077   4.635   4.726   5.392   6.110
trt:
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  3.480   4.388   4.850   4.860   5.390   6.340
```

We see that the treatment has larger median and larger mean. Is this significant? When looking at the data in Fig. 9.3 there could be some doubts.

To answer the question whether the difference in means is significant or not, we fit a linear model and print the ANOVA table:

```
Analysis of Variance Table

Response: weight
          Df  Sum Sq Mean Sq F value Pr(>F)
group      1  0.1782 0.17822  0.2599 0.6131
Residuals 38 26.0535 0.68562
```

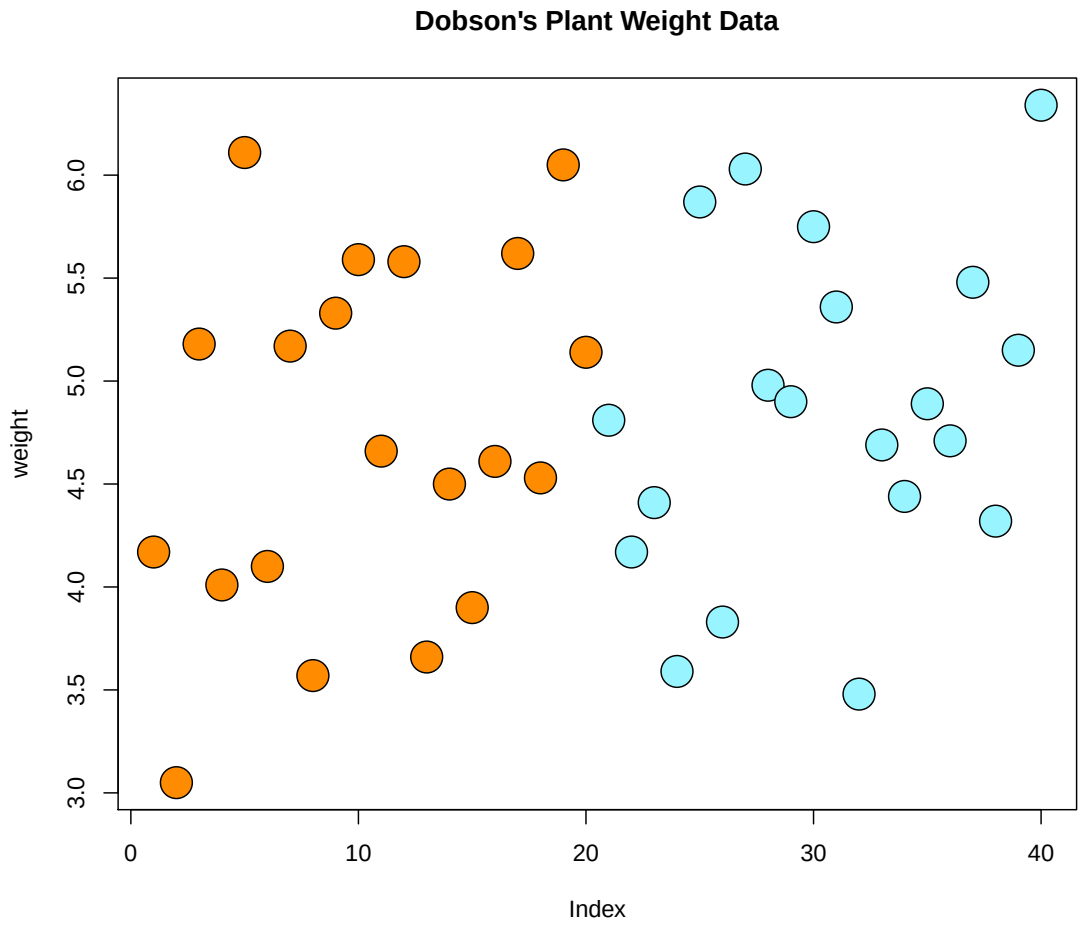

Figure 9.3: Dobson's dried plant data: orange indicates the control and blue the treatment group.

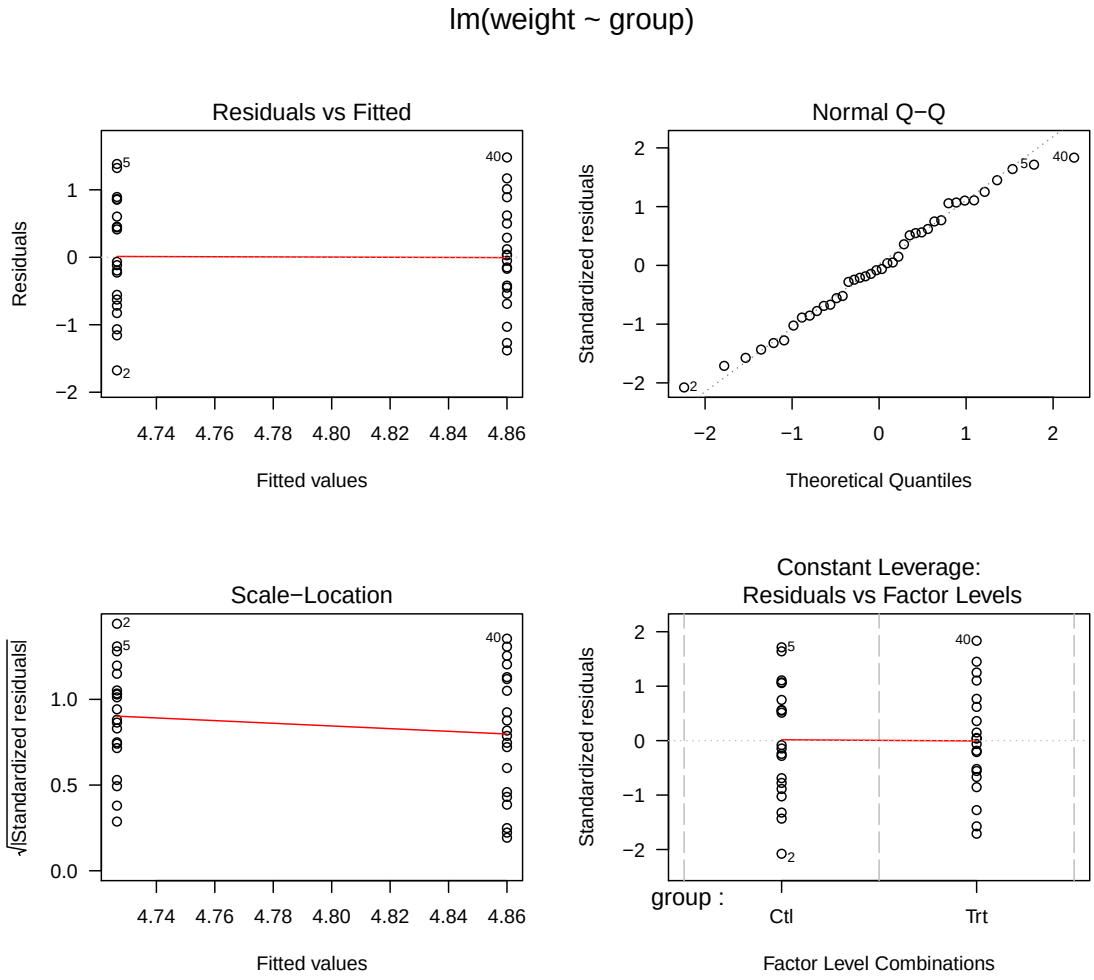

Figure 9.4: Results of ANOVA for dried plant data.

The difference in means between treatment and control is not significant, i.e. the treatment did not show more or less average yield. We shown the results in Fig. 9.4.

Next we fit a model without an intercept

```
Residuals:
     Min       1Q   Median       3Q      Max
-1.67650 -0.57400 -0.05825  0.60763  1.48000

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
groupCtl    4.7265     0.1852   25.53   <2e-16 ***
groupTrt    4.8600     0.1852   26.25   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

	Control	Treatment A	Treatment B
	4.17	4.81	6.31
	5.58	4.17	5.12
	5.18	4.41	5.54
	6.11	3.59	5.50
	4.50	5.87	5.37
	4.61	3.83	5.29
	5.17	6.03	4.92
	4.53	4.89	6.15
	5.33	4.32	5.80
	5.14	4.69	5.26
$\sum_i y_i$	50.32	46.61	55.26
$\sum_i y_i^2$	256.27	222.92	307.13

Table 9.5: Dried weight of plants grown under three conditions from Dobson [2002], page 101, data of Table 6.6.

```
Residual standard error: 0.828 on 38 degrees of freedom
Multiple R-squared:  0.9724,    Adjusted R-squared:  0.971
F-statistic: 670.3 on 2 and 38 DF,  p-value: < 2.2e-16
```

The intercept is replaced by the groups because always one of them is present. Therefore both groups are significantly different from zero (sure: dried plants have a weight), however there is no difference between the groups.

9.2.3.2 Extended Dried Plants

The second example extends the first example and is from Dobson [2002], page 101, data of Table 6.6. The results of plant weights in grams for three groups (control, treatment A, treatment B) are shown in Tab. 9.5 and in Fig. 9.5. Plants from treatment B group (green) seem to be larger than the others. We will check whether this impression also holds after fitting a linear model and analyzing the results.

The ANOVA models are fitted:

```
Analysis of Variance Table

Response: weight
          Df  Sum Sq Mean Sq F value  Pr(>F)
group      2  3.7663  1.8832  4.8461 0.01591 *
Residuals 27 10.4921  0.3886
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

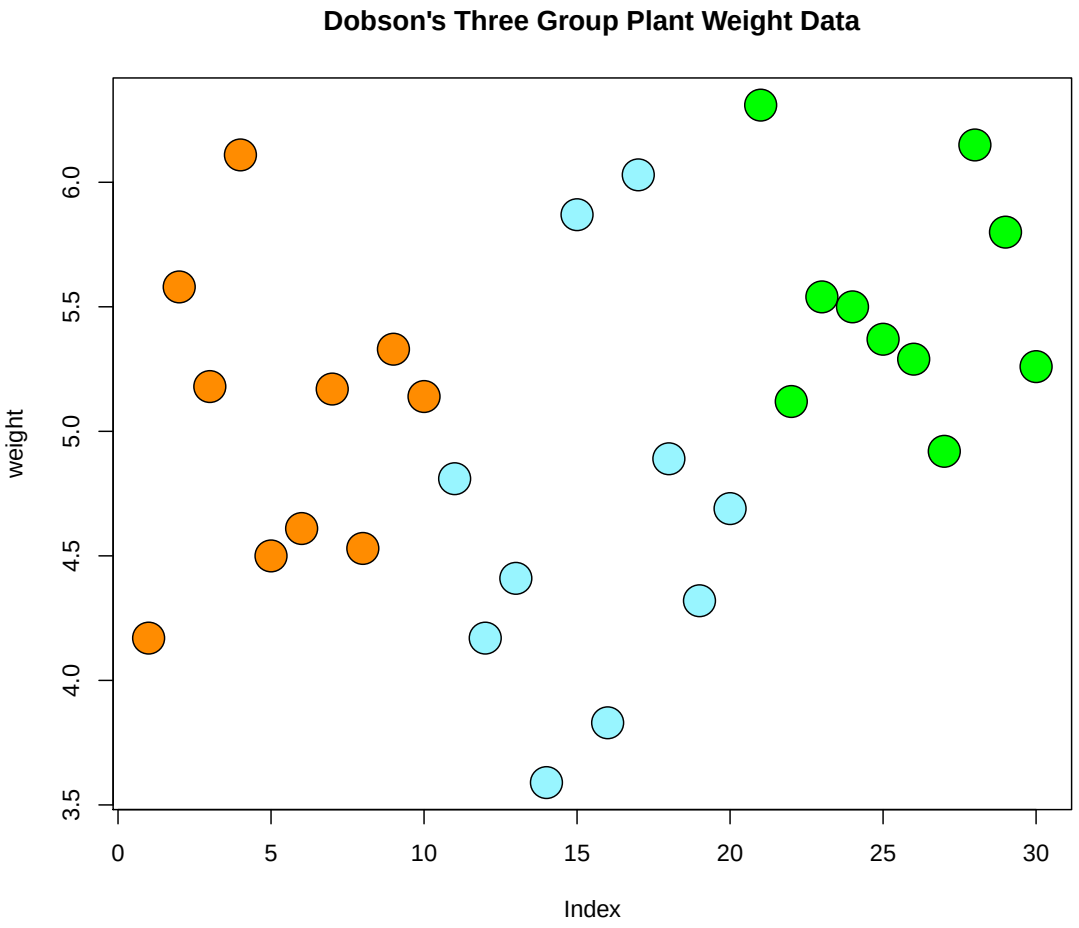

Figure 9.5: Dobson's dried plant data for three groups: orange indicates the control, blue the treatment A, and green treatment B group. Treatment B group seem to be larger than the others.

```
Residuals:
    Min      1Q  Median      3Q     Max
-1.0710 -0.4180 -0.0060  0.2627  1.3690

##################################################
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   5.0320     0.1971  25.527   <2e-16 ***
groupA       -0.3710     0.2788  -1.331   0.1944
groupB        0.4940     0.2788   1.772   0.0877 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6234 on 27 degrees of freedom
Multiple R-squared:  0.2641,    Adjusted R-squared:  0.2096
F-statistic: 4.846 on 2 and 27 DF,  p-value: 0.01591

##################################################
(Intercept)      groupA      groupB
      5.032      -0.371       0.494

            Estimate Std. Error   t value     Pr(>|t|)
(Intercept)    5.032  0.1971284 25.526514 1.936575e-20
groupA        -0.371  0.2787816 -1.330791 1.943879e-01
groupB         0.494  0.2787816  1.771996 8.768168e-02
```

Group B can be distinguished best from other groups. Its coefficient has a p-value of 0.09 which is almost significant. The F-statistic and its p-value of 0.016 shows that the groups together are significant. The estimated parameters show that group B is larger (0.494) and group A smaller (-0.371) than the control group.

9.2.3.3 Two-Factor ANOVA Toy Example

This example for a two-way ANOVA problem is from Dobson [2002], page 106, data of Table 6.9. The fictitious data is shown in Tab. 9.6, where factor A has 3 levels and factor B has 2 levels. This gives $2 \times 3 = 6$ subgroups which form all combinations of A and B levels. Each subgroup has 2 replicates. The data is shown in Fig. 9.6.

Questions for this data set can be:

- are there interaction effects?,
- are there different responses for different levels of factor A?,
- are there different responses for different levels of factor B?

Each question corresponds to a hypothesis.

We analyze this data by an ANOVA table:

	Levels of factor B		
Levels of factor A	B_1	B_2	Total
A_1	6.8, 6.6	5.3, 6.1	24.8
A_2	7.5, 7.4	7.2, 6.5	28.6
A_3	7.8, 9.1	8.8, 9.1	34.8
Total	45.2	43.0	88.2

Table 9.6: Fictitious data for two-factor ANOVA with equal numbers of observations in each subgroup from Dobson [2002].

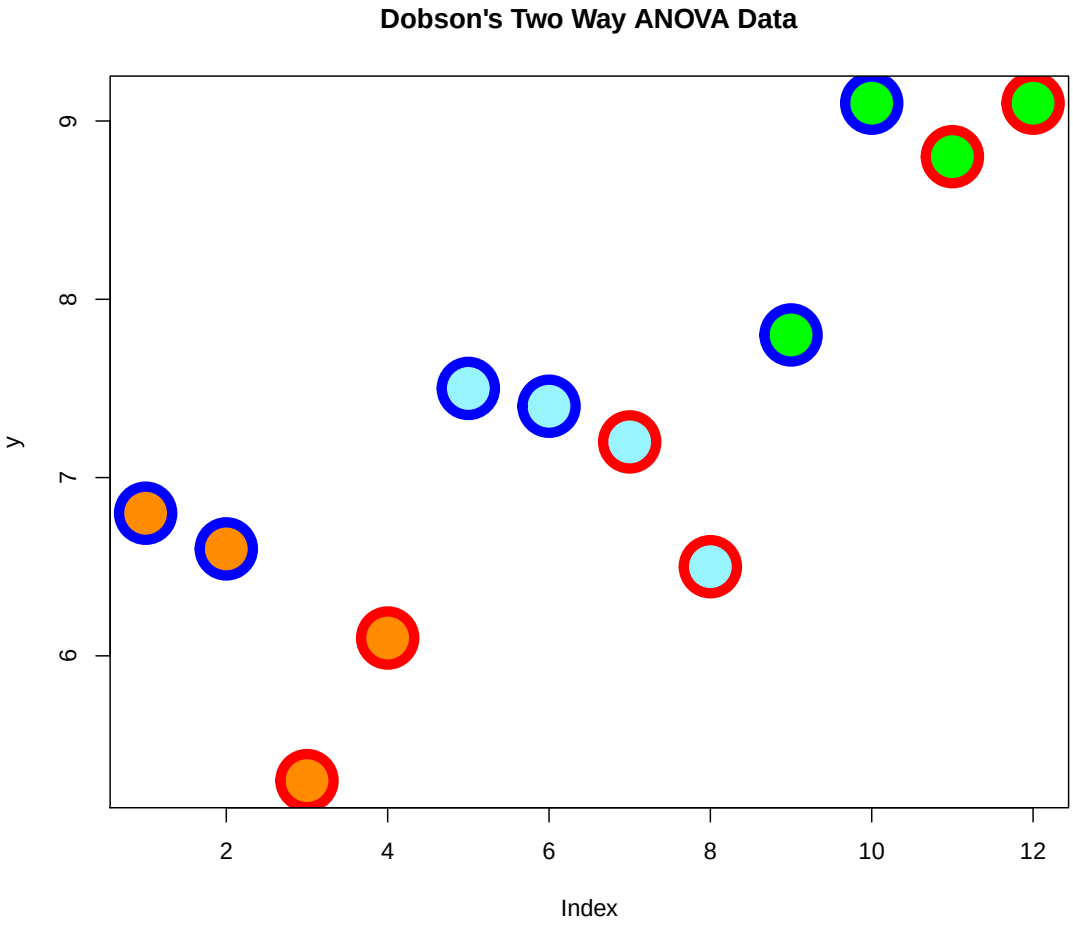

Figure 9.6: Fictitious data for two-factor ANOVA with equal numbers of observations in each subgroup from Dobson [2002]. Levels of factor A are indicated by the interior color of the circles while levels of factor B are indicated by the border color of the circles.

```
Analysis of Variance Table

Response: y
          Df  Sum Sq Mean Sq F value   Pr(>F)
a          2 12.7400  6.3700 25.8243 0.001127 **
b          1  0.4033  0.4033  1.6351 0.248225
a:b        2  1.2067  0.6033  2.4459 0.167164
Residuals  6  1.4800  0.2467
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

The is no evidence against the hypothesis that the levels of factor B do not influence the response. Similar there is no evidence against the hypothesis that the interaction effect does not influence the response. Therefore we conclude that the response is mainly affected by differences in the levels of factor A.

9.3 Analysis of Covariance

9.3.1 The Model

We now consider models that combine covariates (variables or regressors measured together with y) and designed or dummy variables as in the ANOVA models. These models are called analysis of covariance (ANCOVA) models. Thus, we know treatment groups but have also additional measurements. The additional measurements, the covariates, reduce the error variance because some variance is explained by them. Therefore, the unexplained variance is reduced before comparing the means of groups which is supposed to increase the performance of the ANOVA models.

The model is

$$\boldsymbol{y} \;=\; \boldsymbol{X\beta} \;+\; \boldsymbol{Zu} \;+\; \boldsymbol{\epsilon}\,, \tag{9.112}$$

where $\boldsymbol{Xb}$ is the same as in the ANOVA model but now the covariate values $\boldsymbol{Z}$ together with their coefficients $\boldsymbol{u}$ are added. The designed $\boldsymbol{X}$ contains zeros and ones while $\boldsymbol{Z}$ contains measured values.

For example, a one-way balanced model with only one covariate is

$$y_{gi} \;=\; \beta_0 \;+\; \beta_g \;+\; u\, z_{gi} \;+\; \epsilon_{gi}\,, 1 \le g \le G,\; 1 \le i \le \tilde{n}\,, \tag{9.113}$$

where β_g is the treatment effect, z_{gi} is the covariate that was observed together with sample y_{gi}, and u is the coefficient or slope for z_{gi}. With q covariates the model is

$$y_{gi} \;=\; \beta_0 \;+\; \beta_g \;+\; \sum_{r}^{q} u_r\, z_{gir} \;+\; \epsilon_{gi}\,, 1 \le g \le G,\; 1 \le i \le \tilde{n} \tag{9.114}$$

which is in matrix notation

$$\boldsymbol{Zu} \;=\; \begin{pmatrix} z_{111} & z_{112} & \cdots & z_{11q} \\ z_{121} & z_{122} & \cdots & z_{12q} \\ \vdots & \vdots & & \vdots \\ z_{G\tilde{n}1} & z_{G\tilde{n}2} & \cdots & z_{G\tilde{n}q} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_q \end{pmatrix}\,. \tag{9.115}$$

The matrices $\boldsymbol{X}$ and $\boldsymbol{Z}$ can be combined:

$$\boldsymbol{y} \;=\; (\boldsymbol{X}, \boldsymbol{Z}) \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{u} \end{pmatrix} \;+\; \boldsymbol{\epsilon}\,. \tag{9.116}$$

The normal equations are

$$\begin{pmatrix} \boldsymbol{X}^T \\ \boldsymbol{Z}^T \end{pmatrix} (\boldsymbol{X}, \boldsymbol{Z}) \begin{pmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\boldsymbol{u}} \end{pmatrix} \;=\; \begin{pmatrix} \boldsymbol{X}^T\boldsymbol{X} & \boldsymbol{X}^T\boldsymbol{Z} \\ \boldsymbol{Z}^T\boldsymbol{X} & \boldsymbol{Z}^T\boldsymbol{Z} \end{pmatrix} \begin{pmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\boldsymbol{u}} \end{pmatrix} \;=\; \begin{pmatrix} \boldsymbol{X}^T\boldsymbol{y} \\ \boldsymbol{Z}^T\boldsymbol{y} \end{pmatrix}\,. \tag{9.117}$$

We obtain two equations:

$$\boldsymbol{X}^T\boldsymbol{X}\,\hat{\boldsymbol{\beta}} \;+\; \boldsymbol{X}^T\boldsymbol{Z}\,\hat{\boldsymbol{u}} \;=\; \boldsymbol{X}^T\boldsymbol{y} \tag{9.118}$$

$$\boldsymbol{Z}^T\boldsymbol{X}\,\hat{\boldsymbol{\beta}} \;+\; \boldsymbol{Z}^T\boldsymbol{Z}\,\hat{\boldsymbol{u}} \;=\; \boldsymbol{Z}^T\boldsymbol{y}\,. \tag{9.119}$$

Solving the first equation for $\hat{\boldsymbol{\beta}}$ gives

$$\begin{aligned}\hat{\boldsymbol{\beta}} &= (\boldsymbol{X}^T\boldsymbol{X})^+\boldsymbol{X}^T\boldsymbol{y} - (\boldsymbol{X}^T\boldsymbol{X})^+\boldsymbol{X}^T\boldsymbol{Z}\,\hat{\boldsymbol{u}} \\ &= \hat{\boldsymbol{\beta}}_0 - (\boldsymbol{X}^T\boldsymbol{X})^+\boldsymbol{X}^T\boldsymbol{Z}\,\hat{\boldsymbol{u}}\,,\end{aligned} \tag{9.120}$$

where $(\boldsymbol{X}^T\boldsymbol{X})^+$ denotes the pseudo inverse of $(\boldsymbol{X}^T\boldsymbol{X})$ and $\hat{\boldsymbol{\beta}}_0 = (\boldsymbol{X}^T\boldsymbol{X})^+\boldsymbol{X}^T\boldsymbol{y}$ is the solution to the normal equations of the model without covariates.

We now substitute this equation for $\hat{\boldsymbol{\beta}}$ into the second equation in order to solve for $\hat{\boldsymbol{u}}$:

$$\boldsymbol{Z}^T\boldsymbol{X}\,\left((\boldsymbol{X}^T\boldsymbol{X})^+\boldsymbol{X}^T\boldsymbol{y} - (\boldsymbol{X}^T\boldsymbol{X})^+\boldsymbol{X}^T\boldsymbol{Z}\,\hat{\boldsymbol{u}}\right) + \boldsymbol{Z}^T\boldsymbol{Z}\,\hat{\boldsymbol{u}} = \boldsymbol{Z}^T\boldsymbol{y}\,. \tag{9.121}$$

We define

$$\boldsymbol{P} = \boldsymbol{X}\,(\boldsymbol{X}^T\boldsymbol{X})^+\boldsymbol{X}^T \tag{9.122}$$

and obtain for $\hat{\boldsymbol{u}}$:

$$\hat{\boldsymbol{u}} = \left(\boldsymbol{Z}^T(\boldsymbol{I} - \boldsymbol{P})\,\boldsymbol{Z}\right)^{-1}\boldsymbol{Z}^T(\boldsymbol{I} - \boldsymbol{P})\,\boldsymbol{y}\,. \tag{9.123}$$

We immediately obtain a solution for $\hat{\boldsymbol{\beta}}$:

$$\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\beta}}_0 - (\boldsymbol{X}^T\boldsymbol{X})^+\boldsymbol{X}^T\boldsymbol{Z}\,\hat{\boldsymbol{u}}\,. \tag{9.124}$$

Different hypotheses can be tested like H_0: $\beta_1 = \beta_2 = \ldots = \beta_G$ (equality of treatment effects), H_0: $\boldsymbol{u} = \boldsymbol{0}$ (slope equal to zero), or H_0: $u_1 = u_2 = \ldots = u_q$ (equal slopes, homogeneity of slopes) Rencher and Schaalje [2008]. Also two-way models with covariates can be constructed Rencher and Schaalje [2008].

9.3.2 Examples

9.3.2.1 Achievement Scores

The data are from Dobson [2002], page 111, data of Table 6.12. The data are listed in Tab. 9.7 which is originally from Winer (1971), page 776. The responses are achievement scores measured at three levels of a factor representing three different training methods. The covariates are aptitude scores measured before training commenced. We want to compare the training methods, taking into account differences in initial aptitude between the three groups of subjects. The data is plotted in Fig. 9.7, where the data points are jittered to avoid data points covering others.

The figure shows that the achievement scores y increase linearly with aptitude x. Further the achievement scores y are generally higher for training methods B and C if compared to A. We want to test the hypothesis that there are no differences in mean achievement scores among the three training methods, after adjustment for initial aptitude.

```
Analysis of Variance Table

Response: y
```

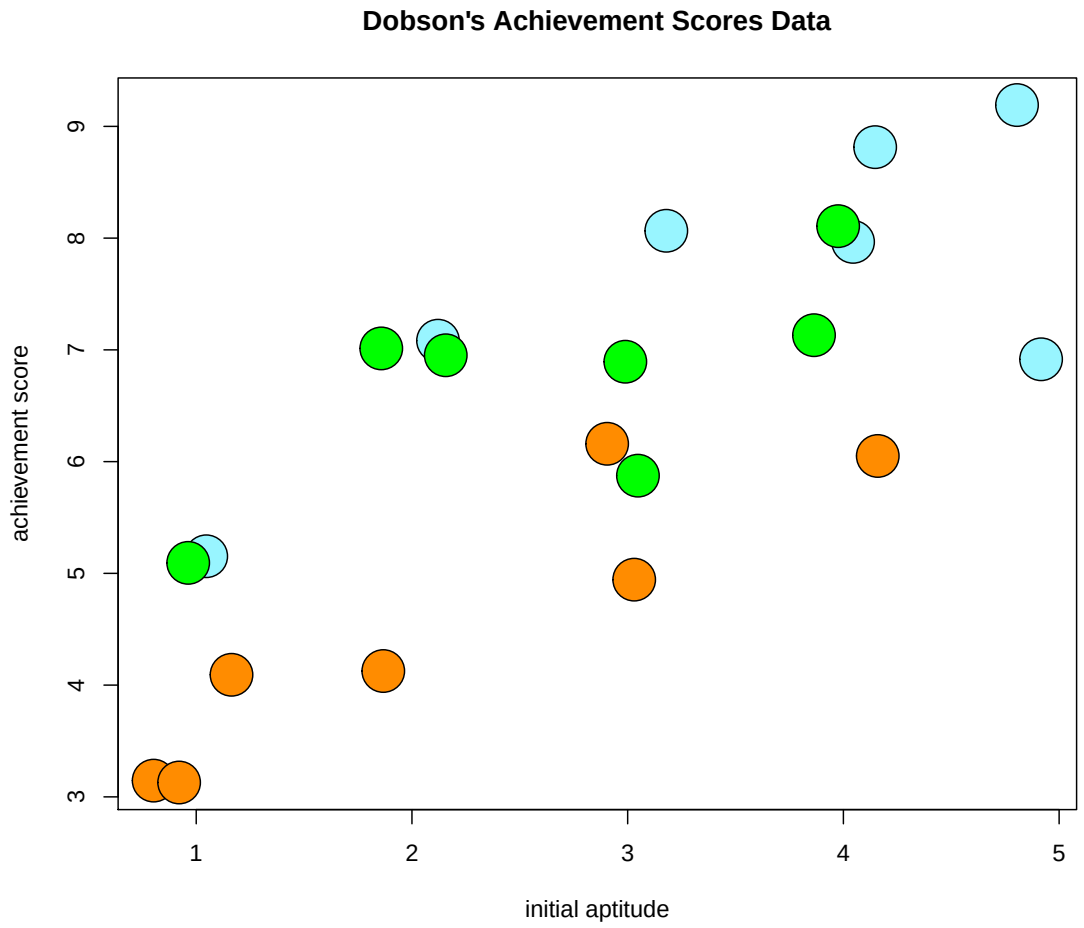

Figure 9.7: Scatter plot of Dobson's achievement scores data. Observations are jittered to avoid data points covering others.

	Training method					
	A		B		C	
	y	x	y	x	y	x
	6	3	8	4	6	3
	4	1	9	5	7	2
	5	3	7	5	7	2
	3	1	9	4	7	3
	4	2	8	3	8	4
	3	1	5	1	5	1
	6	4	7	2	7	4
$\sum x$ / $\sum y$	31	15	53	24	47	19
$\sum x^2$ / $\sum y^2$	147	41	413	96	321	59
$\sum xy$	75		191		132	

Table 9.7: The responses are achievement scores measured at three levels of a factor representing three different training methods. The data is from Dobson [2002] and originally from Winer (1971), p. 776.

```
          Df Sum Sq Mean Sq F value    Pr(>F)
x          1 36.575  36.575  60.355 5.428e-07 ***
m          2 16.932   8.466  13.970 0.0002579 ***
Residuals 17 10.302   0.606
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Of course, the initial aptitude x is significant for the achievement scores y. More importantly, the training methods, which are given by m, show significant differences concerning the achievement scores. We obtain the same result by looking at the ANOVA table of different models:

```
Analysis of Variance Table

Model 1: y ~ x + m
Model 2: y ~ x
  Res.Df    RSS Df Sum of Sq     F    Pr(>F)
1     17 10.302
2     19 27.234 -2   -16.932 13.97 0.0002579 ***
```

Again we see that the training methods show significant differences after adjusting for the initial aptitude.

9.3.2.2 Birthweights of Girls and Boys

The data set is from Dobson [2002], page 30, data of Table 2.3. Birthweights (in grams) and estimated gestational ages (in weeks) of 12 male and female babies are sampled. Tab. 9.8 shows

	Boys		Girls	
	Age	Birthweight	Age	Birthweight
	40	2968	40	3317
	38	2795	36	2729
	40	3163	40	2935
	35	2925	38	2754
	36	2625	42	3210
	37	2847	39	2817
	41	3292	40	3126
	40	3473	37	2539
	37	2628	36	2412
	38	3176	38	2991
	40	3421	39	2875
	38	2975	40	3231
Means	38.33	3024.00	38.75	2911.33

Table 9.8: Birthweight and gestational age for boys and girls from Dobson [2002].

the data. The mean ages are almost the same for both sexes but the mean birthweight for boys is higher than the mean birthweight for girls. The data are shown in a scatter plot in Fig. 9.8. There is a linear trend of birth weight increasing with gestational age and the girls tend to weigh less than the boys of the same gestational age. The question of interest is whether the rate of increase of birthweight with gestational age is the same for boys and girls.

For analysis we fit a linear model where the groups are male and female and the covariate is the age:

```
Residuals:
    Min      1Q  Median      3Q     Max
-257.49 -125.28  -58.44  169.00  303.98

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -1610.28     786.08  -2.049   0.0532 .
sexFemale    -163.04      72.81  -2.239   0.0361 *
age           120.89      20.46   5.908 7.28e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 177.1 on 21 degrees of freedom
Multiple R-squared:   0.64,     Adjusted R-squared:  0.6057
F-statistic: 18.67 on 2 and 21 DF,  p-value: 2.194e-05

Correlation of Coefficients:
          (Intercept) sexFemale
```

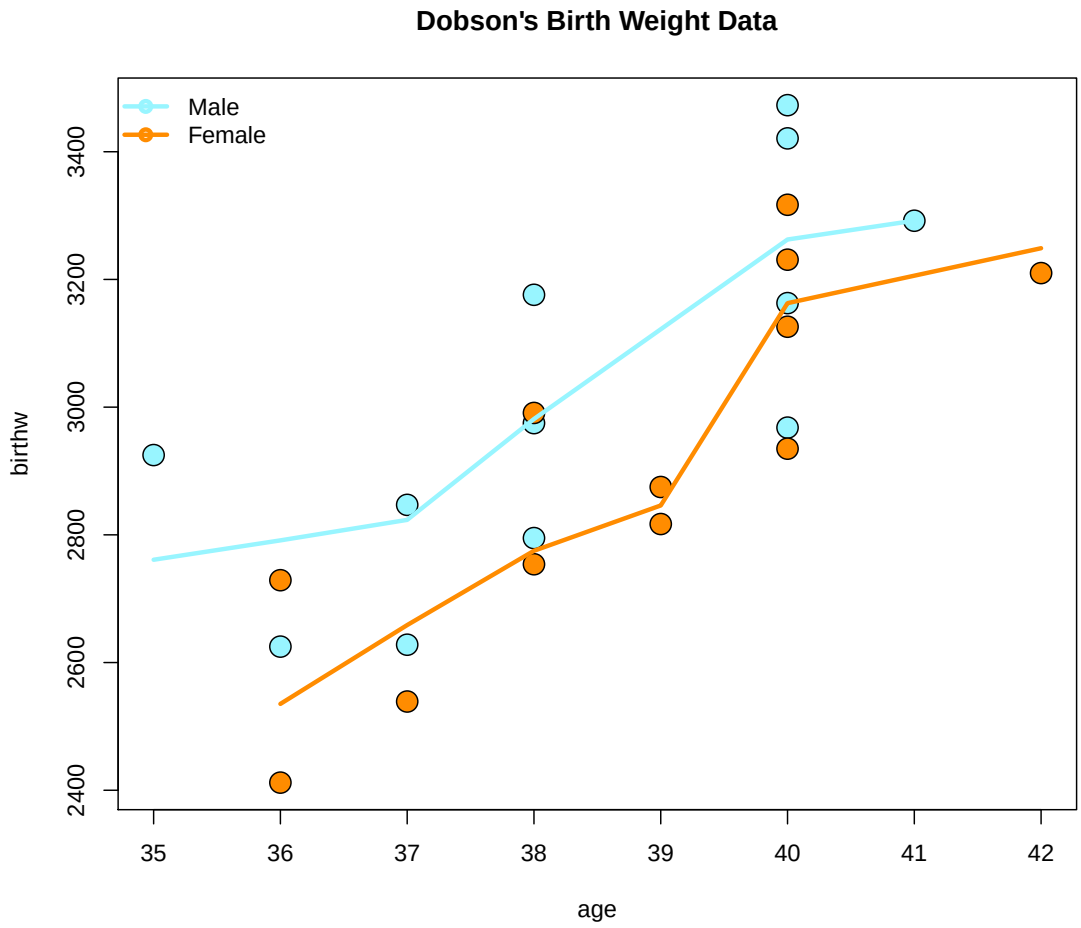

Figure 9.8: Scatter plot of Dobson's birthweight data. Regression lines are shown.

```
sexFemale  0.07
age       -1.00       -0.12
```

Of course, the birthweight depends on the age, which is highly significant. However also the sex is significant at a level of 0.05. Females weigh less than males as the coefficient for females is -163.04.

The intercept was not important, we fit the model without an intercept:

```
Residuals:
    Min      1Q  Median      3Q     Max
-257.49 -125.28  -58.44  169.00  303.98

Coefficients:
          Estimate Std. Error t value Pr(>|t|)
sexMale   -1610.28     786.08  -2.049   0.0532 .
sexFemale -1773.32     794.59  -2.232   0.0367 *
age         120.89      20.46   5.908 7.28e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 177.1 on 21 degrees of freedom
Multiple R-squared:  0.9969,    Adjusted R-squared:  0.9965
F-statistic:  2258 on 3 and 21 DF,  p-value: < 2.2e-16

Correlation of Coefficients:
          sexMale sexFemale
sexFemale  1.00
age       -1.00   -1.00
```

The intercept is now attributed to the males. This is in agreement to the result in previous setting, where the males were the reference group. Either the reference group effect or the constant offset (the intercept) is set to zero.

We compare the models by an ANOVA table:

```
Analysis of Variance Table

Model 1: birthw ~ sex + age
Model 2: birthw ~ sex + age - 1
  Res.Df    RSS Df  Sum of Sq F Pr(>F)
1     21 658771
2     21 658771  0 1.5134e-09
```

The intercept is not required.

Next we fit a more complex model which contains the interaction of factor sex with variable age:

```
Residuals:
    Min      1Q  Median      3Q     Max
-246.69 -138.11  -39.13  176.57  274.28

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
sexMale       -1268.67    1114.64  -1.138 0.268492
sexFemale     -2141.67    1163.60  -1.841 0.080574 .
sexMale:age     111.98      29.05   3.855 0.000986 ***
sexFemale:age   130.40      30.00   4.347 0.000313 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 180.6 on 20 degrees of freedom
Multiple R-squared:  0.9969,    Adjusted R-squared:  0.9963
F-statistic:  1629 on 4 and 20 DF,  p-value: < 2.2e-16

Correlation of Coefficients:
              sexMale sexFemale sexMale:age
sexFemale      0.00
sexMale:age   -1.00    0.00
sexFemale:age  0.00   -1.00      0.00
```

The interaction terms (interaction of factor sex with variable age) explain significant variance in the data. However the interaction factors are driven by age. Thus, age is now less significant as it is divided into two interaction factors.

The ANOVA table shows

```
Analysis of Variance Table

Model 1: birthw ~ sex + sex:age - 1
Model 2: birthw ~ sex + age - 1
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     20 652425
2     21 658771 -1   -6346.2 0.1945 0.6639
```

The difference between the models is not significant. Only age is separated into the combined factors containing the sex.

9.4 Mixed Effects Models

So far, we considered only the noise $\boldsymbol{\epsilon}$ as random variable given $\boldsymbol{x}$. Thus, only $\boldsymbol{\epsilon}$ could explain the variance of $p(y \mid \boldsymbol{x})$. We now assume there is a second source of variation which is represented by a hidden or latent variable $\boldsymbol{u}$. If the variance of $\boldsymbol{u}$ is not known, then the parameter estimation becomes more complicated. The error variance has to be distinguished from the variance through $\boldsymbol{u}$. So far the parameters could be estimated without knowing the error variance. We assumed that the errors have the same spherical variance. Therefore this variance would factor out in the objective and the normal equations would not change. For mixed effect models that is no longer the case.

For each observation y there is a corresponding latent variable $\boldsymbol{u}$:

$$y \;=\; \boldsymbol{x}^T\boldsymbol{\beta} \;+\; \boldsymbol{z}^T\boldsymbol{u} \;+\; \epsilon\,. \tag{9.125}$$

$\boldsymbol{z}$ is a vector indicating the presence of the latent variable, which can be sampled with y or be designed via dummy variables.

We assume that

$$\begin{aligned}
\mathrm{E}(\boldsymbol{u}) \;&=\; \mathbf{0}\,, && (9.126)\\
\mathrm{E}(\epsilon) \;&=\; \mathbf{0}\,, && (9.127)\\
\mathrm{Var}(\boldsymbol{u}) \;&=\; \boldsymbol{G}\,, && (9.128)\\
\mathrm{Var}(\boldsymbol{\epsilon}) \;&=\; \boldsymbol{R}\,, && (9.129)\\
\mathrm{Cov}(\boldsymbol{\epsilon}, \boldsymbol{u}) \;&=\; \mathbf{0}\,. && (9.130)
\end{aligned}$$

The model in matrix notation is

$$\boldsymbol{y} \;=\; \boldsymbol{X}\boldsymbol{\beta} \;+\; \boldsymbol{Z}\boldsymbol{u} \;+\; \boldsymbol{\epsilon}\,. \tag{9.131}$$

The design matrix is $\boldsymbol{X}$ with $\boldsymbol{\beta}$ as the coefficient vector of fixed effects. $\boldsymbol{u}$ is the vector of random effects with $\boldsymbol{Z}$ as fixed predictor matrix. $\boldsymbol{Z}$ is often used to specify group memberships or certain measurement conditions.

We have

$$\begin{aligned}
\mathrm{E}(\boldsymbol{y}) \;&=\; \boldsymbol{X}\boldsymbol{\beta}\,, && (9.132)\\
\mathrm{Var}(\boldsymbol{y}) \;&=\; \boldsymbol{Z}^T\boldsymbol{G}\boldsymbol{Z} \;+\; \boldsymbol{R}\,. && (9.133)
\end{aligned}$$

These properties of $\boldsymbol{y}$ follow immediately from the assumptions.

9.4.1 Approximative Estimator

We want to find an estimator for both $\boldsymbol{e}$ and $\boldsymbol{u}$. The estimator for $\boldsymbol{u}$ is the posterior, that is, the distribution of $\boldsymbol{u}$ after having seen the observation, while the prior is the distribution of $\boldsymbol{u}$ without an observation.

9.4.1.1 Estimator for Beta

We assume that both $\boldsymbol{G} = \sigma_u^2 \boldsymbol{I}$ and $\boldsymbol{R} = \sigma^2 \boldsymbol{I}$ as well as normal distributed errors. Then, we approximate $\boldsymbol{G}$ by $\hat{\sigma}_u^2 \boldsymbol{I}$ and $\boldsymbol{R}$ by $\hat{\sigma}^2 \boldsymbol{I}$. We have to find an estimator $\hat{\sigma}_u^2$ for σ_u^2 and an estimator $\hat{\sigma}^2$ for σ^2. One approach for this estimate is the restricted (or residual) maximum likelihood (REML) estimator.

We define

$$\boldsymbol{K} \;=\; \boldsymbol{C}(\boldsymbol{I} \;-\; \boldsymbol{P}) \;=\; \boldsymbol{C}\left(\boldsymbol{I} \;-\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{+}\boldsymbol{X}^T\right)\;, \tag{9.134}$$

where $\boldsymbol{C}$ is a full-rank transformation of the rows of $(\boldsymbol{I} \;-\; \boldsymbol{P})$. We immediately see that

$$\boldsymbol{K}\;\boldsymbol{X} \;=\; \boldsymbol{0}\,. \tag{9.135}$$

We define

$$\boldsymbol{\Sigma} \;=\; \sigma_u^2\;\boldsymbol{Z}\boldsymbol{Z}^T \;+\; \sigma^2 \boldsymbol{I}_n\,. \tag{9.136}$$

We know the distribution of $\boldsymbol{K}\;\boldsymbol{y}$:

$$\boldsymbol{K}\;\boldsymbol{y} \;\sim\; \mathcal{N}\left(\boldsymbol{0}\,,\; \boldsymbol{K}\;\boldsymbol{\Sigma}\;\boldsymbol{K}^T\right)\;. \tag{9.137}$$

Using this distribution, estimators for σ^2 and σ_u^2 can be obtained by solving the equations:

$$\mathrm{Tr}\left(\boldsymbol{K}^T\left(\boldsymbol{K}\;\boldsymbol{\Sigma}\;\boldsymbol{K}^T\right)^{-1}\boldsymbol{K}\right) \;=\; \boldsymbol{y}^T\boldsymbol{K}^T\left(\boldsymbol{K}\;\boldsymbol{\Sigma}\;\boldsymbol{K}^T\right)^{-1}\boldsymbol{K}\;\boldsymbol{K}^T\left(\boldsymbol{K}\;\boldsymbol{\Sigma}\;\boldsymbol{K}^T\right)^{-1}\boldsymbol{K}\;\boldsymbol{y} \tag{9.138}$$

$$\mathrm{Tr}\left(\boldsymbol{K}^T\left(\boldsymbol{K}\;\boldsymbol{\Sigma}\;\boldsymbol{K}^T\right)^{-1}\boldsymbol{K}\boldsymbol{Z}\boldsymbol{Z}^T\right) \;=\; \boldsymbol{y}^T\boldsymbol{K}^T\left(\boldsymbol{K}\;\boldsymbol{\Sigma}\;\boldsymbol{K}^T\right)^{-1}\boldsymbol{K}\;\boldsymbol{Z}\boldsymbol{Z}^T\boldsymbol{K}^T\left(\boldsymbol{K}\;\boldsymbol{\Sigma}\;\boldsymbol{K}^T\right)^{-1}\boldsymbol{K}\;\boldsymbol{y}\,. \tag{9.139}$$

These equations are obtained by setting the derivatives of the likelihood of $\boldsymbol{K}\boldsymbol{y}$ with respect to σ^2 and to σ_u^2 to zero.

The solution of these equations are the estimators $\hat{\sigma}^2$ and $\hat{\sigma}_u^2$ for σ^2 and σ_u^2, respectively. Using these estimators, we define

$$\hat{\boldsymbol{\Sigma}} \;=\; \hat{\sigma}_u^2\;\boldsymbol{Z}\boldsymbol{Z}^T \;+\; \hat{\sigma}^2 \boldsymbol{I}_n\,. \tag{9.140}$$

to obtain an estimator for $\boldsymbol{\beta}$ as

$$\hat{\boldsymbol{\beta}} \;=\; \left(\boldsymbol{X}^T\hat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{X}\right)^{+}\boldsymbol{X}^T\hat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{y}\,. \tag{9.141}$$

This is the estimated generalized least squares (EGLS) estimator. The EGLS estimator is only asymptotically the minimum variance unbiased estimator (MVUE).

Similarly, an approximated estimate for the covariance of $\boldsymbol{\beta}$ is

$$\mathrm{Var}(\hat{\boldsymbol{\beta}}) \;=\; \left(\boldsymbol{X}^T\hat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{X}\right)^{+}\boldsymbol{X}^T\hat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{X}\;\left(\boldsymbol{X}^T\hat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{X}\right)^{+}\;. \tag{9.142}$$

For full rank $\boldsymbol{X}$ that is

$$\mathrm{Var}(\hat{\boldsymbol{\beta}}) \;=\; \left(\boldsymbol{X}^T\hat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{X}\right)^{-1}\;. \tag{9.143}$$

Large-sample estimator. Using this approximation for the variance, we can approximate $100(1-\alpha)\%$ confidence intervals by

$$\boldsymbol{a}^T\boldsymbol{\beta} \in \boldsymbol{a}^T\hat{\boldsymbol{\beta}} \pm z_{\alpha/2}\sqrt{\boldsymbol{a}^T\left(\boldsymbol{X}^T\hat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{X}\right)^+\boldsymbol{a}}\,. \tag{9.144}$$

Using the $\boldsymbol{a} = \boldsymbol{e}_j$ gives a confidence interval for β_j. However this confidence interval is not valid for a small number of samples. For a small number of samples we use a different approach.

Small-sample estimator. For

$$t = \frac{\boldsymbol{a}^T\hat{\boldsymbol{\beta}}}{\sqrt{\boldsymbol{a}^T\left(\boldsymbol{X}^T\hat{\boldsymbol{\Sigma}}^{-1}\boldsymbol{X}\right)^+\boldsymbol{a}}} \tag{9.145}$$

often a t-distribution with unknown degrees of freedom is assumed. The task is to estimate the degrees of freedom to compute confidence intervals or p-values. Different approaches exist to estimate the degrees of freedom Rencher and Schaalje [2008].

9.4.1.2 Estimator for u

If $\boldsymbol{u}$ is normally distributed, then we know the posterior $p(\boldsymbol{u} \mid \boldsymbol{y})$.

We use the following connection between two normally distributed variables:

$$\begin{aligned}
&\boldsymbol{u} \sim \mathcal{N}(\boldsymbol{\mu}_u, \boldsymbol{\Sigma}_{uu})\ ,\ \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}_y, \boldsymbol{\Sigma}_{yy})\ , \\
&\boldsymbol{\Sigma}_{uv} = \mathrm{Cov}(\boldsymbol{y}, \boldsymbol{u}) \ \text{ and } \ \boldsymbol{\Sigma}_{vu} = \mathrm{Cov}(\boldsymbol{u}, \boldsymbol{y}) : \\
&\boldsymbol{u} \mid \boldsymbol{y} \sim \mathcal{N}\left(\boldsymbol{\mu}_u + \boldsymbol{\Sigma}_{vu}\boldsymbol{\Sigma}_{yy}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}_y)\ ,\ \boldsymbol{\Sigma}_{uu} - \boldsymbol{\Sigma}_{vu}\boldsymbol{\Sigma}_{yy}^{-1}\boldsymbol{\Sigma}_{uv}\right)
\end{aligned} \tag{9.146}$$

The covariance between $\boldsymbol{u}$ and $\boldsymbol{y}$ is

$$\mathrm{Cov}(\boldsymbol{u}, \boldsymbol{y}) = \mathrm{Cov}(\boldsymbol{u}\ ,\ \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{u} + \boldsymbol{\epsilon}) = \boldsymbol{G}\,\boldsymbol{Z}^T\,. \tag{9.147}$$

and we have

$$\mathrm{E}(\boldsymbol{u}) = \boldsymbol{0}\,, \tag{9.148}$$
$$\mathrm{Var}(\boldsymbol{u}) = \boldsymbol{G}\,, \tag{9.149}$$
$$\mathrm{E}(\boldsymbol{y}) = \boldsymbol{X}\boldsymbol{\beta}\,, \tag{9.150}$$
$$\mathrm{Var}(\boldsymbol{y}) = \boldsymbol{Z}^T\boldsymbol{G}\boldsymbol{Z} + \boldsymbol{R}\,. \tag{9.151}$$

Therefore we obtain

$$\boldsymbol{u} \mid \boldsymbol{y} \sim \mathcal{N}\left(\boldsymbol{G}\,\boldsymbol{Z}^T\left(\boldsymbol{Z}^T\boldsymbol{G}\boldsymbol{Z} + \boldsymbol{R}\right)^{-1}(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})\ ,\ \boldsymbol{G} - \boldsymbol{G}\,\boldsymbol{Z}^T\left(\boldsymbol{Z}^T\boldsymbol{G}\boldsymbol{Z} + \boldsymbol{R}\right)^{-1}\boldsymbol{Z}\,\boldsymbol{G}^T\right)\,. \tag{9.152}$$

This posterior can be computed if $\boldsymbol{G}$, $\boldsymbol{R}$, and $\boldsymbol{\beta}$ are known. We can use above approximation for these values

$$\boldsymbol{G} = \hat{\sigma}_u^2\boldsymbol{I}, \tag{9.153}$$
$$\boldsymbol{R} = \hat{\sigma}^2\boldsymbol{I}, \tag{9.154}$$
$$\boldsymbol{Z}^T\boldsymbol{G}\boldsymbol{Z} + \boldsymbol{R} = \hat{\boldsymbol{\Sigma}} \tag{9.155}$$

to estimate the posterior.

9.4.2 Full Estimator

Here we consider the full estimator and not only an approximation. Henderson's "mixed model equations" (MME) are:

$$\begin{pmatrix} \boldsymbol{X}^T\boldsymbol{R}^{-1}\boldsymbol{X} & \boldsymbol{X}^T\boldsymbol{R}^{-1}\boldsymbol{Z} \\ \boldsymbol{Z}^T\boldsymbol{R}^{-1}\boldsymbol{X} & \boldsymbol{Z}^T\boldsymbol{R}^{-1}\boldsymbol{Z} \,+\, \boldsymbol{G}^{-1} \end{pmatrix} \begin{pmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\boldsymbol{u}} \end{pmatrix} \;=\; \begin{pmatrix} \boldsymbol{X}^T\boldsymbol{R}^{-1}\boldsymbol{y} \\ \boldsymbol{Z}^T\boldsymbol{R}^{-1}\boldsymbol{y} \end{pmatrix} . \tag{9.156}$$

The solution to these equations are best linear unbiased estimates (BLUE).

Mixed effect models can also be fitted by the EM algorithm. Variance components are considered as unobserved variables which are estimated in the E-step. The M-step maximizes the other parameters. For example, the R function `lme()` ("linear mixed effect") of the R package `nlme` ("non-linear mixed effect") implements such an EM algorithm.

For $\boldsymbol{R} = \sigma^2\boldsymbol{I}$ we obtain for the MME

$$\begin{pmatrix} \boldsymbol{X}^T\boldsymbol{X} & \boldsymbol{X}^T\boldsymbol{Z} \\ \boldsymbol{Z}^T\boldsymbol{X} & \boldsymbol{Z}^T\boldsymbol{Z} \,+\, \sigma^{-2}\,\boldsymbol{G}^{-1} \end{pmatrix} \begin{pmatrix} \hat{\boldsymbol{\beta}} \\ \hat{\boldsymbol{u}} \end{pmatrix} \;=\; \begin{pmatrix} \boldsymbol{X}^T\boldsymbol{y} \\ \boldsymbol{Z}^T\boldsymbol{y} \end{pmatrix} . \tag{9.157}$$

9.5 Generalized Linear Models

So far, we assumed spherical and often normal errors. However other distributions may be possible — even discrete or count distributions. For example with counts the error corresponds to the deviation from the mean count. The error-free model is obtained by the expectation of the observation y_i: $\mathrm{E}(y_i) = \mu_i = \boldsymbol{x}_i^T\boldsymbol{\beta}$. We now generalize this relation by introducing a link function g. The link function g relates the mean $\mathrm{E}(y_i) = \mu_i$ to the linear component $g(\mu_i) = \boldsymbol{x}_i^T\boldsymbol{\beta}$.

Generalized linear models require

(i) a *random component* or an error distribution which specifies the probability distribution of the response y,

(i) a *systematic component* which is a linear function of the explanatory variables / regressors,

(i) a *link function* which determines the functional relation between the expectation of the random variable and the systematic component, i.e. the linear function.

For the *exponential dispersion model* with the natural parameter θ_i and dispersion parameter ϕ, the density is

$$f(y_i \mid \theta_i, \phi) \;=\; \exp\left(\frac{y_i\theta_i \;-\; b(\theta_i)}{a(\phi)} \;+\; c(y_i, \phi) \right) , \tag{9.158}$$

where

$$b(\theta_i) \;=\; a(\phi)\; \ln \int \exp\left(\frac{y_i\theta_i}{a(\phi)} \;+\; c(y_i, \phi) \right)\, dy_i \, . \tag{9.159}$$

The function b is a normalizing constant (in y_i) that ensures f to be a distribution:

$$\int f(y_i \mid \theta_i, \phi)\, dy_i \;=\; \frac{\int \exp\left(\frac{y_i\theta_i}{a(\phi)} \;+\; c(y_i,\phi)\right)\, dy_i}{\int \exp\left(\frac{y_i\theta_i}{a(\phi)} \;+\; c(y_i,\phi)\right)\, dy_i} \;=\; 1\,. \tag{9.160}$$

Using this density we can derive Rao and Toutenburg [1999]:

$$\mathrm{E}(y_i) \;=\; \mu_i \;=\; b'(\theta_i) \tag{9.161}$$

$$\mathrm{Var}(y_i) \;=\; b''(\theta_i)\, a(\phi)\,. \tag{9.162}$$

These equations can be derived as follows. For the mean we have

$$\begin{aligned}
\frac{\partial b(\theta_i)}{\partial \theta_i} \;&=\; a(\phi)\, \frac{\int (y_i/a(\phi)) \exp\left(\frac{y_i\theta_i}{a(\phi)} \;+\; c(y_i,\phi)\right)\, dy_i}{\int \exp\left(\frac{y_i\theta_i}{a(\phi)} \;+\; c(y_i,\phi)\right)\, dy_i} \\
&=\; a(\phi)\, \frac{\int (y_i/a(\phi)) \exp\left(\frac{y_i\theta_i}{a(\phi)} \;+\; c(y_i,\phi)\right)\, dy_i}{exp(b(\theta_i)/a(\phi))} \\
&=\; \int y_i \exp\left(\frac{y_i\theta_i \;-\; b(\theta_i)}{a(\phi)} \;+\; c(y_i,\phi)\right)\, dy_i \\
&=\; \mu_i\,.
\end{aligned} \tag{9.163}$$

and for the variance we obtain

$$\begin{aligned}
\frac{\partial^2 b(\theta_i)}{\partial \theta_i^2} \;&=\; -\,\frac{1}{a(\phi)}\, \frac{\left(\int y_i \,\exp\left(\frac{y_i\theta_i}{a(\phi)} \;+\; c(y_i,\phi)\right)\, dy_i\right)^2}{\left(\int \exp\left(\frac{y_i\theta_i}{a(\phi)} \;+\; c(y_i,\phi)\right)\, dy_i\right)^2} \;+ \\
&\qquad \frac{1}{a(\phi)}\, \frac{\int y_i^2 \,\exp\left(\frac{y_i\theta_i}{a(\phi)} \;+\; c(y_i,\phi)\right)\, dy_i}{\int \exp\left(\frac{y_i\theta_i}{a(\phi)} \;+\; c(y_i,\phi)\right)\, dy_i} \\
&=\; \frac{1}{a(\phi)}\left(-\,\mu_i^2 \;+\; \mathrm{E}(y_i^2)\right) \;=\; \frac{1}{a(\phi)}\mathrm{Var}(y_i)\,.
\end{aligned} \tag{9.164}$$

The log-likelihood is

$$\ln \mathcal{L} \;=\; \sum_{i=1}^{n} \ln \mathcal{L}_i \;=\; \sum_{i=1}^{n} \left(\frac{y_i\theta_i \;-\; b(\theta_i)}{a(\phi)} \;+\; c(y_i,\phi)\right)\,, \tag{9.165}$$

where $\mathcal{L}_i = f(y_i \mid \theta_i, \phi)$ is the conditional likelihood of y_i given $\boldsymbol{x}_i$.

The derivative of the log-likelihood with respect to the coefficient β_j is

$$\frac{\partial \ln \mathcal{L}_i}{\partial \beta_j} \;=\; \frac{\partial \ln \mathcal{L}_i}{\partial \theta_i}\, \frac{\partial \theta_i}{\partial \mu_i}\, \frac{\partial \mu_i}{\partial g(\mu_i)}\, \frac{\partial g(\mu_i)}{\partial \beta_j}\,. \tag{9.166}$$

We only applied the chain rule a couple of times.

The derivatives which appear in the chain rule can be computed separately. We compute these derivatives, where we use $\mu_i = b'(\theta_i)$:

$$\frac{\partial \ln \mathcal{L}_i}{\partial \theta_i} = \frac{y_i - b'(\theta_i)}{a(\phi)} = \frac{y_i - \mu_i}{a(\phi)} \tag{9.167}$$

$$\frac{\partial \theta_i}{\partial \mu_i} = \left(\frac{\partial \mu_i}{\partial \theta_i}\right)^{-1} = \left(b''(\theta_i)\right)^{-1} = \frac{a(\phi)}{\mathrm{Var}(y_i)} \tag{9.168}$$

$$\frac{\partial \mu_i}{\partial g(\mu_i)} = \left(\frac{\partial g(\mu_i)}{\partial \mu_i}\right)^{-1} \tag{9.169}$$

$$\frac{\partial g(\mu_i)}{\partial \beta_j} = x_{ij} \; . \tag{9.170}$$

For finding the maximum, the derivative of the log-likelihood is set to zero

$$\frac{\partial \ln \mathcal{L}}{\partial \beta_j} = \sum_{i=1}^{n} \frac{(y_i - \mu_i)\, x_{ij}}{\mathrm{Var}(y_i)} \left(\frac{\partial g(\mu_i)}{\partial \mu_i}\right)^{-1} = 0 \; . \tag{9.171}$$

The maximum likelihood solution is obtained by solving this equation for the parameters $\boldsymbol{\beta}$.

Since $g(\mu_i) = \boldsymbol{x}_i^T \boldsymbol{\beta}$, $\mu_i = b'(\theta_i)$, and $\mathrm{Var}(y_i) = b''(\theta_i)\, a(\phi)$, this equation is nonlinear in $\boldsymbol{\beta}$ depending on the functions g and b. Therefore numerical methods are used to solve this equation. The probability function is determined by the functions a and b while the link function is given by g. A popular method to solve this equation is the iteratively re-weighted least squares algorithm. Using

$$w_i = \frac{\left(\frac{\partial \mu_i}{\partial g(\mu_i)}\right)^2}{\mathrm{Var}(y_i)} \tag{9.172}$$

and the diagonal matrix $\boldsymbol{W} = \mathrm{diag}(w_i)$ the iterative algorithm is

$$\left(\boldsymbol{X}^T \boldsymbol{W}^{(k)} \boldsymbol{X}\right) \boldsymbol{\beta}^{(k+1)} = \left(\boldsymbol{X}^T \boldsymbol{W}^{(k)} \boldsymbol{X}\right) \boldsymbol{\beta}^{(k)} + \frac{\partial \ln \mathcal{L}}{\partial \boldsymbol{\beta}^{(k)}} \; . \tag{9.173}$$

Here $\left(\boldsymbol{X}^T \boldsymbol{W}^{(k)} \boldsymbol{X}\right)$ approximates the Fisher information matrix $\mathcal{F}$:

$$\mathcal{F} \approx \boldsymbol{X}^T \boldsymbol{W}^{(k)} \boldsymbol{X} \; . \tag{9.174}$$

If $\boldsymbol{X}$ has full rank then the update rule becomes

$$\boldsymbol{\beta}^{(k+1)} = \boldsymbol{\beta}^{(k)} + \left(\boldsymbol{X}^T \boldsymbol{W}^{(k)} \boldsymbol{X}\right)^{-1} \frac{\partial \ln \mathcal{L}}{\partial \boldsymbol{\beta}^{(k)}} \; . \tag{9.175}$$

If different models are fitted, then the maximum likelihood solutions of different models can be compared by a likelihood ratio test. The likelihood ratio test is of interest when using reduced models to test which variables are relevant. Also the interaction of variables might be tested.

Tab. 9.9 shows commonly used generalized linear models described by their distribution and link function. The last three models are known as *logistic regression* and *multinomial logistic regression* for more than two classes.

distribution	link function	link name	support	application
normal	$\boldsymbol{X}\boldsymbol{\beta} = g(\mu) = \mu$	identity	real, $(-\infty, +\infty)$	linear response
exponential	$\boldsymbol{X}\boldsymbol{\beta} = g(\mu) = -\mu^{-1}$	inverse	real, $(0, +\infty)$	exponential response
Gamma	$\boldsymbol{X}\boldsymbol{\beta} = g(\mu) = -\mu^{-1}$	inverse	real, $(0, +\infty)$	exponential response
inv. Gaussian	$\boldsymbol{X}\boldsymbol{\beta} = g(\mu) = -\mu^{-2}$	inv. squared	real, $(0, +\infty)$	
Poisson	$\boldsymbol{X}\boldsymbol{\beta} = g(\mu) = \ln(\mu)$	log	integer, $[0, +\infty)$	count data
Bernoulli	$\boldsymbol{X}\boldsymbol{\beta} = g(\mu) = \ln\left(\frac{\mu}{1-\mu}\right)$	logit	integer, $[0, 1]$	two classes, occurrence
binomial	$\boldsymbol{X}\boldsymbol{\beta} = g(\mu) = \ln\left(\frac{\mu}{1-\mu}\right)$	logit	integer, $[0, n]$	two classes, count
categorical	$\boldsymbol{X}\boldsymbol{\beta} = g(\mu) = \ln\left(\frac{\mu}{1-\mu}\right)$	logit	integer, $[0, K]$	K classes, occurrence
multinomial	$\boldsymbol{X}\boldsymbol{\beta} = g(\mu) = \ln\left(\frac{\mu}{1-\mu}\right)$	logit	integer, $[0, n]^K$	K classes, count

Table 9.9: Commonly used generalized linear models described by their distribution and link function. The probability distribution and the link function are given. Further the support of the distribution, the short-cut name for the link, and the typical application.

Commonly used link functions are: “logit”, “probit”, “cauchit”, “cloglog”, “identity”, “log”, “sqrt”, “inverse squared”, and “inverse”.

The “cloglog” is the “complementary log log function” given as

$$g(x) \ = \ \log\left(- \ \log(x)\right) \ . \tag{9.176}$$

The “cloglog” model is similar to the logit models around 0.5 but differs near 0 or 1. Following models are common:

```
binomial:         link = logit
gaussian:         link = identity
Gamma:            link = inverse
inverse.gaussian: link = 1/mu^2
poisson:          link = og
quasi:            link = identity, variance = constant
quasibinomial:    link = logit
quasipoisson:     link = log
```

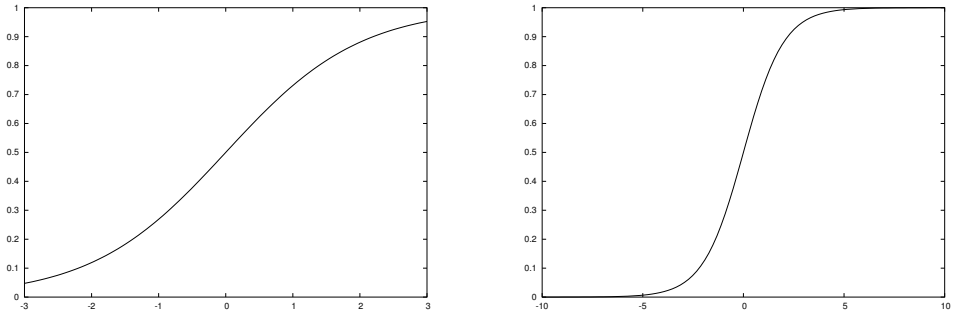

Figure 9.9: The sigmoid function $\frac{1}{1+\exp(-x)}$.

9.5.1 Logistic Regression

9.5.1.1 The Model

The inverse of the logit function

$$g(x) \;=\; \ln\left(\frac{x}{1\;-\;x}\right) \tag{9.177}$$

is the sigmoid function:

$$f(x) \;=\; \frac{1}{1\;+\;e^{-\,x}}\;, \tag{9.178}$$

which is depicted in Fig. 9.9.

Since

$$1\;-\;\frac{1}{1\;+\;e^{-\,x}} \;=\; \frac{e^{-\,x}}{1\;+\;e^{-\,x}}\;, \tag{9.179}$$

we obtain the probabilities

$$p(y\;=\;1\mid \boldsymbol{x};\boldsymbol{\beta}) \;=\; \frac{1}{1\;+\;e^{-\,\boldsymbol{x}^T\boldsymbol{\beta}}} \tag{9.180}$$

and

$$p(y\;=\;0\mid \boldsymbol{x};\boldsymbol{\beta}) \;=\; \frac{e^{-\,\boldsymbol{x}^T\boldsymbol{\beta}}}{1\;+\;e^{-\,\boldsymbol{x}^T\boldsymbol{\beta}}}\;. \tag{9.181}$$

The logit as link function gives

$$\boldsymbol{x}^T\boldsymbol{\beta} \;=\; \ln\left(\frac{p(y\;=\;1\mid \boldsymbol{x})}{1\;-\;p(y\;=\;1\mid \boldsymbol{x})}\right)\;. \tag{9.182}$$

9.5.1.2 (Regularized) Logistic Regression is Strictly Convex

Following Jason D. M. Rennie, we show that linear Logistic Regression is strictly convex.

For labels $y \in +1, -1$ we have

$$\frac{\partial \mathcal{L}}{\partial \beta_j} \;=\; -\sum_{i=1}^{n} y_i x_{ij} \;(1 \;-\; p(y_i \mid \boldsymbol{x}; \boldsymbol{\beta}))\;. \tag{9.183}$$

The second derivatives of the objective L that is minimized are

$$H_{jk} \;=\; \frac{\partial L}{\partial \beta_j \, \partial \beta_k} \;=\; \sum_{i=1}^{n} (y_i)^2 \; x_{ij} \, x_{ik} \, p(y_i \mid \boldsymbol{x}; \boldsymbol{\beta}) \;(1 \;-\; p(y_i \mid \boldsymbol{x}; \boldsymbol{\beta}))\;, \tag{9.184}$$

where $\boldsymbol{H}$ is the Hessian. The Hessian is the Fisher information matrix $\mathcal{F} = \left(\boldsymbol{X}^T \boldsymbol{W} \boldsymbol{X}\right)$, where $\boldsymbol{W}$ contains the variance of the binomial $p(1-p)$.

Since $p(1-p) \geq 0$ for $p \leq 1$, we can define

$$\rho_{ij} \;=\; x_{ij} \, \sqrt{p(y_i \mid \boldsymbol{x}; \boldsymbol{\beta}) \;(1 \;-\; p(y_i \mid \boldsymbol{x}; \boldsymbol{\beta}))}\,. \tag{9.185}$$

The bilinear form of the Hessian with a vector $\boldsymbol{a}$ is

$$\boldsymbol{a}^T \, \boldsymbol{H} \, \boldsymbol{a} \;=\; \sum_{i=1}^{n} \sum_{j=1}^{d} \sum_{k=1}^{d} x_{ij} \, x_{ik} \, a_j \, a_k \, p(y_i \mid \boldsymbol{x}; \boldsymbol{\beta}) \;(1 \;-\; p(y_i \mid \boldsymbol{x}; \boldsymbol{\beta})) \;= \tag{9.186}$$

$$\sum_{i=1}^{n} \sum_{j=1}^{d} a_j \, x_{ij} \, \sqrt{p(y_i \mid \boldsymbol{x}; \boldsymbol{\beta}) \;(1 \;-\; p(y_i \mid \boldsymbol{x}; \boldsymbol{\beta}))}$$

$$\sum_{k=1}^{d} a_k \;\; x_{ik} \, \sqrt{p(y_i \mid \boldsymbol{x}; \boldsymbol{\beta}) \;(1 \;-\; p(y_i \mid \boldsymbol{x}; \boldsymbol{\beta}))} \;=$$

$$\sum_{i=1}^{n} \left(\boldsymbol{a}^T \, \boldsymbol{\rho}_i\right) \left(\boldsymbol{a}^T \, \boldsymbol{\rho}_i\right) \;=\; \sum_{i=1}^{n} \left(\boldsymbol{a}^T \, \boldsymbol{\rho}_i\right)^2 \;\geq\; 0\,.$$

Since we did not impose any restriction on $\boldsymbol{a}$, the Hessian is positive definite. Adding a term like $\frac{1}{2} \, \boldsymbol{\beta}^T \, \boldsymbol{\beta}$ to the objective for regularization, then the Hessian of the objective is strictly positive definite.

9.5.1.3 Maximizing the Likelihood

We aim at maximizing the likelihood by gradient ascent. Therefore we have to compute the gradient, that is, the first order derivatives of the likelihood $\mathcal{L}$ with respect to the parameters β_j.

The log-likelihood for iid sampled data is

$$\ln \mathcal{L}(\{(y_i, \boldsymbol{x}_i)\}; \boldsymbol{\beta}) \;=\; \sum_{i=1}^{n} \ln p(y_i, \boldsymbol{x}_i; \boldsymbol{\beta}) \;= \tag{9.187}$$

$$\sum_{i=1}^{n} \ln p(y_i \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \;+\; \sum_{i=1}^{n} \ln p(\boldsymbol{x}_i)\,.$$

Only the first sum depends on the parameters, therefore maximum likelihood maximizes the sum of the conditional probabilities

$$\sum_{i=1}^{n} \ln p(y_i \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \,, \tag{9.188}$$

This term is often called the conditional likelihood.

Next we will consider the derivative of the log-likelihood. First we will need some algebraic properties:

$$\begin{aligned} &\frac{\partial}{\partial \beta_j} \ln p(y = 1 \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \;=\; \frac{\partial}{\partial \beta_j} \ln \frac{1}{1 \,+\, e^{-\,\boldsymbol{x}_i^T \boldsymbol{\beta}}} \;= \\ &\left(1 \,+\, e^{-\,\boldsymbol{x}_i^T \boldsymbol{\beta}}\right) \left(- \frac{e^{-\,\boldsymbol{x}_i^T \boldsymbol{\beta}}}{\left(1 \,+\, e^{-\,\boldsymbol{x}_i^T \boldsymbol{\beta}}\right)^2} \right) \frac{\partial\, \boldsymbol{x}_i^T \boldsymbol{\beta}}{\partial \beta_j} \;= \\ &-\; \frac{e^{-\,\boldsymbol{x}_i^T \boldsymbol{\beta}}}{1 \,+\, e^{-\,\boldsymbol{x}_i^T \boldsymbol{\beta}}} \frac{\partial\, \boldsymbol{x}_i^T \boldsymbol{\beta}}{\partial \beta_j} \;=\; - \, p(y \,=\, 0 \mid \boldsymbol{x}_i; \boldsymbol{\beta})\, x_{ij} \end{aligned} \tag{9.189}$$

and

$$\begin{aligned} &\frac{\partial}{\partial \beta_j} \ln p(y = 0 \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \;=\; \frac{\partial}{\partial \beta_j} \ln \frac{e^{-\,\boldsymbol{x}_i^T \boldsymbol{\beta}}}{1 \,+\, e^{-\,\boldsymbol{x}_i^T \boldsymbol{\beta}}} \;= \\ &\frac{1 \,+\, e^{-\,\boldsymbol{x}_i^T \boldsymbol{\beta}}}{e^{-\,\boldsymbol{x}_i^T \boldsymbol{\beta}}} \left(\frac{e^{-\,\boldsymbol{x}_i^T \boldsymbol{\beta}}}{1 \,+\, e^{-\,\boldsymbol{x}_i^T \boldsymbol{\beta}}} \;-\; \frac{e^{-\,2\,\boldsymbol{x}_i^T \boldsymbol{\beta}}}{\left(1 \,+\, e^{-\,\boldsymbol{x}_i^T \boldsymbol{\beta}}\right)^2} \right) \frac{\partial\, \boldsymbol{x}_i^T \boldsymbol{\beta}}{\partial \beta_j} \;= \\ &\frac{1}{1 \,+\, e^{-\,\boldsymbol{x}_i^T \boldsymbol{\beta}}}\, x_{ij} \;=\; p(y \,=\, 1 \mid \boldsymbol{x}_i; \boldsymbol{\beta})\, x_{ij} \,. \end{aligned} \tag{9.190}$$

We can rewrite the likelihood as

$$\begin{aligned} &\sum_{i=1}^{n} \ln p(y_i \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \;= \\ &\sum_{i=1}^{n} y_i \ln p(y = 1 \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \;+\; \sum_{i=1}^{n} (1 \;-\; y_i) \;\ln p(y = 0 \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \end{aligned} \tag{9.191}$$

which gives for the derivative

$$\begin{aligned}
&\frac{\partial}{\partial \beta_j} \sum_{i=1}^{n} \ln p(y_i \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \; = \\
&\sum_{i=1}^{n} y_i \frac{\partial}{\partial \beta_j} \ln p(y = 1 \mid \boldsymbol{x}_i; \boldsymbol{\beta}) + \\
&\sum_{i=1}^{n} (1 \; - \; y_i) \; \frac{\partial}{\partial \beta_j} \ln p(y = 0 \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \; = \\
&\sum_{i=1}^{n} - \, y_i \, p(y \; = \; 0 \mid \boldsymbol{x}_i; \boldsymbol{\beta}) x_{ij} + \\
&\sum_{i=1}^{n} (1 \; - \; y_i) \, p(y \; = \; 1 \mid \boldsymbol{x}_i; \boldsymbol{\beta}) x_{ij} \; = \\
&\sum_{i=1}^{n} (- \, y_i \; (1 \; - \; p(y \; = \; 1 \mid \boldsymbol{x}_i; \boldsymbol{\beta})) \\
&(1 \; - \; y_i) \, p(y \; = \; 1 \mid \boldsymbol{x}_i; \boldsymbol{\beta})) \; x_{ij} \; = \\
&\sum_{i=1}^{n} (p(y \; = \; 1 \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \; - \; y_i) \; x_{ij} \; ,
\end{aligned} \tag{9.192}$$

where

$$p(y \; = \; 1 \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \; = \; \frac{1}{1 \; + \; e^{- \, \boldsymbol{x}_i^T \boldsymbol{\beta}}} \tag{9.193}$$

For computing the maximum, the derivatives have to be zero

$$\forall_j : \; \sum_{i=1}^{n} (p(y \; = \; 1 \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \; - \; y_i) \, x_{ij} \; = \; 0 \; . \tag{9.194}$$

A gradient ascent based method may be used to find the solutions to this equation.

Alternative formulation with $y \in +1, -1$

We now give an alternative formulation of logistic regression with $y \in +1, -1$. We remember

$$p(y \; = \; 1 \mid \boldsymbol{x}; \boldsymbol{\beta}) \; = \; \frac{1}{1 \; + \; e^{- \, \boldsymbol{x}^T \boldsymbol{\beta}}} \tag{9.195}$$

and

$$p(y \; = \; -1 \mid \boldsymbol{x}; \boldsymbol{\beta}) \; = \; \frac{e^{- \, \boldsymbol{x}^T \boldsymbol{\beta}}}{1 \; + \; e^{- \, \boldsymbol{x}^T \boldsymbol{\beta}}} \; = \; \frac{1}{1 \; + \; e^{\, \boldsymbol{x}^T \boldsymbol{\beta}}} \; . \tag{9.196}$$

Therefore we have

$$- \ln p(y \; = \; y_i \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \; = \; \ln \left(1 \; + \; e^{- \, y_i \, \boldsymbol{x}_i^T \boldsymbol{\beta}} \right) \tag{9.197}$$

and the objective which is minimized to find the maximum likelihood solution is

$$\mathcal{L} \;=\; -\sum_{i=1}^{n} \ln p(y_i \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \;=\; \sum_{i=1}^{n} \ln \left(1 \;+\; e^{-\,y_i\, \boldsymbol{x}_i^T \boldsymbol{\beta}}\right) \tag{9.198}$$

The derivatives of the objective with respect to the parameters are

$$\begin{aligned} \frac{\partial L}{\partial \beta_j} \;&=\; -\sum_{i=1}^{n} y_i \, \frac{\partial\, \boldsymbol{x}_i^T \boldsymbol{\beta}}{\partial \beta_j} \, \frac{e^{-\,y_i\, \boldsymbol{x}_i^T \boldsymbol{\beta}}}{1 \;+\; e^{-\,y_i\, \boldsymbol{x}_i^T \boldsymbol{\beta}}} \;= \\ &-\sum_{i=1}^{n} y_i \, x_{ij} \;(1 \;-\; p(y_i \mid \boldsymbol{x}; \boldsymbol{\beta}))\;. \end{aligned} \tag{9.199}$$

The last equation is similar to Eq. (9.192). In matrix notation we have for the gradient:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\beta}} \;=\; -\sum_{i=1}^{n} y_i \,(1 \;-\; p(y_i \mid \boldsymbol{x}; \boldsymbol{\beta}))\; \boldsymbol{x}_i\;. \tag{9.200}$$

We showed that the objective, the log likelihood, of logistic regression is strictly convex. Therefore efficient gradient-based techniques to find the maximum likelihood solution can be used.

9.5.2 Multinomial Logistic Regression: Softmax

9.5.2.1 The Method

For multi-class problems logistic regression can be generalized to Softmax. We assume K classes with $y \in \{1, \ldots, K\}$ and the probability of $\boldsymbol{x}$ belonging to class k is

$$p(y \;=\; k \mid \boldsymbol{x}; \boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_K) \;=\; \frac{e^{\boldsymbol{x}^T \boldsymbol{\beta}_k}}{\sum_{j=1}^{K} e^{\boldsymbol{x}^T \boldsymbol{\beta}_j}} \tag{9.201}$$

which gives a multinomial distribution across the classes.

The objective, which is minimized in order to maximize the likelihood, is

$$L \;=\; -\sum_{i=1}^{n} \ln p(y = y_i \mid \boldsymbol{x}_i; \boldsymbol{\beta}) \;=\; \sum_{i=1}^{n} \ln \left(\sum_{j=1}^{K} e^{\boldsymbol{x}^T \boldsymbol{\beta}_j} \right) \;-\; \boldsymbol{x}^T \boldsymbol{\beta}_{y_i}\;. \tag{9.202}$$

In the following we set

$$p(y \;=\; k \mid \boldsymbol{x}; \boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_K) \;=\; p(k \mid \boldsymbol{x}; \boldsymbol{W})\;, \tag{9.203}$$

where $\boldsymbol{W} = (\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_K)$ is the matrix of parameters.

The derivatives are

$$\frac{\partial \mathcal{L}}{\partial \beta_{kt}} \;=\; \sum_{i=1}^{n} \frac{\partial \boldsymbol{x}_i^T \boldsymbol{\beta}_k}{\partial \beta_{kt}} p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \;-\; \delta_{y_i = k} \sum_{i=1}^{n} \frac{\partial \boldsymbol{x}_i^T \boldsymbol{\beta}_k}{\partial \beta_{kt}} \tag{9.204}$$

$$=\; \sum_{i=1}^{n} x_{it} p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \;-\; \delta_{y_i = k} \sum_{i=1}^{n} x_{it}\;. \tag{9.205}$$

Next we show that the objective of Softmax is strictly convex.

9.5.2.2 (Regularized) Softmax is Strictly Convex

Following Jason D. M. Rennie, we show that linear Softmax is strictly convex.

The derivatives are

$$\frac{\partial \mathcal{L}}{\partial \beta_{kt}} \;=\; \sum_{i=1}^{n} x_{it} p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \;-\; \delta_{y_i=k} \sum_{i=1}^{n} x_{it} \;. \tag{9.206}$$

To compute the second derivatives of the objective, we need the derivatives of the probabilities with respect to the parameters:

$$\begin{aligned} \frac{\partial p(v \mid \boldsymbol{x}_i; \boldsymbol{W})}{\partial \beta_{vm}} \;&=\; x_{im}\, p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \;(1 \;-\; p(k \mid \boldsymbol{x}_i; \boldsymbol{W})) \\ \frac{\partial p(k \mid \boldsymbol{x}_i; \boldsymbol{W})}{\partial \beta_{vm}} \;&=\; x_{im}\, p(k \mid \boldsymbol{x}_i; \boldsymbol{W})\, p(v \mid \boldsymbol{x}_i; \boldsymbol{W})) \;. \end{aligned} \tag{9.207}$$

The second derivatives of $\mathcal{L}$ with respect to the components of the parameter vector $\boldsymbol{\beta}$ are

$$\begin{aligned} H_{kt,vm} \;&=\; \frac{\partial \mathcal{L}}{\partial \beta_{kt} \partial \beta_{vm}} \;= \\ &\sum_{i=1}^{n} x_{it}\, x_{im}\, p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \;(\delta_{k=v}\,(1 \;-\; p(k \mid \boldsymbol{x}_i; \boldsymbol{W})) \;- \\ &(1 \;-\; \delta_{k=v}) p(v \mid \boldsymbol{x}_i; \boldsymbol{W}) \;) \;. \end{aligned} \tag{9.208}$$

Again we define a vector $\boldsymbol{a}$ with components a_{uj} (note, the double index is considered as single index so that a matrix is written as vector).

We consider the bilinear from

$$
\begin{aligned}
&\boldsymbol{a}^T \boldsymbol{H}\, \boldsymbol{a} \;= \qquad (9.209)\\
&\sum_{k,t}\sum_{v,m}\sum_{i} a_{kt}\, a_{vm}\, x_{it}\, x_{im}\, p(k \mid \boldsymbol{x}_i; \boldsymbol{W})\; (\delta_{k=v}\,(1 \;-\; p(k \mid \boldsymbol{x}_i; \boldsymbol{W})) \;-\\
&(1 \;-\; \delta_{k=v})\, p(v \mid \boldsymbol{x}_i; \boldsymbol{W})) \;=\\
&\sum_{k,t}\sum_{i} a_{kt}\, x_{it}\, p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \sum_{m} x_{im} \left(a_{km} \;-\; \sum_{v} a_{vm} p(v \mid \boldsymbol{x}_i; \boldsymbol{W}) \right) \;=\\
&\sum_{i}\sum_{t} x_{it} \sum_{k} a_{kt}\, p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \sum_{m} x_{im} \left(a_{km} \;-\; \sum_{v} a_{vm} p(v \mid \boldsymbol{x}_i; \boldsymbol{W}) \right) \;=\\
&\sum_{i} - \left\{ \left(\sum_{t} x_{it} \sum_{k} a_{kt}\, p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \right) \left(\sum_{m} x_{im} \sum_{v} a_{vm} p(v \mid \boldsymbol{x}_i; \boldsymbol{W}) \right) \right\} +\\
&\left\{ \sum_{t} x_{it} \sum_{k} a_{kt}\, p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \sum_{m} x_{im}\, a_{km} \right\} \;=\\
&\sum_{i} - \left\{ \left(\sum_{t} x_{it} \sum_{k} a_{kt}\, p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \right)^2 \right\} +\\
&\left\{ \sum_{k} p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \left(\sum_{t} x_{it}\, a_{kt} \right) \left(\sum_{m} x_{im}\, a_{km} \right) \right\} \;=\\
&\sum_{i} - \left\{ \left(\sum_{t} x_{it} \sum_{k} a_{kt}\, p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \right)^2 \right\} +\\
&\left\{ \sum_{k} p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \left(\sum_{t} x_{it}\, a_{kt} \right)^2 \right\} .
\end{aligned}
$$

If for each summand of the sum over i

$$
\begin{aligned}
&\sum_{k} p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \left(\sum_{t} x_{it}\, a_{kt} \right)^2 \;-\; \left(\sum_{k} \; p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \sum_{t} x_{it}\, a_{kt} \right)^2 \qquad (9.210)\\
&\geq\; 0
\end{aligned}
$$

holds, then the Hessian $\boldsymbol{H}$ is positive semidefinite. This holds for arbitrary numbers of samples as each term corresponds to a sample.

In the last equation the $p(k \mid \boldsymbol{x}_i; \boldsymbol{W})$ can be viewed as a multinomial distribution over k. The terms $\sum_t x_{it}\, a_{kt}$ can be viewed as functions depending on k.

In this case $\sum_k p(k \mid \boldsymbol{x}_i; \boldsymbol{W})\, (\sum_t x_{it}\, a_{kt})^2$ is the second moment and the squared expectation is $(\sum_k \; p(k \mid \boldsymbol{x}_i; \boldsymbol{W}) \sum_t x_{it}\, a_{kt})^2$. Therefore the left hand side of inequality Eq. (9.210) is the second central moment, which is larger than zero.

Alternatively inequality Eq. (9.210) can be proved by applying Jensen's inequality with the square function as a convex function.

distribution	parameters	pmf $\Pr(X = k)$	μ	Var	$r = \mu/\text{Var}$	r
binomial	$n \in \mathbb{N}, p$	$\binom{n}{k} p^k (1-p)^{n-k}$	np	$np(1-p)$	$1/(1-p)$	> 1
Poisson	$0 < \lambda$	$\frac{\lambda^k e^{-\lambda}}{k!}$	λ	λ	1	$= 1$
negative binomial	$0 < r, p$	$\binom{k+r-1}{k} (1-p)^r p^k$	$\frac{pr}{1-p}$	$\frac{pr}{(1-p)^2}$	$(1-p)$	< 1

Table 9.10: Commonly used distributions to model count data. The parameter $p \in [0, 1]$ is the probability of a success. The probability mass function ("pmf"), the mean μ, the variance Var, and the ratio $r = \frac{\mu}{\text{Var}}$ of mean to variance are given. The last column indicates whether r is larger or smaller than 1.

We have proved that the Hessian $\boldsymbol{H}$ is positive semidefinite.

Adding a term like $\frac{1}{2} \sum_k \boldsymbol{\beta}_k^T \boldsymbol{\beta}_k$ to the objective for regularization, then the Hessian of the objective is strictly positive definite.

9.5.3 Poisson Regression

To model count data, three distributions are popular: the binomial (variance smaller than the mean), Poisson (variance equal to the mean), negative binomial (variance larger than the mean). Tab. 9.10 shows these distributions.

In many cases the observations can be described by a rate θ and the number of trials n: $\lambda = \theta n$. An observation is the number of successes or failures out of n trials or exposures. Depending on the kind of applications and the problem which should be modeled, either the rate θ changes or the number of exposures changes. For example, n may be the number of kilometers which an individual drives with a car, while θ is the probability of having an accident. In this case, different individuals drove a different number of kilometers, that is, the exposure changes. For another task, all persons drive on a test track 100 km, however, different persons consumed a different amount of alcohol. Therefore, θ, the probability of having an accident, is different for each individual. Consequently, either θ or n can be modeled by a linear regression.

Poisson regression models the case were the rate changes and can be estimated by a linear model using the explanatory variables. We have

$$\mathrm{E}(y_i) \;=\; \lambda_i \;=\; n_i\, \theta_i \;=\; n_i\, e^{\boldsymbol{x}_i^T \boldsymbol{\beta}} \tag{9.211}$$

$$\log \lambda_i \;=\; \log n_i \;+\; \boldsymbol{x}_i^T \boldsymbol{\beta}\,. \tag{9.212}$$

The term $\log n_i$ is an additional offset.

Hypotheses tests can be based on the Wald statistics or on a likelihood ratio statistic. Reduced models allow to test the relevance of different variables for explaining the response variable. Also the combination and interactions of variables can be tested.

The standard error is

$$\mathrm{SE}(\hat{\boldsymbol{\beta}}) \;=\; \sqrt{\frac{1}{\mathcal{F}}}\,, \tag{9.213}$$

where $\mathcal{F}$ is the Fisher information matrix. Confidence intervals can be estimated using

$$\frac{\hat{\beta}_j \; - \; \beta_j}{\mathrm{SE}(\hat{\beta}_j)} \; \sim \; \mathcal{N}(0,1) \; . \tag{9.214}$$

The estimated values are

$$e_i \; = \; n_i \, e^{\boldsymbol{x}_i^T \hat{\boldsymbol{\beta}}} \tag{9.215}$$

giving the estimated standard deviation $\sqrt{e_i}$. The variance is equal to the mean for a Poisson. The Pearson residuals are

$$r_i \; = \; \frac{o_i \; - \; e_i}{\sqrt{e_i}} \; , \tag{9.216}$$

where o_i are the observed counts. These residuals can be standardized by

$$r_{pi} \; = \; \frac{o_i \; - \; e_i}{\sqrt{e_i} \, \sqrt{1 - P_{ii}}} \; , \tag{9.217}$$

where P_{ii} is the leverage which is the i-th element of the main diagonal of the hat matrix $\boldsymbol{P}$.

The goodness of fit, that is, the error or the objective is chi-squared distributed because

$$\sum_i r_i^2 \; = \; \sum_i \frac{(o_i \; - \; e_i)^2}{e_i} \; , \tag{9.218}$$

which is the definition of a chi-squared statistic.

The Poisson regression is an example of *log-linear models*:

$$\log \mathrm{E}(y_i) = \; c \; + \; \boldsymbol{x}_i^T \boldsymbol{\beta} \; . \tag{9.219}$$

This includes models like

$$\log \mathrm{E}(y_{jk}) = \; \log n \; + \; \log \theta_{j.} \; + \; \log \theta_{.k} \tag{9.220}$$

or

$$\log \mathrm{E}(y_{jk}) = \; \log n \; + \; \log \theta_{jk.} \tag{9.221}$$

which is similar to

$$\log \mathrm{E}(y_{jk}) = \; \mu \; + \; \alpha_j \; + \; \beta_k \; + \; (\alpha\beta)_{jk} \; . \tag{9.222}$$

These models show that ANOVA like approaches are possible in the context of generalized linear models.

9.5.4 Examples

9.5.4.1 Birthweight Data: Normal

We revisit Dobson's birthweight data set from Section 9.3.2.2.

The first model `l0` in Section 9.3.2.2 was a linear model estimated by least squares. This model is a generalized linear model with Gaussian error, therefore `l0` can also be produced by a glm:

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-257.49  -125.28   -58.44   169.00   303.98

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -1610.28     786.08  -2.049   0.0532 .
sexFemale    -163.04      72.81  -2.239   0.0361 *
age           120.89      20.46   5.908 7.28e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 31370.04)

    Null deviance: 1829873  on 23  degrees of freedom
Residual deviance:  658771  on 21  degrees of freedom
AIC: 321.39

Number of Fisher Scoring iterations: 2
```

The model without intercept is:

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-257.49  -125.28   -58.44   169.00   303.98

Coefficients:
          Estimate Std. Error t value Pr(>|t|)
sexMale   -1610.28     786.08  -2.049   0.0532 .
sexFemale -1773.32     794.59  -2.232   0.0367 *
age         120.89      20.46   5.908 7.28e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 31370.04)

    Null deviance: 213198964  on 24  degrees of freedom
```

```
Residual deviance:    658771  on 21  degrees of freedom
AIC: 321.39

Number of Fisher Scoring iterations: 2
```

We compare these models by an ANOVA table:

```
Model 1: birthw ~ sex + age
Model 2: birthw ~ sex + age - 1
  Resid. Df Resid. Dev Df    Deviance
1        21     658771
2        21     658771  0 -1.1642e-10
```

The scatter plot in Fig. 9.8 shows that the observation (35,2925) of a male baby looks like an outlier. If we check the residuals, we see

```
          1           2           3           4           5           6
-257.490545 -188.701891  -62.490545  303.981090 -116.913237  -15.807564
          7           8           9          10          11          12
 -54.384872  247.509455 -234.807564  192.298109  195.509455   -8.701891
         13          14          15          16          17          18
 254.548758  150.126066 -127.451242  -66.662588  -94.239896 -124.556915
         19          20          21          22          23          24
  63.548758 -160.768261 -166.873934  170.337412  -66.556915  168.548758
```

Indeed the fourth observation has the largest residual. We now investigate a subset of the data by removing the observation no. 4. We remove oberservation no. 4:

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-253.86  -129.46   -53.46   165.04   251.14

Coefficients:
          Estimate Std. Error t value Pr(>|t|)
sexMale   -2318.03     801.57  -2.892  0.00902 **
sexFemale -2455.44     803.79  -3.055  0.00625 **
age         138.50      20.71   6.688 1.65e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 26925.39)

    Null deviance: 204643339  on 23  degrees of freedom
Residual deviance:    538508  on 20  degrees of freedom
AIC: 304.68

Number of Fisher Scoring iterations: 2
```

Now all regressors are more significant.

Next we add an interaction term:

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-246.69  -138.11   -39.13   176.57   274.28

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
sexMale       -1268.67    1114.64  -1.138 0.268492
sexFemale     -2141.67    1163.60  -1.841 0.080574 .
age             111.98      29.05   3.855 0.000986 ***
sexFemale:age    18.42      41.76   0.441 0.663893
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 32621.23)

    Null deviance: 213198964  on 24  degrees of freedom
Residual deviance:    652425  on 20  degrees of freedom
AIC: 323.16

Number of Fisher Scoring iterations: 2
```

These results are already known from Section 9.3.2.2: the interaction does not help. The ANOVA table tells the same story:

```
Analysis of Deviance Table

Model 1: birthw ~ sex + age - 1
Model 2: birthw ~ sex + age + sex:age - 1
  Resid. Df Resid. Dev Df Deviance
1        21     658771
2        20     652425  1   6346.2
```

9.5.4.2 Beetle Mortality: Logistic Regression

An example for logistic regression is found in Dobson [2002], page 124, data of Table 7.2. The numbers of dead beetles are counted after five hours exposure to gaseous carbon disulfide at various concentrations. The data stems from Bliss (1935). The data are shown in Tab. 9.11 and as a scatter plot in Fig. 9.10. The dose is actually the logarithm of the quantity of carbon disulfide. For the scatter plot the response was the percentage of dead beetles from all beetles.

The data are binomial because from all beetles a certain number is dead. We produce count data as pairs of (dead,alive). We start with logistic regression, that is the distribution is binomial and the link function is logit:

Dose ($\log_{10}$ $CS_2 mgl^{-1}$)	Number of beetles	Number killed
1.6907	59	6
1.7242	60	13
1.7552	62	18
1.7842	56	28
1.8113	63	52
1.8369	59	53
1.8610	62	61
1.8839	60	60

Table 9.11:

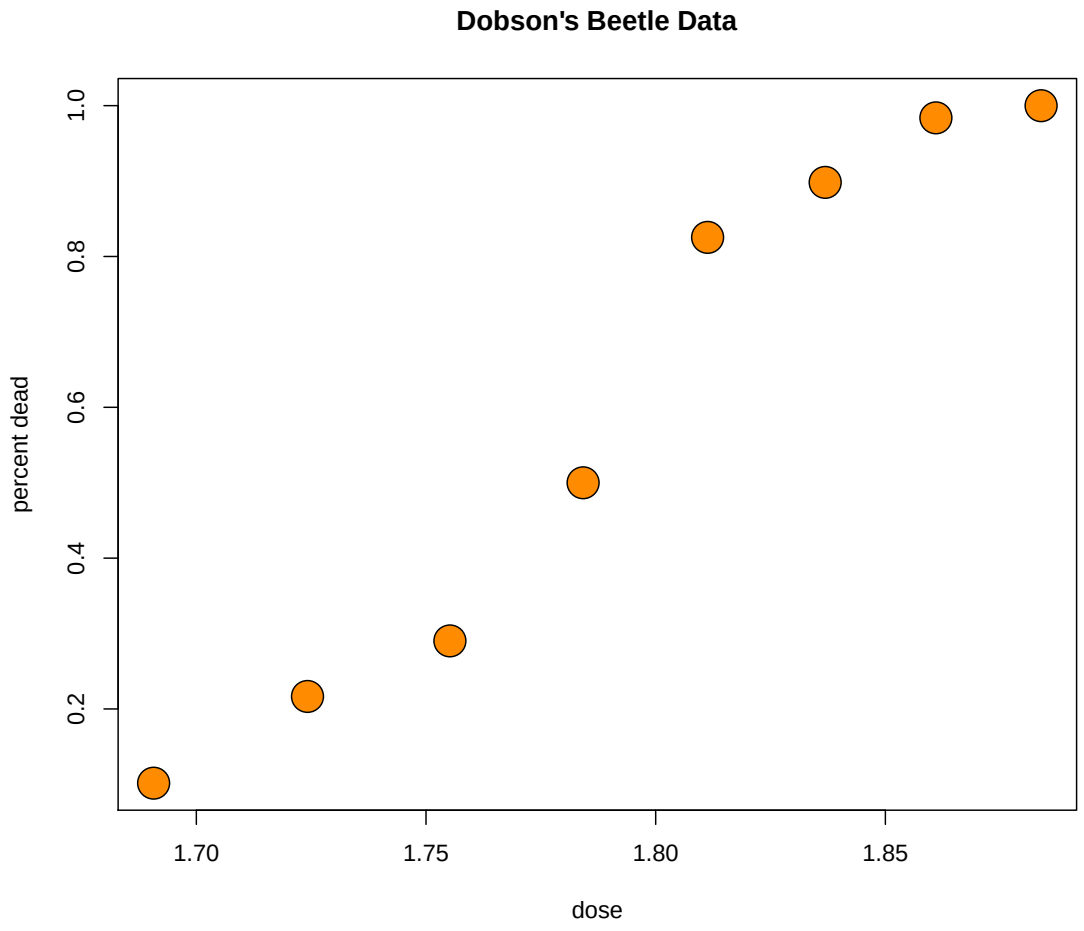

Figure 9.10: Scatter plot of Dobson's beetle data for logistic regression.

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5941  -0.3944   0.8329   1.2592   1.5940

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -60.717      5.181  -11.72   <2e-16 ***
dose          34.270      2.912   11.77   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 284.202  on 7  degrees of freedom
Residual deviance:  11.232  on 6  degrees of freedom
AIC: 41.43

Number of Fisher Scoring iterations: 4
```

Both intercept and dose are significant. The mean is not around zero, therefore the intercept has to move it. The significance of the dose shows that the number of dead beetles indeed depends on the dose of carbon disulfide.

The next link function, that we try, is the probit.

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5714  -0.4703   0.7501   1.0632   1.3449

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -34.935      2.648  -13.19   <2e-16 ***
dose          19.728      1.487   13.27   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 284.20  on 7  degrees of freedom
Residual deviance:  10.12  on 6  degrees of freedom
AIC: 40.318

Number of Fisher Scoring iterations: 4
```

The result is very similar to the logit link function.

We now test the cloglog link function:

```
Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.80329  -0.55135   0.03089   0.38315   1.28883

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -39.572      3.240  -12.21   <2e-16 ***
dose          22.041      1.799   12.25   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 284.2024  on 7  degrees of freedom
Residual deviance:   3.4464  on 6  degrees of freedom
AIC: 33.644

Number of Fisher Scoring iterations: 4
```

For this cloglog link function the residual deviance is 3.4464 while it was 11.232 and 10.12 for the logit and probit link function, respectively. Also the AIC (Akaike information criterion) of the last model is lower. This hints at the fact that the last model fits the data better. The fitting of the different link functions is shown in Fig. 9.11, where it is clear that the cloglog link function fits the data best.

9.5.4.3 Embryogenic Anthers: Logistic Regression

Another example for logistic regression is found in Dobson [2002], page 128, data of Table 7.5. The data are taken from Sangwan-Norrell (1977) and are shown in Tab. 9.12. The authors counted the embryogenic anthers of the plant species *Datura innoxia* Mill. obtained from a particular number of anthers prepared. The embryogenic anthers were obtained under different conditions. The first factor has two levels which relate to the storage type, which is either a control storage or a storage at 3 °C for 48 hours. The second factor has three levels corresponding to the centrifuging forces. The data is shown in Fig. 9.12. The task is to compare the treatment and the control storage type after adjusting for the centrifuging force.

f gives the centrifuging force and g the storage type. We first fit a full model:

```
Deviance Residuals:
       1         2         3         4         5         6
 0.08269  -0.12998   0.04414   0.42320  -0.60082   0.19522

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.1456719  0.1975451   0.737   0.4609
g2           0.7963143  0.3125046   2.548   0.0108 *
f           -0.0001227  0.0008782  -0.140   0.8889
```

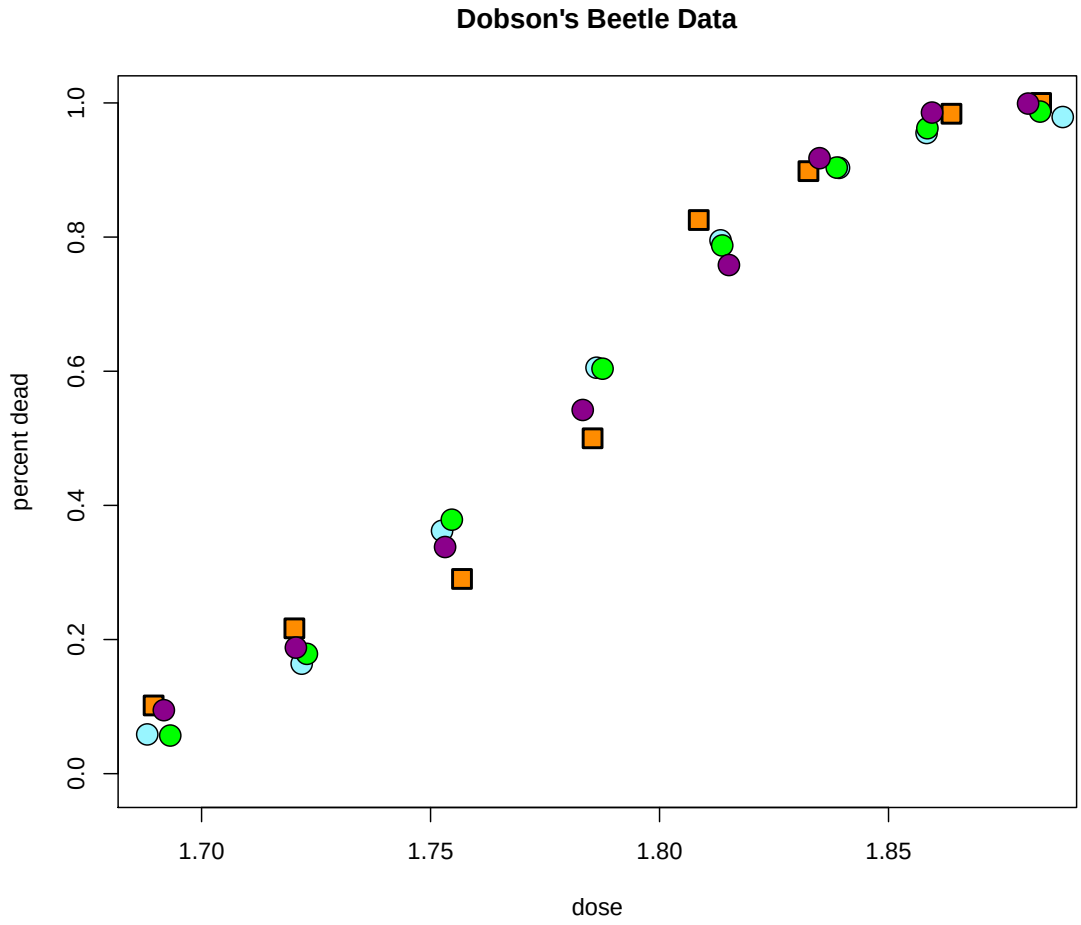

Figure 9.11: Fitting of Dobson's beetle data with different link functions. Orange rectangles are the original data, blue circles are the fitted points with logistic link function, green circles are the fitted points with the probit link function, and the magenta circles are the fitted points with the cloglog link function. The x-axis values are jittered. The cloglog link function fits the points best.

		Centrifuging force (g)		
Storage condition		40	150	350
Control	y	55	52	57
	n	102	99	108
Treatment	y	55	50	50
	n	76	81	90

Table 9.12: Dobson's embryogenic anther data taken from Sangwan-Norrell (1977).

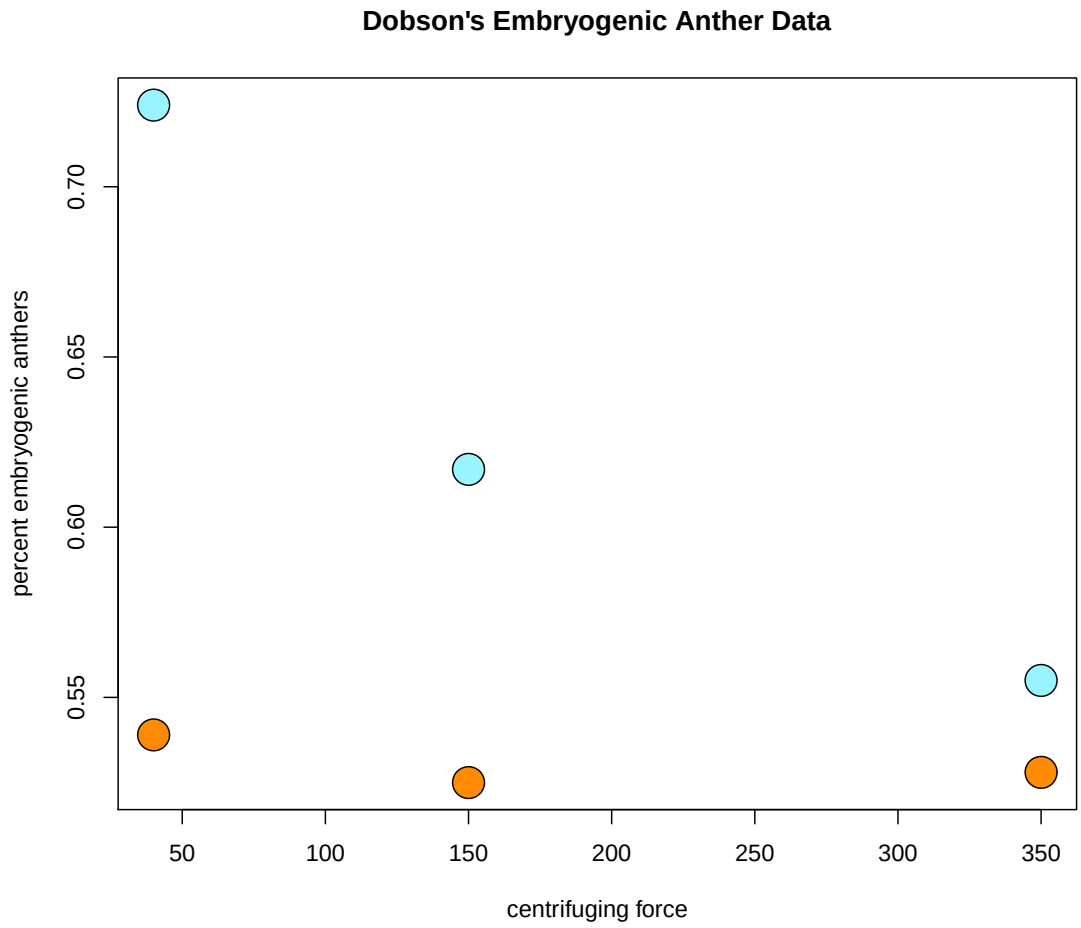

Figure 9.12: Dobson's embryogenic anther data taken from Sangwan-Norrell (1977). The color mark the groups, which are the storage types.

```
g2:f        -0.0020493  0.0013483  -1.520   0.1285
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 10.45197  on 5  degrees of freedom
Residual deviance:  0.60387  on 2  degrees of freedom
AIC: 38.172

Number of Fisher Scoring iterations: 3
```

Next we do not consider the interaction effect between centrifuging force and storage type:

```
Deviance Residuals:
      1        2        3        4        5        6
-0.5507  -0.2781   0.7973   1.1558  -0.3688  -0.6584

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.306643   0.167629   1.829   0.0674 .
g2           0.405554   0.174560   2.323   0.0202 *
f           -0.000997   0.000665  -1.499   0.1338
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 10.4520  on 5  degrees of freedom
Residual deviance:  2.9218  on 3  degrees of freedom
AIC: 38.49

Number of Fisher Scoring iterations: 3
```

The centrifuging force seems not to be relevant for explaining the yield in embryogenic anthers. Therefore we only consider the group effects, that is the different storage conditions:

```
Deviance Residuals:
       1         2         3         4         5         6
 0.17150  -0.10947  -0.06177   1.77208  -0.19040  -1.39686

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.1231     0.1140   1.080   0.2801
g2            0.3985     0.1741   2.289   0.0221 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

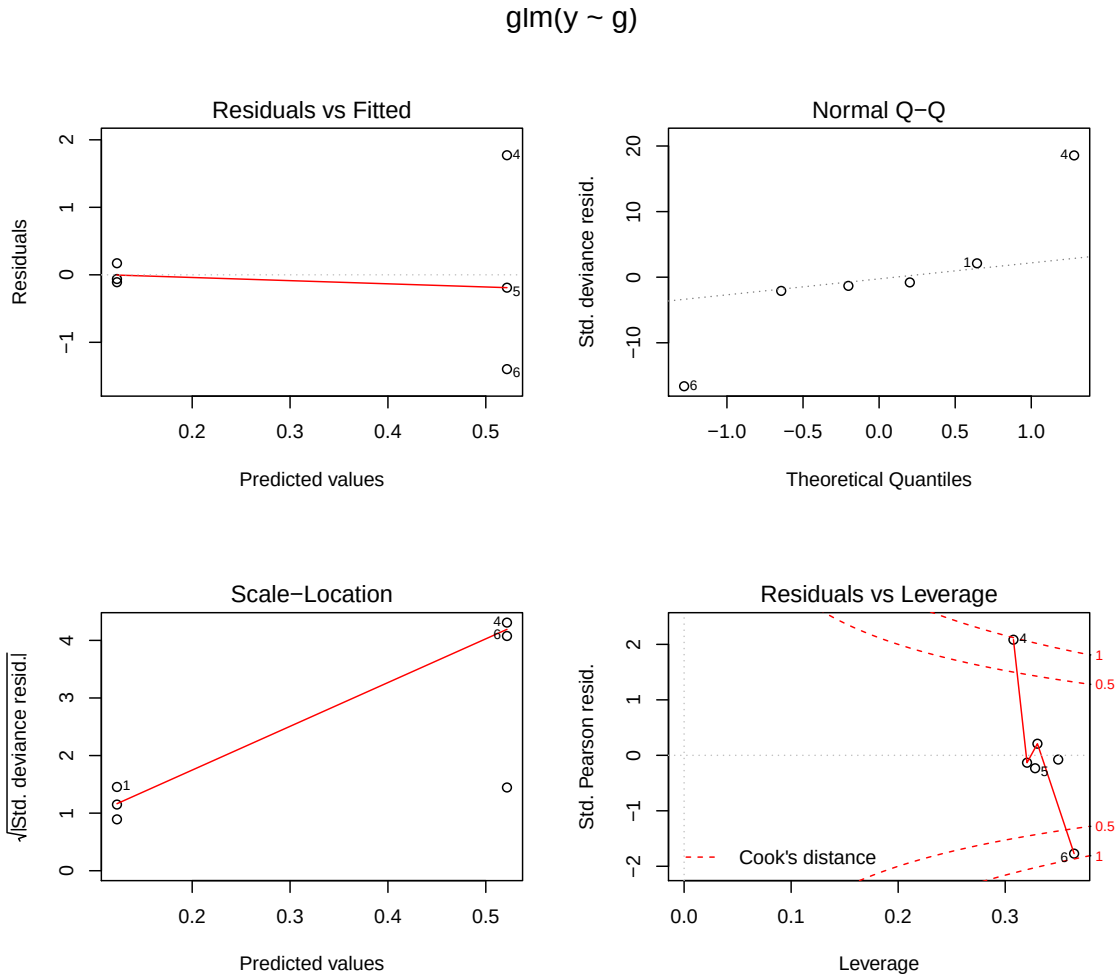

Figure 9.13: The best best model for Dobson's embryogenic anther data with respect to the AIC considers only the groups. The groups are the storage type.

```
(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 10.452  on 5  degrees of freedom
Residual deviance:  5.173  on 4  degrees of freedom
AIC: 38.741

Number of Fisher Scoring iterations: 3
```

This best model with respect to the AIC, which only considers the groups, is analyzed in Fig. 9.13.

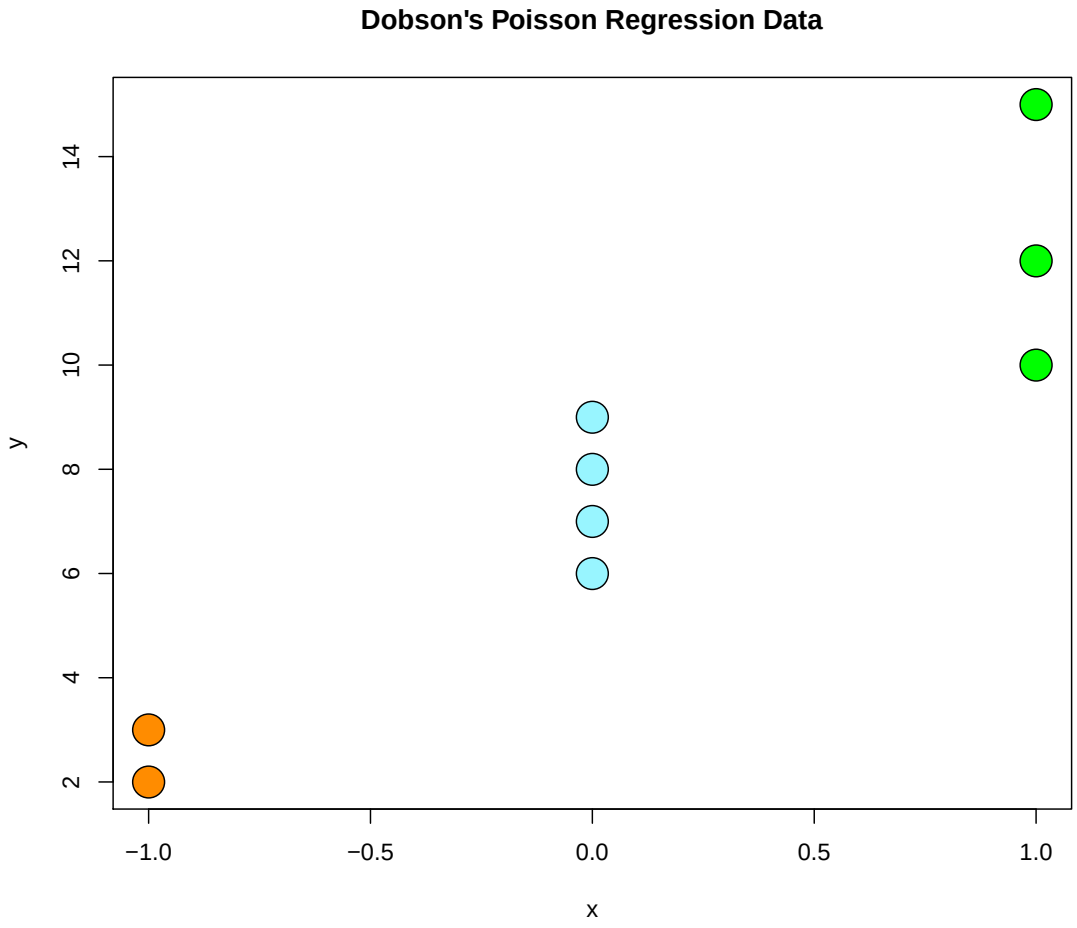

Figure 9.14: Scatter plot of Dobson's toy data for Poisson regression.

9.5.4.4 Toy Example 1: Poisson Regression

For Poisson regression we present a toy example from Dobson [2002], page 71, data of Table 4.3. The data are shown in Fig. 9.14. There is a clear relation between x and the count data y as counts for $x = 1.0$ are larger than counts for $x = 0.0$ which in turn are larger than counts for $x = -1.0$.

On the data

```
x <- c(-1,-1,0,0,0,0,1,1,1)
y <- c(2,3,6,7,8,9,10,12,15)
```

we performe Poisson regression:

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.7019  -0.3377  -0.1105   0.2958   0.7184
```

	Outcome			
Treatment	O_1	O_2	O_3 Total	
T_1	18	17	15	50
T_2	20	10	20	50
T_3	25	13	12	50
Total	63	40	47	

Table 9.13: Toy data from Dobson [1990] for randomized controlled trial analyzed by Poisson regression. Outcomes are indicated by the border color of the circles (O_1=blue,O_2=red,O_3=magenta). Treatments are indicated by the interior color of the circles (T_1=orange,T_3=blue,T_3=green).

```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   7.4516     0.8841   8.428  < 2e-16 ***
x             4.9353     1.0892   4.531 5.86e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 18.4206  on 8  degrees of freedom
Residual deviance:  1.8947  on 7  degrees of freedom
AIC: 40.008

Number of Fisher Scoring iterations: 3
```

Both the intercept and the coefficient are significant. The intercept must move x into the range of the count data.

9.5.4.5 Toy Example 2: Poisson Regression

This is another example for Poisson regression from Dobson [1990] (the first edition), page 93. This example is a randomized controlled trial with two factors. Both factors, outcome and treatment, have three levels. The data is listed in Tab. 9.13. Each treatment group contains 50 samples. Fig. 9.13 shows the data. Outcomes are indicated by the border color of the circles (O_1=blue,O_2=red,O_3=magenta). Treatments are indicated by the interior color of the circles (T_1=orange,T_3=blue,T_3=green). The counts for outcome O_1 are larger than the other two.

We analyze the data by Poisson regression:

```
Deviance Residuals:
       1         2         3         4         5         6         7         8
```

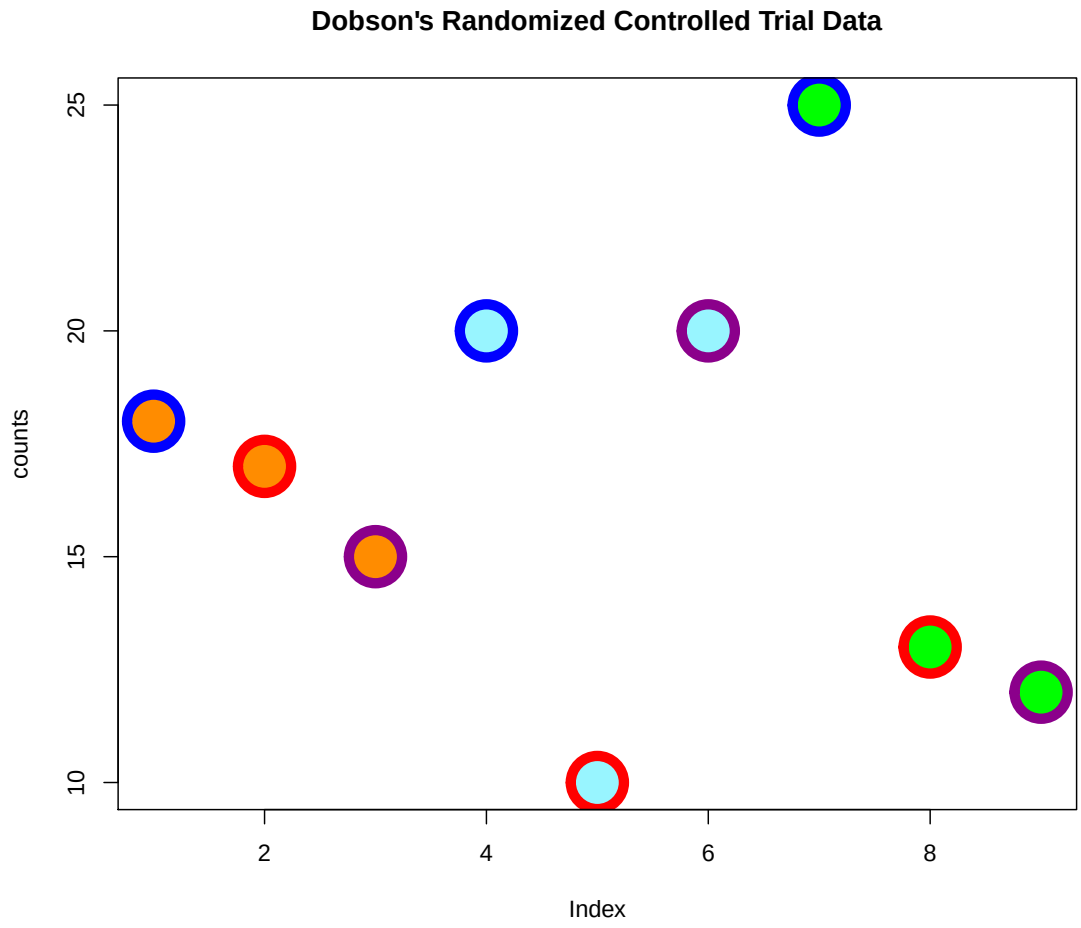

Figure 9.15: Toy data from Dobson [1990] for randomized controlled trial analyzed by Poisson regression.

```
-0.67125   0.96272  -0.16965  -0.21999  -0.95552   1.04939   0.84715  -0.09167
       9
-0.96656

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.045e+00  1.709e-01  17.815   <2e-16 ***
outcome2    -4.543e-01  2.022e-01  -2.247   0.0246 *
outcome3    -2.930e-01  1.927e-01  -1.520   0.1285
treatment2   8.717e-16  2.000e-01   0.000   1.0000
treatment3   4.557e-16  2.000e-01   0.000   1.0000
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 10.5814  on 8  degrees of freedom
Residual deviance:  5.1291  on 4  degrees of freedom
AIC: 56.761

Number of Fisher Scoring iterations: 4
```

Of course the intercept is significant as the data is not centered around zero. Outcome 1 and treatment 1 are the reference. Treatment does not have influence on the counts because they are all the same. Outcome O_2 is significant for a level of 0.05. That can be seen in Fig. 9.13 because the reference outcome O_1 indicated by blue border circles is larger than outcome O_2 indicated by red border circles.

9.5.4.6 Detergent Brand: Poisson Regression

These data were reported by Ries & Smith (1963), analyzed by Cox & Snell (1989) and described in Modern Applied Statistics with S+. The user preference for brand M or X is counted. At analyzing these data, different factors are considered. Explanatory variables (regressors, features) are “user of M”, “temperature”, and “water”. The data are presented in Tab. 9.14.

The results of Poisson regression with

```
formula = Fr ~ M.user * Temp * Soft + Brand
```

are

```
     Min        1Q    Median        3Q       Max
-2.20876  -0.99190  -0.00126   0.93542   1.97601

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   4.01524    0.10034  40.018  < 2e-16 ***
```

user of M?	No				Yes			
temperature	Low		High		Low		High	
preference	X	M	X	M	X	M	X	M
water softness								
hard	68	42	42	30	37	52	24	43
medium	66	50	33	23	47	55	23	47
soft	63	53	29	27	57	49	19	29

Table 9.14: Data set on detergent brand preference from Ries & Smith (1963) and analyzed by Cox & Snell (1989).

```
M.userY       -0.21184    0.14257  -1.486  0.13731
TempHigh      -0.42381    0.15159  -2.796  0.00518 **
SoftMedium     0.05311    0.13308   0.399  0.68984
SoftSoft       0.05311    0.13308   0.399  0.68984
BrandM        -0.01587    0.06300  -0.252  0.80106
M.userY:TempHigh              0.13987    0.22168   0.631  0.52806
M.userY:SoftMedium            0.08323    0.19685   0.423  0.67245
M.userY:SoftSoft              0.12169    0.19591   0.621  0.53449
TempHigh:SoftMedium          -0.30442    0.22239  -1.369  0.17104
TempHigh:SoftSoft            -0.30442    0.22239  -1.369  0.17104
M.userY:TempHigh:SoftMedium   0.21189    0.31577   0.671  0.50220
M.userY:TempHigh:SoftSoft    -0.20387    0.32540  -0.627  0.53098
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 118.627  on 23  degrees of freedom
Residual deviance:  32.826  on 11  degrees of freedom
AIC: 191.24

Number of Fisher Scoring iterations: 4
```

Besides the intercept only temperature is significant but not the water characteristic nor the previous use of the brand.

We now try another model with

```
formula = Fr ~ M.user * Temp * Soft + Brand * M.user * Temp
```

which gives

```
Deviance Residuals:
     Min        1Q    Median        3Q       Max
```

```
-0.91365  -0.35585   0.00253   0.33027   0.92146

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   4.14887    0.10603  39.128  < 2e-16 ***
M.userY      -0.40521    0.16188  -2.503  0.01231 *
TempHigh     -0.44275    0.17121  -2.586  0.00971 **
M.userY:TempHigh            -0.12692    0.26257  -0.483  0.62883
SoftMedium    0.05311    0.13308   0.399  0.68984
SoftSoft      0.05311    0.13308   0.399  0.68984
M.userY:SoftMedium           0.08323    0.19685   0.423  0.67245
M.userY:SoftSoft             0.12169    0.19591   0.621  0.53449
TempHigh:SoftMedium         -0.30442    0.22239  -1.369  0.17104
TempHigh:SoftSoft           -0.30442    0.22239  -1.369  0.17104
M.userY:TempHigh:SoftMedium  0.21189    0.31577   0.671  0.50220
M.userY:TempHigh:SoftSoft   -0.20387    0.32540  -0.627  0.53098
BrandM       -0.30647    0.10942  -2.801  0.00510 **
M.userY:BrandM               0.40757    0.15961   2.554  0.01066 *
TempHigh:BrandM              0.04411    0.18463   0.239  0.81119
M.userY:TempHigh:BrandM      0.44427    0.26673   1.666  0.09579 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 118.627  on 23  degrees of freedom
Residual deviance:   5.656  on  8  degrees of freedom
AIC: 170.07

Number of Fisher Scoring iterations: 4

Correlation of Coefficients:

(Intercept)                 1
M.userY                     , 1
TempHigh                    , . 1
M.userY:TempHigh            . , , 1
SoftMedium                  , . .   1
SoftSoft                    , . .   . 1
M.userY:SoftMedium          . ,   . , . 1
M.userY:SoftSoft            . ,   . . , . 1
TempHigh:SoftMedium         .   , . . . .   1
TempHigh:SoftSoft           .   , . . .   . . 1
M.userY:TempHigh:SoftMedium   . . . .   , . , . 1
M.userY:TempHigh:SoftSoft     . . .   . . , . , . 1
BrandM                                    .       1
```

```
M.userY:BrandM                                .       , 1
TempHigh:BrandM                                 . .   . . 1
M.userY:TempHigh:BrandM                         . .   . . , 1
attr(,"legend")
[1] 0 ‘ ’ 0.3 ‘.’ 0.6 ‘,’ 0.8 ‘+’ 0.9 ‘*’ 0.95 ‘B’ 1
```

Besides the temperature also the brand becomes significant and also, to a lesser degree, the previous use of brand M and the combined previous use of brand M plus brand M.

Finally we compare the two models by an ANOVA table:

```
Analysis of Deviance Table

Model 1: Fr ~ M.user * Temp * Soft + Brand
Model 2: Fr ~ M.user * Temp * Soft + Brand * M.user * Temp
  Resid. Df Resid. Dev Df Deviance
1        11     32.826
2         8      5.656  3    27.17
```

9.5.4.7 Tumor Data: Poisson Regression

In Dobson [2002], page 162, in Table 9.4, data from Roberts et al. (1981) are presented. The data are from a cross-sectional study of patients with a form of skin cancer called malignant melanoma. For a sample of $n = 400$ patients, the site of the tumor and its histological type were determined. The counts of patients with each combination of tumor type and body site, are given in Tab. 9.15. The patients are categorized by the type of tumor they have, which corresponds to the first factor with four levels: freckle, superficial, nodular, indeterminate. The patients are also categorized by the body site where the tumor was found, which corresponds to the second factor with three levels: head, trunk, extremities. The association between tumor type and site should be investigated.

Fig. 9.16 shows the data, where the four tumor types are indicated by the interior color of the circles (orange=freckle, blue=superficial, green=nodular, indeterminate=wood). The three locations at the body are indicated by the border color of the circles (head=blue,trunk=red,extremities=magenta).

We analyze these data by a Poisson regression:

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.0453  -1.0741   0.1297   0.5857   5.1354

Coefficients:
   Estimate Std. Error z value Pr(>|z|)
(Intercept)         1.7544     0.2040   8.600  < 2e-16 ***
typesuperficial     1.6940     0.1866   9.079  < 2e-16 ***
typenodular         1.3020     0.1934   6.731 1.68e-11 ***
typeindeterminate   0.4990     0.2174   2.295  0.02173 *
sitetrunk           0.4439     0.1554   2.857  0.00427 **
siteextremities     1.2010     0.1383   8.683  < 2e-16 ***
```

	Site			
Tumor type	Head & neck	Trunk	Extrem -ities	Total
Hutchinson's melanotic freckle	22	2	10	34
Superficial spreading melanoma	16	54	115	185
Nodular	19	33	73	125
Indeterminate	11	17	28	56
Total	68	106	226	400

Table 9.15: Dobson's malignant melanoma data: frequencies for tumor type and site (Roberts et al., 1981).

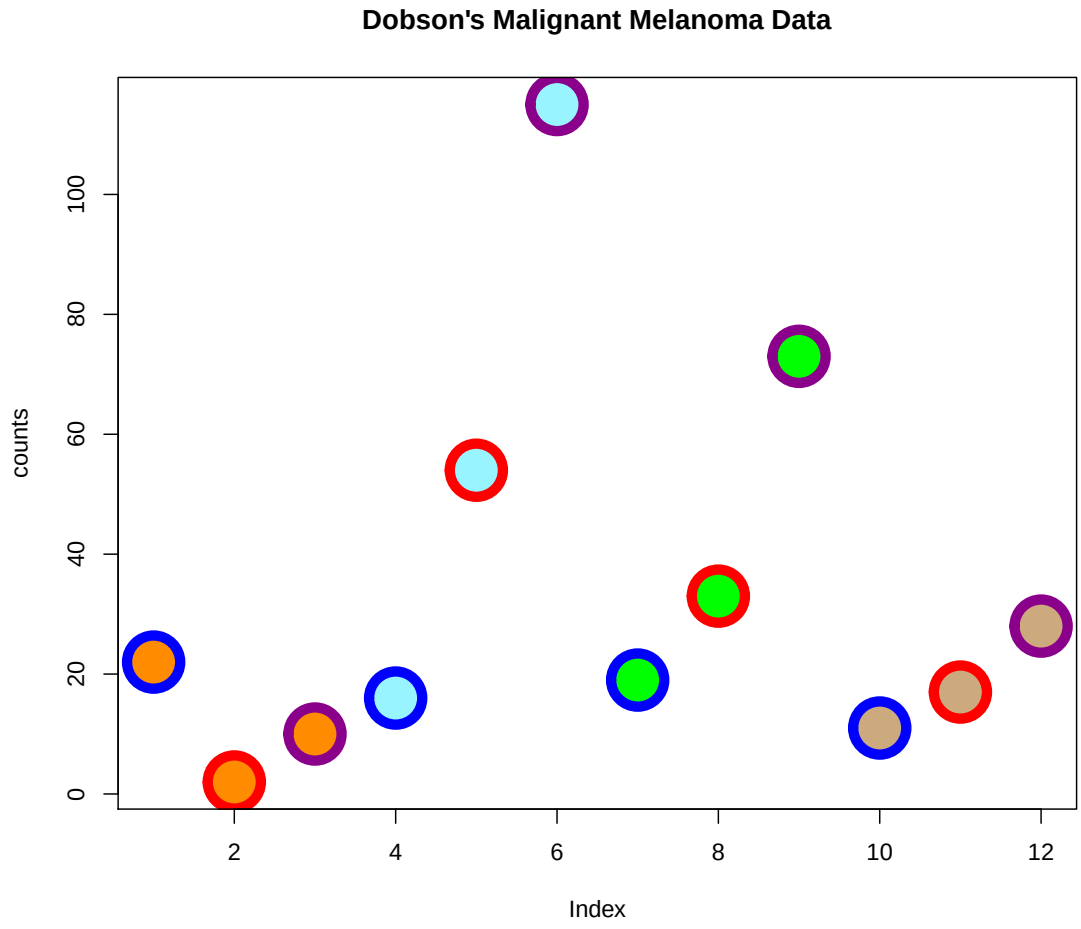

Figure 9.16: Dobson's malignant melanoma data where tumor types are counted. The four tumor types are indicated by the interior color of the circles (orange=freckle, blue=superficial, green=nodular, indeterminate=wood). The three locations at the body are indicated by the border color of the circles (head=blue,trunk=red,extremities=magenta).

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 295.203  on 11  degrees of freedom
Residual deviance:  51.795  on  6  degrees of freedom
AIC: 122.91

Number of Fisher Scoring iterations: 5
```

This means that type superficial and nodular are highly significant if compared to the counts of type freckle while indeterminate is less significant. This result can be confirmed in Fig. 9.16, where the blue and green interior (blue=superficial, green=nodular) circles have clearly higher counts if compared to freckle. The counts of indeterminate are not so clearly larger. The site extremities is also highly significant. In Fig. 9.16 data points corresponding to counts for extremities have magenta borders. The two largest counts belong to extremities of which one has tumor type superficial and one type nodular. To a lesser degree the site trunk is significant. Also this is confirmed in Fig. 9.16, where the third and fourth largest counts with a red border belong to the site trunk.

9.5.4.8 Ulcers and Aspirin Use: Logistic Regression

This example is a case-control study of gastric and duodenal ulcers and aspirin use from Dobson [2002], page 165/166, with data in Table 9.7. In this retrospective case-control study ulcer patients were compared to controls which are matched with respect to age, sex and socio-economic status. The data is from Duggan et al. (1986). The individuals are categorized:

(1) ulcer cases or controls,

(2) site of the ulcer: gastric or duodenal,

(3) aspirin use or not.

The data is shown in Tab. 9.16 and in Fig. 9.16.

Questions which are of interest for this data set are:

1. Is gastric ulcer associated with aspirin use?
2. Is duodenal ulcer associated with aspirin use?
3. Is any association with aspirin use the same for both ulcer sites?

We first look at a model without interaction effects using

```
formula = y ~ group + type
```

	Aspirin use		
	Non-user	User	Total
Gastric ulcer			
Control	62	6	68
Cases	39	25	64
Duodenal ulcer			
Control	53	8	61
Cases	49	8	57
Total	203	47	250

Table 9.16: Dobson's gastric and duodenal ulcers and aspirin use from Duggan et al. (1986).

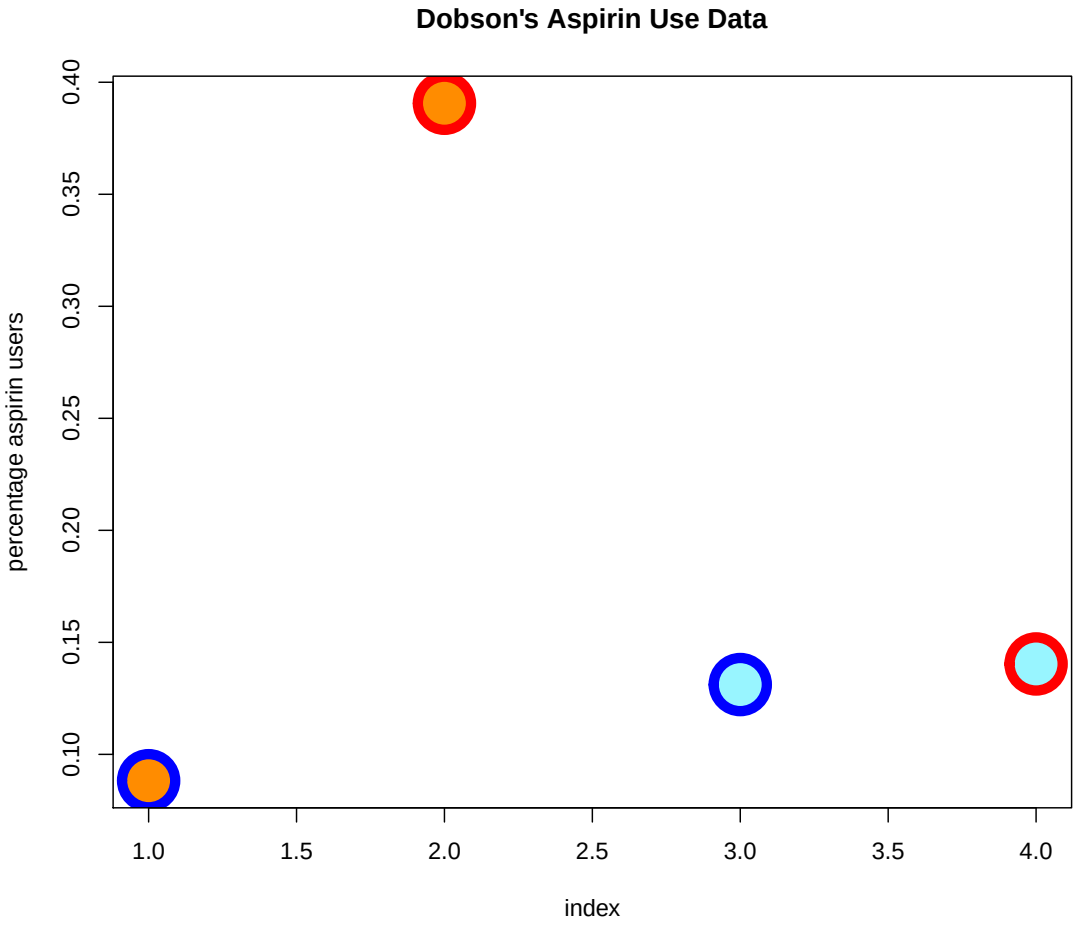

Figure 9.17: Dobson's gastric and duodenal ulcers and aspirin use. The border color indicates ulcer patients, the cases (red), and controls (blue). The interior color indicates the type of ulcer for the cases: gastric (orange) or duodenal (blue).

which gives

```
Deviance Residuals:
      1        2        3        4
 1.2891  -0.9061  -1.5396   1.1959

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)    1.8219     0.3080   5.916  3.3e-09 ***
groupcases    -1.1429     0.3521  -3.246  0.00117 **
typeduodenal   0.7000     0.3460   2.023  0.04306 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 21.789  on 3  degrees of freedom
Residual deviance:  6.283  on 1  degrees of freedom
AIC: 28.003

Number of Fisher Scoring iterations: 4

Correlation of Coefficients:
             (Intercept) groupcases
groupcases   -0.73
typeduodenal -0.38       -0.05
```

As the count data are not centered, the intercept is significant. Most significant is the group cases for aspirin use. The rate is the percentage of the first count of all counts, that is the rate of aspirin non-users. The coefficient of group cases is -1.14 which means the rate of non-users is smaller than the rate for controls. This means that for cases the percentage of aspirin use is larger than for controls. Less significant and almost not significant is the type of ulcer where gastric is more related to aspirin users.

Next, we investigate the linear model with interaction effects using

```
formula = y ~ group*type
```

which gives

```
Deviance Residuals:
[1]  0  0  0  0

Coefficients:
         Estimate Std. Error z value Pr(>|z|)
(Intercept)               2.3354     0.4275   5.462  4.7e-08 ***
groupcases               -1.8907     0.4984  -3.793 0.000149 ***
```

```
typeduodenal              -0.4445     0.5715  -0.778 0.436711
groupcases:typeduodenal    1.8122     0.7333   2.471 0.013460 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2.1789e+01  on 3  degrees of freedom
Residual deviance: 2.3981e-14  on 0  degrees of freedom
AIC: 23.72

Number of Fisher Scoring iterations: 3
```

Again cases are significantly associated with aspirin use. Further cases with gastric are more related to aspirin use.

We compare these two models by an ANOVA table:

```
Analysis of Deviance Table

Model 1: y ~ group + type
Model 2: y ~ group * type
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1         1      6.283
2         0      0.000  1    6.283  0.01219 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

The deviance shows that the interaction model is significantly better at fitting the data. However, the AIC tells that this may only be due to overfitting to the data.

9.6 Regularization

In machine learning and statistics it is important to avoid that the model is too much fitted to the data. In this case only data specific features are modeled but not the structure in the data. This is called overfitting. Overfitting reduces generalization capabilities because other, new data will not have the specific features of the current data but only the general structures. To avoid overfitting, simple models should be selected Hochreiter and Schmidhuber [1995, 1994, 1997], Hochreiter and Obermayer [2006], Hochreiter et al. [2007], Knebel et al. [2008]. Simple models are models from low-complex model classes and as such cannot capture specific data characteristics but only general structures in the data. To prefer simple models during model selection is called *regularization*. In the following we present some regularization methods for linear models.

9.6.1 Partial Least Squares Regression

The first kind of regularization is based on models which are based on a $l < m$ variables. This means that regularization is achieved by fitting a model in a lower dimensional space. The idea of *partial least squares* (PLS) is to factorize both the response matrix $\boldsymbol{Y}$ and the regression matrix $\boldsymbol{X}$:

$$\boldsymbol{X} = \boldsymbol{T}\,\boldsymbol{P}^T + \boldsymbol{E} \tag{9.223}$$

$$\boldsymbol{Y} = \boldsymbol{U}\,\boldsymbol{Q}^T + \boldsymbol{F}\,, \tag{9.224}$$

where the covariance between $\boldsymbol{T}$ and $\boldsymbol{U}$ is maximized. $\boldsymbol{X}$ is an $n \times m$ matrix of predictors. $\boldsymbol{Y}$ is an $n \times p$ matrix of responses. $\boldsymbol{T}$ and $\boldsymbol{U}$ are $n \times l$ matrices that are, respectively, projections of $\boldsymbol{X}$ and projections of $\boldsymbol{Y}$. $\boldsymbol{P}$ and $\boldsymbol{Q}$ are, respectively, $m \times l$ and $p \times l$ orthogonal matrices. $\boldsymbol{E}$ and $\boldsymbol{F}$ are additive noise terms which are assumed to be independently normally distributed.

Iterative partial least squares finds projection vectors $\boldsymbol{w}$ for $\boldsymbol{X}$ and $\boldsymbol{v}$ for $\boldsymbol{Y}$ which have maximal covariance:

$$\max_{\|\boldsymbol{w}\|=\|\boldsymbol{v}\|=1} \mathrm{Cov}(\boldsymbol{X}\boldsymbol{w}, \boldsymbol{Y}\boldsymbol{v})\,. \tag{9.225}$$

Iterative partial least squares is closely related to *canonical correlation analysis* (CCA) which finds projection vectors $\boldsymbol{w}$ for $\boldsymbol{X}$ and $\boldsymbol{v}$ for $\boldsymbol{Y}$ which have maximal correlation coefficient:

$$\max_{\|\boldsymbol{w}\|=\|\boldsymbol{v}\|=1} \mathrm{corr}(\boldsymbol{X}\boldsymbol{w}, \boldsymbol{Y}\boldsymbol{v})\,. \tag{9.226}$$

PLS takes the variance into account while CCA only looks at the correlation.

For *partial least squares regression* (PLSR) the score matrix $\boldsymbol{T}$ is orthogonal:

$$\boldsymbol{T}^T\boldsymbol{T} = \boldsymbol{I}\,. \tag{9.227}$$

PLSR defines a *linear inner relation*, which is basically a regression:

$$\boldsymbol{U} = \boldsymbol{T}\,\boldsymbol{D} + \boldsymbol{H}\,, \tag{9.228}$$

where $\boldsymbol{D}$ is a diagonal matrix. Via this regression the covariance between $\boldsymbol{T}$ and $\boldsymbol{U}$ is maximized. This regression gives

$$\begin{aligned} \boldsymbol{Y} &= \boldsymbol{T}\,\boldsymbol{D}\,\boldsymbol{Q}^T + \boldsymbol{H}\,\boldsymbol{Q}^T + \boldsymbol{F} && (9.229)\\ &= \boldsymbol{T}\,\boldsymbol{C}^T + \boldsymbol{F}', && (9.230) \end{aligned}$$

where $\boldsymbol{C}^T = \boldsymbol{D}\boldsymbol{Q}^T$ are the regression coefficients and $\boldsymbol{F}' = \boldsymbol{H}\boldsymbol{Q}^T + \boldsymbol{F}$ is the noise. We obtained a least squares estimate with projections $\boldsymbol{T}$ from orthogonal matrices.

For the noise free case, we have the decompositions

$$\begin{aligned} \boldsymbol{X} &= \boldsymbol{T}\,\boldsymbol{P}^T, && (9.231)\\ \boldsymbol{T} &= \boldsymbol{X}\,\boldsymbol{W}, && (9.232)\\ \hat{\boldsymbol{Y}} &= \boldsymbol{T}\,\boldsymbol{D}\,\boldsymbol{Q}^T, && (9.233)\\ \boldsymbol{U} &= \hat{\boldsymbol{Y}}\,\boldsymbol{Q}. && (9.234) \end{aligned}$$

The matrix $\hat{\boldsymbol{Y}}$ approximates $\boldsymbol{Y}$, the columns of $\boldsymbol{T}$ are the "latent vectors", $\boldsymbol{D}$ are the "regression weights" (see Eq. (9.228)) and $\boldsymbol{Q}$ is the "weight matrix" of the dependent variables $\boldsymbol{Y}$.

$\boldsymbol{W}$ is pseudo inverse of $\boldsymbol{P}^T$ which leads to the following equations:

$$\begin{aligned} \boldsymbol{T}^T\boldsymbol{T} &= \boldsymbol{I}, && (9.235)\\ \boldsymbol{Q}^T\boldsymbol{Q} &= \boldsymbol{I}, && (9.236)\\ \boldsymbol{W} &= (\boldsymbol{P}^T)^+, && (9.237)\\ \boldsymbol{U} &= \boldsymbol{T}\,\boldsymbol{D}, && (9.238)\\ \boldsymbol{D} &= \boldsymbol{T}^T\,\boldsymbol{U}. && (9.239) \end{aligned}$$

Using these equations the partial least squares regression algorithm Alg. 9.1 can be derived.

Partial least squares regression can be based on the singular value decomposition of $\boldsymbol{X}^T\boldsymbol{Y}$. If noise terms are ignored then we have

$$\boldsymbol{X}^T\boldsymbol{Y} = \boldsymbol{P}\,\boldsymbol{T}^T\boldsymbol{U}\,\boldsymbol{Q}^T = \boldsymbol{P}\,\boldsymbol{D}\,\boldsymbol{Q}^T, \tag{9.240}$$

where the second equality follows from Eq. (9.228). The largest singular value gives the first $\boldsymbol{w}$ and the first $\boldsymbol{q}$. The first $\boldsymbol{t}$ is the first eigenvector of $\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{Y}\boldsymbol{Y}^T$ and the first $\boldsymbol{u}$ is the first eigenvector of $\boldsymbol{Y}\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{X}^T$.

If $\boldsymbol{T}$ are the projections onto the first l principal components of $\boldsymbol{X}$ then this is called *principal components regression*.

9.6.2 Ridge Regression

Ridge regression is also known as *Tikhonov regularization* for ill-posed problems. The objective of the least squares estimate is the sum of squares

$$\|\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y}\|^2. \tag{9.241}$$

Algorithm 9.1 Partial least squares regression

Given: matrix $\boldsymbol{X}$, matrix $\boldsymbol{Y}$
initialization
initialize $\boldsymbol{u}$ by random values
$\boldsymbol{A}$ is set to the column centered and column normalized $\boldsymbol{X}$
$\boldsymbol{B}$ is set to the column centered and column normalized $\boldsymbol{Y}$
main loop
while $\boldsymbol{A}$ is not the null matrix **do**
while not converged **do**
$\boldsymbol{w} = \boldsymbol{A}^T\boldsymbol{u}$ (estimate $\boldsymbol{X}$ weights)
$\boldsymbol{t} = \boldsymbol{A}\boldsymbol{w}$ (estimate $\boldsymbol{X}$ factor scores)
$\boldsymbol{t} = \boldsymbol{t}/\|\boldsymbol{t}\|$ (normalize factor scores)
$\boldsymbol{q} = \boldsymbol{B}^T\boldsymbol{t}$ (estimate $\boldsymbol{Y}$ weights)
$\boldsymbol{q} = \boldsymbol{q}/\|\boldsymbol{q}\|$ (normalize weights)
$\boldsymbol{u} = \boldsymbol{B}\boldsymbol{q}$ (estimate $\boldsymbol{Y}$ factor scores)
use $\boldsymbol{w}$ to test if loop has converged
end while
$d = \boldsymbol{t}^T\boldsymbol{u}$
$\boldsymbol{p} = \boldsymbol{A}^T\boldsymbol{t}$
$\boldsymbol{A} = \boldsymbol{A} - \boldsymbol{t}\boldsymbol{p}^T$ (partial out the effect of $\boldsymbol{t}$ from $\boldsymbol{X} \sim \boldsymbol{A}$)
$\boldsymbol{B} = \boldsymbol{B} - d\boldsymbol{t}\boldsymbol{q}^T$ (partial out the effect of $\boldsymbol{t}$ from $\boldsymbol{Y} \sim \boldsymbol{B}$)
store all computed values $\boldsymbol{t}$, $\boldsymbol{u}$, $\boldsymbol{w}$, $\boldsymbol{q}$, $\boldsymbol{p}$ in the corresponding matrices
store d as diagonal element of $\boldsymbol{D}$
end while
result
training: $\hat{\boldsymbol{Y}} = \boldsymbol{T}\boldsymbol{D}\boldsymbol{Q}^T$
prediction: $\boldsymbol{\tau} = \boldsymbol{x}^T\boldsymbol{W}$ ($\boldsymbol{x}$ is normalized like $\boldsymbol{A}$); $\hat{\boldsymbol{y}} = \boldsymbol{\tau}\boldsymbol{D}\boldsymbol{Q}^T$

If the number of regressors is large, then overfitting is a problem. Overfitting refers to the fact that specific observations are fitted even if they are noisy or outliers. In this case the estimated parameters are adjusted to specific characteristics of the observed data which reduces the generalization to new unknown data. To avoid overfitting simple models should be selected even if they do not fit the observed data as well as the model with minimal squared error. Regularization fits the data while preferring simple models, that is, there is a trade-off between simple models and small squared error. This trade-off is controlled by a hyperparameter.

Regularization can be performed by an additional objective on the parameters, like a squared term in the parameters:

$$\|\boldsymbol{X}\boldsymbol{\beta} \; - \; \boldsymbol{y}\|^2 \; + \; \|\boldsymbol{\Gamma}\,\boldsymbol{\beta}\|^2 \,. \tag{9.242}$$

The estimator for ridge regression $\hat{\boldsymbol{\beta}}$, which minimizes this objective, is

$$\hat{\boldsymbol{\beta}} \; = \; \left(\boldsymbol{X}^T\boldsymbol{X} \; + \; \boldsymbol{\Gamma}^T\boldsymbol{\Gamma}\right)^{-1} \boldsymbol{X}^T\boldsymbol{y} \,. \tag{9.243}$$

Often $\boldsymbol{\Gamma} = \sqrt{\gamma}\boldsymbol{I}$ is used, where γ is a hyperparameter of the method which has to be adjusted. γ controls the trade-off between simple models and low squared error. For $\boldsymbol{\Gamma} = \sqrt{\gamma}\boldsymbol{I}$ we have the estimator

$$\hat{\boldsymbol{\beta}} \; = \; \left(\boldsymbol{X}^T\boldsymbol{X} \; + \; \gamma\boldsymbol{I}\right)^{-1} \boldsymbol{X}^T\boldsymbol{y} \,. \tag{9.244}$$

The variance of the ridge regression estimator is:

$$\mathrm{Var}(\hat{\boldsymbol{\beta}}) \; = \; \sigma^2 \; \left(\boldsymbol{X}^T\boldsymbol{X} \; + \; \gamma\boldsymbol{I}\right)^{-1} \boldsymbol{X}^T\boldsymbol{X} \left(\boldsymbol{X}^T\boldsymbol{X} \; + \; \gamma\boldsymbol{I}\right)^{-1} \,. \tag{9.245}$$

The bias of ridge regression estimator is:

$$\mathrm{bias}(\hat{\boldsymbol{\beta}}) \; = \; -\,\gamma \; \left(\boldsymbol{X}^T\boldsymbol{X} \; + \; \gamma\boldsymbol{I}\right)^{-1} \boldsymbol{\beta} \,. \tag{9.246}$$

It has been shown, that there is always a γ for which the parameter mean squared error of ridge regression is smaller than this error of least squares. However, this γ is not known. Ridge regression is consistent if $\gamma/n \xrightarrow{n} 0$ Knight and Fu [2000].

Ridge regression is an L^2-norm regularizer, that is the squares of the parameters (or products of them) are weighted and summed up and thereby penalized. Therefore small absolute parameter values around zero are preferred by ridge regression. However, in general the ridge regression estimator has its parameters not exactly at zero. The regularizing term hardly changes if the values are already small because the derivatives are proportional to the values. If very small parameter values still improve the squared error, they will be kept. Setting these small parameters to zero would increase the error more than it would decrease the regularization term. On the other hand, larger values are very strongly penalized.

Ridge regression gives a solution even if the parameters are under-determined for few data points because $\left(\boldsymbol{X}^T\boldsymbol{X} \; + \; \boldsymbol{\Gamma}^T\boldsymbol{\Gamma}\right)^{-1}$ always exists. This means that ridge regression has a unique solution.

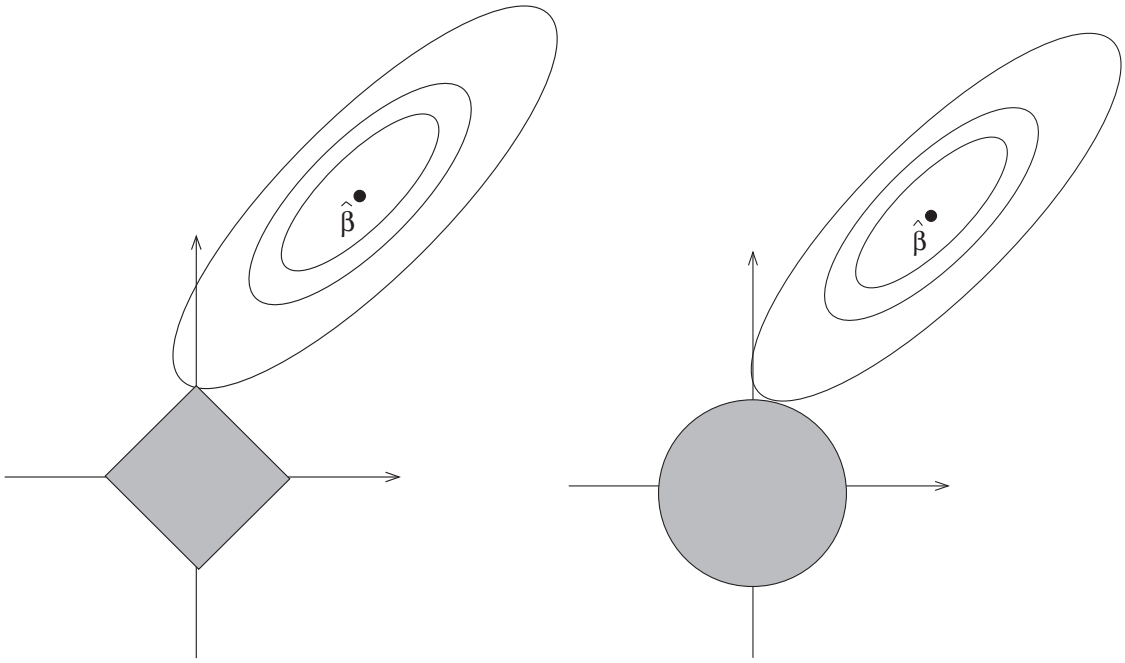

Figure 9.18: Optimization LASSO (left) vs. ridge regression (right). The error objective, the ellipse, touches in most cases a corner of the L^1-norm where at least one component is zero. In contrast the L^2-norm does not possess corners as all points with the same regularization value are on a hyperball.

9.6.3 LASSO

Least absolute shrinkage and selection operator (LASSO) Tibshirani [1996] performs a L^1-norm regularization. The objective is

$$\|\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y}\|^2 + \gamma \|\boldsymbol{\beta}\|_1 . \tag{9.247}$$

In contrast to ridge regression, the LASSO estimate has many zero components (see Fig. 9.18). The decrease of the regularization term if the absolute values of parameters are made smaller, does not depend on the current values of the parameters. Thus, small parameter values are pushed toward zero. Therefore LASSO is often used for feature selection because features, of which the corresponding parameters are zero, can be removed from the model without changing regression result.

The minimization of the LASSO objective is a quadratic optimization problem. It can be solved by techniques of constrained quadratic optimization. An alternative method for finding a solution is the *forward stepwise regression algorithm*:

1. Start with all coefficients β_j equal to zero.
2. Find the predictor x_j which is most correlated with y, and add it to the model. Take residuals $r = y - \hat{y}$.
3. Continue, at each stage adding to the model the predictor which is most correlated with r.

4. Until: all predictors are in the model

A even better approach to finding the LASSO estimator is the *least angle regression procedure*. In contrast to forward stepwise regression, a predictor is not fully added to the model. The coefficient of that predictor is increased only until that predictor is no longer the one which is most correlated with the residual r. Then some other competing predictor is pushed by increasing its parameter.

1. Start with all coefficients β_j equal to zero.

2. Find the predictor x_j most correlated with y. Increase the coefficient β_j in the direction of the sign of its correlation with y. Take residuals $r = y - \hat{y}$ and compute correlations. Stop when some other predictor x_k has the same correlation with r than x_j.

3. Increase (β_j, β_k) in their joint least squares direction, until some other predictor x_m has the same correlation with the residual r.

4. Until: all predictors are in the model

This procedure gives the entire path of LASSO solutions if one modification is made. This modification is: if a non-zero coefficient is set to zero, remove it from the active set of predictors and recompute the joint direction.

Lasso is consistent if $\gamma/n \xrightarrow{n} 0$ Knight and Fu [2000].

LASSO is implemented in the R package `lars` which can be used to fit least angle regression, LASSO, and infinitesimal forward stagewise regression models.

9.6.4 Elastic Net

The L^1-norm has also disadvantages. For many features m and few samples n, only the first n features are selected. For correlated variables LASSO only selects one variable and does not use the others. *Elastic net* is a compromise between ridge regression and LASSO. It has both an L^1-norm as well as an L^2-norm regularizer. The objective is

$$\|\boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{y}\|^2 + \gamma \|\boldsymbol{\beta}\|_1 + \delta \|\boldsymbol{\beta}\|_2^2 . \tag{9.248}$$

The elastic net estimator minimizes this objective.

The problem is that now two hyperparameters are introduced. If $\gamma = 0$ then the elastic net is ridge regression. If $\delta = 0$ then the elastic net is LASSO.

The elastic net is consistent if $\gamma/n \xrightarrow{n} 0$ Knight and Fu [2000].

Elastic net is implemented in the R package `glmnet`. This package allows to fit a generalized linear model via penalized maximum likelihood. The regularization path is computed for the LASSO or elastic net penalty at a grid of values for the regularization parameter γ.

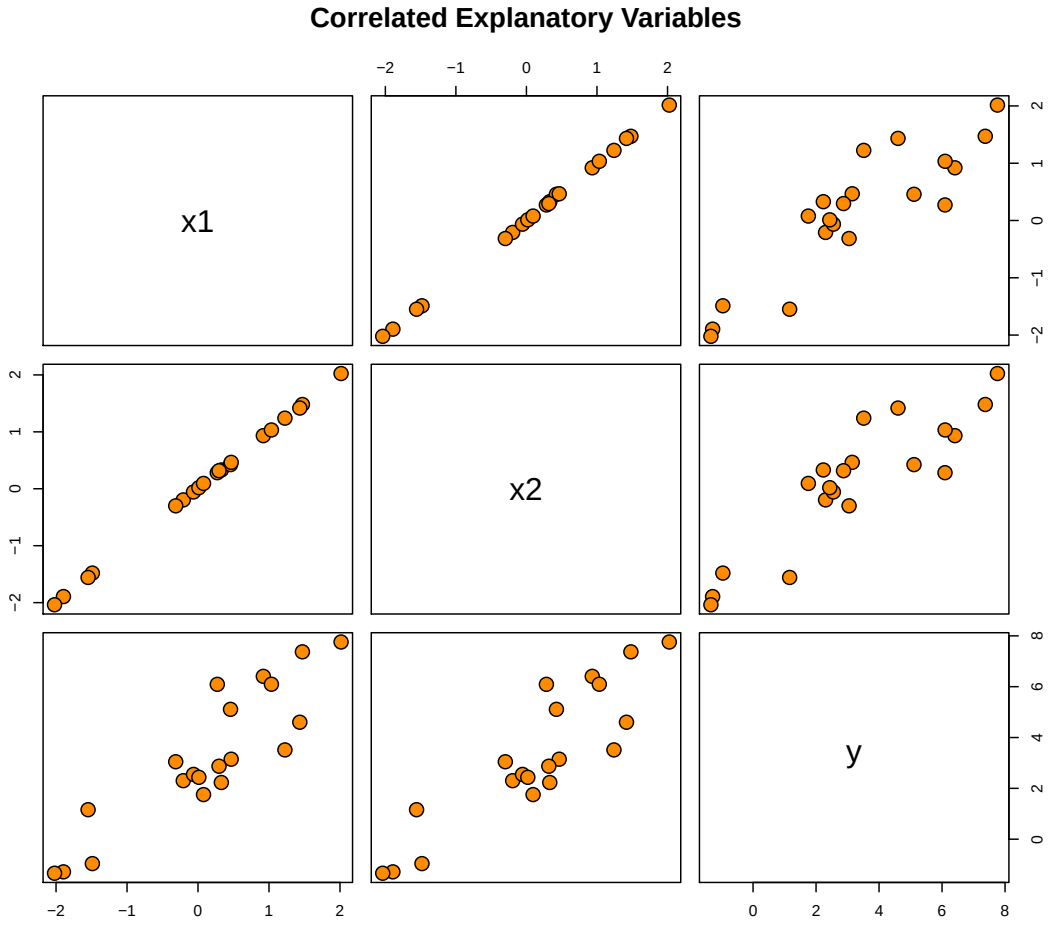

Figure 9.19: An Example for highly correlated explanatory variables.

9.6.5 Examples

9.6.5.1 Example: Ridge Regression, LASSO, Elastic Net

We generate data with highly correlated explanatory variables, where the correlation is as follows:

```
          x1        x2        y
x1 1.0000000 0.9999319 0.8927331
x2 0.9999319 1.0000000 0.8919416
y  0.8927331 0.8919416 1.0000000
```

The data is shown as pairs of scatter plots in Fig. 9.19.

First we fit a standard linear model:

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-12.710  -4.842   3.027   1.723   8.941  14.850
```

```
(Intercept)          x1          x2
   3.026583   14.854954  -12.711132
```

Next we fit the model with ridge regression:

```
              x1       x2
2.985240 1.051382 1.011735
```

The ridge regression is much closer to the true parameter values. Fig. 9.20 shows the results. The response data are the wooden-colored squares. Standard least squares gives the green circles while ridge regression gives the orange circles. The noise free data is indicated by crosses. Ridge regression is less prone to overfitting and closer to the crosses and, therefore, it generalizes better.

We are interested in the LASSO solution:

```
R-squared: 0.801
Sequence of LASSO moves:
     x1 x2
Var   1  2
Step  1  2


LARS/LASSO
Call: lars(x = cbind(x1, x2), y = y)
  Df     Rss      Cp
0  1 138.062 67.3827
1  2  28.030  1.3351
2  3  27.489  3.0000


         x1        x2
0  0.000000   0.00000
1  2.116893   0.00000
2 14.854954 -12.71113
```

The last call supplies the intercept for the LASSO solutions. Since in step 2 the residual does not change much compared to step 3 which all variables, we select step 2 solution $y = 2.982374 + 2.116893 * x_1$. The solution is shown in Fig. 9.21. LASSO is almost as good as ridge regression since the orange circles are covered by the blue circles obtained from LASSO. However, LASSO used only one explanatory variable.

A call to elastic net, where the L^1 and the L^2 norms are equally weighted ($\alpha = 0.5$), gives:

```
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
0.002078 0.014340 0.098850 0.628400 0.681200 4.691000
```

λ is the factor which weighs the penalty term that includes both L^1 and the L^2 norm.

We choose a small penalty term $s = 0.004$:

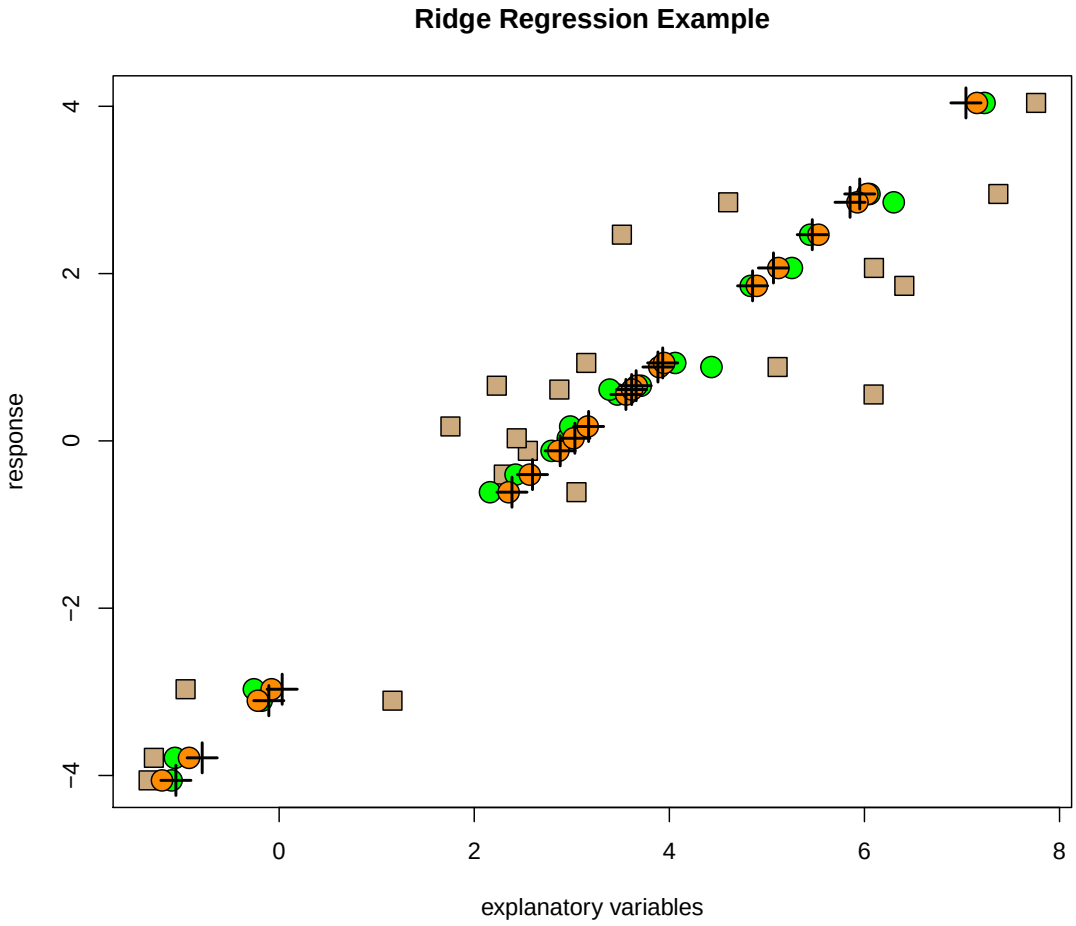

Figure 9.20: Example of ridge regression. The response data are the wooden-colored squares. Standard least squares gives the green circles while ridge regression gives the orange circles. The noise free data is indicated by crosses. Ridge regression is less prone to overfitting and closer to the crosses and, therefore, it generalizes better.

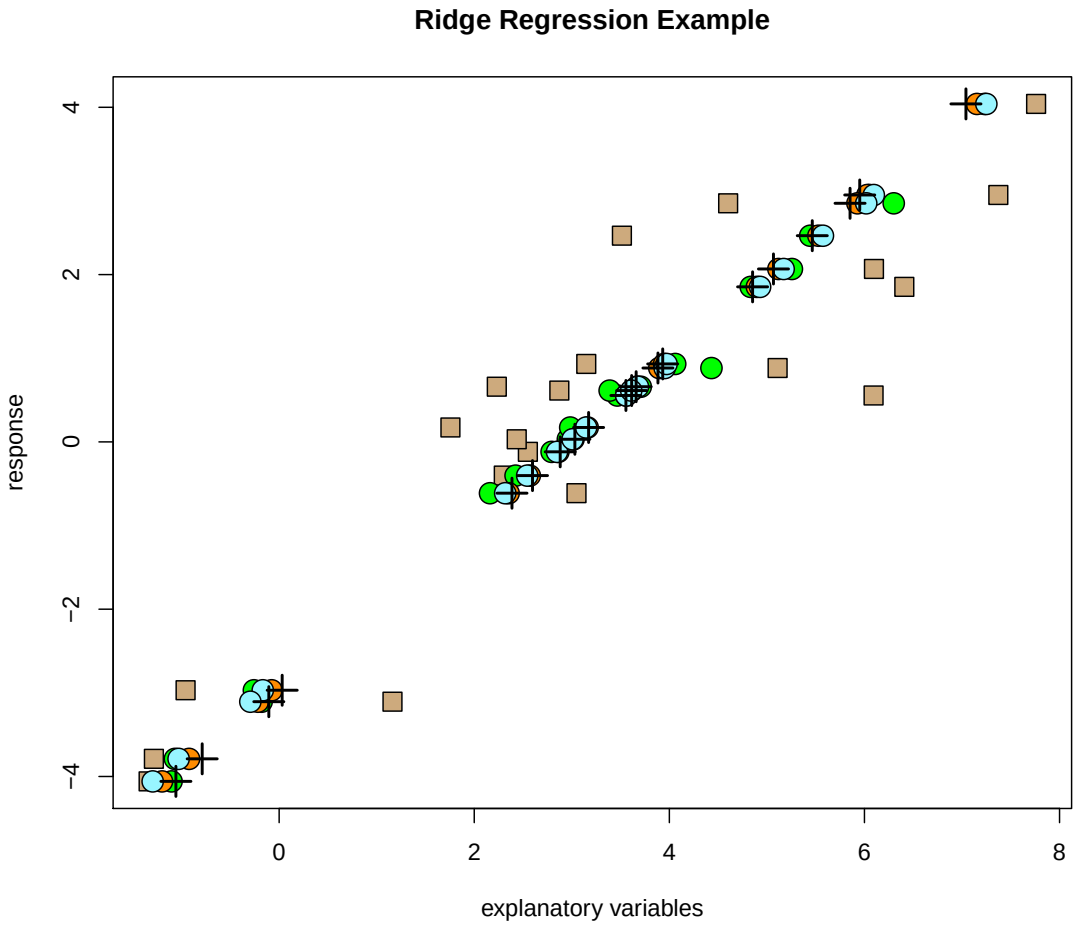

Figure 9.21: Example of LASSO. The same figure as Fig. 9.20 except that now LASSO with only one variable is shown (blue circles). This solution is almost as good as the ridge regression because the orange circles are covered by the blue circles. However, LASSO used only one explanatory variable.

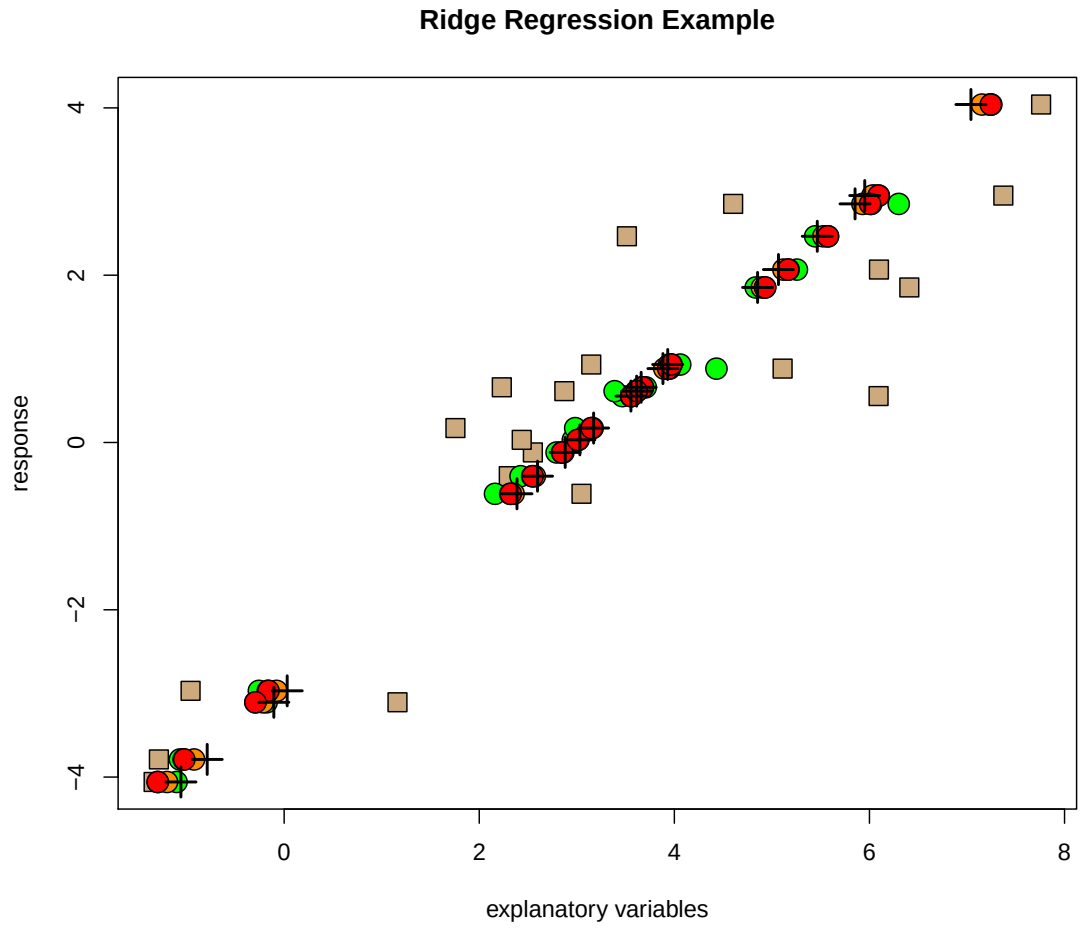

Figure 9.22: Example of elastic net. The same figure as Fig. 9.21 except that now elastic net with $\alpha = 0.5$ is shown (red circles). This solution does not differ much from the LASSO solution because the red circles overlay the blue circles.

```
(Intercept) 2.981441
x1          1.738632
x2          0.374484
```

The elastic net solution is shown in Fig. 9.22. This solution does not differ much from the LASSO solution because the red circles overlay the blue circles.

9.6.5.2 Example: Diabetes using Least Angle Regression

The data contain blood and other measurements in diabetics and are taken from Efron, Hastie, Johnstone and Tibshirani (2003) "Least Angle Regression", *Annals of Statistics*. The diabetes data frame has 442 rows and 3 columns:

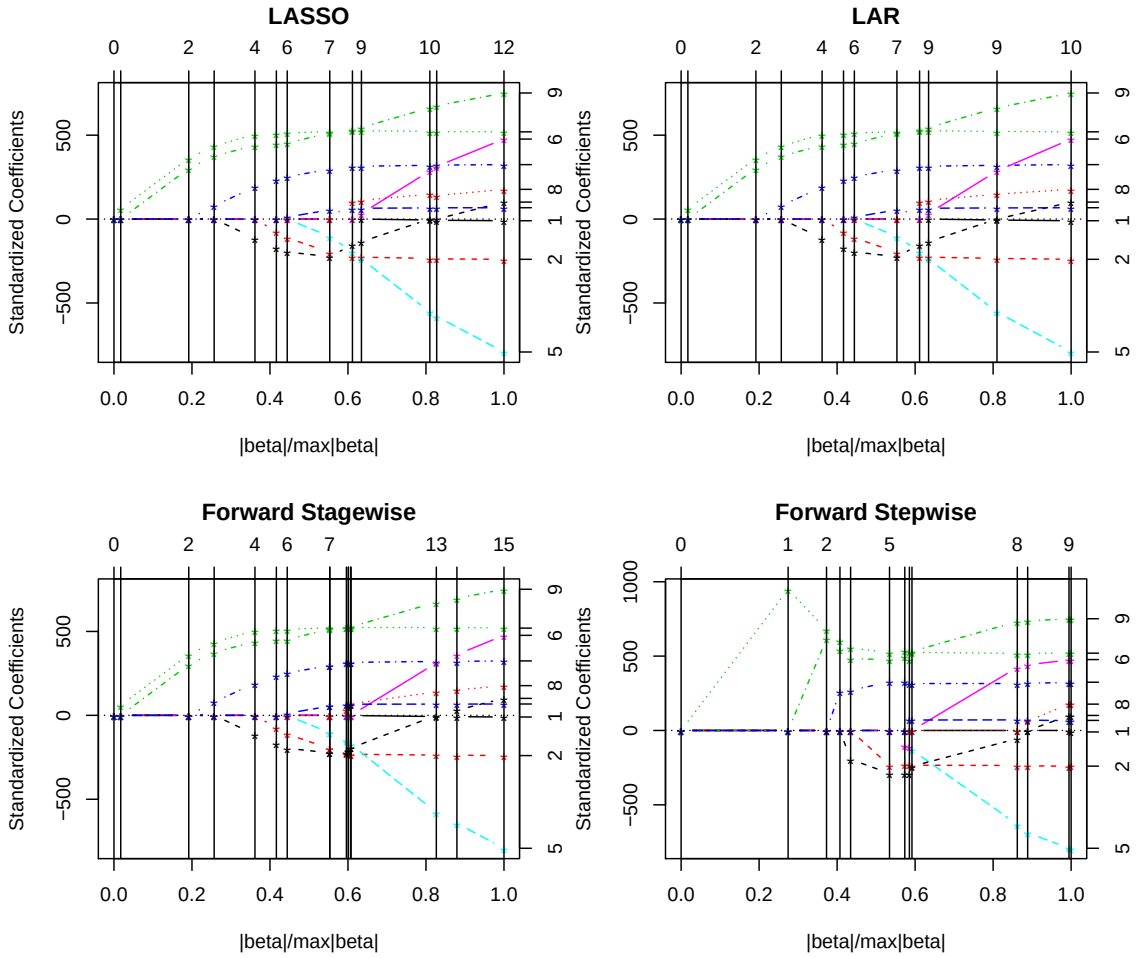

Figure 9.23: Example of least angle regression. The diabetes data set was fitted by LASSO, least angle regression, forward stagewise, and forward stepwise. The figure shows the coefficients that obtain certain values at certain steps.

1. x: a matrix with 10 columns with explanatory variables “age”, “sex2”, “bmi”, “map”, “tc2”, “ldl”, “hdl”, “tch”, “ltg”, “glu”. That is age, sex, body mass index (bmi), and blood measurements like cholesterol levels (ldl and hdl) etc.

2. y: a numeric vector,

3. x_2: a matrix with 64 columns which contains all explanatory variables, their squared values, and measurements of interaction effects.

The x matrix has been standardized to have unit L^2 norm in each column and zero mean. The matrix x_2 consists of x plus certain interactions. Fig. 9.23 shows coefficients at different steps for the diabetes data set fitted by LASSO, least angle regression, forward stagewise, and forward stepwise.

In the following, the different solution paths for the different methods are listed (LASSO, least angle regression, forward stagewise, and forward stepwise):

```
       age  sex  bmi  map   tc  ldl  hdl  tch  ltg  glu
 [1,]    0    0    0    0    0    0    0    0    0    0
 [2,]    0    0   60    0    0    0    0    0    0    0
 [3,]    0    0  362    0    0    0    0    0  302    0
 [4,]    0    0  435   79    0    0    0    0  375    0
 [5,]    0    0  506  191    0    0 -114    0  440    0
 [6,]    0  -75  511  234    0    0 -170    0  451    0
 [7,]    0 -112  512  253    0    0 -196    0  452   12
 [8,]    0 -198  522  297 -104    0 -224    0  515   55
 [9,]    0 -226  527  314 -195    0 -152  106  530   64
[10,]    0 -227  526  315 -237   34 -135  111  545   65
[11,]   -6 -234  523  320 -554  287    0  149  663   66
[12,]   -7 -237  521  322 -580  314    0  140  675   67
[13,]  -10 -240  520  324 -792  477  101  177  751   68

       age  sex  bmi  map   tc  ldl  hdl  tch  ltg  glu
 [1,]    0    0    0    0    0    0    0    0    0    0
 [2,]    0    0   60    0    0    0    0    0    0    0
 [3,]    0    0  362    0    0    0    0    0  302    0
 [4,]    0    0  435   79    0    0    0    0  375    0
 [5,]    0    0  506  191    0    0 -114    0  440    0
 [6,]    0  -75  511  234    0    0 -170    0  451    0
 [7,]    0 -112  512  253    0    0 -196    0  452   12
 [8,]    0 -198  522  297 -104    0 -224    0  515   55
 [9,]    0 -226  527  314 -195    0 -152  106  530   64
[10,]    0 -227  526  315 -237   34 -135  111  545   65
[11,]  -10 -240  520  324 -792  477  101  177  751   68

       age  sex  bmi  map   tc  ldl  hdl  tch  ltg  glu
 [1,]    0    0    0    0    0    0    0    0    0    0
 [2,]    0    0   60    0    0    0    0    0    0    0
 [3,]    0    0  362    0    0    0    0    0  302    0
 [4,]    0    0  435   79    0    0    0    0  375    0
 [5,]    0    0  506  191    0    0 -114    0  440    0
 [6,]    0  -75  511  234    0    0 -170    0  451    0
 [7,]    0 -112  512  253    0    0 -196    0  452   12
 [8,]    0 -198  522  297 -104    0 -224    0  515   55
 [9,]    0 -198  522  297 -104    0 -224    0  515   55
[10,]    0 -230  522  313 -148    0 -224   35  524   65
[11,]    0 -231  522  315 -159    0 -211   50  526   66
[12,]    0 -231  522  315 -159    0 -211   50  526   66
[13,]   -1 -232  523  316 -172    0 -195   68  528   66
[14,]   -1 -232  523  316 -172    0 -195   68  528   66
[15,]   -8 -238  523  322 -644  362   31  151  697   67
```

x_1	-2	2	-2	2
x_2	3	-3	1	-1
t	1	-1	-1	1

Table 9.17: A toy example where variable x_1 is relevant because $t = x_1 + x_2$ but has no target correlation.

```
[16,]  -10 -240  520  324 -792  477  101  177  751   68

       age  sex  bmi  map   tc  ldl  hdl  tch  ltg  glu
 [1,]   0    0    0    0    0    0    0    0    0    0
 [2,]   0    0  949    0    0    0    0    0    0    0
 [3,]   0    0  675    0    0    0    0    0  615    0
 [4,]   0    0  603  262    0    0    0    0  544    0
 [5,]   0    0  555  270    0    0 -194    0  485    0
 [6,]   0 -236  524  326    0    0 -289    0  474    0
 [7,]   0 -227  538  328    0 -103 -291    0  498    0
 [8,]   0 -233  527  315    0 -111 -289    0  479   70
 [9,]   0 -236  518  316 -632  423  -55    0  732   71
[10,]   0 -241  520  322 -791  474  100  177  750   66
[11,]  -10 -240  520  324 -792  477  101  177  751   68
```

The final solution is the same. The variables that were selected first and second agree between the different methods. The first variable that has been selected is body mass index followed by "lgt" and then "map" and thereafter "hdl".

The features that are selected for the combined variables in x_2 are:

```
[1] "bmi"
[1] "bmi" "ltg"
[1] "bmi" "map" "ltg"
[1] "bmi" "map" "hdl" "ltg"
[1] "bmi" "map" "hdl" "ltg" "bmi:map"
[1] "bmi" "map" "hdl" "ltg" "age:sex" "bmi:map"
```

The most important variables are the variables which were identified previously.

9.6.5.3 Example: Relevant Variable but No Correlation to Response

We demonstrate on a toy example that relevant variables may be not correlated to the response / target variable. The toy example is shown in Tab. 9.17.

We now perform least squares regression, ridge regression, and LASSO:

```
Correlation:
          t         x1         x2
```

x_1	0	1	-1	1
x_2	-1	1	0	0
x_3	0	0	-1	1
t	-1	1	-1	1

Table 9.18: A toy example where variable x_1 is irrelevant because $t = x_2 + x_3$ but has high target correlation.

```
t  1.0000000  0.0000000  0.4472136
x1 0.0000000  1.0000000 -0.8944272
x2 0.4472136 -0.8944272  1.0000000

least squares:
  (Intercept)            x1            x2
-8.326673e-17  1.000000e+00  1.000000e+00

ridge regression with lambda=1:
  x1        x2
0.0000000 0.2622951 0.3278689
```

Fig. 9.24 shows the solution paths for different LASSO fitting methods. The variable x_1 is always selected in the second step even if it is not correlated to the response variable.

9.6.5.4 Example: Irrelevant Variable but High Correlation to Response

We demonstrate on a toy example that irrelevant variables may be correlated to the response / target variable. The toy example is shown in Tab. 9.18.

Again we fit the data by least squares regression, ridge regression, and LASSO:

```
Correlation:
           t        x1        x2        x3
t  1.0000000 0.9045340 0.7071068 0.7071068
x1 0.9045340 1.0000000 0.4264014 0.8528029
x2 0.7071068 0.4264014 1.0000000 0.0000000
x3 0.7071068 0.8528029 0.0000000 1.0000000

least squares:
  (Intercept)            x1            x2            x3
-1.171607e-16  4.686428e-16  1.000000e+00  1.000000e+00

ridge regression:
    x1         x2         x3
-0.1043478  0.4173913  0.6330435  0.4660870
```

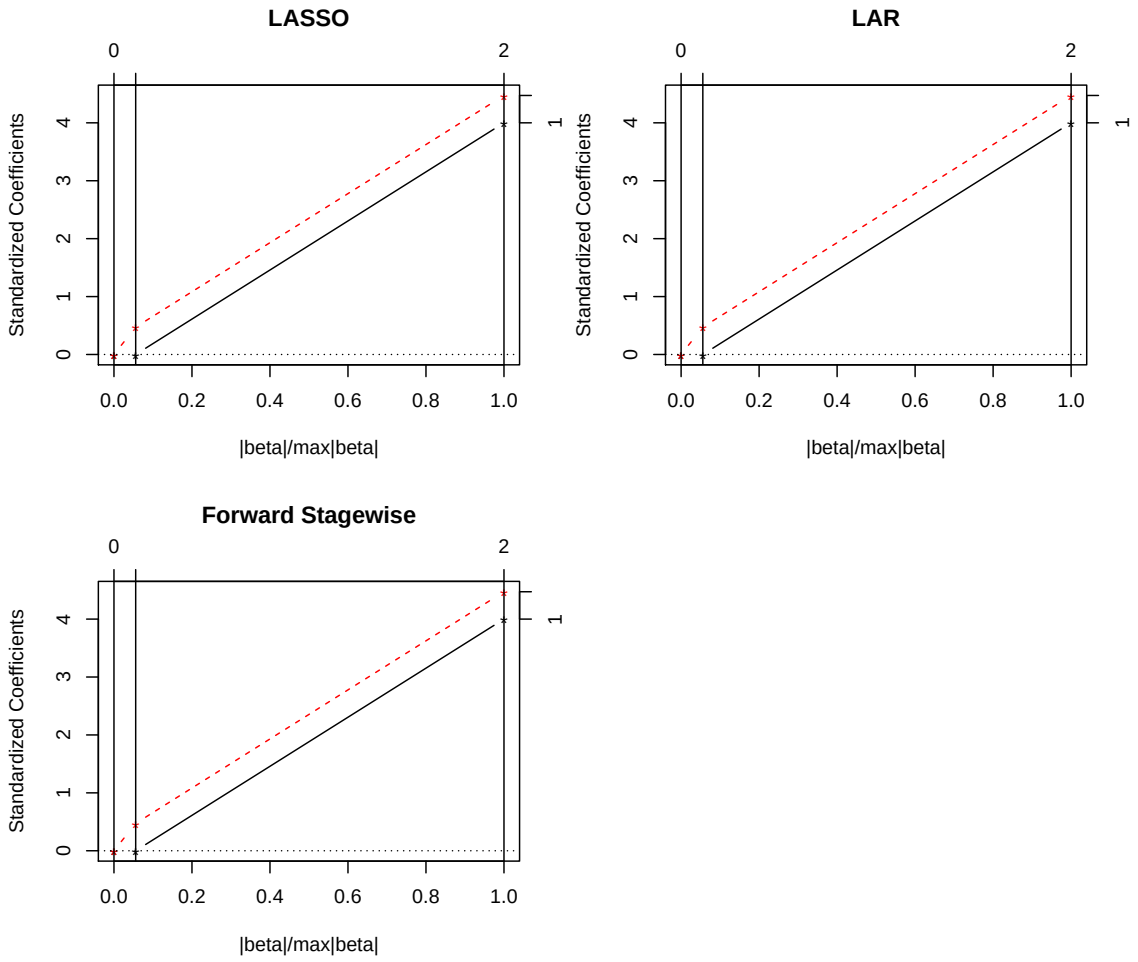

Figure 9.24: A toy example where variable x_1 is relevant because $t = x_1 + x_2$ but has not target correlation. The solution paths for different LASSO fitting methods. The variable x_1 is selected in the second step.

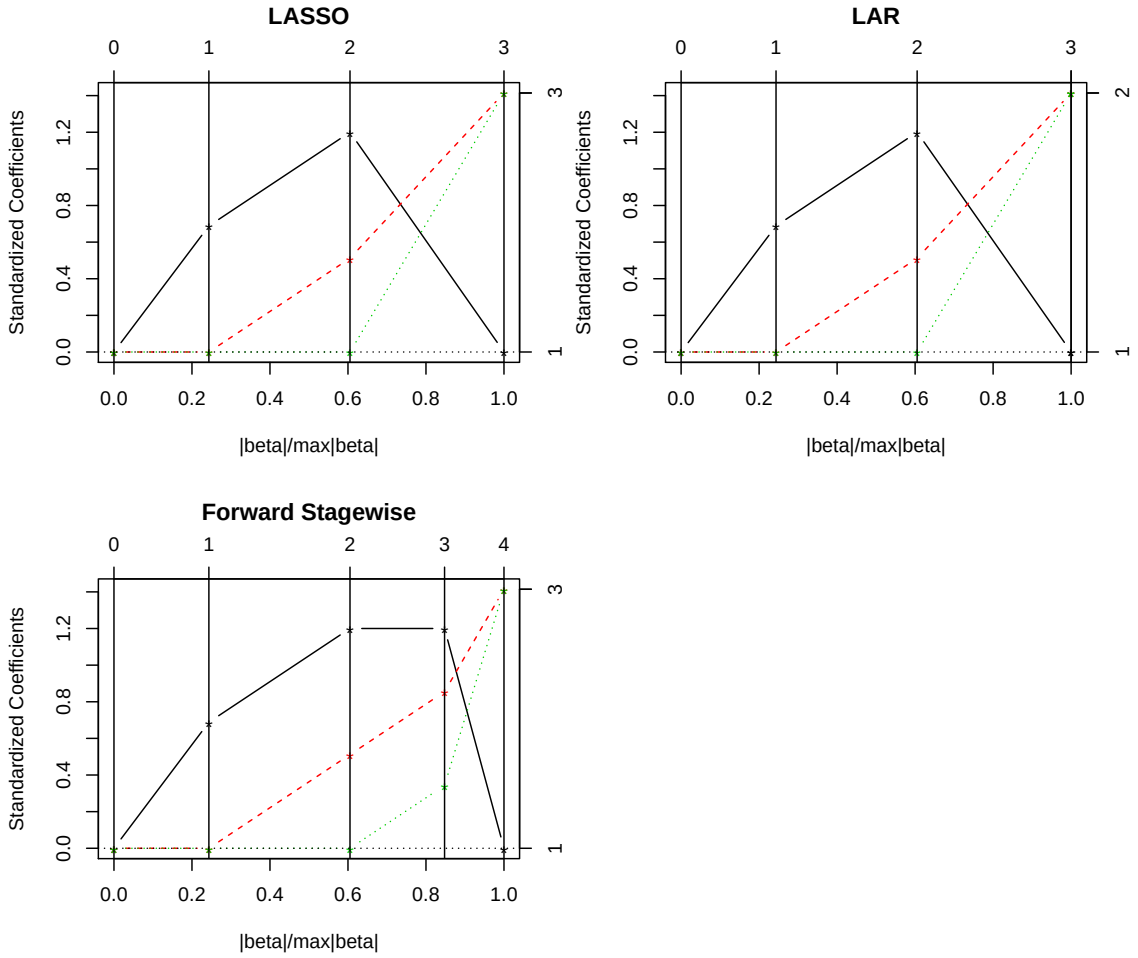

Figure 9.25: A toy example where variable x_1 is irrelevant because $t = x_2 + x_3$ but has high target correlation. The solution paths for different LASSO fitting methods are shown. The variable x_1 is selected first but in the last step correctly removed.

Least squares finds the correct solution while ridge regression uses the highly correlated variable to reduce the overall squared sum of coefficients (to obtain small regularization terms). Fig. 9.25 shows the solution paths for different LASSO fitting methods. The variable x_1 is selected first but in the last step correctly removed.

9.6.5.5 Gas Vapor: Ridge Regression and LASSO

This data set is from Rencher and Schaalje [2008] page 182, Ex. 7.53, Table 7.3, and originally from Weisberg (1985), page 138. When gasoline is pumped into the tank of a car, vapors are vented into the atmosphere. An experiment was conducted to determine whether the response y, the amount of vapor, can be predicted using the following four variables based on initial conditions of the tank and the dispensed gasoline:

y	x_1	x_2	x_3	x_4	y	x_1	x_2	x_3	x_4
29	33	53	3.32	3.42	40	90	64	7.32	6.70
24	31	36	3.10	3.26	46	90	60	7.32	7.20
26	33	51	3.18	3.18	55	92	92	7.45	7.45
22	37	51	3.39	3.08	52	91	92	7.27	7.26
27	36	54	3.20	3.41	29	61	62	3.91	4.08
21	35	35	3.03	3.03	22	59	42	3.75	3.45
33	59	56	4.78	4.57	31	88	65	6.48	5.80
34	60	60	4.72	4.72	45	91	89	6.70	6.60
32	59	60	4.60	4.41	37	63	62	4.30	4.30
34	60	60	4.53	4.53	37	60	61	4.02	4.10
20	34	35	2.90	2.95	33	60	62	4.02	3.89
36	60	59	4.40	4.36	27	59	62	3.98	4.02
34	60	62	4.31	4.42	34	59	62	4.39	4.53
23	60	36	4.27	3.94	19	37	35	2.75	2.64
24	62	38	4.41	3.49	16	35	35	2.59	2.59
32	62	61	4.39	4.39	22	37	37	2.73	2.59

Table 9.19: Rencher's gas vapor data from Rencher and Schaalje [2008] and originally from Weisberg (1985).

1. $x_1 =$ tank temperature (°F),
2. $x_2 =$ gasoline temperature (°F),
3. $x_3 =$ vapor pressure in tank (psi),
4. $x_4 =$ vapor pressure of gasoline (psi).

The data are given in Tab. 9.19.

Correlation of the variables often give a first impression which variables might be helpful for prediction:

```
          [,1]      [,2]      [,3]      [,4]      [,5]
[1,] 1.0000000 0.8260665 0.9093507 0.8698845 0.9213333
[2,] 0.8260665 1.0000000 0.7742909 0.9554116 0.9337690
[3,] 0.9093507 0.7742909 1.0000000 0.7815286 0.8374639
[4,] 0.8698845 0.9554116 0.7815286 1.0000000 0.9850748
[5,] 0.9213333 0.9337690 0.8374639 0.9850748 1.0000000
```

The response y is highly correlated with all explanatory variables which in turn are correlated among themselves. y is most correlated with x_4 followed by x_2. x_4 is very highly correlated with x_3 and least with x_2.

We start with standard least squares regression:

```
Coefficients:
(Intercept)           x1           x2           x3           x4
    1.01502     -0.02861      0.21582     -4.32005      8.97489

anova(l1)
Analysis of Variance Table

Response: y
          Df  Sum Sq Mean Sq F value    Pr(>F)
x          4 2520.27  630.07   84.54 7.249e-15 ***
Residuals 27  201.23    7.45
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

The variables x_3 and x_4 seem to be relevant. We know that they are highly correlated and lead to overfitting effects.

The relevance of the variables is checked by ridge regression which deals with these highly correlated variables:

```
                    x1          x2          x3          x4
 0.72339986 -0.04937793  0.27780519  0.35225191  3.74029965
```

Here variable x_4 sticks out.

Next we analyze the data set by LASSO:

```
[1] 0.0000000 0.0000000 0.0000000 0.4963341

[1] 0.0000000 0.2695754 0.0000000 3.5437050

[1] -0.06804859  0.27044138  0.00000000  4.48953562
```

Here it becomes clear that x_4 is the most important variable and next a less correlated variable x_2 is selected.

We perform feature selection and use only the variables x_2 and x_4:

```
Coefficients:
  (Intercept)  x[, c(2, 4)]1  x[, c(2, 4)]2
       0.1918         0.2747         3.6020


Analysis of Variance Table

Response: y
             Df  Sum Sq Mean Sq F value    Pr(>F)
x[, c(2, 4)]  2 2483.11 1241.56  151.04 4.633e-16 ***
```

```
Residuals    29  238.39    8.22
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

We now compare the full model with the model where only two features are selected:

```
Analysis of Variance Table

Model 1: y ~ x
Model 2: y ~ x[, c(2, 4)]
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     27 201.23
2     29 238.39 -2   -37.159 2.4929 0.1015
```

The model with only two features does not perform significantly worse.

We want to check which model is better suited by Akaike's information criterion (AIC):

```
AIC(l1):
[1]  5.00000 68.83842
AIC(l3):
[1]  3.00000 70.26103
```

The model with only two variables should be chosen.

9.6.5.6 Chemical Reaction: Ridge Regression and LASSO

This data set is from Rencher and Schaalje [2008] page 182, Ex. 7.54, Table 7.4 and originally from Box and Youle (1955) and was also used in Andrews and Herzberg (1985), page 188. The yield in a chemical reaction should be maximized, therefore the values of the following variables were used to control the experimenter:

1. $x_1 =$ temperature (°C),
2. $x_2 =$ concentration of a reagent (%),
3. $x_3 =$ time of reaction (hours).

The response variables were:

1. $y_1 =$ percent of unchanged starting material,
2. $y_2 =$ percent converted to the desired material.

The data are given in Tab. 9.20.

First we check the correlation among the variables:

y_1	y_2	x_1	x_2	x_3
41.5	45.9	162	23	3
33.8	53.3	162	23	8
27.7	57.5	162	30	5
21.7	58.8	162	30	8
19.9	60.6	172	25	5
15.0	58.0	172	25	8
12.2	58.6	172	30	5
4.3	52.4	172	30	8
19.3	56.9	167	27.5	6.5
6.4	55.4	177	27.5	6.5
37.6	46.9	157	27.5	6.5
18.0	57.3	167	32.5	6.5
26.3	55.0	167	22.5	6.5
9.9	58.9	167	27.5	9.5
25.0	50.3	167	27.5	3.5
14.1	61.1	177	20	6.5
15.2	62.9	177	20	6.5
15.9	60.0	160	34	7.5
19.6	60.6	160	34	7.5

Table 9.20: Rencher's chemical reaction data from Rencher and Schaalje [2008].

```
          [,1]       [,2]       [,3]       [,4]       [,5]
[1,]  1.0000000 -0.60782343 -0.67693865 -0.22472586 -0.45253956
[2,] -0.6078234  1.00000000  0.40395099  0.07998377  0.39273121
[3,] -0.6769387  0.40395099  1.00000000 -0.46200145 -0.02188275
[4,] -0.2247259  0.07998377 -0.46200145  1.00000000  0.17665667
[5,] -0.4525396  0.39273121 -0.02188275  0.17665667  1.00000000
```

The first response variable has negative correlation to the first regressor and less negative correlation to the third regressor. The second response variable is negatively correlated to the first response variable which was to be expected. The second response variable is equally correlated to the first and third regressor.

We start with a least square estimator:

```
Coefficients:
(Intercept)           x1           x2           x3
    332.111       -1.546       -1.425       -2.237


Analysis of Variance Table

Response: y1
          Df  Sum Sq Mean Sq F value    Pr(>F)
```

```
x          3 1707.16  569.05  106.47 2.459e-10 ***
Residuals 15   80.17    5.34
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

All variables are relevant for prediction. x_3 is the most relevant variable.

We perform regularization using ridge regression:

```
    x1         x2         x3
307.512361  -1.424838  -1.279060  -2.179261
```

The figure did not change compared to standard least squares estimation. This is a hint that indeed all variables are required.

Next we perform LASSO:

```
[1] -0.3518723  0.0000000  0.0000000

[1] -0.5182233  0.0000000 -0.6334936
```

The first and last variable seem to be the most relevant ones.

We fit a least squares model with the two most important variables:

```
Coefficients:
  (Intercept)  x[, c(1, 3)]1  x[, c(1, 3)]2
      222.957         -1.101         -2.853

Analysis of Variance Table

Response: y1
             Df  Sum Sq Mean Sq F value    Pr(>F)
x[, c(1, 3)]  2 1209.61  604.81   16.75 0.0001192 ***
Residuals    16  577.72   36.11
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

An ANOVA table shows that all variables are required to predict the response:

```
Analysis of Variance Table

Model 1: y1 ~ x
Model 2: y1 ~ x[, c(1, 3)]
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1     15  80.17
2     16 577.72 -1   -497.55 93.088 7.988e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

We move on the second response variable y_2, that is, the converted material to the desired product.

```
formula = y2 ~ x

Coefficients:
(Intercept)           x1           x2           x3
   -26.0353       0.4046       0.2930       1.0338

anova(l12)
Analysis of Variance Table

Response: y2
          Df Sum Sq Mean Sq F value  Pr(>F)
x          3 151.00  50.334  3.0266 0.06235 .
Residuals 15 249.46  16.631
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Again x_3 is the most relevant variable but now even more dominant.

```
      x1          x2          x3
-19.9403245   0.3747668   0.2617700   0.9933463
```

The figure remains the same for ridge regression.

We perform fitting with LASSO:

```
[1] 0.008327752 0.000000000 0.000000000

[1] 0.1931751 0.0000000 0.7039310
```

Interestingly, x_1 is selected before x_3. Looking at the correlation matrix, we see that indeed x_1 is more correlated to y_2 than x_3 (0.40 vs. 0.39).

If we select the two variables which would be first selected by LASSO, then we have for the least squares fit:

```
formula = y2 ~ x[, c(1, 3)]

Coefficients:
  (Intercept)  x[, c(1, 3)]1  x[, c(1, 3)]2
      -3.5856         0.3131         1.1605

anova(l32)
Analysis of Variance Table
```

```
Response: y2
             Df Sum Sq Mean Sq F value  Pr(>F)
x[, c(1, 3)]  2 129.96  64.978  3.8433 0.04334 *
Residuals    16 270.51  16.907
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Comparing the full model with the model, where only two features are selected by an ANOVA table gives:

```
Analysis of Variance Table

Model 1: y2 ~ x
Model 2: y2 ~ x[, c(1, 3)]
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     15 249.46
2     16 270.51 -1   -21.047 1.2655 0.2783
```

Therefore, the model with only two features is not significantly worse than the full model.

9.6.5.7 Land Rent: Ridge Regression and LASSO

This data set is from Rencher and Schaalje [2008] page 184, Ex. 7.55, Table 7.5 and originally from Weisberg (1985) page 162. For 34 counties in Minnesota the following variables were recorded in 1977:

1. y: average rent paid per acre of land with alfalfa,
2. x_1: average rent paid per acre for all land,
3. x_2: average number of dairy cows per square mile,
4. x_3: proportion of farmland in pasture.

The data is shown in Tab. 9.21. A relevant question is: can the rent for alfalfa land be predicted from the other three variables?

We check the correlation:

```
           [,1]       [,2]      [,3]       [,4]
[1,]  1.0000000  0.8868392 0.2967901 -0.3838808
[2,]  0.8868392  1.0000000 0.0296753 -0.5212982
[3,]  0.2967901  0.0296753 1.0000000  0.4876448
[4,] -0.3838808 -0.5212982 0.4876448  1.0000000

Standard deviatins of the variables:
1:
[1] 21.53698
```

y	x_1	x_2	x_3	y	x_1	x_2	x_3
18.38	15.50	17.25	.24	8.50	9.00	8.89	.08
20.00	22.29	18.51	.20	36.50	20.64	23.81	.24
11.50	12.36	11.13	.12	60.00	81.40	4.54	.05
25.00	31.84	5.54	.12	16.25	18.92	29.62	.72
52.50	83.90	5.44	.04	50.00	50.32	21.36	.19
82.50	72.25	20.37	.05	11.50	21.33	1.53	.10
25.00	27.14	31.20	.27	35.00	46.85	5.42	.08
30.67	40.41	4.29	.10	75.00	65.94	22.10	.09
12.00	12.42	8.69	.41	31.56	38.68	14.55	.17
61.25	69.42	6.63	.04	48.50	51.19	7.59	.13
60.00	48.46	27.40	.12	77.50	59.42	49.86	.13
57.50	69.00	31.23	.08	21.67	24.64	11.46	.21
31.00	26.09	28.50	.21	19.75	26.94	2.48	.10
60.00	62.83	29.98	.17	56.00	46.20	31.62	.26
72.50	77.06	13.59	.05	25.00	26.86	53.73	.43
60.33	58.83	45.46	.16	40.00	20.00	40.18	.56
49.75	59.48	35.90	.32	56.67	62.52	15.89	.05

Table 9.21: Rencher's land rent data from Rencher and Schaalje [2008].

```
2:
[1] 22.45614
3:
[1] 14.21056
4:
[1] 0.1532131
```

We also computed the standard deviations of the variables because x_3 has smaller values than the other variables. x_3 is about a factor of 100 smaller than the other variables.

We start with a least squares regression:

```
formula = y ~ x

Coefficients:
(Intercept)           x1           x2           x3
     0.6628       0.7803       0.5031     -17.1002

anova(l1)
Analysis of Variance Table

Response: y
          Df  Sum Sq Mean Sq F value    Pr(>F)
x          3 13266.9  4422.3  65.037 3.112e-13 ***
Residuals 30  2039.9    68.0
```

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

x_3 has the largest coefficient but it has to be divided by a factor of 100 to be in the range of the other variables. Thus, x_3 has actually the smallest influence on the response variable after fitting by least squares.

Ridge regression confirms the observations we had for the least squares estimator:

```
y ~ x with lambda=1
                     x1          x2          x3
  2.1360609   0.7542789   0.4955992 -18.2104311
```

Since ridge regression penalizes the coefficients for the standardized variables, the absolute coefficient for x_3 even increases. The other two coefficients decrease as they are pushed toward zero by ridge regression.

LASSO confirms our findings:

```
[1] 0.5832042 0.0000000 0.0000000

[1] 0.7872064 0.3223731 0.0000000
```

The first two explanatory variables are the most relevant. From the correlations we see that the first explanatory variable has largest correlation with the response and is therefore selected first. Interestingly, x_3 has the second largest correlation to the response variable but is not selected. The reason for this is that x_3 has also large correlation to x_1 and does not bring in much new information. In contrast to x_3, x_2 has low correlation to x_1 and brings in new information.

We again fit a least squares model, but now with only the first two explanatory variables:

```
formula = y ~ x[,c(1,2)]

Coefficients:
  (Intercept)  x[, c(1, 2)]1  x[, c(1, 2)]2
      -3.3151         0.8428         0.4103

anova(l3)
Analysis of Variance Table

Response: y
             Df  Sum Sq Mean Sq F value    Pr(>F)
x[, c(1, 2)]  2 13159.3  6579.6  94.981 6.015e-14 ***
Residuals    31  2147.5    69.3
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Comparing the full model with the model that has only the first two variables shows that the error difference is not significant:

```
Analysis of Variance Table

Model 1: y ~ x
Model 2: y ~ x[, c(1, 2)]
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     30 2039.9
2     31 2147.5 -1   -107.58 1.5821 0.2182
```

Therefore the reduce model may be chosen for analysis.

Bibliography

N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68(3):337–404, 1950.

Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. B*, 57(1):289–300, 1995.

Y. Benjamini and D. Yekutieli. The control of the false discovery rate in multiple testing under dependency. *The Annals of Statistics*, 29(4):1165–1188, 2001.

A. Berlinet and C. Thomas. *Reproducing kernel Hilbert spaces in Probability and Statistics*. Kluwer Academic Publishers, 2004.

C. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

C. E. Bonferroni. Il calcolo delle assicurazioni su gruppi di teste. Studii in Onore del Profesor S. O. Carboni, 1936. Roma.

F. Cucker and S. Smale. On the mathematical foundations of learning. *Bulletin of the American Mathematical Society*, 39(1):1–49, 2002.

A. J. Dobson. *An Introduction to Generalized Linear Models*. Chapman & Hall, Boca Raton, London, New York, Washington D.C., 1 edition, 1990. ISBN 0412311100 / 9780412311109.

A. J. Dobson. *An Introduction to Generalized Linear Models*. Texts in Statistical Science. Chapman & Hall / CRC, London, 2 edition, 2002. ISBN 1-58488-165-8.

S. Hochreiter, M. Heusel, and K. Obermayer. Fast model-based protein homology detection without alignment. *Bioinformatics*, 23(14):1728–1736, 2007.

S. Hochreiter and K. Obermayer. Support vector machines for dyadic data. *Neural Computation*, 18(6):1472–1510, 2006.

S. Hochreiter and J. Schmidhuber. Flat minimum search finds simple nets. Technical Report FKI-200-94, Fakultät für Informatik, Technische Universität München, 1994.

S. Hochreiter and J. Schmidhuber. Simplifying nets by discovering flat minima. In G. Tesauro, D. S. Touretzky, and T. K. Leen, editors, *Advances in Neural Information Processing Systems 7*, pages 529–536. MIT Press, Cambridge MA, 1995.

S. Hochreiter and J. Schmidhuber. Flat minima. *Neural Computation*, 9(1):1–42, 1997.

G. Kimeldorf and G. Wahba. Some results on Tchebycheffian spline functions. *J. Mathematical Analysis and Applications*, 33(1):82–95, 1971.

T. Knebel, S. Hochreiter, and K. Obermayer. An SMO algorithm for the potential support vector machine. *Neural Computation*, 20:271–287, 2008.

K. Knight and W. Fu. Asymptotics for lasso-type estimators. *The Annals of Statistics*, 28(5): 1356–1378, 2000.

R. Neal. *Bayesian Learning for Neural Networks*. Springer Verlag, New York, 1996.

E. E. Osuna, R. Freund, and F. Girosi. Support vector machines: Training and applications. Technical Report AIM No. 1602, CBCL no. 144, MIT, 1997.

C. R. Rao and H. Toutenburg. *Linear Models — Least Squares and Alternatives*. Springer Series in Statistics. Springer, New York, Berlin, Heidelberg, London, Tokyo, 2 edition, 1999. ISBN 0-387-98848-3.

A. C. Rencher and G. B. Schaalje. *Linear Models in Statistics*. Wiley, Hoboken, New Jersey, 2 edition, 2008.

J. Rissanen. Modeling by shortest data description. *Automatica*, 14:465–471, 1978.

B. Schölkopf and A. J. Smola. *Learning with kernels – Support Vector Machines, Reglarization, Optimization, and Beyond*. MIT Press, Cambridge, 2002.

R. Tibshirani. Regression shrinkage and selection via the lasso. *J. Royal. Statist. Soc B.*, 58(1): 267–288, 1996.

G. Wahba. Spline models for observational data. *SIAM*, 1990.

C. S. Wallace and D. M. Boulton. An information measure for classification. *Computer Jrnl.*, 11 (2):185–194, 1968.