Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Applied Statistical Regression

Marcel Dettling

HS 2011

September 26, 2011

1 Introduction

In science, but also in everyday life one often asks the question how a target (value) of special interest depends on several other factors or causes. Examples are numerous, e.g.:

- how fertilizer and soil quality affects the growth of plants
- how size, location, furnishment and age affect apartment rents
- how take-off-weight, distance and weather affect airplane fuel consumption

In all quantitative settings, regression techniques can provide an answer to these questions. They describe the relation between some *explanatory* or *predictor variables* and a variable of special interest, called the *response* or *target variable*. Regression techniques are of high practical importance, and probably the most widely used statistical methodology.

1.1 The Linear Model

One of mathematically simplest and most appealing ways to describe the relation between target and predictor variables is to use a linear function, which is specified up to some unknown parameters and a random error component. We will see later, that even under the restriction of linear functions, we obtain a very versatile modeling tool. When we write the target variable as Y, and the predictor variables as $x_1,...,x_p$, the linear regression model is as follows:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p + \varepsilon .$$

Here, $\beta_0, \beta_1, ..., \beta_p$ are unknown parameters, and ε is the random error term. The goal now is to estimate the unknown parameters, such that the error term is minimized according to some criterion. Mostly, the criterion will be the sum of squared residuals.

In order to perform estimation, we need data, i.e. the predictor and the target value need to be observed on a sufficient number of instances. We assume that we are given n such observations numbered from $1,...,n$, introduce an additional index i to account for this and write the regression function as follows:

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip} + \varepsilon_i .$$

In the standard framework, one also assumes that the error terms ε_i are independent and identically distributed, have expectation zero and finite variance. We will get back to these assumptions.

1.2 Goals with Linear Modeling

There are a variety of reasons to perform regression analysis. The two most prominent ones are:

- *Gaining some understanding on the causal relation, i.e. doing inference*

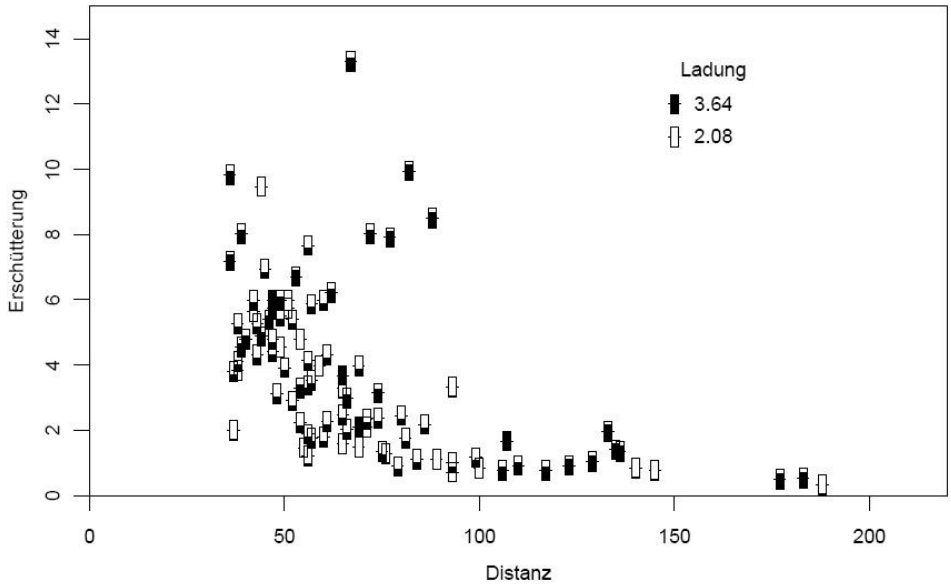
  In the “growth of plants” example from above, one might be interested in the question whether there is a benefit in growth, caused by the fertilizer, potentially regarding the influence of several co-variables. We will see that regression analysis offers tools to answer the question whether the fertilizer influence is beneficial in statistically significant way. Drawing conclusions on true causal relationship, however, is a somewhat different matter.

- *Target value prediction as a function of new explanatory variables*

  In the “fuel consumption” example from above, an airplane crew or the ground staff may want to determine the amount of fuel that is necessary for a particular flight, given its parameters. Regression analysis incorporates the previous experience in that matter and yields a quantitative prediction. It also results in prediction intervals which give a hint on the uncertainty such a prediction has. In practice, the latter might be very useful for the amount of reserve fuel that needs to be loaded.

1.3 The Versatility of Linear Modeling

At a first glance, it might seem very restrictive to use linear models only. However, the function only needs to be linear in the parameters, but we are free to transform the predictor variables as we wish. As we can see in the example and text below, linear regression modeling is an extremely versatile tool that is sufficient for a wide range of data analysis problems.

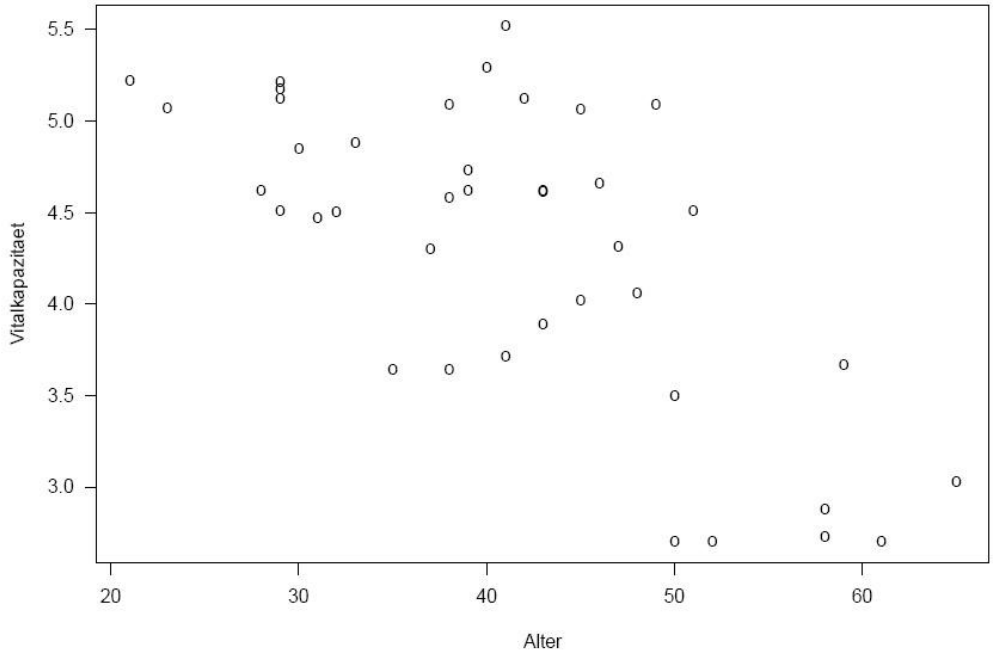


This example shows the shock measured after a detonation when a road tunnel was built in an urban area, depending on the distance and on the amount of explosive material that was used. We see that the relation is far from what is a straight line, i.e. what is commonly perceived as a “linear function”. Yet in this example, the relation was estimated with a linear regression model.

We will get back to this example later and now focus on a summary of the content of the remaining chapters of this scriptum, in order to show the versatility of (generalized) linear regression modeling altogether.

Simple Linear Regression

In simple linear regression, we are after the relation between two continuous variables Y and x. Below is an example, where the target variable Y is the vital capacity of the lung of workers in industry who are exposed to cadmium polluted air, depending on their age, which is variable x.



The task here is to fit a straight line into this scatter plot, i.e. a function $Y = \beta_0 + \beta_1 x$, where β_0 is the intercept, and β_1 is the slope. We will also discuss parameter estimation, inference, confidence and prediction intervals, residual analysis, variable transformations and erroneous input variables.

Multiple Linear Regression

This is an extension of the simple linear regression model in the sense that there is still one continuous target, but more than one (continuous) predictor variable. In the case of two explanatory variables, we are in a 3d-space and fit a plane. With more than two predictors, it is a hyper plane in a higher dimensional space that cannot be visualized anymore.

The topics which are discussed are similar to simple linear regression: estimation, inference, prediction and model diagnostics. However, a variety of new aspects come into play here, and we also discuss topics such as weighted regression and some thoughts on robustness.

Extending the Linear Model

The restriction to continuous predictor variables above was somewhat artificial. Using transformed input, binary or categorical variables is well within the envelope of multiple linear regression analysis. This chapter deals with some special aspects about these variable types.

Model Choice

In practice, it is often the case that there are a lot of potential explanatory variables. The regression setting can be used to determine which are the most influential on the response variable. We will get to know techniques for identifying the most relevant predictors, and how to skip others. Finally, we leave the area of multiple linear regression modeling here and conclude with some general remarks on modeling strategies.

Generalized Linear Models

As soon as the target variable is no longer continuous (e.g. binary or categorical data, count data, etc.), the above regression framework does no longer fit. However, it can be extended. In this introductory chapter on generalized linear modeling we explain what the benefits of this extension are, and also give a (not very technical) sketch of why and how the extension works.

Binary Logistic Regression

This is somewhat the simplest case of a generalized linear model: the response variable is binary, i.e. “yes/no”, “survived/died”, “success/no success”, etc. Such situations are often met in practice. We show how models are fit, how goodness-of-fit can be measured here and also talk about model diagnostics, all of which are quite different of what we saw before.

Ordinal and Nominal Response

In spirit, this is similar to the binary case above, but now, the response variable has more than just two levels. It can be unordered (the nominal case, e.g. which party somebody votes for, “SP”, “CVP”, “FDP”, “SVP”, “Others”) or ordered (the ordinal case, i.e. if a patient is affected by side effects “weakly”, “normally”, “strongly”). Again, we talk about estimation, fitting and interpretation.

Poisson Regression for Count Data

Here, the response variable is a number, and the explanatory variables can be either continuous or categorical. Examples include situations where one tries to model “the number of incidents” on factors such as weekday, season, weather, etc. There is also a close relation to modeling the content of contingency tables which will be discussed.

Modern Regression Techniques

The (generalized) linear model is very flexible and useful. However, there are “regression type” situations where one reaches its boundaries. High-dimensional problems where one has more predictor variables than examples are such a case, or also situations with inherent strong non-linearity. Here, we give a sketch on some alternative approaches such as recursive partitioning with classification and regression trees, random forest and boosting.

Synopsis

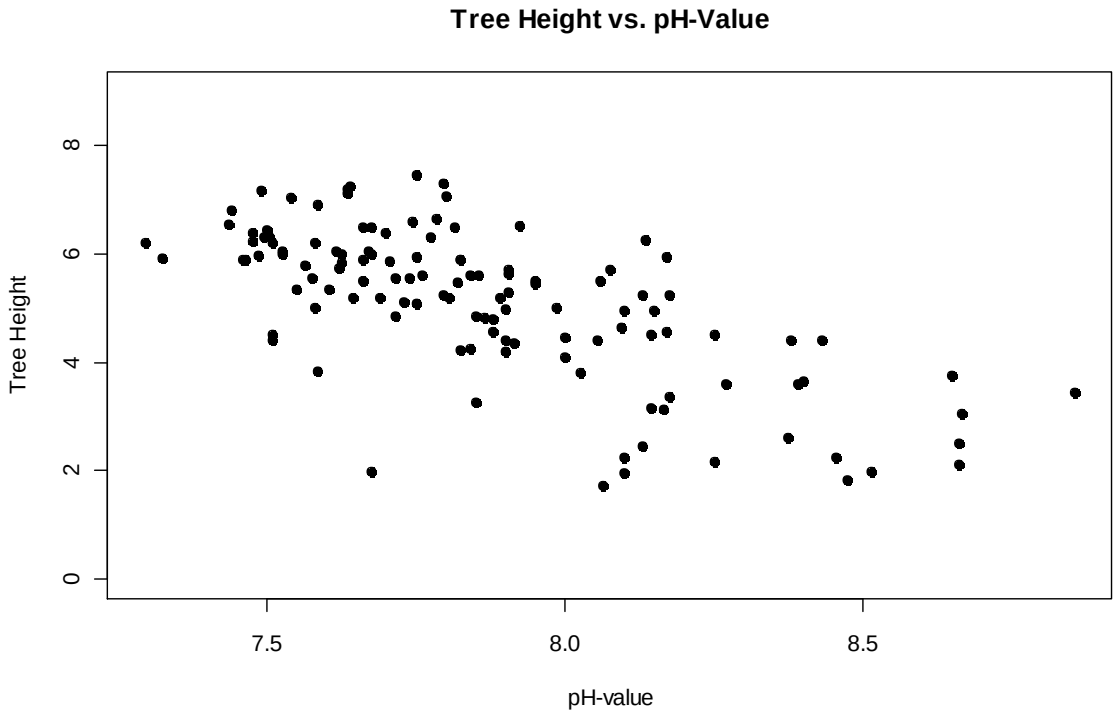
Finally, after naming a quite large number of techniques and situations, we will try to boil it down again to a few characteristics questions that are at the roots of every data analysis and regression. They include:

- Is a regression analysis the right way to go with my data?
- How do we estimate parameters and their confidence intervals?
- What assumptions are behind the fitted models, and are they met?
- Does my model fit? What can I improve it if that’s not the case?
- How can identify the “best” model, and how to choose it?

2 Simple Linear Regression

2.1 Introduction Example

In India, it was observed that alkaline soil (i.e. soil with high pH-value) hampers plant growth. This gave rise to a search for tree species which show high tolerance against these conditions. An outdoor trial was performed, where 123 trees of a particular species were planted on a big field with considerable soil pH-value variation. After 3 years of growth, every trees height Y_i was measured. Additionally, the pH-value of the soil in the vicinity of each tree was known and recorded as variable x_i. The best way to display the data is a scatter plot.



What could be the goal of an analysis? Well, all is targeted towards understanding the relation between pH-value and tree height. Thus, in the first place we would want to know how the tree height typically changes, when the pH-value increases by 1 unit. Moreover, it would be interesting to know whether there is a statistically significant relation between the two variables. Also of interest is the expected tree height, including a confidence interval, given the soil condition.

2.2 The Simple Linear Regression Model

The relation between an explanatory variable x and the response variable Y is, given a set of n observation, written as:

$$Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \text{, for all } i = 1,...,n.$$

The meaning of the quantities above is as follows:

Y_i is the *response* or *target variable* of the ith observation. In our example, this is the height of the ith tree. Note that the response is a random variable.

x_i is the *explanatory* or *predictor variable*, measured on the ith observation. In our example, it is the pH-value of the soil the tree grew on. The predictor is treated as a fixed, deterministic variable.

β_0, β_1 are unknown parameters, and are called *regression coefficients*. These are to be estimated by using the data points which are available. β_0 is called *intercept*, whereas β_1 is the *slope*. The latter indicates by how much the response changes, if the x-value is increased by 1 unit.

ε_i is the *random error term* or *residual*. It is a random variable, or more precisely, the random difference between the observed value y_i (which is the realization of a random variable) and the model value fitted by the regression.

Assumptions for this model

We always require zero expectation for the error term, i.e.

$$E[\varepsilon_i] = 0 .$$

This means that the relation between predictor and response is a linear function, or in our example: a straight line is the correct fit. Furthermore, we require constant variance of the error term, i.e.

$$Var(\varepsilon_i) = \sigma_\varepsilon^2 .$$

Finally, there must not be any correlation among the errors for different instances, which boils down to the fact that the observations do not influence each other, and that there are no hidden factors (e.g. time) that do so. In particular,

$$Cov(\varepsilon_i, \varepsilon_j) = 0 \text{ for all } i \neq j .$$

Also note that for now, there is no distributional assumption (e.g. Gaussian distribution) for the residuals. This model is called *simple* (or *univariate*) *linear regression*, since there is only one predictor.

However, it is important to perceive that we here talk about linear modeling not because we fit a straight line, but because the model equation is linear in the two parameters β_0 and β_1. Thus for example, also

$$Y_i = \beta_0 + \beta_1 x_i^2 + \varepsilon_i$$

is a linear regression model, because it is linear in the parameters, even though a parabola is fitted to the data points. On the other hand,

$$Y_i = \beta_0 + \beta_1 x_i^{\beta_2} + \varepsilon_i$$

is not a linear regression problem anymore, because it is not linear in the parameters.

2.3 Parameter Estimation

Parameter estimation means asking the question which line best fits through the n data pairs (x_i, y_i). For each data point, we consider the vertical difference to the regression line, i.e. the *residual*

$$r_i = y_i - (\beta_0 + \beta_1 x_i)$$

The regression line, and thus the parameters will be such that that the sum of squared residuals is minimal. This is known as the *least squares approach*.

The minimization problem can either be tackled by setting the partial derivatives to zero and solving for the parameters, or also by a geometrically motivated projection idea. Note that in both cases there is a very strong analogy to what is known as least squares adjustments in linear algebra. Also here, we need to solve the so-called normal equations to find the solution:

$$(X^T X)\beta = X^T y\,.$$

As long as the matrix X has full rank, which is the case for any "reasonable setup" with a non-singular design (in simple linear regression this is: we have at least two data points with different values for x_i), the least squares estimator is unique and can be written explicitly as

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \text{ and } \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}\,.$$

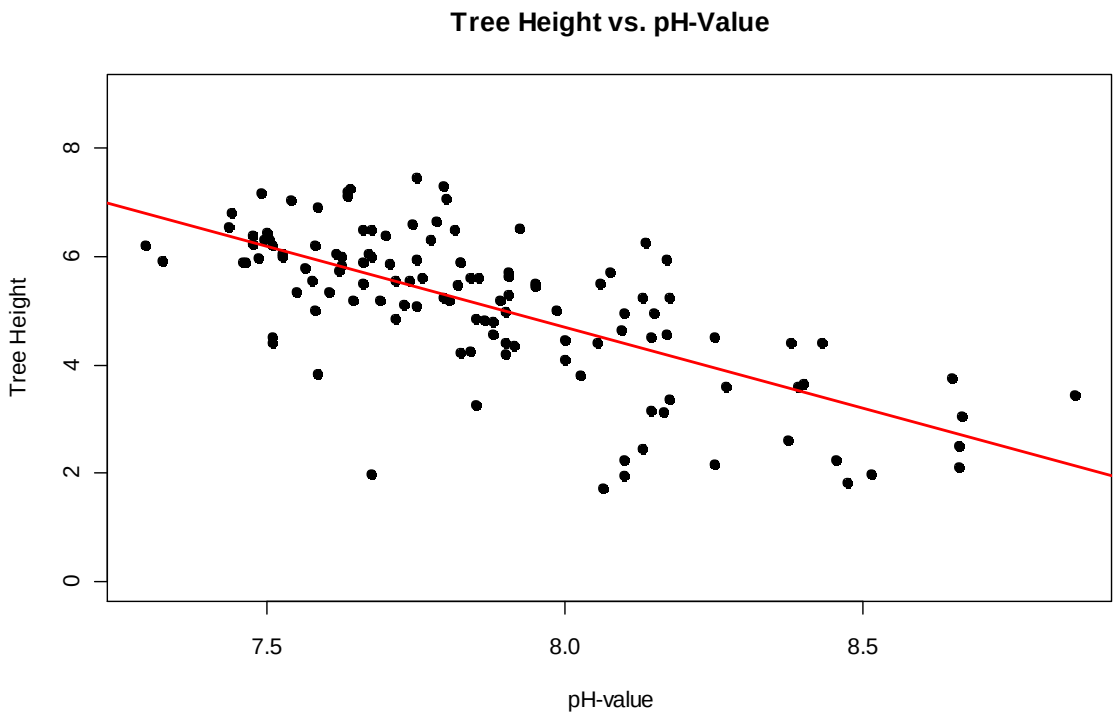
Using these estimated parameters, we obtain the regression line, defined as

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i \text{ for all } i = 1,...,n\,.$$

It can be visualized within the scatter plot, see the figure below. Here, $\hat{y}_i$ is the model value for the response of observation i, and is called *fitted* or *predicted* value. Please note again, that the residuals are the difference between fitted and observed values.

You may find it somewhat arbitrary that we chose the sum of squares residuals as the criterion to minimize. The reasons are mainly two-fold. First, this criterion results in a solution that can be written explicitly, and does not require sophisticated numerical optimization, which was important in a historical context. Moreover, we will see below that there is some mathematical justification ("optimality") for the use of least squares, especially if the errors have Gaussian distribution.

However, sometimes one also relies e.g. on minimizing the sum of absolute residuals, which is also known as L_1-regression. While it requires numerical optimization, the resulting procedure is more robust against outlying observations.



We turn our attention back to the least squares method and study some properties of the estimates. This also serves as further legitimating for minimizing the sum of squared residuals.

Gauss Markov Theorem

Under the model assumptions from section 2.1 (zero expected value and constant variance for the residuals, uncorrelated errors), the estimates $\hat{\beta}_0, \hat{\beta}_1$ are unbiased (i.e. $E[\hat{\beta}_0] = \beta_0$ and $E[\hat{\beta}_1] = \beta_1$). Moreover, they have minimal variance among all unbiased, linear estimators, meaning that they are most precise. It can be shown that:

$$Var(\hat{\beta}_0) = \sigma_\varepsilon^2 \cdot \left(\frac{1}{n} + \frac{\bar{x}}{\sum_{i=1}^{n} (x_i - \bar{x})^2} \right), \text{ and}$$

$$Var(\hat{\beta}_1) = \frac{\sigma_\varepsilon^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2}.$$

These results also show how a good experimental design can help to improve the quality of the estimates, or in other words, how we can obtain a more precisely determined regression line:

- we can rise the number of observations n.
- we have to make sure that the predictors x scatter well.
- by using a well-chosen predictor, we can keep σ_ε^2 small.
- for $\hat{\beta}_0$ it helps, if the average predictor value $\bar{x}$ is close to zero.

Estimation of σ_ε^2

Besides the regression coefficients, we also need to estimate the variance of the residuals. We require it for doing inference on the estimated parameters. The estimate is based on the *residual sum of squares* (abbreviation: RSS), in particular:

$$\hat{\sigma}_\varepsilon^2 = \frac{1}{n-2} \cdot \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

2.4 Inference on the Parameters

So far, we did not make any distributional assumptions. Let us remark again that we do not need them for proving the Gauss Markov Theorem, which holds independent of the error distribution. However now, we want to do inference on the estimated parameters, i.e. check if the predictor variable x has significant influence on the response Y. This is only possible by making a distributional assumption, hence we assert:

$\varepsilon_i \sim N(0, \sigma_\varepsilon^2)$, i.i.d..

A word of caution: please note that if one wants to rely on tests and confidence intervals for the parameters, the above assumptions (Gaussian distribution and independence) need to be met, and thus checked by using the methods that will be discussed in section 2.6 on residual diagnostics. If they are violated, one often draws false conclusions.

For finding out whether the predictor x has a statistically significant influence on the response variable Y, one tests the null hypothesis $H_0 : \beta_1 = 0$ against the alternative hypothesis $H_A : \beta_1 \neq 0$. As a test statistics, we use

$$T = \frac{\hat{\beta}_1 - E[\hat{\beta}_1]}{\sqrt{Var(\hat{\beta}_1)}} = \frac{\hat{\beta}_1 - 0}{\sqrt{\hat{\sigma}_\varepsilon^2 \Big/ \sum_{i=1}^{n} (x_i - \bar{x})^2}}.$$

It has a Student distribution with $n-2$ degrees of freedom, which can be used to determine acceptance and rejection regions, as well as the p-value. If one comes to the conclusion that the null hypothesis needs to be rejected, we have a statistically significant relation between predictor and response, i.e. the slope of

the regression line is significantly different from zero. For inferring the intercept, the procedure is analogous.

Output of Statistical Software Packages

When performing simple linear regression, one often relies on statistical software. The output looks similar, no matter what suite is used. We here show the output that is produced by R. It provides the points estimates for β_0, β_1 (column "Estimate"), as well as their standard deviations (column „*Std. Error*"), the value of the test statistic T (column „*t value*"), and the p-value for the respective null hypotheses (column „*Pr(>|t|)*").

```
> summary(fit)
Call:
lm(formula = height ~ ph, data = dat)

Residuals:
     Min       1Q   Median       3Q      Max
-3.70195 -0.54712  0.08745  0.66626  2.00330

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  28.7227     2.2395   12.82   <2e-16 ***
ph           -3.0034     0.2844  -10.56   <2e-16 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.008 on 121 degrees of freedom
Multiple R-squared: 0.4797,  Adjusted R-squared: 0.4754
F-statistic: 111.5 on 1 and 121 DF,  p-value: < 2.2e-16
```

Moreover, also the point estimate for σ_ε^2 is given („*Residual standard error*") with corresponding degrees of freedom $n-2$ („*degrees of freedom*"), from which one directly concludes on the number of observations that were present. Finally, *Multiple R-squared* is defined as

$$R^2 = \frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \in [0,1]$$

and shows the proportion of the total variance which has been explained by the predictor. In case of simple linear regression, the last line with the *F-statistic* does not provide any further value. Its meaning will be discussed in the context of multiple linear regression, see section 5.

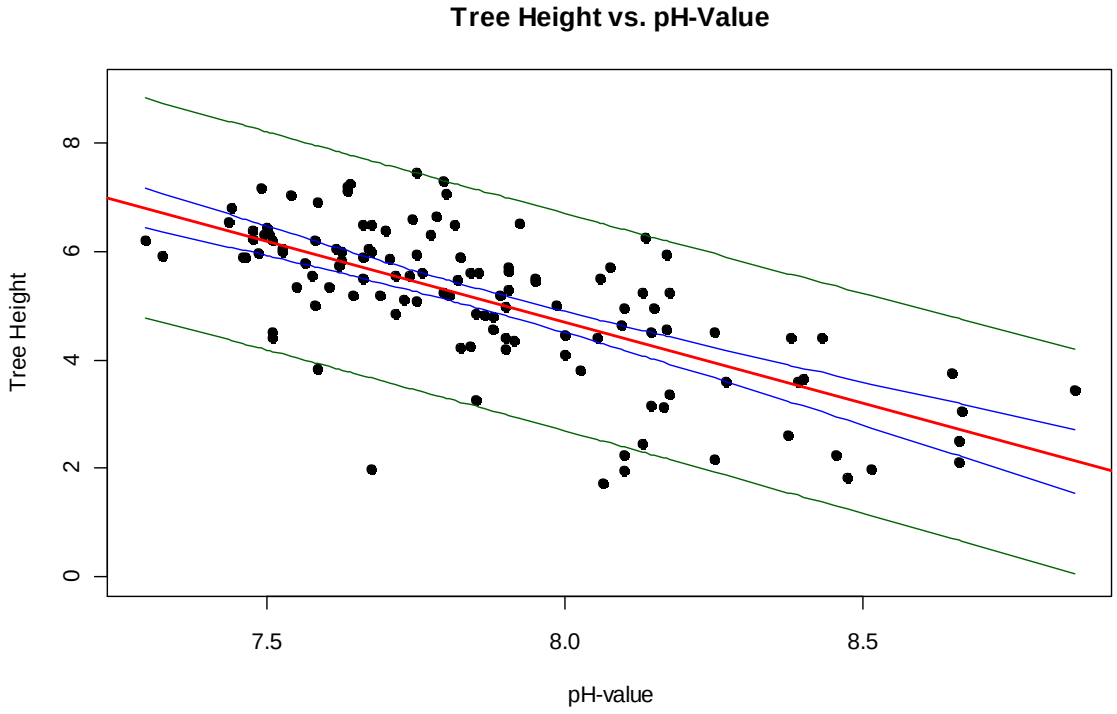
2.5 Prediction, Confidence and Prediction Intervals

The estimated parameters, i.e. the regression line can now be used for predicting the target value at an arbitrary (new) value x^*. We simply do as follows:

$$\hat{y}^* = \hat{\beta}_0 + \hat{\beta}_1 x^*$$

It is important to note that usually only a prediction within the range of x-values that were present for fitting is sensible. This is called interpolation. On the other hand, extrapolation, i.e. a prediction beyond the boundaries of the x-values that were present when fitting, has to be treated with great care.

Example: For a pH-value of 8.0, we expect a tree height of $28.7227 + (-3.0034 \cdot 8.0) = 4.4955$ units. However, it wouldn't be a good idea to use the regression line for predicting the tree height on soil with a pH-value of 5.0. It is very questionable that the relation we found on our data also holds for such acid ground.



We can now determine a 95% confidence interval for the predicted value $\hat{y}^*$. It is as follows:

$$\hat{\beta}_0 + \hat{\beta}_1 x^* \pm t_{0.975;n-2} \cdot \hat{\sigma}_\varepsilon \cdot \sqrt{\frac{1}{n} + \frac{(x^* - \bar{x})^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2}}$$

This confidence interval can be computed for arbitrary x^*, and can be displayed in the scatter plot (see above, in blue) as a confidence region for the fitted

regression line. This region is larger towards the boundaries of the present x-values, as it is easy to comprehend from the formula.

It is very important to note that the above confidence region does not tell us, to which height an additional tree place somewhere might grow. The reason is that (also within the training data), the true values scatter around their expected value. We can, however, derive a 95% prediction interval for y^*:

$$\hat{\beta}_0 + \hat{\beta}_1 x^* \pm t_{0.975;n-2} \cdot \hat{\sigma}_\varepsilon \cdot \sqrt{1 + \frac{1}{n} + \frac{(x^* - \bar{x})^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2}} \; .$$

Again, we can compute this interval for arbitrary x^* and display it in the scatter plot (see above, in green). It is clearly wider than the confidence region for the regression line.

2.6 Residual Diagnostics

After every regression fit, in order to avoid drawing any false conclusions, we need to check the model assumption stated under section 2.2, plus potentially the normality assumption. In summary, we have to check for:

- at least an approximately linear relation between x and Y, i.e. the expected value of the errors ε_i is zero over the entire x-range.
- the errors ε_i show constant variation σ_ε^2 and are (serially) uncorrelated.
- if tests and/or confidence/prediction intervals are computed, the errors ε_i also need to be normally distributed.

These checks are usually done by plotting the residuals r_i against various other variables. We here present the two most important plots, and refer to section 6 for further details on model diagnostics.

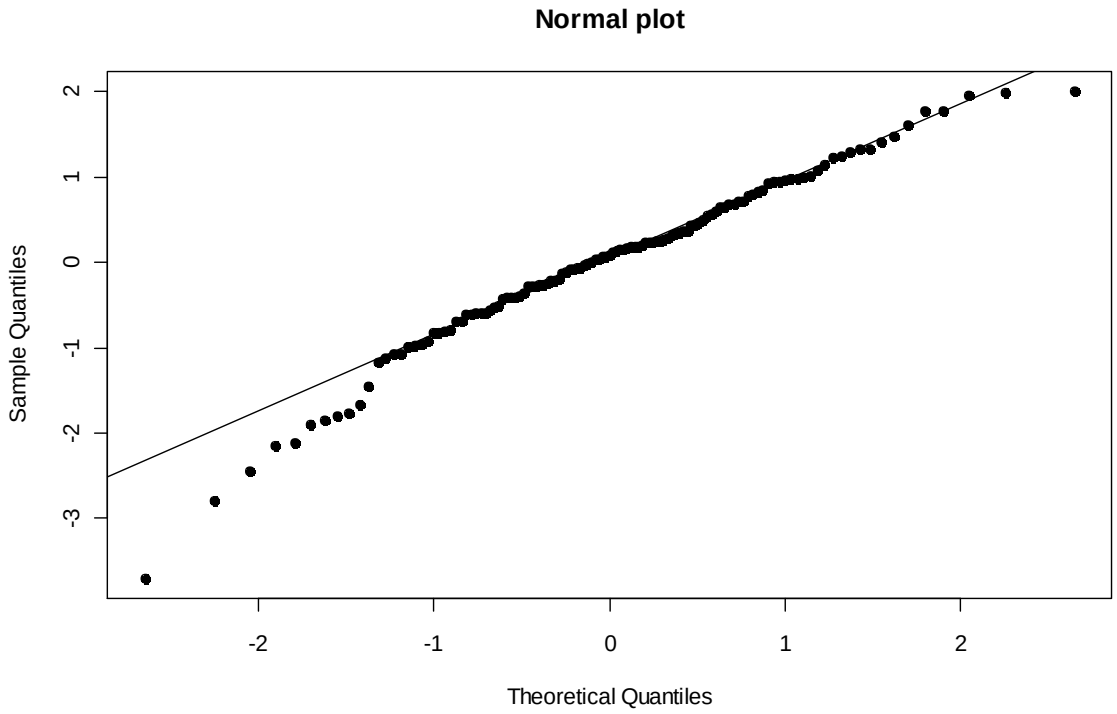
Normal Plot

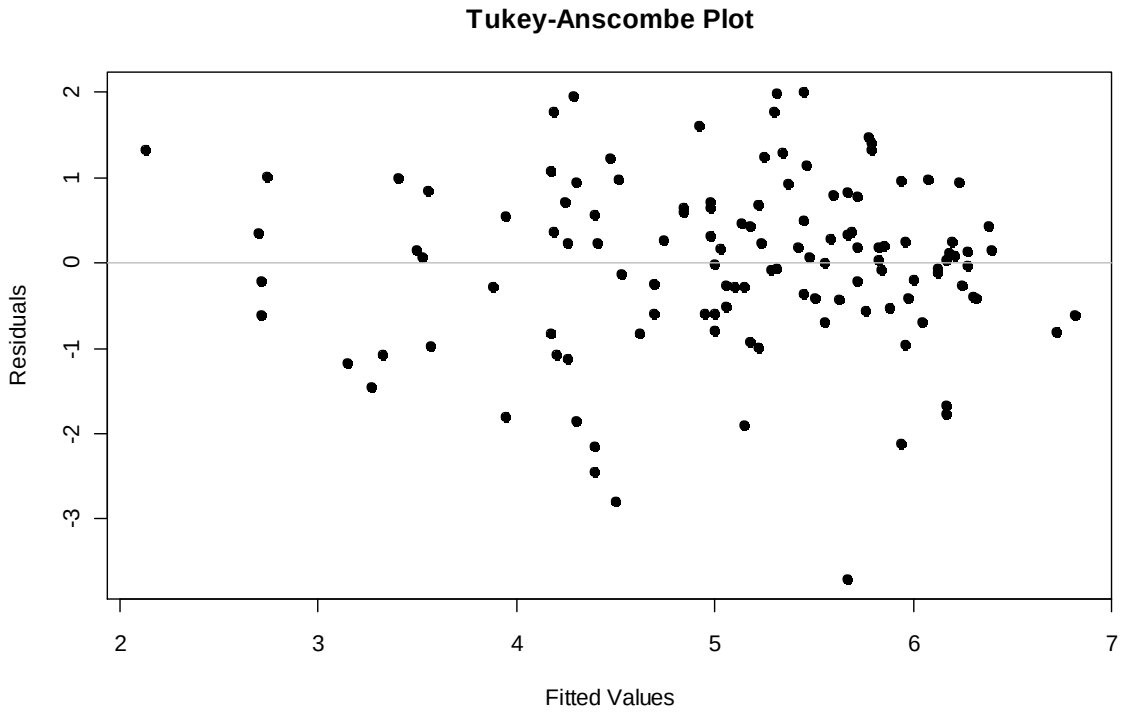
The assumption of normally distributed errors can be checked with the *normal plot*, i.e. we plot the ordered residuals against the corresponding quantiles of the Gaussian distribution. If the errors ε_i are normally distributed, then this also holds for the residuals r_i. Thus, the normal plot should (nearly) be a straight line.

Example: The normal plot of the tree growth problem (see plot below) shows a few negative residuals which are bigger than normally distributed errors would suggest. However, in real life things “never look perfect”. In this example, we would judge the normality assumption to be “reasonably fulfilled”, and would thus trust in test results and confidence intervals.

In cases where the distribution of residuals is skewed to the right, the situation may be improved by a square-root or log-transformation of the response variable.

If there are just a few large residuals (outliers), then we recommend checking whether these are caused by typing errors or other, unwanted influences. If that is the case, then the corresponding data points are either corrected, or omitted. For some further strategies in dealing with systematic deviations from the normality assumption, we refer to the later chapters of this scriptum.





Tukey-Anscombe-Plot

With this popular plot, violations of the zero expected value and constant variance conditions can be unveiled at a glance. On the x-axis, one plots the fitted values, whereas the y-axis shows the residuals. The optimal case is when all residuals fall within a horizontal layer of constant width, and show random scatter. On the other hand, any non-random structure that is visible should be treated as suspicious for a model violation.

Example: The Tukey-Anscombe-Plot (see below) in the tree growth example shows no gross violations of the model assumptions. An outlier with negative residual value is apparent, though. It is advisable to check this instance for typos or other irregularities. If nothing is found, there is no reason to be overly worried here, and the data point can be kept in the model.

In cases of non-constant variance, a transformation (usually of the response variable) may help to improve the situation. An alternative can be to use weighted regression, see section 6.6. If the Tukey-Anscombe-Plot shows a non-linear relation between predictor and response, it may be that a transformation clears the problem, or additional predictors need to be incorporated into the model.

2.7 Erroneous Input Variables

There are cases where the predictor variable is not deterministic, but like the response, subject to random error, too. We then write

$Y_i = \eta_i + \varepsilon_i$, for all $i = 1,...n$, with $E[\varepsilon_i] = 0$ and $Var(\varepsilon_i) = \sigma_\varepsilon^2$,

$X_i = \xi_i + \delta_i$, for all $i = 1,...n$, with $E[\delta_i] = 0$ and $Var(\delta_i) = \sigma_\delta^2$.

We then have a linear relation $\eta_i = \beta_0 + \beta_1 \xi_i$, but η_i and ξ_i cannot be observed. Such variables are called latent variables, and only X_i and Y_i can be observed. We can shift the terms around and obtain:

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i - \beta_1 \delta_i$$

If we estimate the slope with the usual least squares estimator in this case, then we generally will not obtain an unbiased estimate for β_1. Under some mild conditions (not shown here), we have:

$$E[\hat{\beta}_1] = \beta_1 \cdot \frac{1}{\left(1 + \sigma_\delta^2 / \sigma_\xi^2\right)}, \text{ where } \sigma_\xi^2 = \frac{1}{n} \cdot \sum (\xi_i - \bar{\xi})$$

From the above formula we conjecture, that the estimate for β_1 is unbiased only if $\sigma_\delta^2 = 0$. However, if σ_δ^2 is small when compared to σ_ξ^2, i.e. if the errors in observing the X's is small compared to the scatter of the X-values, then the bias can be neglected and we would still use the least squares approach. In all other cases, we refer to the work of Draper (1992, Straight line regression when both variables are subject to error).

However, if the goal in regression analysis is not inference but “only” prediction, then errors in the explanatory variables might be ignored altogether. The reason for this is that the predicted values are unbiased, as long as the error structure on the input variables does not change.

3 Multiple Linear Regression

3.1 Introduction and Example

Often, the response variable is, or may be, influenced by various predictors at a time. In our previous tree growth example, such multiple predictors could potentially be other properties of the soil besides the pH-value, the amount of water that the tree was drained with, etc.

What to do with such cases, where multiple predictor variables are available? The poor man's approach would be to do many simple linear regressions on each of the predictors separately. This has the somewhat doubtful advantage that the relation between each predictor and the response can be displayed by a 2d-scatter plot. However, and this is really important, doing many simple regressions is clearly not advisable, the reason is explained just below.

The appropriate tool to include the effects of more than one predictor on a response variable is *multiple linear regression*. Geometrically spoken, it fits the least-squares hyper plane in the $(p+1)$-dimensional space, where p is the number of predictors that are present. Generally, this fit cannot be visualized when $p > 2$.

It is important to note that doing many simple regressions is not equivalent to a multiple regression. The results will generally be different, i.e. they are only identical, if the predictor variables are orthogonal – and this is almost never the case with data from observational studies.

Example

The chapter on multiple linear regression will be illustrated with the *mortality dataset*. Researchers at General Motors collected data on 59 US Standard Metropolitan Statistical Areas in a study of whether air pollution contributes to mortality. The data include predictors measuring demographic characteristics of the cities, predictors measuring climate characteristics, and finally predictors recording the concentration of three different air pollutants in the ambient air: hydrocarbons (HC), nitrous oxide (NO_x), and sulfur dioxide (SO_2).

3.2 The Multiple Linear Regression Model

The multiple linear regression model specifies the relation between the response variable Y and the predictor variables $x_1, ..., x_p$. We assume that we are given n instances, where response and predictors were observed. We then write the model as:

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip} + \varepsilon_i \text{, for all } i = 1, ..., n.$$

As we had explained before, $\beta_1, ..., \beta_p$ are the unknown regression parameters. It will be our goal to estimate these from the data. Again, ε_i is the error term, on

which we make same assumptions as in simple linear regression. However, we restate them here:

$$E[\varepsilon_i] = 0 .$$

Again this means that the relation between predictors and response is a linear function, or in other words: the hyper plane is the correct fit. Furthermore, we require constant variance of the error term, i.e.

$$Var(\varepsilon_i) = \sigma_\varepsilon^2 .$$

Finally, there must not be any correlation among the errors for different instances, which boils down to the fact that the observations do not influence each other, and that there are no hidden factors (e.g. time) that do so. In particular,

$$Cov(\varepsilon_i, \varepsilon_j) = 0 \text{ for all } i \neq j .$$

As in simple linear regression, we do not require any specific distribution for parameter estimation and certain optimality results of the least squares approach. The distributional assumption only comes into play when we do inference on the parameters.

Example

We turn back our attention to the mortality dataset. While there are more predictor variables, we first focus on only three of them, plus the response. As stated above, the data from 59 cities are available:

Y_i Mortality rate, i.e. number of deaths per 100'000 people, in city i.

x_{i1} Average SO_2 concentration in city i.

x_{i2} Percentage of non-white population in city i.

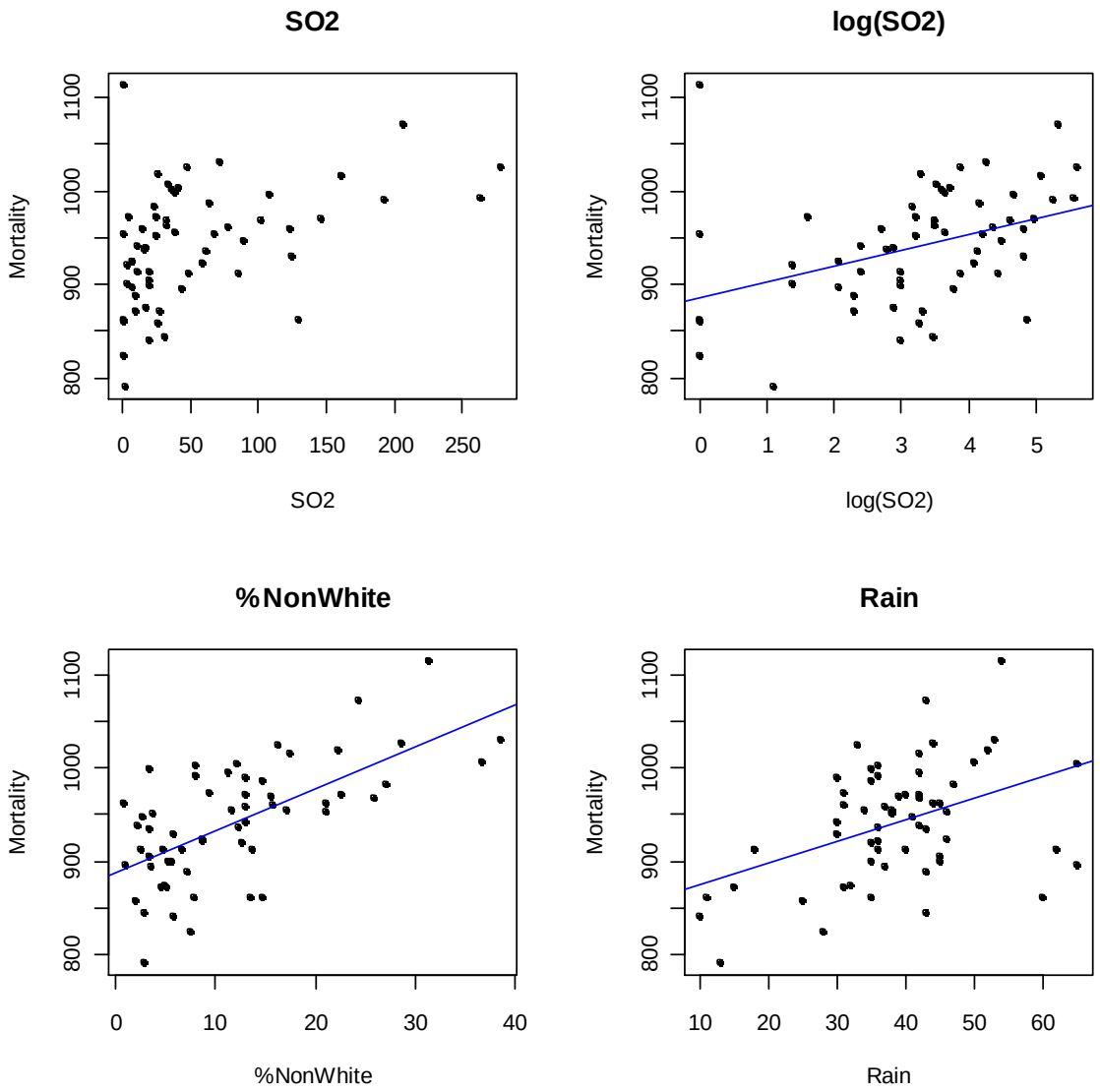
x_{i3} Average yearly precipitation in inches, in city i.

The plot below shows scatter plots of the response versus each of the predictors, together with the fit from a simple linear regression. Since the SO_2-values show a skewed distribution, and because the relation to the mortality rate does not seem very linear, we apply a log-transform on them. This improves the situation. The equations for the simple linear regressions are as follows:

$\log(SO_2)$: $\hat{y} = 886.34 + 16.86 \cdot \log(SO_2)$

$NonWhite$: $\hat{y} = 887.90 + 4.49 \cdot NonWhite$

$Rain$: $\hat{y} = 851.22 + 2.34 \cdot Rain$



However, as we have learned, we must fit a multiple linear regression in this case, where 3 predictor variables are available. The R code for doing so is as follows:

```
> lm(Mortality ~ log(SO2) + NonWhite + Rain, data=mortality)

Coefficients:

(Intercept)      log(SO2)      NonWhite          Rain
    773.020        17.502         3.649         1.763
```

We observe that as in simple linear regression, the function `lm()` does the job. The response variable is written first on the left hand side, and after the tilde, we list the predictors, which are separated by a '+'. Finally, we have to specify the data frame where the variables are taken from. We obtain the coefficient estimates, and thus, the regression equation:

$$\hat{y} = 773.020 + 17.502 \cdot \log(SO_2) + 3.649 \cdot NonWhite + 1.763 \cdot Rain$$

As blatantly claimed in the introduction, the parameters $\hat{\beta}_j$ from the simple regressions are not equal to the ones from multiple regression. The differences are not even that prominent here, but note that they can be arbitrarily big. We now turn our attention to the question what the meaning of the coefficients is in the case of multiple linear regression?

> *The regression coefficient $\hat{\beta}_j$ is the increase in the response Y, if the predictor x_j increases by 1 unit, but all other predictors remain unchanged.*

3.3 Matrix Notation

Multiple linear regression is much easier to comprehend when the matrix notation is used. We can write the model very simply as

$$Y = X\beta + \varepsilon .$$

The elements in this equation are as follows:

- Y is a $(n \times 1)$ column vector that holds the responses for all n cases.
- X is the design matrix with dimension $(n \times (p+1))$. Each column of X holds a predictor variable, with all its observations on the n cases. The first column is special. It consists of 1 only, and it is there such that we have an intercept in the model
- β is a $((p+1) \times 1)$ column vector that holds the regression coefficients. Note that these are unknown, and it is the goal to estimate these from the data we have observed.
- ε is a $(n \times 1)$ column vector with the errors. Also the errors are unobservable, they will be estimated by the residuals, i.e. the difference between the observed and fitted values. For the error terms, we assume that $E[\varepsilon] = 0$ and $Cov(\varepsilon) = \sigma^2 I$.

4 Estimation with Multiple Linear Regression Models

While we did already fit a multiple regression model to the mortality data in the example above, we did not specify how to do this yet. This will be the content of this section. We will also discuss the properties of the estimates, and some problems that can arise during the estimation process.

4.1 Least Squares Approach and Normal Equations

As in simple linear regression, we will again estimate the regression coefficients by the least squares approach. Thus, we have to determine the residuals

$$r_i = y_i - (\beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip}) .$$

Then, we choose the parameters $\beta_0, ..., \beta_p$ such that the sum of squared residuals

$$\sum_{i=1}^{n} r_i^2$$

is minimal. This problem can be tackled by taking partial derivatives and setting them to zero. This again results in the so-called *normal equations*. We do now take full advantage of the matrix notation that was introduced above in section 3.3 and can thus write them as

$$(X^T X)\beta = X^T y .$$

If $X^T X$ is regular, we can obtain the least squares estimates of the regression coefficients by some simple matrix calculus as

$$\hat{\beta} = (X^T X)^{-1} \cdot X^T y .$$

As long as the regularity condition for $X^T X$ is fulfilled, there is a unique and explicit solution for the regression coefficients $\hat{\beta}$, and thus no numerical optimization is needed. A side remark: in software packages, the inverse of $X^T X$ is usually not computed for numerical reasons, but the computations will be based on a QR- or similar decompositions of $X^T X$.

4.2 Identifiability

We claimed above that the normal equations have a unique solution if and only if $X^T X$ is regular and thus invertible. This is the case if X has full rank, i.e. all columns of that matrix, or in other words, all predictor variables are linearly independent. This is the standard case, and whenever the full rank condition for X is fulfilled, we are fine.

On the other hand, there will also be cases where X does not have full rank and $X^T X$ is singular. Then, there usually are infinitely many solutions. Is this a problem? And how does it occur? The answer to the first question is “yes”. When the design matrix X does not have full rank, the model is “badly formulated”, such

that the regression coefficients β are at least partially unidentifiable. It is mandatory to improve the design, in order to obtain a unique solution, and regression coefficients with a clear meaning. How can it happen?

1) **Duplicated variables**

   It could be that we use a person's height both in meters and centimeters as a predictor. This information is redundant, and the two variables are linearly dependent. One thus has to remove one of the two.

2) **Circular variables**

   Another example is when the number of years of pre-university education, the number of years of university education and also the total number of years of education are recorded and included in the model. These predictors will be linearly dependent, thus X does not have full rank.

3) **More predictors than cases**

   Note that a necessary (but not sufficient) condition for the regularity of $X^T X$ is $p < n$. Thus, we need more observations than we have predictors! This makes sense, because the regression is over-parameterized (or super-saturated) else and will not have a (unique) solution.

What does R do in non-identifiable problems?

Generally, statistics packages handle non-identifiability differently. Some may return error messages; some may even fit models because rounding errors kill the exact linear dependence. R handles this a bit different: it recognizes unidentifiable models and fits the largest identifiable one by removing the excess predictors in reverse order of appearance in the model formula. The removed predictors will still appear in the summary, but all their values are `NA`, and a message also says "`Coefficients: k not defined because of singularities`"). While this still results in a fit, it is generally better in such cases to rethink the formulation of the regression problem, and remove the non-needed predictors manually.

4.3 Properties of the Least Squares Estimates

What are the properties of the least squares estimates, in cases where there is a unique solution? Well, the Gauss-Markov-Theorem from simple linear regression (see section 2.3) also holds here, under the general conditions stated at the beginning of section 3.2. It tells us that the regression coefficients are unbiased estimates, and they fulfill the optimality condition of minimal variance among all linear, unbiased estimators. In particular, we have:

$$E[\hat{\beta}] = \beta \text{ and } Cov(\beta) = \sigma_\varepsilon^2 \cdot (X^T X)^{-1},$$

As in simple linear regression, the precision of the regression coefficients depends on the design and the number of observations.

4.4 Estimating the Error Variance σ_ε^2

An unbiased estimate for the unknown error variance σ_ε^2 can be obtained by standardizing the sum of squared residuals with the appropriate degrees of freedom, which is the number of observations n minus the number of estimated parameters. With p predictors variables and an intercept, this number of estimated parameters is $p+1$, and the error variance estimate is:

$$\hat{\sigma}_\varepsilon^2 = \frac{1}{n-(p+1)} \sum_{i=1}^{n} r_i^2 \ .$$

4.5 The Hat Matrix H

We will now take further advantage of the matrix notation and the estimated regression coefficient. They allow us to write the fitted values $\hat{y}$ very simply:

$$\hat{y} = X\hat{\beta}$$

We now do some further calculus and plug-in the solution for $\hat{\beta}$ from above. We then observe that the fitted values $\hat{y}$ are obtained by multiplying the *hat matrix* H, with the observed response values Y:

$$\hat{y} = X\hat{\beta} = X(X^T X)^{-1} X^T Y = HY$$

The matrix H is called hat matrix, because "it puts a hat on the Y's", i.e. transforms the observed values into fitted values. We can also use this matrix for computing the residuals:

$$r = Y - \hat{Y} = (I - H)Y$$

If we compute expected value and variance in the two formulas above, then, regarding the fact that the predictors X are fixed, non-random values, we obtain:

$E[\hat{y}] = y$ and $E[r] = 0$, respectively

$Var(\hat{y}) = \sigma_\varepsilon^2 H$ and $Var(r) = \sigma_\varepsilon^2 (I - H)$.

This shows to us that the residuals r_i, which are estimates of the unobservable errors ε_i, have zero expectation, but usually do not have equal variance. Moreover, they are usually correlated. Note that this is fundamentally different from the assumption we imposed on the errors, where we required equal variance and no correlation.

4.6 Additional Properties under Gaussian Distribution

While all of the above statements hold for arbitrary error distribution, we obtain some more, very useful properties by assuming i.i.d. Gaussian errors. Then, and only then, the estimators for the regression coefficients will have a Normal distribution:

$$\hat{\beta} \sim N\left(\beta, \sigma_\varepsilon^2 (X^T X)^{-1}\right)$$

When doing inference, i.e. performing hypothesis tests and computing confidence intervals, one routinely assumes Gaussian errors (as we also did for inference in simple linear regression) and makes use of the above result. Under Gaussian errors, also the distribution of the fitted values and the error variance estimate is known:

$$\hat{y} \sim N(X\beta, \sigma_\varepsilon^2 H)$$

$$\hat{\sigma}_\varepsilon^2 \sim \frac{\sigma_\varepsilon^2}{n-p} \chi_{n-p}$$

In practice, the normality assumption of the errors ε_i needs to be carefully checked. We refer to section 6 for how to do this. But what to do if the assumption is not (well) fulfilled? For very large number of observations n, we can rely on the central limit theorem, which tells us that the result of normally distributed parameters will still approximately hold for large sample sizes n.

This is the usual justification in practice to use the above formulae for constructing confidence intervals and tests for the regression coefficients. However, while small deviations from normality may be tolerable for large sample sizes, it is often much better and safer to use robust methods (see section **Fehler! Verweisquelle konnte nicht gefunden werden.**) in case of clearly non-Gaussian errors.

5 Inference with Multiple Linear Regression Models

If we assume normally distributed errors, we have seen above that also the regression coefficients have a joint Gaussian distribution, and thus also marginals. We make use of this for computing confidence intervals and performing hypothesis tests.

5.1 Individual Parameter Tests

If we are interested whether the j^{th} predictor variable is relevant, we can test the hypothesis $H_0 : \beta_j = 0$ against the alternative hypothesis $H_A : \beta_j \neq 0$. We can the easily derive from the normal distribution that

$$\frac{\hat{\beta}_j}{\sqrt{\sigma_\varepsilon^2 (X^T X)^{-1}_{jj}}} \sim N(0,1)$$

Since σ_ε^2 is unknown, this quantity is not useful. However, if we substitute the unknown error variance with the estimate $\hat{\sigma}_\varepsilon^2$, we obtain the so-called t-test statistic

$$T = \frac{\hat{\beta}_j}{\sqrt{\hat{\sigma}_\varepsilon^2 (X^T X)^{-1}_{jj}}} \sim t_{n-(p+1)},$$

which has a slightly different distribution than the standard Normal. The present Student distribution with $n-(p+1)$ degrees of freedom has a bit more mass in the tails, this is to account for the effect of the estimated parameters which are used for standardization.

In practice, we can now quantify the relevance of each individual predictor variable by looking at its test statistic, or the corresponding p-value. Note that the latter is usually more informative. However, there are 2 problems which arise:

1) The *multiple testing problem*: if we repeatedly do hypothesis testing on the α=5% significance level, our total type II error (i.e. at least one of the tested hypotheses is falsely rejected) increases. In particular, for p hypothesis tests, it is $1-(1-\alpha)^p$.

2) It can happen that all individual tests do not reject the null hypothesis (say at the 5% significance level), although it is in fact true that some predictor variables have a significant effect on the response. This paradoxon can occur because of correlation among predictor variables.

Finally, we come to the interpretation of an individual parameter test: it quantifies the effect of the predictor x_j on the response Y after having subtracted the linear effect of all other predictor variables on Y. This is different from the corresponding test in a simple linear regression, which infers the isolated one-to-one relation between x_j and Y.

5.2 Global F-Test

Another question which is of major interest in multiple linear regression analysis is whether there is *any* relation between predictors and response. This can be formulated with the null hypothesis

$$H_0 : \beta_1 = \beta_2 = ... = \beta_p = 0 ,$$

which is tested against the alternative hypothesis that at least one regression coefficient is different from zero:

$$H_A : \beta_j \neq 0 \text{ for at least one } j \in \{1, 2, ..., p\} .$$

A test statistic can be developed by using the so-called analysis of variance (ANOVA) table, which decomposes the total scatter of the Y-values around the global mean into a first portion explained by the regression, and a second which remains with the residuals. Under the global null hypothesis of no predictor influence, the first portion cancels out. If we divide the total scatter (the left hand side of the equation) by $\hat{\sigma}_\varepsilon^2$, we obtain a scale-free quantity that serves as a test statistic for the global null:

$$F = \frac{n-(p+1)}{p} \cdot \frac{\sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \sim F_{p, n-(p+1)}$$

Under the null, F has a F-distribution with p and $n-(p+1)$ degrees of freedom. We can use it for computing the p-value.

5.3 Coefficient of Determination

The coefficient of determination, also called *multiple R-squared*, is aimed at describing the goodness-of-fit of the multiple linear regression model. It is defined exactly as it was in simple linear regression:

$$R^2 = \frac{\sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \in [0,1] ,$$

and still shows the proportion of the total variance which has been explained by the predictors. We have $R^2 = 0$ if $\beta_j = 0$ for all $j = 1, ..., p$, and $R^2 = 1$ if the fit is perfect, i.e. all residuals are equal to zero.

As a goodness-of-fit measure, the coefficient of determination needs to be taken with a grain of salt. If we add more and more predictor variables to the model, it can only increase, but never decreases. However, adding more and more variables to the model will always improve the fit on the present training dataset,

but may lead to an increased generalization error (see section **Fehler! Verweisquelle konnte nicht gefunden werden.** for further reference on this topic). Thus, one often considers an adjusted coefficient of determination, which is also found in the R-output:

$$adjR^2 = 1 - \frac{n-1}{n-(p+1)} \cdot \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \in [0,1].$$

5.4 Confidence and Prediction Intervals

One more thing that we did not discuss yet is the construction of a confidence interval for the expected value of Y, as well as a prediction interval for a future observation with given predictor values $x_1^*, ..., x_p^*$. In section 2.5, we did so for simple linear regression, and we could also neatly visualize these intervals.

Note that we can still compute the intervals for multiple linear regression fits, but we cannot display them anymore. The reason is just that we now work in a high-dimensional space. In spirit and interpretation, however, the intervals are equivalent with what we had in simple linear regression.

A 95% confidence interval for $E[Y^*]$ is given by

$$\hat{y}^* \pm t_{0.975;n-(p+1)} \cdot \hat{\sigma}_\varepsilon \cdot \sqrt{\left(x^*\right)^T \left(X^T X\right)^{-1} \left(x^*\right)},$$

where $(x^*)^T = (1, x_1^*, ..., x_p^*)$ is the predictor vector. The 95% prediction interval for a future observation with such predictor values is then given by:

$$\hat{y}^* \pm t_{0.975;n-(p+1)} \cdot \hat{\sigma}_\varepsilon \cdot \sqrt{1 + \left(x^*\right)^T \left(X^T X\right)^{-1} \left(x^*\right)}.$$

5.5 R-Output

All the quantities that were discussed in this section on inference and the previous on estimation are returned when the `summary()` function is applied on a linear model fit:

```
> summary(lm(Mortality~log(SO2)+NonWhite+Rain, data=mort…))

Residuals:
     Min       1Q   Median       3Q      Max
  -76.04   -24.45     0.58    22.59   130.33
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 773.0197    22.1852  34.844  < 2e-16 ***
log(SO2)     17.5019     3.5255   4.964 7.03e-06 ***
NonWhite      3.6493     0.5910   6.175 8.38e-08 ***
Rain          1.7635     0.4628   3.811 0.000352 ***
---
Residual standard error: 38.4 on 55 degrees of freedom
Multiple R-squared: 0.641,  Adjusted R-squared: 0.6214
F-statistic: 32.73 on 3 and 55 DF,  p-value: 2.834e-12
```

As for simple linear regression, the R output provides the point estimates for $\beta_0, \beta_1, ..., \beta_p$ (column "Estimate"), as well as their standard deviations (column „*Std. Error*"), the value of the test statistic T (column „*t value*"), and the p-value for the respective null hypotheses (column „*Pr(>|t|)*").

Moreover, also the point estimate for σ_ε^2 is given („*Residual standard error*") with corresponding degrees of freedom $n-(p+1)$ („*degrees of freedom*"), from which one directly concludes on the number of observations that were present (here: 59 observations). Finally, the result for the global F-test is presented, too.

5.6 Example and Fitting in R

We observe that for our mortality example, all three individual parameter tests, as well as the global F-test show very small p-values, and are thus highly statistically significant. Can we thus conjecture that the logged SO_2 concentration really affects the mortality rate?

The answer is: not quite. And the reason is: there are only 3 predictors, and there may be a confounding effect. However, the more (statistically significant) predictor variables are present in the model, the stronger the evidence for a causal relation between a single predictor and the response gets, since the low observed p-value is obtained under the presence of all other variables. Thus, we will now add some more predictors for explaining the mortality:

JanTemp	Average temperature in January (in F)
JulyTemp	Average temperature in July (in F)
RelHum	Average relative humidity, measured daily at 1pm
Rain	Average yearly rainfall, in inches
Educ	Median of the years of school a person visited, in years
Dens	Population Density per Square Mile
NonWhite	Percentage of non white population
WhiteCollar	Percentage of white collar workers
Pop	Number of inhabitants in the city
House	Average number of persons per household
Income	Median income

We now fit this extended model and obtain the following R output:

```
> summary(lm(Mortality ~ JanTemp + JulyTemp + RelHum + Rain +
             Educ + Dens + NonWhite + WhiteCollar + Pop +
             House + Income + log(SO2), data=mortality))

Residuals:
    Min      1Q  Median      3Q     Max
 -70.92  -20.94   -2.77   18.86  105.93

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.164e+03  2.939e+02   3.960 0.000258 ***
JanTemp     -1.669e+00  7.930e-01  -2.105 0.040790 *
JulyTemp    -1.167e+00  1.939e+00  -0.602 0.550207
RelHum       7.017e-01  1.105e+00   0.635 0.528644
Rain         1.224e+00  5.490e-01   2.229 0.030742 *
Educ        -1.108e+01  9.449e+00  -1.173 0.246981
Dens         5.623e-03  4.482e-03   1.255 0.215940
NonWhite     5.080e+00  1.012e+00   5.019 8.25e-06 ***
WhiteCollar -1.925e+00  1.264e+00  -1.523 0.134623
Pop          2.071e-06  4.053e-06   0.511 0.611799
House       -2.216e+01  4.040e+01  -0.548 0.586074
Income       2.430e-04  1.328e-03   0.183 0.855617
log(SO2)     6.833e+00  5.426e+00   1.259 0.214262
---
Residual standard error:  36.2 on 46 degrees of freedom
Multiple R-squared: 0.7333,  Adjusted R-squared: 0.6637
F-statistic: 10.54 on 12 and 46 DF,  p-value: 1.417e-09
```

When we add more predictors, we observe that the logged SO_2 concentration with a p-value of 0.214 is no longer significant. Do we now have to face the fact that ambient air pollution has no effect on mortality? We should not be too quick with such a conjecture, and do some model diagnostics first (see section 6)

Additionally, collinearity, i.e. correlation among predictor variables can hamper interpretation even further. Note that if two predictors x_1 and x_2 are uncorrelated, then the estimated regression coefficients $\hat{\beta}_1$ and $\hat{\beta}_2$ remain the same, no matter whether only one of the two, or both variables are included in the model. For collinear predictors, this unfortunately is not the case.

With collinear predictors, it can also happen that the global F-test shows a highly significant result, while all the individual parameter tests are not even rejected. The reason is that one single variable does not add much if all the others are already included in the model. The ensemble, however, still has an effect on the response.

Thus, one may conjecture that uncorrelated predictors are preferable. This is true. However, while this may be achieved in designed experiments, it will almost never be the case with observational studies. There, we have to live with collinear input variables. The only thing we can do is to check the “amount of collinearity” in our data. This is done by a multiple linear regression of all remaining predictors on x_j,

and computing the respective coefficient of determination R_j^2. Instead of interpreting this quantity, one often regards the so-called variance inflation factor:

$$VIF_j = \frac{1}{1-R_j^2}.$$

As a rule of the thumb, a $VIF > 10$ is dangerous. It means that inference (i.e. interpreting p-values from individual parameter tests and the global F-test) should be "handled with care", and drawing conclusions on causality should be left out. However, the fitted values are not affected by this, and also prediction with a model fitted from collinear predictors is always fine.

5.7 Partial F-Tests

So far, we discussed individual parameter tests, as well as the global F-test. Thus, we either infer the influence of only one single predictor at a time or of all p predictors simultaneously. The question is whether we could also check if a group of predictors has a significant effect on the response. In our mortality example, we could e.g. ask the question, whether the subset of *all* meteorological variables has a significant effect on the response.

Thus, our goal is to test the effect of $p-q$ predictors simultaneously. For doing so, we partition the parameter vector β and the design matrix X into two parts each:

$$\beta = \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_q \\ \beta_{q+1} \\ \vdots \\ \beta_p \end{pmatrix}, \text{ and } X = [X_1 \; X_2].$$

Here, the dimensions are $n \times (q+1)$ for X_1 and $n \times (p-q)$ for X_2. We can the rewrite the model as:

$$Y = X\beta + \varepsilon = X_1\beta_1 + X_2\beta_2 + \varepsilon.$$

We want to infer whether the collective subset of $x_{q+1}, ..., x_p$ has an influence on the response. This leads to the null and alternative hypotheses:

$$H_0 : \beta_2 = 0 \text{ versus } H_A : \beta_2 \neq 0.$$

In words, the null hypothesis means "the last $p-q$ predictors in my model do not have an effect on the response", whereas the alternative is "at least one of the last $p-q$ predictors is meaningful".

In fact, we do perform and compare two multiple linear regression analyses. The one under the alternative hypothesis H_A is including the full set of p predictors, whereas the one under the null hypothesis H_0 is with the reduced set of only the first q predictors.

Naturally, if the differences in the quality of the two fits are small, we would prefer the smaller model, while a large difference would speak for the larger model. Thus, a test statistics could be based on the difference in the residual sum of squares (RSS) between the two models, relative to the RSS from the large model:

$$\frac{RSS_{H_0} - RSS_{H_A}}{RSS_{H_A}}, \text{ where } RSS = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

with $\hat{y}_i$ from the respective model. While this is almost it, we also need to take the number of observations as well as the difference in the number of predictors into account for a formal test. It can be shown that:

$$F = \frac{n-(p+1)}{p-q} \cdot \frac{RSS_{H_0} - RSS_{H_A}}{RSS_{H_A}} \sim F_{p-q,n-(p+1)}.$$

Thus, the relative difference in the residual sum of squares needs to be multiplied with the degrees of freedom of the large model, and divided by the difference in the number of predictors. When this is small, we cannot reject the null hypothesis, and the small model is appropriate.

Indeed the test statistic has an F-distribution with $p-q$ and $n-(p+1)$ degrees of freedom. If the realized value F exceeds the 95th percentile of that distribution, the null hypothesis is rejected.

Example

Using the above methodology, we can now test whether the subset of meteorological variables “jantemp”, “julytemp”, “relhum” and “rain” affect the mortality as a collective. We obtain a test statistic of $F = 2.92$, which has an F-distribution with 4 and 46 degrees of freedom. The resulting p-value is 0.031, we thus reject the null hypothesis and conclude that meteorology has a significant effect on the mortality.

6 Model Diagnostics

6.1 Why Model Diagnostics?

We need to check the assumptions we made for fitting a multiple linear regression model. Why? One reason is because we want to make sure that the estimates we produce and the inference we draw is valid. This seems rather technical and also somewhat fussy and boring.

However, there is a second, usually even more important reason to perform model diagnostics: any potential deviations that appear can help us to improve the model. In fact, we can even go as far as saying “it is all in the residuals”, i.e. most of what we can learn about how to enhance a regression analysis is derived from some clever diagnostics plots.

Such enhancement include response and/or predictor transformations, inclusion of further predictors or interactions between predictors into the model, weighted regression or using more generally formulated, robust models, which can really deal with the problem at hand. This is what explorative data analysis is like – we fit a model, try some ideas, check the results and try to improve.

6.2 What Do We Need to Check For, and How?

We restate the assumptions we made for the multiple linear regression model. The goal in model diagnostics is to detect potential deviations from them.

$E[\varepsilon_i] = 0$,

$Var(\varepsilon_i) = \sigma_\varepsilon^2 I$,

$Cov(\varepsilon_i, \varepsilon_j) = 0$ for all $i \neq j$,

$\varepsilon_i \sim N(0, \sigma_\varepsilon^2 I),\ i.i.d$.

Please remember that while the first three conditions are necessary for performing least square estimation, the last condition is only required for any hypothesis tests, as well as confidence and prediction intervals.

There are graphical and numerical diagnostic techniques. While the former are far more flexible and versatile, they require some expertise in interpretation. The latter require no intuition, but are much narrower in scope, and often lack of the power to really detect what is important – we thus focus on graphical diagnostics. This is in line with our view that regression analysis is an interactive and iterative process.

6.3 Checking Error Assumptions

We wish to check the independence, constant variance and normality of the errors. The errors ε_i themselves are not observable, but we can examine the residuals r_i, which are estimates of the errors. However, the two are not the same, and also have somewhat different properties. Even under the assumption of $Var(\varepsilon) = \sigma_\varepsilon^2 I$, we have $Var(r) = (I - H)\sigma_\varepsilon^2$.

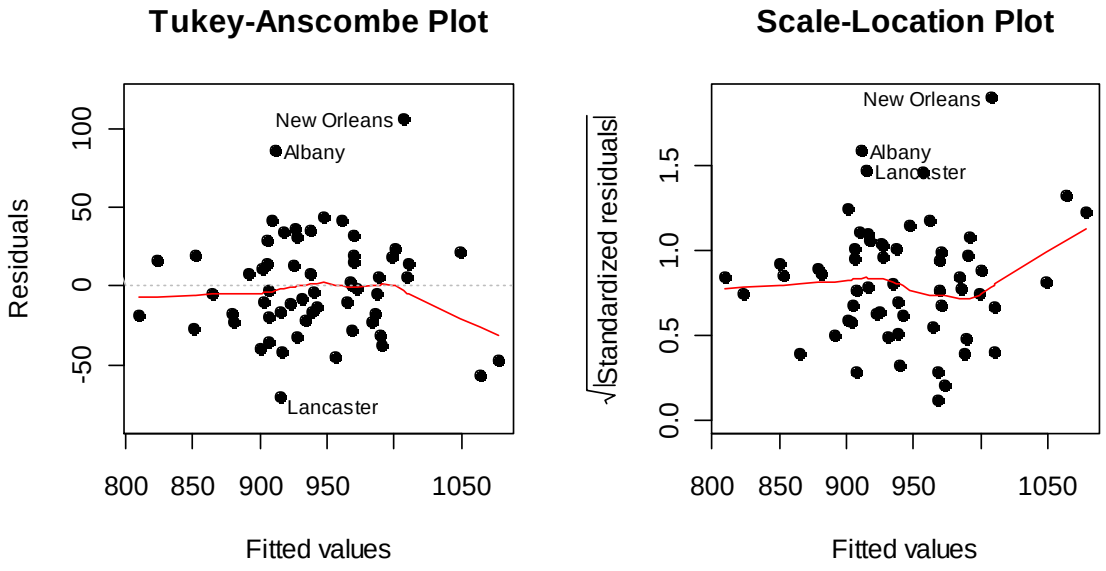
Thus, although the errors may have equal variance and be uncorrelated, the residuals do not. Fortunately, the impact of this is usually small, and diagnostics are often applied to the residuals in order to check the assumptions on the error. The alternative is to use the so-called standardized or studentized residuals r_i^*:

$$r_i^* = \frac{r_i}{\hat{\sigma}_\varepsilon \sqrt{1 - h_{ii}}}$$

Constant Variance

It is not possible to check the assumption of constant variance just by examining the residuals alone – some will be large and some will be small, but this proves nothing. We need to check whether the variance in the residuals is related to some other quantity, and this should not be the case, no matter what that quantity is.

The most popular diagnostic means is the Tukey-Anscombe plot, where the residuals r_i are plotted against the fitted values $\hat{y}_i$. If all is well, we should see constant variation in the vertical direction, and the scatter should be symmetric around zero. Things to look for are heteroscedasticity (non-constant variance) and non-linearity. The red smoother line that is plotted in R aids for detecting non-linearity.



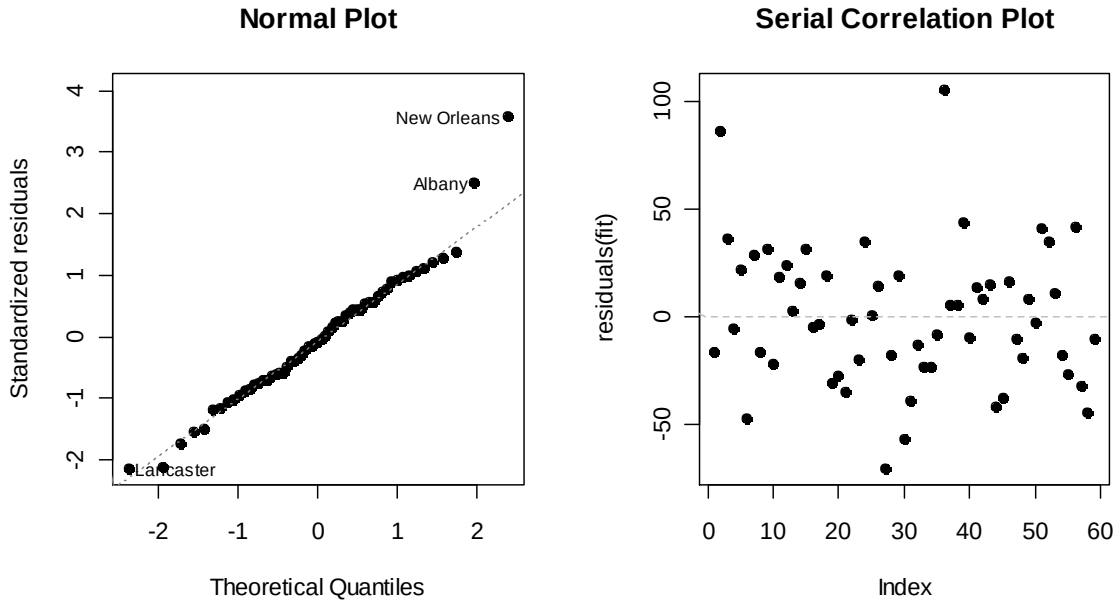
The scale-location plot is similar to the Tukey-Anscombe plot. It also has the fitted values on the x-axis, but the y-axis holds the square root of the standardized

residuals, in place of the raw residuals r_i. Thus, it folds over the bottom half of the first plot to increase the resolution for detecting non-constant variance. The red smoother line again helps in detecting violations. While we observe some increase on the right, we consider this as only slight evidence for non-constant variance.

Moreover, we can and should also plot the residuals r_i against x_i, i.e. predictors that are both in and out of the model. We must not see structures in any of these plots. If we find some, that means the model is inadequate.

Normality

The assumption of normally distributed errors can be checked with the *normal plot*, i.e. we plot the ordered residuals against the corresponding quantiles of the Gaussian distribution. If the errors ε_i are normally distributed, then this also holds for the residuals r_i. Thus, the normal plot should (nearly) be a straight line.



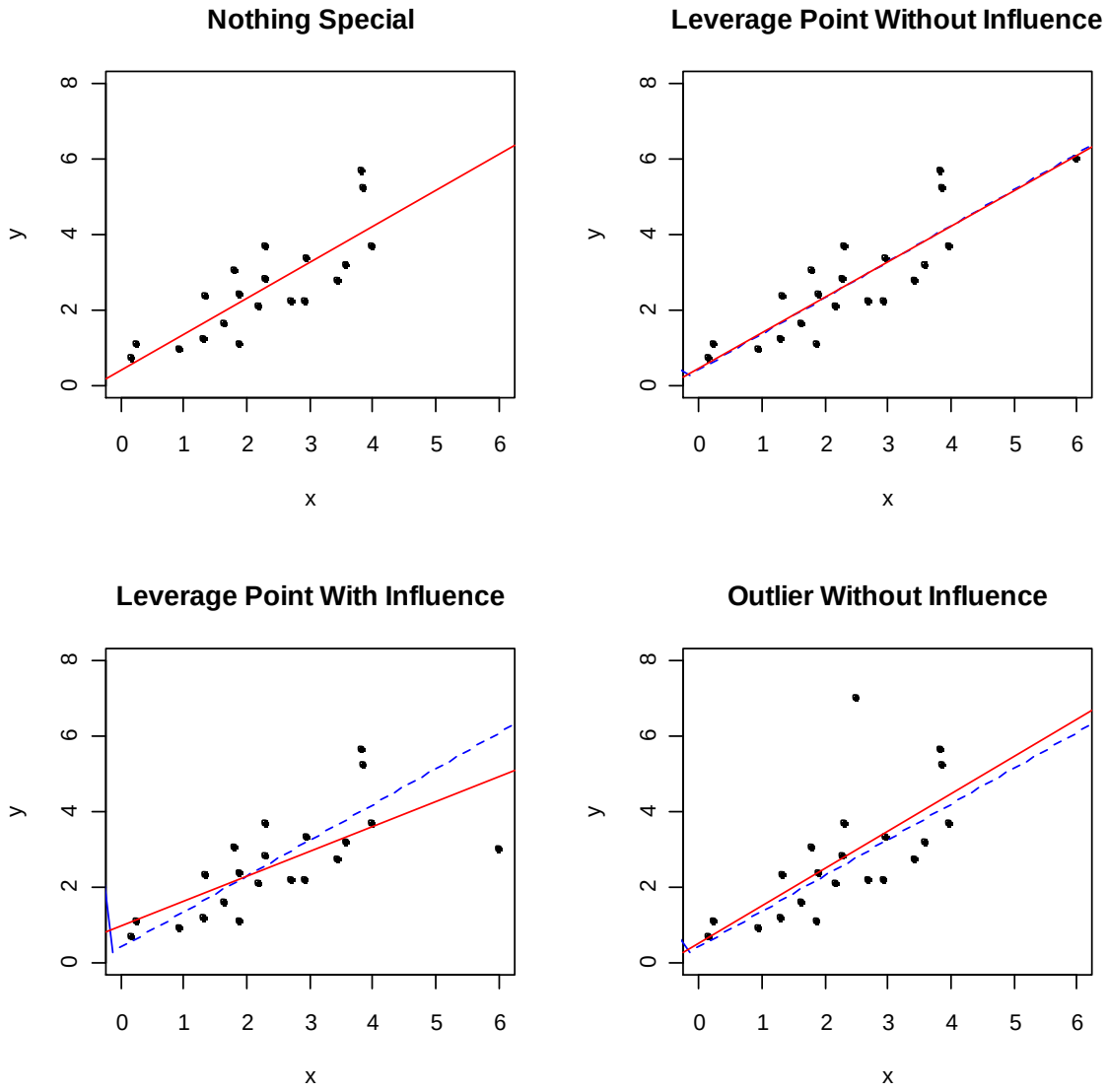
The assumption of Gaussian errors is well fulfilled here, except for two outliers: New Orleans and Albany. We will discuss these in depth in section 6.5

Correlated Errors

For checking the independence of the errors we can plot the residuals r_i versus their observation number i, or if available, versus the time t_i of recording. Here, the residuals vary randomly around the zero line, there is thus no indication for serial correlation. If there was a non-random structure, this would be a model violation, and we might need to consider time series regression, respectively the generalized least squares approach, which is not discussed in this course.

6.4 Influential Data Points and Outliers

There are situations where the regression coefficient estimates are strongly influenced by one single, or just a few data points. If that is the case, it was beneficial to identify these. However, the residual plots mostly only show them, if they are not only influential points, but also outliers.



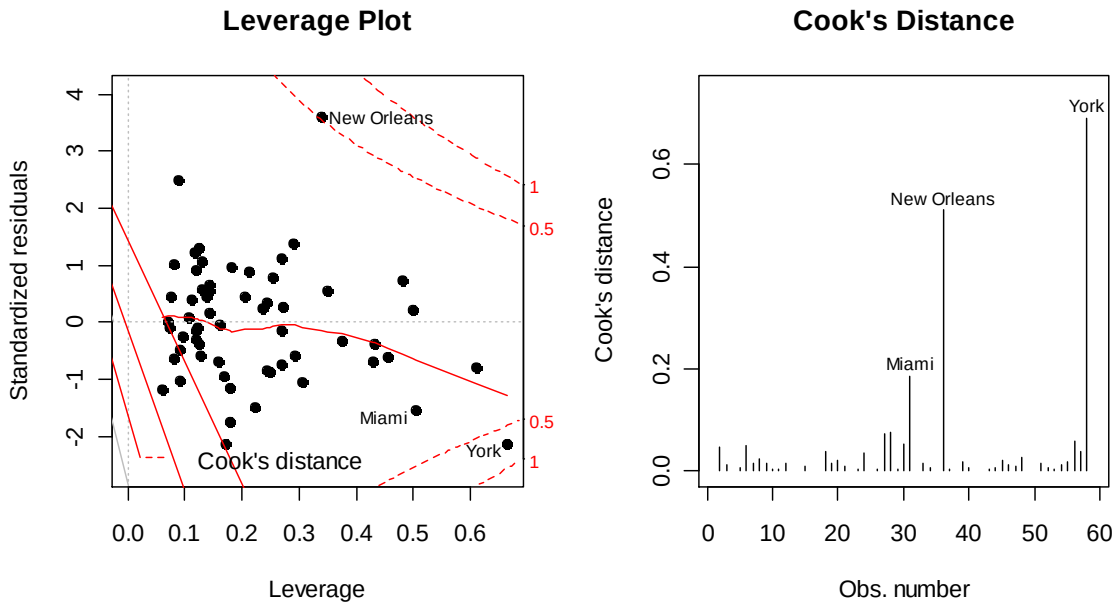
A crucial sub-category of influential data points are the so-called leverage points. These are data with some extreme x-values. The plots below illustrate this: the top left panel shows a “normal” situation without any specialties. In the top right panel, there is a leverage point; however it is not influential on the regression line. This is different in the bottom left panel: the leverage point now has considerable influence, i.e. the red regression line differs markedly from the blue one, which was computed by omitting the leverage point. Finally, the bottom right panel shows an outlier, which has only little influence on the regression line. This is because it has an x-value which is close to $\bar{x}$.

Leverage

Thus, a simple strategy for identifying data points which are influential on estimates and hypothesis test would be to just repeat the analysis after having them deleted. However, doing this would be quite laborious, and is in fact not necessary. We can compute the so-called *leverages* and the *Cook's distance*, and by using them, are able to identify influential data points in a single computing step.

The leverages are the diagonal elements h_{ii} of the hat matrix H which was introduced in section 0. Their meaning is as follows: if we change y_i by Δy_i, then $h_{ii}\Delta y_i$ is the change in the fitted value $\hat{y}_i$. Thus, a high leverage for the i^{th} data point means that it strongly forces the regression line to fit well to it.

We have $0 \le h_{ii} \le 1$ for all i, and $\sum h_{ii} = p+1$. All data points with values exceeding $h_{ii} > 2(p+1)/n$ are regarded as leverage points. As we have seen above, especially the ones with high leverage and high residual r_i are dangerous, i.e. have high potential to strongly affect the results of our regression analysis. Plotting the residuals r_i versus the leverages h_{ii} can thus be very beneficial.



The Leverage Plot shows how strongly a data point forces the regression line to fit through it. Again, it's the two cities of York and New Orleans which lie within the "danger zone". The rule of the thumb is that all data points exceeding the 0.5 line (in Cook's Distance, see below) in the plot are to be treated as suspicious, whereas points exceeding the 1.0 line require (mandatory) further investigation.

Cook's Distance

An even more direct measure for the change in the regression line by omitting the i^{th} data point is Cook's distance. For data point i, it is defined by

$$D_i = \frac{\sum (\hat{y}_j - y_{j(i)})^2}{(p+1)\sigma_\varepsilon^2}$$

Note that $\hat{y}_{j(i)}$ is the fitted value for the j^{th} instance, when the regression is done without the i^{th} data point. Does this mean that we now really have to perform $(n+1)$ regressions to find the D_i. The answer is no, because there is a relation between D_i, h_{ii} and r_i:

$$D_i = \frac{h_{ii}}{1-h_{ii}} \cdot \frac{r_i^{*2}}{(p+1)},$$

Where r_i^* is the so-called standardized or studentized residual. The differences between r_i^* and r_i are usually small, or can even be neglected. The definition is:

$$r_i^* = \frac{r_i}{\hat{\sigma}_\varepsilon \sqrt{1-h_{ii}}}.$$

Data points where the Cook's distance $D_i > 1$ need further investigation; because it might well be that they spoil your regression analysis. In the mortality dataset, the Cook's Distance plot shows that omitting the city of York from the regression analysis would change the results most. New Orleans, and to a much lesser extent, Miami, seem to be influential, too. However, none of the data points exceeds the limit of $D_i > 1$.

Outliers

We have seen above that the "most dangerous" data points are the ones that are leverage points and outliers at the same time. Also, we explained that Cook's Distance is a well suited measure to identify such points. However, here are some more things to consider about the presence of outliers:

1) Two or more adjacent outliers can hide each other.

2) An outlier in one model may not be an outlier in another when the variables have been changed or transformed. One usually needs to reinvestigate the question of outliers when the model is changed.

3) The error distribution may not be Gaussian and thus, larger residuals may be expected. For example, day-to-day changes in stock indices seem Gaussian over large periods of times, but large changes also happen once in a while.

4) A single or very few outliers are usually much less of a problem in larger datasets. A single point will mostly not have the leverage to affect the fit very much. It is still worth identifying outliers if these types of observations are worth knowing about in the particular application.

Suppose that you detected one or several outliers in your data. What to do with them? The following can serve as a practical guide:

a) Check for typos first. These are relatively common. However, this requires the original source of the data. Make sure you do not lose it, or lose contact to it.

b) Examine the physical context – why did it happen? Sometimes, the discovery of an outlier may be of singular interest. On the other hand, it was often the case that scientific discoveries arose from noticing unexpected aberrations.

c) Exclude the outlier(s) from the analysis, and re-fit the model. The differences can be substantial and make the difference between getting a statistically significant result, or having some "garbage" that cannot be published. To avoid any suggestion of dishonesty always report the existence of outliers that were removed from the final model.

d) Suppose there are outliers that cannot be reasonably identified as mistakes or aberrations, but are viewed as naturally occurring, e.g. due to long-tailed error distribution. Rather than excluding these instances and the using least squares, it is more efficient and reliable to use robust regression, as explained in section 6.7

6.5 Example: Mortality Dataset

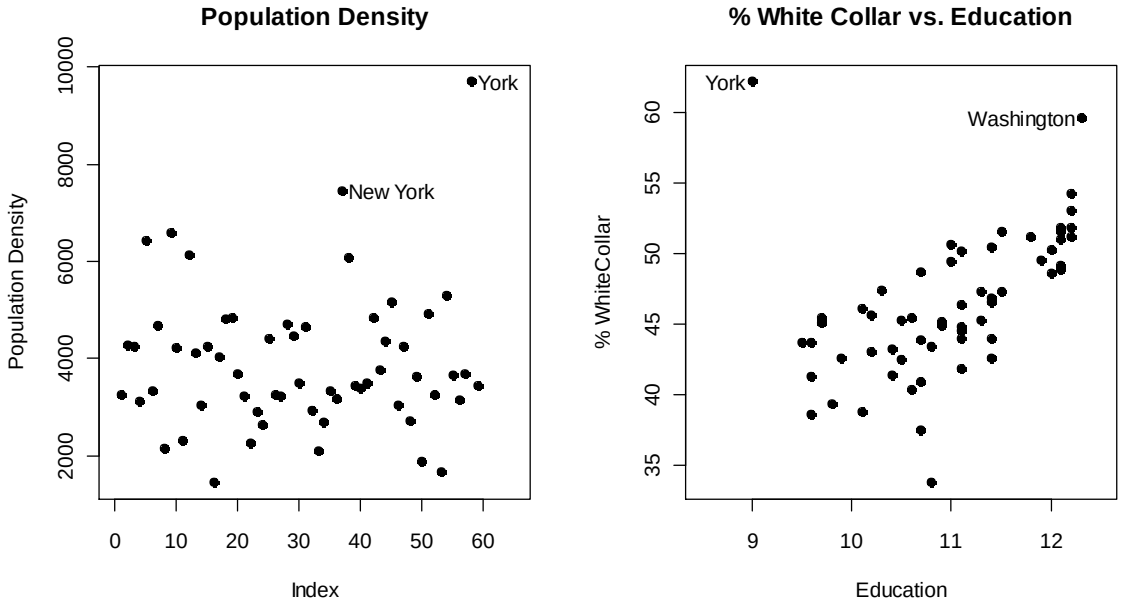
From the model diagnostics, we conjecture that York and New Orleans are the most influential data points. To be on the safe side, it is reasonable to re-run the regression analysis without these two data points. The results are presented here, please compare to section 5.6:

```
> summary(lm(Mortality ~ JanTemp + JulyTemp + RelHum + Rain +
           Educ + Dens + NonWhite + WhiteCollar + Pop + House+
           Income + log(SO2), data = mortality[-c(36, 58), ])

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  9.025e+02  2.564e+02   3.521 0.001016 **
JanTemp     -1.246e+00  6.714e-01  -1.856 0.070168 .
JulyTemp    -1.317e-01  1.693e+00  -0.078 0.938339
RelHum       3.984e-01  9.286e-01   0.429 0.670023
Rain         1.399e+00  4.630e-01   3.022 0.004174 **
Educ        -5.788e+00  9.571e+00  -0.605 0.548430
Dens         9.360e-03  4.210e-03   2.223 0.031377 *
NonWhite     3.651e+00  9.021e-01   4.048 0.000206 ***
WhiteCollar -1.046e+00  1.371e+00  -0.763 0.449775
Pop         -1.175e-06  3.478e-06  -0.338 0.737058
House        1.390e+00  3.430e+01   0.041 0.967857
Income      -9.580e-05  1.118e-03  -0.086 0.932089
log(SO2)     1.388e+01  5.151e+00   2.695 0.009926 **
---
Residual standard error: 30.31 on 44 degrees of freedom
Multiple R-squared: 0.7929,  Adjusted R-squared: 0.7364
F-statistic: 14.04 on 12 and 44 DF,  p-value: 2.424e-11
```

The most important observations from this analysis are that the residual standard error is now smaller, and the coefficient of determination increased. Thus, the fit is better now. Moreover, the logged SO_2 is now significant again. Might it be that the pollution has an influence on the mortality?

We now turn our attention to the interesting question why the cities of York and New Orleans were influential data points. Plotting some of the predictors, maybe even against other predictors and identifying outlying data points may help. In the plots below, we observe that the city of York has a considerably higher population density than all the other towns. It turned out that the definition of districts with which the population density was defined was somewhat suboptimal.

Moreover, it is also striking that the average years of education in York are much lower than elsewhere, but the percentage of white collar workers is higher. This anomaly is explained by the predominance of Amish people in that region. It is thus, an inhomogeneity of the sample.



6.6 Weighted Regression

We consider the following generalization of the multiple linear regression model:

$$Y = X\beta + \varepsilon \text{, where } \varepsilon \sim N(0, \sigma_\varepsilon^2 \Sigma) \text{, with } \Sigma = I$$

Thus, the errors do no longer have constant variance, and may, in case of a non-diagonal Σ, even be correlated. While the case of correlated errors can be dealt with using the generalized least squares approach, it is beyond the scope of this scriptum. We will here focus on the simpler case where Σ is a diagonal matrix, which can be solved by weighted linear regression. Let

$$\Sigma = diag\left(\frac{1}{w_1}, \frac{1}{w_2}, ..., \frac{1}{w_n}\right)$$

The w_i can be interpreted as weights. They should be such that observations with large variance have low weight, and vice versa. This makes sense, because the instances with low variance will typically have smaller errors and should thus have more impact on the fit. How could we choose the weights?

- If the Y_i are means of n_i observations, we choose $w_i = n_i$. This could be the case if the response is the average over measurements from different days, but the number of days is not equal for all instances.
- There are cases where it is clear that the variance is proportional to some predictor x. In that case, we would choose $w_i = 1/x_i$.
- In all other cases, where we "just" observe non-constant variance in the diagnostic plots and do not have an idea about its source, we would estimate the weights from a non-weighted least squares regression.

The regression coefficients in weighted regression are obtained by minimizing the sum of weighted least squares:

$$\sum_{i=1}^{n} w_i r_i^2$$

This causes some changes in the normal equations, but there is still an explicit and unique solution for given weight, provided the design matrix X has full rank.

6.7 Robust Regression

When the errors are normally distributed, least squares regression is clearly the best way to proceed. But what if they are not Gaussian? Well, then, other methods need to be considered. Of particular concern are long-tailed error distributions.

The poor man's approach is to declare the largest residuals as being outliers remove them from the dataset and still use least squares. However, this should only be applied when one is convinced that the outliers represent truly incorrect instances. In cases, where they are non-faulty observations, it is much better to use a robust regression method that down-weights the effect of larger errors.

The Huber method is the default choice of the `rlm()` function in `library(MASS)`. It is beyond the scope of this course to give further details than to say that this is a procedure for limiting the effect of outliers. The use of the function is exactly as the one of `lm()`.

We have seen in the Normal plots above that there were some outliers in the mortality dataset. Thus, using robust regression on this dataset is justified, and it will be interesting to compare results. The summary output is:

```
> summary(fit.rlm)

Call: rlm(Mortality ~ JanTemp + JulyTemp + RelHum + Rain +
          Educ + Dens + NonWhite + WhiteCollar + Pop +
          House + Income + log(SO2), data = mortality)
```

```
Coefficients:
             Value    Std. Error t value
(Intercept) 945.4414 251.6184     3.7574
JanTemp      -1.2313   0.6788    -1.8139
JulyTemp     -0.1605   1.6598    -0.0967
RelHum        0.5576   0.9461     0.5894
Rain          1.3230   0.4699     2.8154
Educ         -3.5682   8.0884    -0.4412
Dens          0.0055   0.0038     1.4461
NonWhite      4.1074   0.8665     4.7403
WhiteCollar  -2.4711   1.0820    -2.2838
Pop           0.0000   0.0000     0.2237
House        -1.3143  34.5831    -0.0380
Income        0.0003   0.0011     0.2212
log(SO2)     13.0484   4.6444     2.8095
---
Residual standard error: 30.17 on 46 degrees of freedom
```

First, we observe that there are some differences in the output. We are missing the coefficient of determination and the global F-test. The reason is because they cannot be calculated with this robust regression model. Similarly, the p-values for the individual parameter tests are not given, although we can use the asymptotic normality of the estimator to make approximate inference using the t-values.

As a next step, we do now compare the coefficient as well as the t-values from the least squares and the robust fit. We have:

```
             coef.lm coef.rlm  t.lm t.rlm
(Intercept) 1163.91   945.44  3.96  3.76
JanTemp       -1.67    -1.23 -2.10 -1.81
JulyTemp      -1.17    -0.16 -0.60 -0.10
RelHum         0.70     0.56  0.63  0.59
Rain           1.22     1.32  2.23  2.82
Educ         -11.08    -3.57 -1.17 -0.44
Dens           0.01     0.01  1.25  1.45
NonWhite       5.08     4.11  5.02  4.74
WhiteCollar   -1.93    -2.47 -1.52 -2.28
Pop            0.00     0.00  0.51  0.22
House        -22.16    -1.31 -0.55 -0.04
Income         0.00     0.00  0.18  0.22
log(SO2)       6.83    13.05  1.26  2.81
```

Except for the intercept, which is of minor importance, the difference in the estimated coefficients is mostly small. There are some notable exceptions concerning `Educ`, `House` and `log(SO2)`. Moreover, the changes in the t-values are mostly unimportant. An exception to this is what we observe for `log(SO2)`. With robust regression, the t-value is 2.81, which would be (asymptotically) significant on the 5%-level. The decision whether pollution contributes to mortality is really controversial!

7 Polynomial Regression and Categorical Input

7.1 Polynomial Regression

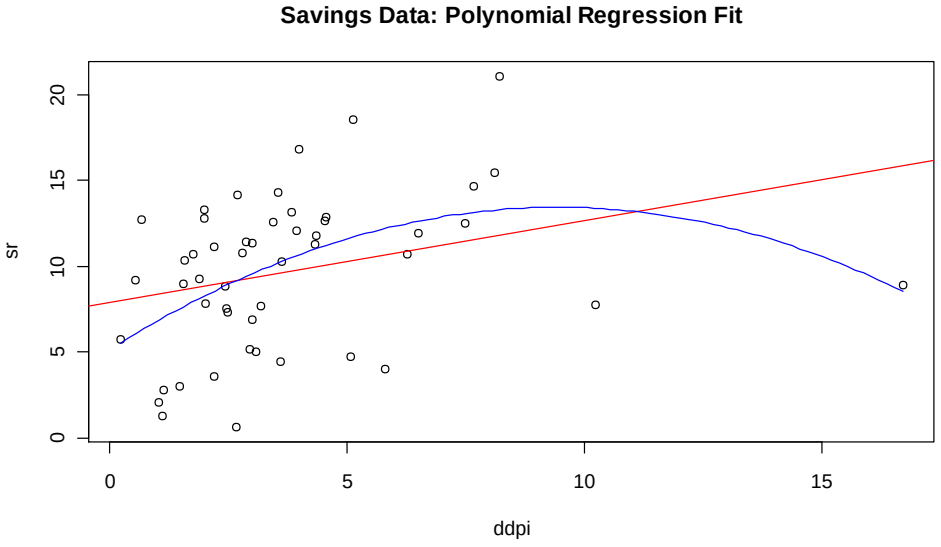
So far, we only considered regression examples where the predictors were continuous, quantitative variables describing totally different aspects of the observations. However, we never made a restriction requiring this. The linear model $Y = X\beta + \varepsilon$ is a general model that is linear in the unknown parameters β. This also includes the important class of polynomial regression models. For example, the polynomial of order d in one variable

$$Y = \beta_0 + \beta_1 x + \beta_2 x^2 + ... + \beta_d x^d + \varepsilon$$

is a multiple linear regression model, too, even though it fits a polynomial. Such models are widely used in cases where the relation to the response is curvilinear, because even complex nonlinear relations can be modeled by polynomials. In other words, our goal here is to improve the fit between x and Y by including quadratic and/or cubic terms, or some of even higher orders, into the model.

7.2 Example: How to Determine the Order d

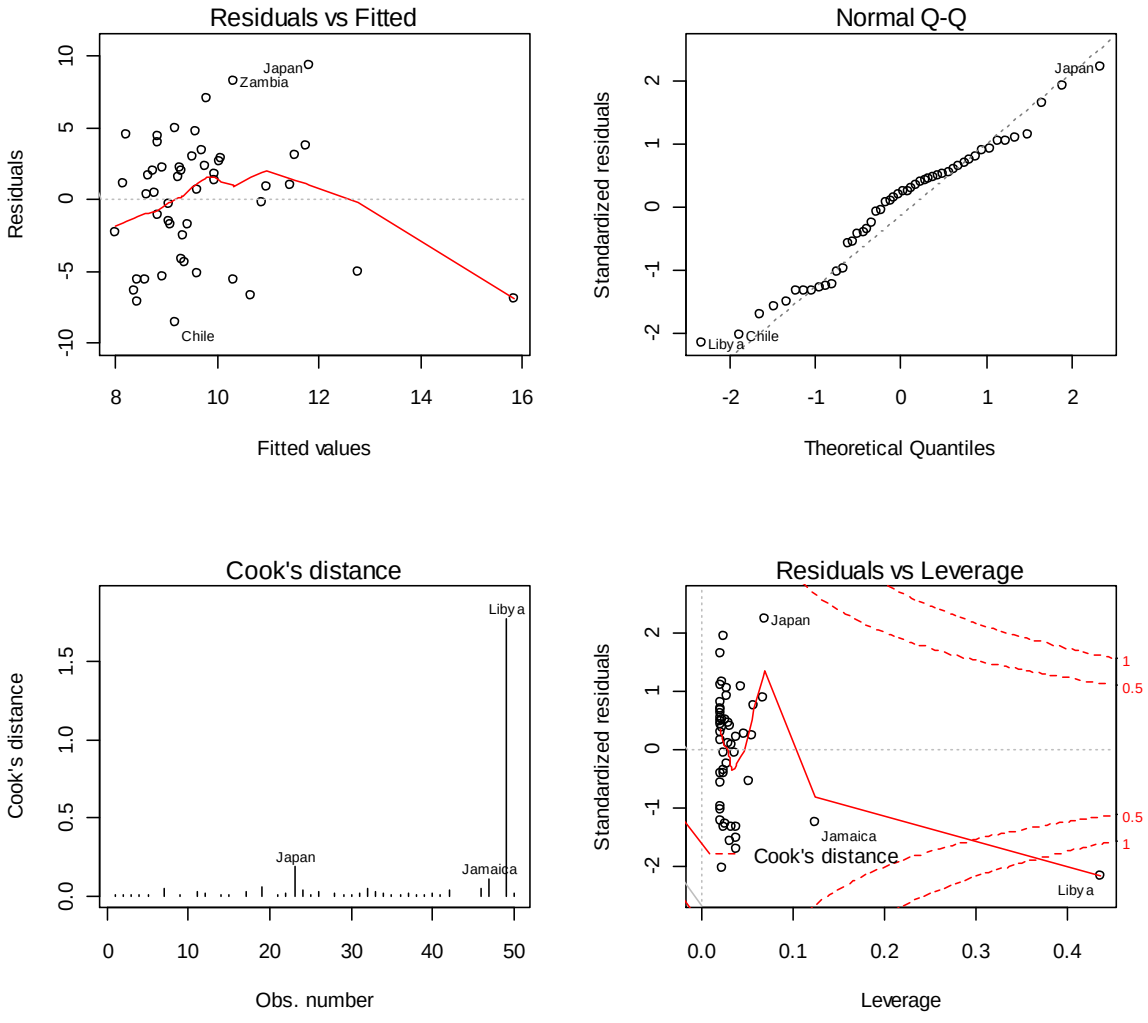
Let us illustrate polynomial regression with an economic dataset comprising observations made in 50 different countries. The data are averages taken over 10 years from 1960 to 1970, in order to remove any business cycles or other unwanted short-term fluctuations. Variable `dpi` is the per capita disposable income in US dollars, `ddpi` is the percentage rate of change in per capita disposable income, and `sr` is the aggregate personal saving divided by disposable income. The percentage of population under 15 (pop15) and over 75 (pop75) are also recorded. The data come from a study of Belsley, Kuh and Welsch ("Regression Diagnostics: Identifying Influential Data and Sources of Collinearity", Wiley, New York, 1980).



We consider `sr` as the response, and `ddpi` as the predictor. The simplest approach is to fit a simple linear regression. The summary is as follows:

```
> summary(lm(sr ~ ddpi, data = savings))

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   7.8830     1.0110   7.797 4.46e-10 ***
ddpi          0.4758     0.2146   2.217   0.0314 *
---
Residual standard error: 4.311 on 48 degrees of freedom
Multiple R-squared: 0.0929,   Adjusted R-squared: 0.074
F-statistic: 4.916 on 1 and 48 DF,  p-value: 0.03139
```

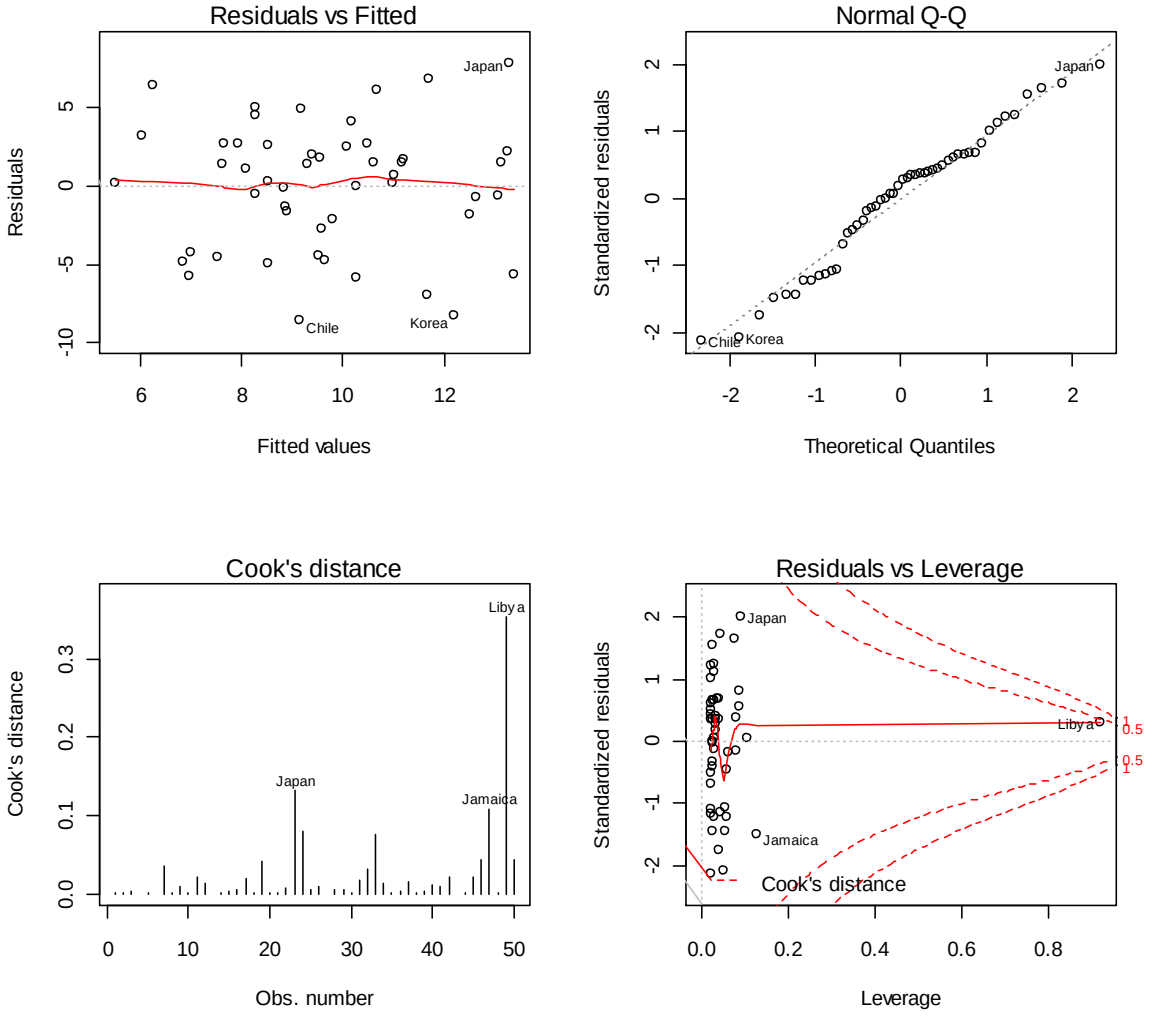
We observe that `ddpi` is weakly significant with a p-value of 0.03. The regression line does not fit the data too badly, but not overly well either. A brief glance at the Tukey-Anscombe plot tells us, that the linear term only does not result in zero expectation for the errors. While the assumption of Gaussian errors does seem to be accurately fulfilled, there is also the country of Libya, which is a very influential data point.



The situation with non-zero expectation for the error term may be much improved by adding the square of `ddpi` into the model. We can then fit a curvilinear relation that is more appropriate for this dataset. The blue line in the scatter plot already indicated this. The summary output is as follows:

```
> summary(lm(sr ~ ddpi + I(ddpi^2), data = savings))

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.13038    1.43472   3.576 0.000821 ***
ddpi         1.75752    0.53772   3.268 0.002026 **
I(ddpi^2)   -0.09299    0.03612  -2.574 0.013262 *
---
Residual standard error: 4.079 on 47 degrees of freedom
Multiple R-squared: 0.205,    Adjusted R-squared: 0.1711
F-statistic: 6.059 on 2 and 47 DF,  p-value: 0.004559
```

Both terms are now significant, the linear term even more strongly than before. Also, the lower estimated value for the error variance and the higher coefficient of determination indicate that the fit is now better.



This impression is confirmed by the diagnostic plots. Tukey-Anscombe and Normal plot are now without any flaws. The country of Libya is still the single most influential data point. However, its Cook Distance is markedly lower than before. Indeed, it has much less of a handle on the fit in the polynomial regression model. Still, it would be interesting to investigate the fit when Libya is omitted. We leave this as an exercise; the changes are not pronounced enough to show them here.

Because the quadratic term improved the fit and both the terms in the model are significant, we may try to add a cubic term. The summary output then is as follows:

```
> summary(lm(sr~ddpi + I(ddpi^2) + I(ddpi^3), data = savings)

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.145e+00  2.199e+00   2.340   0.0237 *
ddpi         1.746e+00  1.380e+00   1.265   0.2123
I(ddpi^2)   -9.097e-02  2.256e-01  -0.403   0.6886
I(ddpi^3)   -8.497e-05  9.374e-03  -0.009   0.9928
---
Residual standard error: 4.123 on 46 degrees of freedom
Multiple R-squared: 0.205,    Adjusted R-squared: 0.1531
F-statistic: 3.953 on 3 and 46 DF,  p-value: 0.01369
```

None of the tree terms is significant anymore. We take this as a signal that the cubic regression is not appropriate here and stick to the quadratic. The global F-test, however, still shows a p-value of 0.01. Thus, the predictors have a significant effect on the response.

7.3 Powers Are Strongly Correlated Predictors

```
> cor(cbind(ddpi, ddpi2=ddpi^2, ddpi3=ddpi^3))
           ddpi     ddpi2     ddpi3
ddpi  1.0000000 0.9259671 0.8174527
ddpi2 0.9259671 1.0000000 0.9715650
ddpi3 0.8174527 0.9715650 1.0000000
```

The reason is that the predictors, i.e. the powers of `ddpi` are strongly correlated, as can be seen from the correlation matrix above. Thus, every of the terms can add a little of its predictive power towards the response, but none of them is really required, as long as the other predictors are in the model.

Having such strongly correlated input variables is a rather unwanted property. In the context of regression, we speak of collinear input predictors. For a general discussion, we refer to section 9.3. Here, we show how the problem can be somewhat mitigated in the context of polynomial regression. Instead of the original predictor `dppi` and its powers, we use the centered variables

$$z_i = (x_i - \bar{x})$$
$$z_i^2 = (x_i - \bar{x})^2$$
$$z_i^3 = (x_i - \bar{x})^3$$

Indeed, the centered variable and its powers show at least partly lower correlation. The matrix is:

```
> cor(cbind(z.ddpi, z2.ddpi, z3.ddpi))
           z.ddpi   z2.ddpi   z3.ddpi
z.ddpi  1.0000000 0.7445202 0.7321169
z2.ddpi 0.7445202 1.0000000 0.9791666
z3.ddpi 0.7321169 0.9791666 1.0000000
```

By working with these variables, the summary output now looks like this:

```
> summary(lm(sr~z.ddpi+I(z.ddpi^2)+I(z.ddpi^3),dat=z.savings)

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.042e+01  8.047e-01  12.946  < 2e-16 ***
z.ddpi       1.059e+00  3.075e-01   3.443  0.00124 **
I(z.ddpi^2) -9.193e-02  1.225e-01  -0.750  0.45691
I(z.ddpi^3) -8.497e-05  9.374e-03  -0.009  0.99281
---
Residual standard error: 4.123 on 46 degrees of freedom
Multiple R-squared: 0.205,    Adjusted R-squared: 0.1531
F-statistic: 3.953 on 3 and 46 DF,  p-value: 0.01369
```

At least the linear term is now significant again. However, the transformation could not fully resolve the issue with the quadratic and cubic terms. Note that this last model has different regression coefficient and different inference results than the one on the original predictors above. On the other hand, the fitted values, residual standard error, coefficient of determination and the global F-test are exactly as they were on the original variables.

We conclude our example on polynomial regression here by making the remark that we can of course have arbitrarily complex models where there are various predictors, some of them with their powers included, and other without. Finally, a word of caution: extrapolation with polynomial models can be extremely hazardous, much more so compared to when only the original predictors are present.

7.4 Dummy Variables

The variables we considered so far were continuous, i.e. temperature, distance, pressure, etc. However, there is no need to do so. It is perfectly valid to use ***categorical predictors***, such as e.g. sex (male or female), status variables (employed or unemployed), shifts (day, evening, night). In general, these

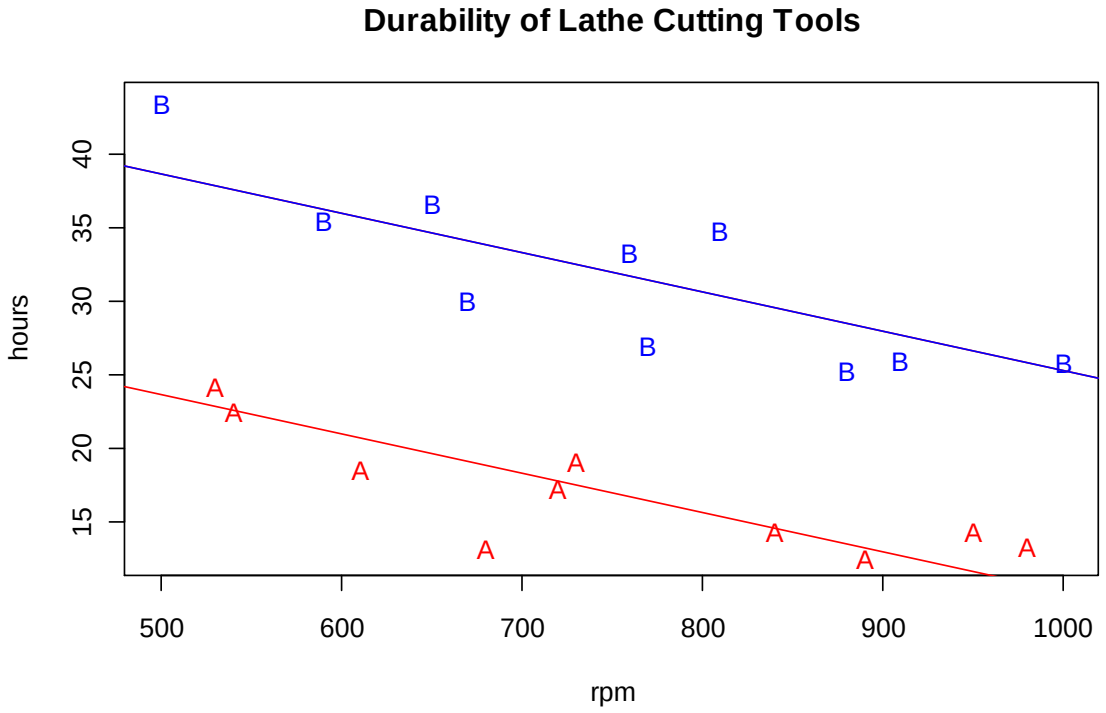
categorical variables have no natural scale of measurement. Thus, we must assign a set of levels to a categorical variable to account for the effect that the variable may have on the response. This is done through the use of indicator variables. In the regression context, they are better known as ***dummy variables***.

7.5 Example: How to Fit with Binary Categorical Variables

Suppose that our goal is to relate the life of a cutting tool (Y) used on a lathe (in German: "Drehbank") in hours to the speed of the machine in rpm (x_1) and the type of cutting tool used (x_2). This second predictor is categorical and here has two levels A and B that codes for two different cutting tools. We will use an indicator variable that takes values 0 and 1 to identify the tool types – this is a so-called dummy variable.

$$x_2 = \begin{cases} 0 & tool\ \ type\ \ A \\ 1 & tool\ \ type\ \ B \end{cases}$$

The choice of 0 and 1 to identify the levels of this categorical predictor is arbitrary. In fact, any two distinct values for x_2 would be satisfactory, although 0 and 1 are the normal choice. We can display the data in a scatter plot of hours vs. rpm, and distinguish the two tool types by different plotting characters.



The plot also contains the regression lines for tool types A and B, respectively. We will now explain how they are found. The regression model for the lathe example does at first sight look no different from one with continuous predictors:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \varepsilon\,.$$

With R, fitting regression models with categorical predictors is straightforward. We do not even need to take care of the dummy variables ourselves. It is sufficient that the categorical predictor is a factor, i.e. `class(lathe$tool)="factor"` The summary output for the regression model is:

```
> summary(lm(hours ~ rpm + tool, data = lathe))

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 36.98560    3.51038  10.536 7.16e-09 ***
rpm         -0.02661    0.00452  -5.887 1.79e-05 ***
toolB       15.00425    1.35967  11.035 3.59e-09 ***
---
Residual standard error: 3.039 on 17 degrees of freedom
Multiple R-squared: 0.9003,  Adjusted R-squared: 0.8886
F-statistic: 76.75 on 2 and 17 DF,   p-value: 3.086e-09
```

We will now turn our attention to the interpretation of this regression model. We first consider an observation i where the tool is of type A. There, we have $x_{i2} = 0$ and thus the model simplifies to:

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 \cdot 0 + \varepsilon_i = \beta_0 + \beta_1 x_{i1} + \varepsilon_i .$$

Thus, the relation between tool life and lathe speed for tool type A is a straight line with intercept $\beta_0 = 36.99$ and slope $\beta_1 = -0.027$. Important: note that the slope is generally not equal to the one we would obtain from a simple linear regression for tools of type A only!

Now conversely, for any observation j with tool type B, we have $x_{j2} = 1$, and thus:

$$Y_j = \beta_0 + \beta_1 x_{j1} + \beta_2 \cdot 1 + \varepsilon_j = (\beta_0 + \beta_2) + \beta_1 x_{j1} + \varepsilon_j$$

That is, for tool type B the relation between tool durability and lathe speed is also a straight line with the same slope $\beta_1 = -0.027$, but different intercept $\beta_0 + \beta_2 = 51.99$. Thus, the model estimates a common, identical slope coefficient for the two tool types.

The regression coefficient β_2 of the dummy variable x_2 accounts for the additive shift in durability of tool type B vs. tool type A, i.e. measures the difference in mean tool life when changing from tool type A to tool type B. Thus, we have the proof for the impression that the two regression lines in the scatter plot above are parallel.

To make the regression analysis complete, we now also have to check the diagnostic plots. We leave this as an exercise, because there are no peculiarities for this specific example. Remember that with categorical input variables, it is most instructive to use different plotting symbols for tool types A and B.

7.6 Interactions

In the above example, the regression line for the tool types A and B had different intercept, but identical slope. In this particular example, the fit seemed to be pretty well even under this restriction. However, we can easily imagine a situation where two parallel regression lines are not appropriate. The question this section deals with is whether and how a model with two different regression lines can be fit. It is possible to model this situation with a single regression equation by using indicator variables. The model is:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2 + \varepsilon .$$

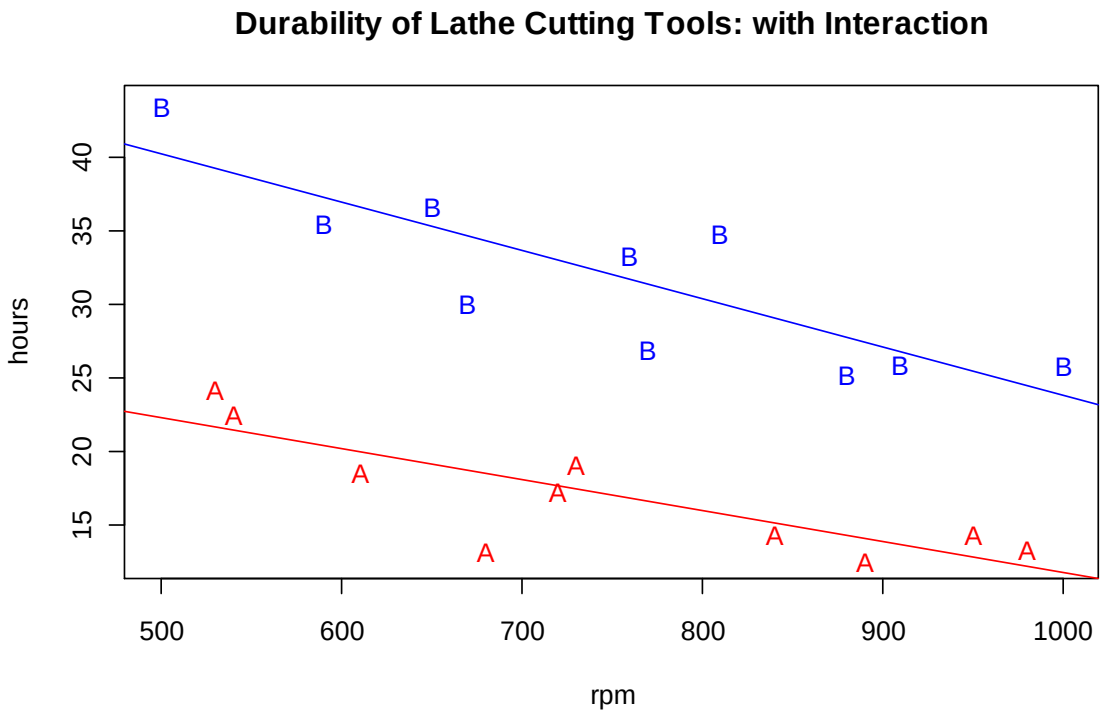
We observe that a cross product between lathe speed x_1 and the indicator variable denoting tool type x_2 has been added to the model. To interpret the parameters in this model, we first consider an observation i with tool type A. Remember; this means that the dummy variable x_2 is equal to 0.

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 \cdot 0 + \beta_3 \cdot 0 + \varepsilon_i = \beta_0 + \beta_1 x_{i1} + \varepsilon_i$$

Thus, this is again the regression line with intercept β_0 and slope β_1. For tool type B, we have $x_2 = 1$ for the dummy variable. Thus, the regression model becomes:

$$Y_j = \beta_0 + \beta_1 x_{j1} + \beta_2 \cdot 1 + \beta_3 x_{j1} \cdot 1 + \varepsilon_j = (\beta_0 + \beta_2) + (\beta_1 + \beta_3) x_{j1} + \varepsilon_j$$

This is a straight-line model with intercept $\beta_0 + \beta_2$ and slope $\beta_1 + \beta_3$. Thus, the interaction model defines two regression lines with different intercepts and different slopes. Therefore the parameter β_2 reflects the change in the intercept associated with changing from tool type A to tool type B, and β_3 indicates the change in the slope associated with this change.



The scatter plot of `hours` vs. `rpm` is shown below, together with the two regression lines that are no longer parallel under the interaction model. Next, we have a look at the summary output. Note that we do no longer separate the predictors by a "+" in function `lm()`, but now use a "*". This means that we do not restrict to the main effects, but also include the interaction term.

```
> summary(lm(hours ~ rpm * tool, data = lathe))

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 32.774760   4.633472   7.073 2.63e-06 ***
rpm         -0.020970   0.006074  -3.452  0.00328 **
toolB       23.970593   6.768973   3.541  0.00272 **
rpm:toolB   -0.011944   0.008842  -1.351  0.19553
---
Residual standard error: 2.968 on 16 degrees of freedom
Multiple R-squared: 0.9105,  Adjusted R-squared: 0.8937
F-statistic: 54.25 on 3 and 16 DF,  p-value: 1.319e-08
```

We have seen in the scatter plot above, that there is a large vertical shift between the two regression lines. The slope however, only differs little. An obvious question is whether fitting two regression lines with different slopes is necessary, i.e. whether the difference is statistically significant. This amounts to the test of

$H_0 : \beta_3 = 0$ against $H_A : \beta_3 \neq 0$.

This is an individual parameter test for the interaction term, and the result can be directly read from the summary output. The p-value is 0.196, thus not statistically significant. If there are no further (practical) reasons strongly speaking for different slopes, we would fit parallel lines.

Note that the (full) interaction model always yields the same result as two separate simple linear regressions on tools of type A, and tools of type B. However, there is still an advantage of using the interaction model: the test whether or not the two regression models are identical is straightforward. We have to test

$H_0 : \beta_2 = \beta_3 = 0$ against $H_A : \beta_2 \neq 0 \ \textit{and / or} \ \beta_3 \neq 0$.

This is a partial F-test, where we try to exclude the interaction and the dummy variable at the same time. The R-code and the output is as follows:

```
> fit1 <- lm(hours ~ rpm, data=lathe)
> fit2 <- lm(hours ~ rpm * tool, data=lathe)
> anova(fit1, fit2)
Model 1: hours ~ rpm
Model 2: hours ~ rpm * tool
  Res.Df     RSS Df Sum of Sq      F    Pr(>F)
1     18 1282.08
2     16  140.98  2    1141.1 64.755 2.137e-08 ***
```

We observe that the p-value is very small, and the partial F-test thus highly significant. While there is no evidence for different slope in this example, there is strong evidence of a difference (in either slope or intercept). Regarding the scatter plot, with the pronounced vertical shift between tool types A and B, this does not surprise us.

Finally, we conclude this section by stating that the use of interaction models is not restricted to a combination of continuous and categorical predictors. In this case, they can be visualized most easily. However, we can have them between any type of predictors. They are appropriate whenever there is, or whenever we suspect a change in the effect of one predictor on the response, conditional on the level of another predictor.

7.7 Categorical Input with More than Two Levels

An obvious extension to the previous example with lathe cutting tools would be to consider three or more types of tools instead of only two. The tool variable then is still categorical, but no longer binary, and we need more dummy variables. For example, suppose that there are three tool type A, B and C. We then require two dummy variables to incorporate them into the model. The coding is as follows:

x_2	x_3	
0	0	*for observations of type A*
1	0	*for observations of type B*
0	1	*for observations of type C*

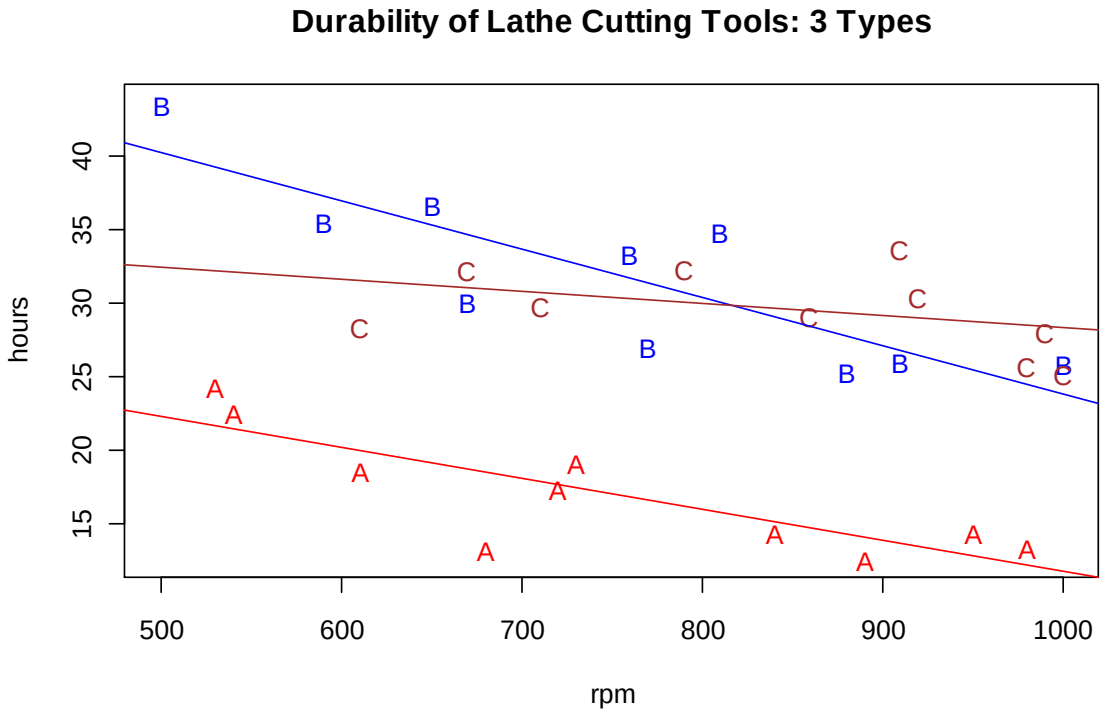
In general, a qualitative variable with a levels is represented by $a-1$ dummy variables, each taking values 0 and 1. Note that with this coding, the first level (here: tool type A) is always the reference. This is also how R codes categorical input variables by default: the first factor level is the reference. There are, however, different options for coding, called contrasts. This is more a topic in analysis of variance, thus we do not discuss that issue here.

The main effects regression model with three types of tools and their respective dummy variables is now:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \varepsilon$$

This would fit three parallel regression lines, where each has a different intercept. However, when we closely observe the scatter plot below, we gain the impression that the durability of tool type C seems to depend much less on `rpm` than the other two. While at slow speeds, its lifetime seems to be inferior to the type B tools, they seem to last longer at faster speeds. Because the main effects model cannot deal with the apparently different slopes, we fit the interaction model:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_1 x_2 + \beta_5 x_1 x_3 + \varepsilon$$



The interpretation of this model is as before with binary categorical input. We leave it as an exercise to write down the cases for observations i, j and k of tool types A, B and C. The regression fit with R is again straightforward; we only need the `tool` variable to be a factor with multiple levels:

```
> summary(lm(hours ~ rpm * tool, data = abc.lathe)

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 32.774760   4.496024   7.290 1.57e-07 ***
rpm         -0.020970   0.005894  -3.558  0.00160 **
toolB       23.970593   6.568177   3.650  0.00127 **
toolC        3.803941   7.334477   0.519  0.60876
rpm:toolB   -0.011944   0.008579  -1.392  0.17664
rpm:toolC    0.012751   0.008984   1.419  0.16869
---
Residual standard error: 2.88 on 24 degrees of freedom
Multiple R-squared: 0.8906,   Adjusted R-squared: 0.8678
F-statistic: 39.08 on 5 and 24 DF,  p-value: 9.064e-11
```

The interpretation of this summary output now needs to be done with care. Individual parameter tests for dummy variable coefficients of categorical predictors with more than two levels are not meaningful. Thus, from the above output, we cannot conjecture that we can do without a different intercept for tool C, because the test for $H_0 : \beta_3 = 0$ is not significant. Moreover, also the coefficients β_4 and β_5 for the interactions have p-values above 0.05. Does that mean that we can do without the interaction? No!

We can only either exclude all the interaction terms at once, i.e. test the hypothesis

$$H_0 : \beta_4 = 0 \ \textit{and} \ \beta_5 = 0 \text{ against } H_A : \beta_4 \neq 0 \ \textit{and / or} \ \beta_5 \neq 0 .$$

This is again a partial F-test. Furthermore, we can also ask the question whether there is a difference between the regression lines of the three tool types altogether. Thus, we also test for the sub-model with only `rpm` as a predictor:

$$H_0 : \beta_2 = \beta_3 = \beta_4 = \beta_5 = 0 \text{ against } H_A : \textit{any of } \beta_2, \beta_3, \beta_4, \beta_5 \neq 0 .$$

While many software packages have troubles with this, R is very convenient and very quick. We can just do `anova(fit.abc)` and obtain the following output:

```
> anova(fit.abc)

Analysis of Variance Table
          Df  Sum Sq Mean Sq F value    Pr(>F)
rpm        1  139.08  139.08 16.7641  0.000415 ***
tool       2 1422.47  711.23 85.7321 1.174e-11 ***
rpm:tool   2   59.69   29.84  3.5974  0.043009 *
Residuals 24  199.10    8.30
```

What we obtain at the row entitled with `tool` is the test statistic and the p-value for the second null hypothesis from above. It is way below 0.05, we thus have very strong evidence that at least the main effect should be kept in the model, i.e. for the fact that there is a (vertical) shift in life time for the different tool types.

Then, the row with title `rpm:tool` contains the partial F-test according to the first null hypothesis from above. It checks whether the interactions can be kicked out of the model, i.e. whether all the three tool types have the same slope. This is weakly significant, there is thus some mild statistical evidence that there is a difference in life time diminishment caused by the speed for tool types A, B and C.

7.8 Categorical Input as a Substitute for Quantitative Predictors

Quantitative Predictors can also be represented by indicator variables. In the lathe example from above, we could for example categorize the continuous predictor `rpm` into bins ranging from 400-600rpm, 600-800rpm, and 800-1000rpm.

There does not seem to be an advantage for doing so, and in this example, there is in fact none. Also, the disadvantage of this approach is that more parameters are required to represent the information of the continuous predictors. Thus, we increase the model complexity by this categorization. However, under the presence of enough data, this is sometimes desired, because it does not require the analyst to make any prior assumptions about the functional form of the relationship between the response and the regressor variable.

Another advantage of the categorization approach is that it allows dealing with missing observations, without having to delete them. If they are numerous in a certain predictor, we could just categorize it, and assign all observations with missing information in that predictor the label “unknown”. Within the model, we would just estimate the effect of unknown status in that predictor.

Such a categorization of continuous predictors is in some fields quite popular among data analysts. The approach is also known as “poor man’s GAM”, an expression you can only understand after we discussed the proper GAMs in chapter **Fehler! Verweisquelle konnte nicht gefunden werden.**.

7.9 More than one Indicator Variable

Of course it is perfectly valid to have regression models where there is more than one categorical input variable. Sometimes even all the predictors can be categorical. Models of this last type are called analysis-of-variance models. While they can still be treated with the regression methodology we have acquainted so far, they are also the subject of more specific ANOVA courses.

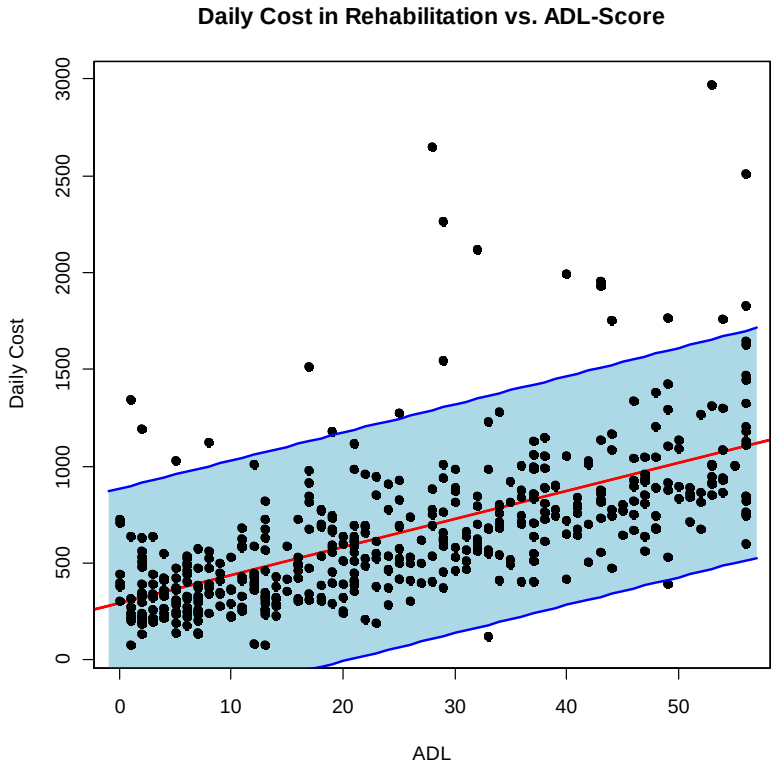
8 Transformations

In the previous chapters, we have discussed some extensions to linear modeling (e.g. polynomial regression, categorical input), as well as some strategies for dealing with violated assumptions (e.g. weighted regression, robust regression).

Here, we will discuss something that is in between: transformations of either the predictors and/or the response broaden the versatility of the linear modeling approach. And such transformations often help in cases where the model assumptions are violated.

8.1 Example: Positive Skewness

We start this section with an example that has several features that are often encountered in practice. The dataset shown here is taken from a study conducted at ZHAW. The goal was to predict the daily cost in rehabilitation with several predictors describing the patient's current state and his socio-demographic background. To keep the example simple and illustrative, we here restrict to one single predictor named ADL. This is a score describing how independent a patient can conduct the *A*ctivities of *D*aily *L*ife (i.e. dressing, eating, washing, etc.). We have data from 469 patients.

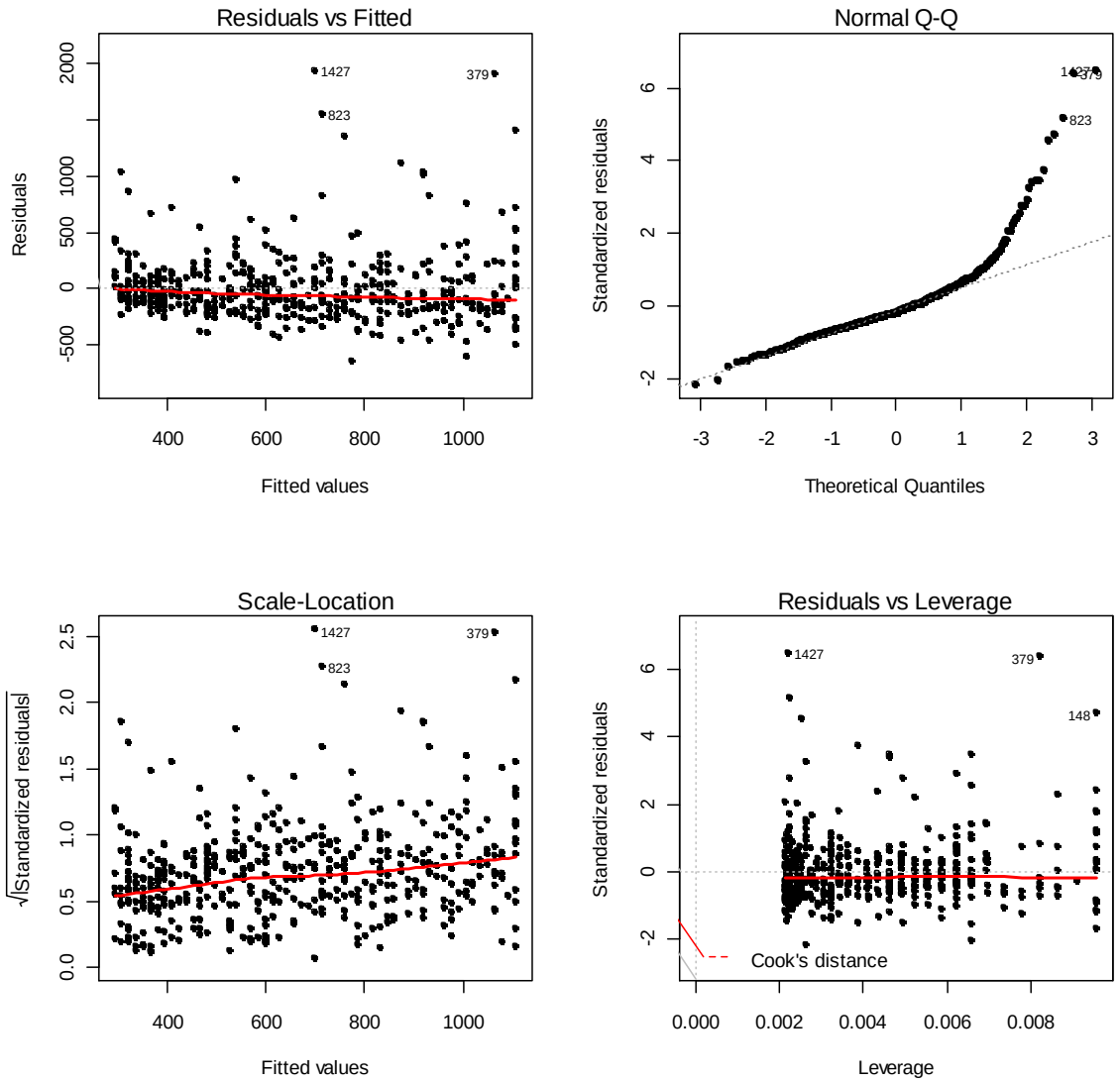


The scatter plot, including the regression fit and a prediction interval is shown above. At first sight, the fit does not seem to be too bad. However, there are some problems that do only appear when one looks at the data more closely, and when one also inspects the diagnostic plots.

1) The Tukey-Anscombe plot shows a smoother which is only slightly off the x-axis, but the conjecture that the expectation for the errors is different from zero is on a solid ground. This is also visible from the scatter plot, although much less clearly so. In any case, this is a first model violation.

2) Then, the Scale-Location plot shows a non-constant error variance. The higher the fitted values are, the bigger the variance is. Moreover, the residuals show a strong positive skewness, and are certainly not normally distributed. All of this is also visible in the scatter plot, but again, much less clearly so. In any case, here we have two further model violations.

3) While there are no influential data points, there is another problem with the regression model from above: the prediction interval takes values which are below zero. Because the response variable is daily cost, this should not be the case. In fact, with our model even the fitted values could be negative, While this is not a violation of the model assumptions in a strict sense, it is still an strongly undesired property.



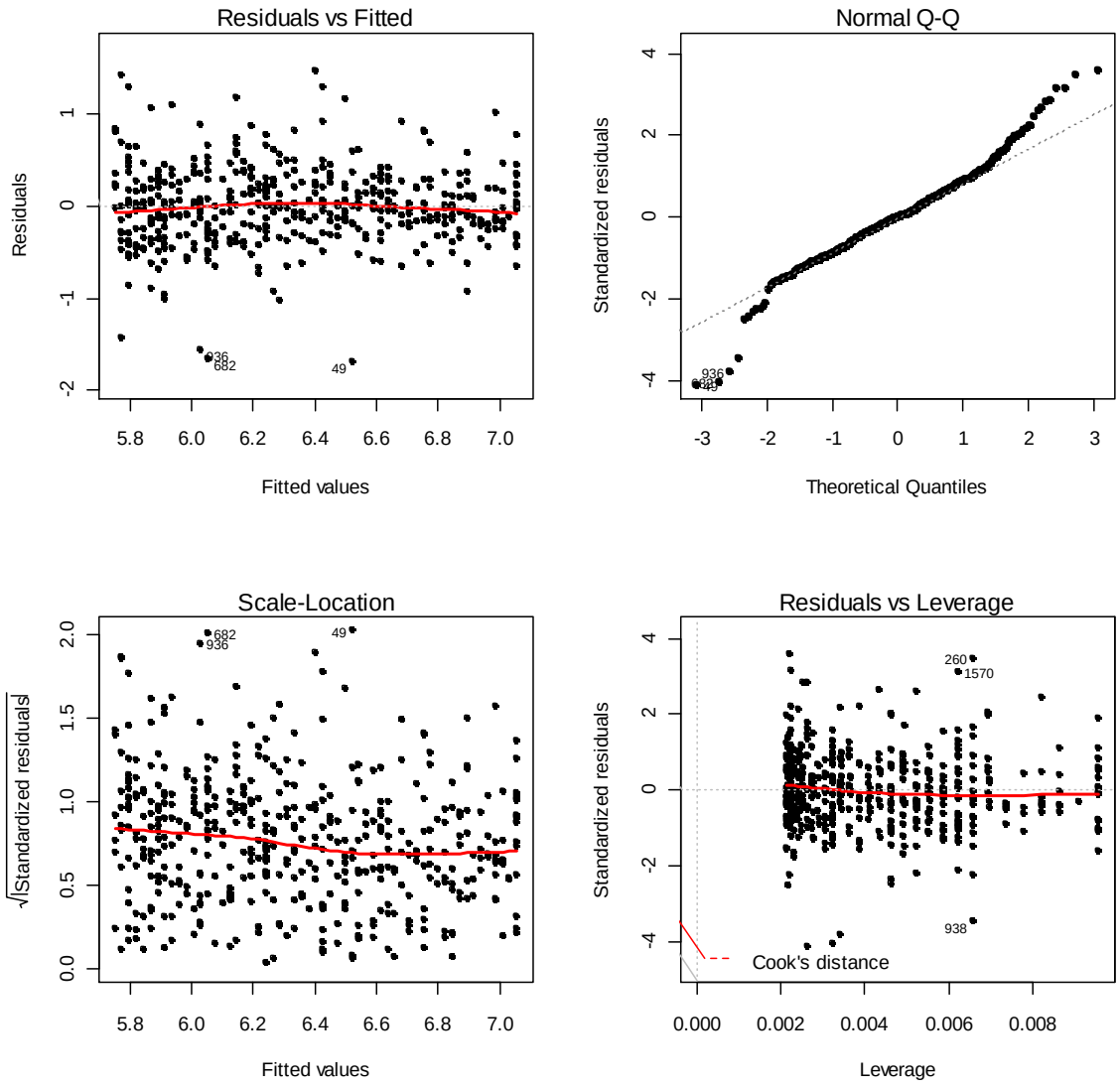
8.2 Logged Response Model

We call a situation like in the example above the “positive skewness syndrome”. Is there a remedy? The answer is yes: it generally helps if we log-transform the response variable, this often improves all our problems simultaneously. However, the logged response model also has quite a few implications, on which we will turn our attention later. Let us first formulate the model:

$$Y' = \log(Y) = \beta_0 + \beta_1 x + \varepsilon$$

In the original scale of the response variable, we can write the logged response model as:

$$Y = \exp(\beta_0 + \beta_1 x) \cdot \exp(\varepsilon).$$



Note that the errors have the usual additive effect only in the logged response model, on the original scale, they are multiplicative. Moreover, because we prefer Gaussian distribution for ε, we usually require the errors on the original scale,

$\exp(\varepsilon)$, having a lognormal distribution. As a next step, we would fit the logged response model to the example from above, and do the diagnostics:

Indeed, the logged response model is much better than the one on the original scale. Zero expectation for the errors seems now plausible. The error distribution is slightly long-tailed, but symmetric. Also the error variance is more constant now, although it seems as if it would even decrease a little with increasing fitted value.

Note that in practice, we did proceed with this model. Some alternative transformations were tried, but no other simple one managed to give an equally good overall result with improving the model violations. For dealing with the long-tailed errors, robust regression was also considered. However, our main focus was on prediction (of daily cost) and less on inference: due to the absence of influential data points, there were hardly any differences between the robust and the ordinary least square regression.

Dealing with Zero Response

There are some peculiarities with the logged response model that need further attention. First, we can only use the logged response model in cases where the response is only positive. However, we often observe positive skewness also in problems where the response can and does take the value 0, too.

Deleting these observations to just run an analysis systematically biases the results, and should in any case be avoided! What we can do is to slightly change the log-transformation, so that we can deal with the zero responses. One usually adds a constant c – this is perfectly valid. We only need to review the choice of c.

The most popular choice is $c = 1$. However, with this choice, the effect of the additive shift depends on the scale of the response. A better choice for c, which avoids these difficulties, is to set it equal to the smallest positive response value.

Back Transforming the Fitted Values

When the response variable has been transformed, the fitted values will be in the transformed scale. However, in our example from above, nobody is interested in the log of daily cost – thus, it is necessary to convert the predicted values back to the original units.

In principle, this is simply a matter of back transforming the fitted values. For example, in the logged response model, we would use

$$\hat{y} = \exp(\hat{y}')$$

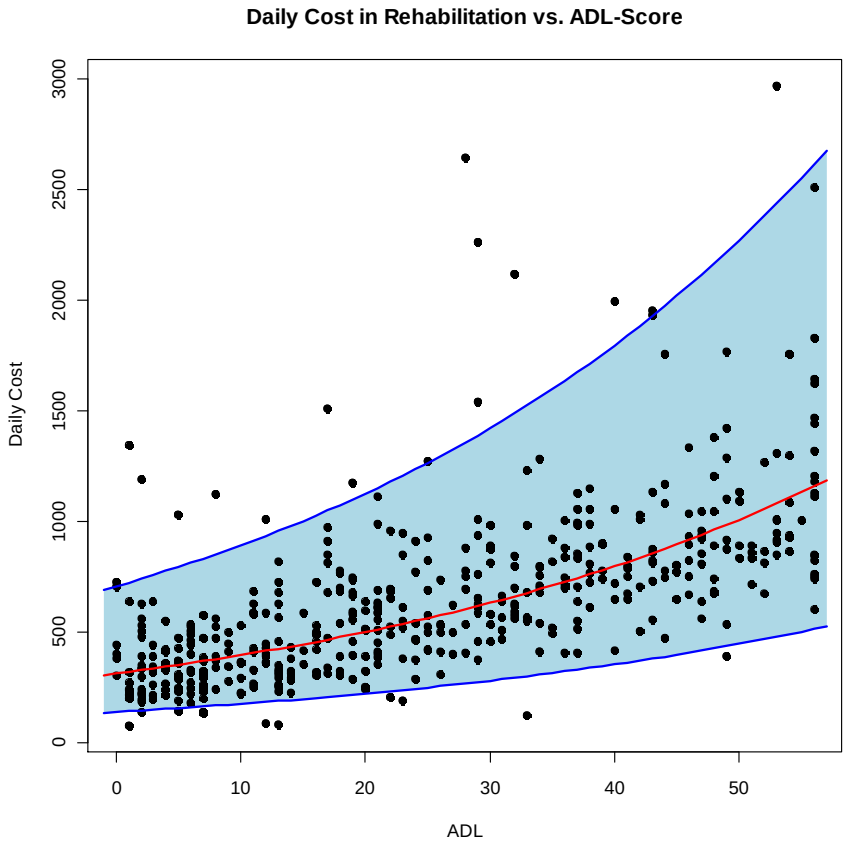
as predictions. This is a valid choice; however, the predicted values are “only” an estimate of the median of the response, but not the mean, i.e. they are biased.

In practical application, unbiased prediction (in the original units) is often a must. This is also the case in our daily cost example, where we want the total sum of the predicted values to be equal to the actual sum of daily cost observations. For the

logged response model, we can compare mean and median of the lognormal distribution. This shows how we need to back transform the fitted values in order to obtain unbiased predictions:

$$\hat{y} = \exp\left(\hat{y}' + \frac{\hat{\sigma}_{\varepsilon}^2}{2}\right)$$

Confidence and prediction intervals can be directly converted from one metric to the other. Thus, if an interval in the logged response model is given by $[l, u]$, then we use $[\exp(l), \exp(u)]$ for the original scale. The reason why this works is because interval estimates are percentiles of a distribution and percentiles are not affected by monotone transformations. However, there is no assurance that the resulting intervals in the original units are the shortest possible intervals.



In the scatter plot shown above, we have the regression line and the prediction interval that were obtained by back transforming from the logged response model. This here seems a lot more reasonable, compared to the scatter plot at the beginning of section 8.1.

Interpretation of the Regression Coefficients

The regression coefficients will need to be interpreted with respect to the transformed scale. There is no straightforward way of back transforming that can be interpreted in the original scale. For the logged response model, we have:

$$\begin{aligned} \log(\hat{y}) &= \hat{\beta}_0 + \hat{\beta}_1 x_1 + ... + \hat{\beta}_p x_p \\ \hat{y} &= \exp(\hat{\beta}_0)\exp(\hat{\beta}_1 x_1)...\exp(\hat{\beta}_p x_p) \end{aligned}$$

An increase by one unit in x_1 would multiply the fitted value in the original scale with $\exp(\hat{\beta}_1)$. Thus, when the logged response model is used, the regression coefficients can be interpreted in a multiplicative rather than an additive manner.

8.3 Variance-Stabilizing Transformations

The logged response model was presented in its own section above because it is the most widely used variance stabilizing transformation, i.e. the one that is often most appropriate for response variables that can only take positive values. However, there are other situations, where different transformations are better.

In applied regression analysis, the following transformations are known as “first aid transformations”. If there are no (practical) reasons strictly speaking against their application, one should always use them:

- **Absolute values and concentrations, where** $y \geq 0$:
  log-transformation: $y' = \log(y)$
- **Count data, where** $y \geq 0$:
  square-root transformation: $y' = \sqrt{y}$
- **Proportions** $(0 \leq y \leq 1)$
  arcsine transformation: $y' = \sin^{-1}\left(\sqrt{y}\right)$

While it is most important to apply these to the response, it is mostly also advisable to transform predictors of these types in exactly the same way.

9 Variable Selection

There is usually a wealth of predictors and potential predictors available to explain a target variable of interest. Here, we show how we can select the "best" subset of predictors. We first motivate why this is useful, then turn our attention to some strategies for finding the subset, and also discuss the meaning of the word "best" in terms of regression modeling.

9.1 Why Variable Selection?

Only in some rare special cases, we do already know the functional form with which a few specified predictors $x_1,...,x_p$ explain the response Y In these cases, we would still be interested in learning about the regression coefficients, do some hypothesis tests, and potentially give some prediction and confidence intervals.

Much more often however, we want to use ***regression*** in an ***explorative fashion***. This is usually when we do not know previously how the relation between response Y and some potential predictors x_j is, usually we do not even know which predictors to use. In these situations it has become standard to collect data from many potential predictors. Our goal with regression analysis will then be to learn not only about the form of the relation between response and predictors, but also about required variable transformations, and probably most importantly, about the ***predictors*** that have a ***relevant impact*** on the outcome.

Thus, there is some motivation for variable selection arising purely from applied aspects. However, there is some more reasoning for keeping a model small, lying more in technical aspects.

1) We generally want to explain the data in the simplest way, and thus remove redundant predictors. This follows the idea that if there are several plausible explanations (i.e. models) for a phenomenon, then the simplest is the best.

2) Unnecessary predictors in a regression model will add noise to the estimation of the coefficients for the other predictors. Or in other words: we need more observations to have the same estimation accuracy.

3) What is stated in 2) above becomes even more pronounced if there is collinearity among the predictors, i.e. if there are too many variables trying to do the same job. Removing excess predictors facilitates interpretation.

4) If the model is to be used for prediction, we will be able to save effort, time and/or money if we do not have to collect data for predictors that are redundant.

Please note that variable selection is not a method. It is a process that cannot even be separated from the rest of the analysis. For example, outliers and influential data points will not only change a particular model – they can even have an impact on the model we select. Also variable transformations will have an

impact on the model that is selected. Some iteration and experimentation is often necessary for variable selection, i.e. to find smaller, but better models.

Example

We will illustrate the variable selection process on a data coming from the US Bureau of Census. They contain information from the 50 US states recorded around 1970. The target variable is life expectancy, and there are 7 continuous predictors:

`Population`: as of July 1, 1975

`Income`: per capita income, estimated 1974

`Illiteracy`: percent of populations, 1970

`Murder`: number of murders per 100'000 people (1976)

`HS.Grad`: percent high-school graduates (1970)

`Frost`: number of days with minimum temperature below freezing

`Area`: land area in square miles

These are raw data where some variable transformations are required. `Population` and `Area` are much skewed predictors. Thus, we apply a log-transformation on them. Moreover, `Murder` and `Frost` are based on counts. This is why a square root transformation is appropriate for these. Finally, `Illiteracy` and `HS.Grad` are proportions, for which we do an arcsine transformation. `Income` is almost symmetrically distributed, and does not need any transformation.

The model diagnostics (not shown here) look fine when one is using the transformed input variables. The summary output is as follows – so now what are the driving predictors for life expectancy?

```
> summary(lm(Life.Exp ~ ., data = state.trsf)

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.878e+01  2.806e+00  24.511  < 2e-16 ***
Population   2.799e-01  1.238e-01   2.261   0.0290 *
Income      -5.601e-05  2.345e-04  -0.239   0.8124
Illiteracy  -5.885e-01  7.663e+00  -0.077   0.9392
Murder      -1.510e+00  2.188e-01  -6.905 1.99e-08 ***
HS.Grad      5.845e+00  2.458e+00   2.378   0.0220 *
Frost       -9.968e-02  4.821e-02  -2.067   0.0449 *
Area         3.361e-02  1.036e-01   0.325   0.7472
---
Residual standard error: 0.7109 on 42 degrees of freedom
Multiple R-squared: 0.7596,   Adjusted R-squared: 0.7195
F-statistic: 18.96 on 7 and 42 DF,    p-value: 3.867e-11
```

We learn from the summary that the signs of some of the coefficients match plausible explanations concerning how the predictors might affect the response. Higher murder rate decreases life expectancy which certainly confirms our a priori ideas. Additionally, we observe that there are some weakly significant variables: `Population`, `HS.Grad` and `Frost`, and a few which are non-significant: `Income`, `Illiteracy`, and `Area`.

The question is now how we could find out which ones are required in this model, and which ones can be omitted. Remember again that is not a valid approach to kick all predictors with non-significant p-values out of the model simultaneously.

9.2 Backward Elimination

We have seen above that reducing the model by more than one variable at a time is problematic. However, we could do some stepwise ***backward elimination***. This is the simplest of all variable selection procedures. It can easily be run without any special software. On the other hand, it can only be conducted if there is a reasonable balance between the number of predictors and the number of observations. There is a general rule of the thumb, saying that there should be at least 5 times as many observations.

We start with a model where all potential predictors are included. We then remove the predictor with the highest p-values greater than α_{crit}. Next, we refit the model and again remove the least significant predictor, provided its p-value is greater than α_{crit}. Sooner or later, all "non-significant" predictors will be removed, and the selection process will be complete. One usually uses the arbitrary $\alpha_{crit} = 0.05$, although for prediction, often a 0.15 or 0.20 cutoff yields better results.

In our example, `Illiteracy` is the least significant predictor with a p-value of 0.939. We will thus omit it, and fit the model again. From the summary output (not shown here), we learn that `Income` is now the least significant predictor and has p-value of 0.804. Thus, income is omitted, and the model gets refitted. Then, `Area` has a p-value of 0.675, and is excluded. Now, the least significant predictor is `Population`. Since it has a p-value as low as 0.012, the backward elimination is terminated.

```
> summary(lm(Life.Exp ~ Population + Murder + HS.Grad +
                        Frost, data = state.trsf)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 68.78767    1.75860  39.115  < 2e-16 ***
Population   0.27663    0.10600   2.610 0.012259 *
Murder      -1.49218    0.17046  -8.754 2.83e-11 ***
HS.Grad      5.83746    1.37130   4.257 0.000104 ***
Frost       -0.09671    0.03669  -2.636 0.011477 *
---
Residual standard error: 0.6888 on 45 degrees of freedom
Multiple R-squared: 0.7582,   Adjusted R-squared: 0.7367
F-statistic: 35.28 on 4 and 45 DF,    p-value: 2.416e-13
```

When comparing this output with the full model from 9.1, we observe that `Murder`, `HS.Grad`, `Frost` and `Population` now have lower p-values than initially. The reason is that some of their predictive power was covered by the removed variables `Area`, `Income` and `Illiteracy`. This does not come as a surprise, because the percentage of high school graduates is certainly correlated with illiteracy, and also income.

Moreover, it is important to understand that the removed variables are still related to the response, as a regression of `Life.Exp` on `Area`, `Income` and `Illiteracy` would show. We do not show the output here, and leave this as an exercise.

9.3 Forward Selection

This is an analogue to the backward elimination. However, forward selection starts with an empty model, i.e. a model where only the intercept, but no predictors are present. The, we add predictors in a stepwise manner: in every step, we add the one which is the most important one, i.e. has the lowest p-value, when added to the model. We do so, until no terms with p-values lower than α_{crit}, usually set equal to 0.05, can be added to the model.

This approach is feasible also in situations where there are more predictors than there are observations. Since it is also computationally cheap, it was popular in the early days of regression analysis.

9.4 Stepwise Regression

This is a combination of backward elimination and forward selection, and is what `R` does by default in function `step()`. It addresses the situation where variables are added or removed early in the process and we want to change our mind about them later. At each stage of the selection process, a variable may be added or removed. As before, we can base our decisions on the p-values from individual hypothesis tests.

9.5 Testing Based Variable Selection

With the backward, forward and stepwise approaches, the decisions for variable selection were based on individual hypothesis tests. While this is computationally cheap, it also has some drawbacks:

1) Because of the “one-at-a-time” nature of adding/dropping predictors, it is possible to miss the “best” model.

2) The p-values should not be treated too literally. We are subject to the multiple testing problem. Moreover, the removal of less significant predictors tends to increase the significance of the remaining ones. One thus often overstates the importance of the remaining predictors.

3) The testing based variable selection procedures are not directly linked to the final objectives of prediction or explanation. With any variable selection method, it is important to keep in mind that model selection cannot be divorced from the underlying purpose of the analysis.

4) The testing based procedures tend to select models that are smaller than desirable for prediction purposes. Consider the following simple example to understand this: in a simple linear regression, we would go with the intercept only if the predictor is not significant. However, for prediction it could still be better to use it.

9.6 Criterion Based Variable Selection: AIC/BIC

On the other hand, if we have some idea about the purpose for which a model is intended, we might propose some measure of how well a given model meets that purpose. It would be appealing to scan a big variety of different models. In the case where there are a fixed number of m predictors, we can build 2^m different regression models: for each variable we can decide, whether it will be included in the model or not.

Obviously, this ***all subsets regression*** approach only works when the number of potential predictors is limited, else it will be too time consuming and we need to economize on computing time. Also note that we here cannot any longer use the p-values from individual hypothesis tests as a criterion, but we need something, that judges the quality of the model more generally. The following quantities are potential candidates:

a) Coefficient of determination R^2

b) Test statistic or p-value of the global F-test

c) Estimated error variance $\hat{\sigma}_\varepsilon^2$

For a fixed number of predictors m', they will all lead to the same order among all possible models. Another property they share is that they somehow judge the goodness-of-fit, and thus generally tend to improve if more terms are added to the model.

However, as we can easily imagine, bigger models are not necessarily better than smaller ones. It would thus be preferable to employ a criterion which is not only based on goodness-of-fit, but also penalizes for the model size. A potential candidate is the adjusted R^2. In practice, one nowadays almost exclusively uses the ***Akaike*** or ***Bayes Information Criteria*** (AIC/BIC), which are defined as follows:

$$\begin{aligned} AIC &= -2\max(\log likelihood) + 2p \\ &= const + n\log(RSS/n) + 2p \end{aligned}$$

and

$$\begin{aligned} BIC &= -2\max(\log likelihood) + p\log n \\ &= const + n\log(RSS/n) + p\log n \end{aligned}$$

Because the constant ($const$) is the same for a given dataset and any assumed error distribution, it can be ignored for regression model comparisons on the same data.

The goal in practice is to find the model which minimizes AIC or BIC. Larger models will fit better, and thus have smaller residual sum of squares RSS. However, they use more parameter and are thus penalized by the terms $2p$ (AIC) and $p\log n$ (BIC). Note that BIC punishes larger models more heavily and so will tend to prefer smaller models in comparison to AIC.

Finally, we note here that the use of AIC/BIC is not limited to all subset regression. These criteria can also be applied in the backward, forward or stepwise approaches. In R, variable selection is generally performed by function `step()`, which by default employs the stepwise approach with AIC as a criterion. We illustrate this with the state data:

Example

```
> step(lm(Life.Exp ~ ., data=state.trsf))
Start:  AIC=-26.84
Life.Exp ~ Population + Income + Illiteracy + Murder +
           HS.Grad + Frost + Area

             Df Sum of Sq    RSS      AIC
- Illiteracy  1    0.0030 21.231 -28.8291
- Income      1    0.0288 21.256 -28.7682
- Area        1    0.0532 21.281 -28.7109
<none>                    21.228 -26.8361
- Frost       1    2.1603 23.388 -23.9903
- Population  1    2.5844 23.812 -23.0918
- HS.Grad     1    2.8591 24.087 -22.5183
- Murder      1   24.0982 45.326   9.0927
```

[Output partly omitted...]

```
Step:  AIC=-32.55
Life.Exp ~ Population + Murder + HS.Grad + Frost

             Df Sum of Sq    RSS     AIC
<none>                    21.347 -32.555
- Population  1     3.231 24.578 -27.508
- Frost       1     3.296 24.643 -27.376
- HS.Grad     1     8.596 29.944 -17.635
- Murder      1    36.352 57.699  15.161
```

When we did a backward elimination based on the AIC criterion, the sequence of predictor removal was exactly the same as when our decision was based on hypothesis testing. The process also stops at the same spot, and the resulting

model is exactly equal to the one we found before. Please note that this is a coincidence – one generally does observe differences when using different variable selection schemes.

A final remark on variable selection: every procedure may yield a different "best" model. However, if we could obtain another sample from the same population, even a fixed procedure might result in another "best" model. Thus, there is an element of chance in this declaration. How can we mitigate this in practice? It is usually advisable to not only consider the "best" model according to a particular procedure, but to check a few more models that did nearly as good, if they exist.

9.7 Correct Treatment of Hierarchical Models and Categorical Predictors

Some regression models have a natural hierarchy. For example in polynomial models, x^2 is a higher order term than x. When selecting variables, it is important to respect this hierarchy. Lower order terms should not be removed from the model before higher order terms in the same variable. As an example, consider the polynomial model:

$$Y = \beta_0 + \beta_1 x + \beta_2 x^2 + \varepsilon$$

Suppose we fit this model and find that the regression summary shows that the term in x is not significant, but the term x^2 is. If we then remove the x term, our reduced model would become

$$Y = \beta_0 + \beta_2 x^2 + \varepsilon$$

However, suppose we make a scale change $x \mapsto x + a$. Then, the above reduced model would look different again:

$$Y = \beta_0 + \beta_2 a^2 + 2\beta_2 a x + \beta_2 x^2 + \varepsilon .$$

Thus, the first order term x has reappeared. Scale changes should not make any important change to any reasonable model, but in this case an additional term has been added. This is not desirable and illustrates why we should not remove lower order terms in the presence of higher order terms: because we do not want interpretation to depend on the choice of scale.

Models with Interactions

For models with interactions, it does not result in a valid model if a main effect is removed, but the interaction is kept within the model. The reasons are similar as with the polynomial models above. We leave it as an exercise to study the effects of removing a main effect, but keeping the interaction.

Categorical Input

When there are categorical predictors, we need dummy variables to incorporate them into the model. Now if a single dummy coefficient is non-significant, we

cannot just kick this term out of the model! Thus, we have to test the entire block of indicator variables. When we work manually and testing based, this will be done with a partial F-test, whose p-value can be compared against the ones from the other variables (even if they may arise from individual hypothesis tests).

When we use a criterion based approach with function `step()`, then `R` deals correctly with categorical predictors, and also with interactions and hierarchical models. Be careful with other software though – not all statistics packages can correctly handle variable selection with such input.

9.8 The Lasso

The result of a variable selection procedure is a subset of predictors that will be included into the model. As stated above, this is a random set. If the data were only slightly changed, or if we obtained another sample from the same population, that set might be completely different. Some predictors x_j might no longer be part of the model; respectively their regression coefficient β_j is suddenly equal to zero. This is a non-continuous behavior that introduces some arbitrariness. However, there is a procedure where variable selection is done in a steadier, smoother way: the Lasso, a ***penalized regression*** approach.

The idea behind is to complement the ordinary least squares criterion for model fitting with a penalty term for the magnitude of the coefficients. Thus, we minimize

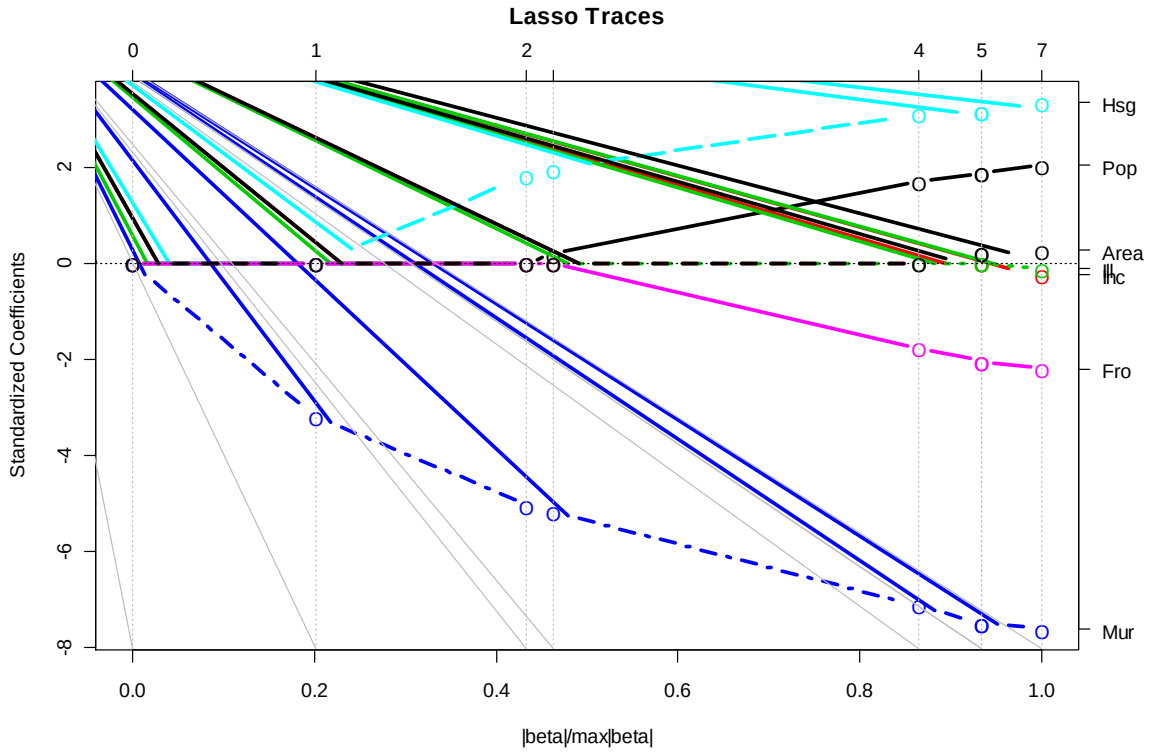
$$Q(\beta, \lambda) = \sum_i r_i^2 + \lambda \sum_i |\beta_j|$$

The procedure has one tuning parameter λ which balances goodness-of-fit against the penalty term. As an alternative formulation, minimization of Q can also be seen as minimizing the sum of squared residuals under the condition that the sum of absolute values of the regression coefficients is $\leq c$.

When c is sufficiently large, then the least square estimates $\hat{\beta}_j$ will fulfill the condition $\sum_i |\hat{\beta}_j| \leq c$, and the solution is the same. However, if we choose smaller c, the regression coefficients will be smaller in magnitude, i.e. they are shrunken towards zero. In order to make good choices for c, it is better to use

$$b = c / \sum_j |\hat{\beta}_j|$$

as a parameter, since this is scale-free and always between 0 and 1. It has been shown mathematically and empirically, that the condition leads to the property that soon the first coefficient will be equal to 0, where it will stay with diminishing c. But not only this: the more we lower c, the more coefficients vanish. We obtain a variable selection procedure that resembles the backward elimination approach, though it is based on entirely different criteria.



When the Lasso is applied to the State dataset, we obtain the same result as with the backward elimination approach. The first three variables which are excluded are again `Illiteracy`, `Income` and `Area`.

10 Missing Data

It is not uncommon in practice that some data points are missing. This causes some problems when doing regression analysis and thus, we need to deal with the missing data. Obviously, finding the missing data points would be the best strategy. If this is a viable option, we strongly recommend doing so.

However, often this will be impractical and we need some alternatives. The first and foremost question you should ask yourself is why the data are missing. This could be:

a) Just randomly, non-informatively for the goal in your analysis. Then fixing up the missing data is comparatively easy.

b) Systematically with respect to the goal of the analysis. For example, patients who dropped out of a drug study because they believed their treatment was not working. Or, a chemical reaction in a certain configuration which just took too long to complete, etc.

Unfortunately, there are no easy solutions for case b), where the findings of the study will be ultimately biased by the missing values. This teaches us that we should avoid this type of missing at any cost. Case a) is somewhat less problematic. We here provide several simple fix-up alternatives for it:

1) When there are plenty of data, the simplest solution is to just omit the incomplete cases. This is a reasonable and valid strategy if only a relatively small number of observations are lost.

2) We could fill in or impute the missing values, i.e. use the rest of the data to predict the missing data point. The easiest method would be to just replace a missing value in a predictor with the average value of that predictor.

3) A more sophisticated filling-in strategy would be to use regression on the other predictors with complete data on the one where the data point is missing.

4) Maximum likelihood methods can be used assuming multivariate normality of the data. The EM algorithm is often used here. We will not elaborate on this here, but the basic idea is to treat missing values as nuisance parameters.

Example

We will randomly delete some five observations from the above state dataset to show and study the effects of fixing alternatives 1)-3). We still use `Life.Exp` as the response variable, and use `Murder` (2 `NA` introduced), `Frost` (3 `NA` introduced), `HS.Grad` and `Population` as predictors. If we try to fit a regression models with that modified dataset, we obtain:

```
> summary(lm(Life.Exp ~ Population + Murder + HS.Grad +
                        Frost, data=state.trsf)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 68.43923    1.91211  35.793  < 2e-16 ***
Population   0.31831    0.11248   2.830 0.007247 **
Murder      -1.43049    0.17821  -8.027 7.26e-10 ***
HS.Grad      5.75964    1.45363   3.962 0.000298 ***
Frost       -0.10537    0.03838  -2.746 0.009006 **
---
Residual standard error: 0.6824 on 40 degrees of freedom
  (5 observations deleted due to missingness)
Multiple R-squared: 0.7515, Adjusted R-squared: 0.7266
F-statistic: 30.24 on 4 and 40 DF,  p-value: 1.293e-11
```

R automagically removes all cases where at least one of the values, be it in the response or one of the predictors, and runs the regression analysis without complaining. It is only visible in the remark of the summary output and with the reduced number of degrees of freedom that some observations were deleted.

So this was alternative 1). When we compare, we see that there are some slight changes in all the numerical values in the summary output, but nothing that seems of tremendous importance. Thus, we can say that 1) would probably be safe here, but note that we in practice cannot usually compare against the fit without any missings. Next, we decide to apply 2), and replace the 3 missing data points in Frost with the mean of that predictor.

```
> missings   <- which(is.na(state.trsf$Frost))
> mean.Frost <- mean(state.trsf$Frost, na.rm=TRUE)
> state.trsf$Frost[missings] <- mean.Frost
```

The replacement value is 9.85, when the removed ones were 0, 10.68 and 13.19 for the states of Hawaii, Kansas and New Hampshire. Especially for Hawaii, the value we replace with seem questionable. In general, we would thus only apply strategy 2) in problems where there are many predictors and in only few, data are missing – then it may be beneficial to profit from the information which is present in the other predictors, and not delete the entire case. We study the output:

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 66.80292    1.98216  33.702  < 2e-16 ***
Population   0.36425    0.12058   3.021 0.004233 **
Murder      -1.34124    0.18860  -7.112 8.87e-09 ***
Frost       -0.03007    0.04800  -0.626 0.534333
HS.Grad      6.15488    1.50475   4.090 0.000185 ***
---
Residual standard error: 0.7298 on 43 degrees of freedom
  (2 observations deleted due to missingness)
Multiple R-squared: 0.7288, Adjusted R-squared: 0.7036
F-statistic: 28.89 on 4 and 43 DF,  p-value: 1.092e-11
```

Now, there are only 2 observations missing, i.e. the ones with `NA` values in Murder. However, our imputation with the mean had quite a big effect on the summary. Predictor `Frost` is no longer significant, and the coefficient decreased, too. It seems as if we would do better with strategy 1) than strategy 2) here. So now, we try alternative 3), and predict the missing observations in `Frost` from a regression of `Frost` on the remaining predictors `Population`, `Murder` and `HS.Grad`:

```
missing <- which(is.na(state.trsf$Frost))
fit.imp <- lm(Frost~Population+Murder+HS.Grad, state.trsf)
predval <- predict(fit.imp, newdata=state.trsf[missing,])
state.trsf$Frost[missing] <- pred.val
```

It is clear that strategy 3) is of no benefit with orthogonal predictors and the more collinearity there is, the better it works. In our case, it seems at least doubtful beforehand if it is possible to predict the number of freezing days from population, percentage of high school graduates and murder rate. Indeed, we have:

```
> pred.val
      HI       KS       NH
11.43693 11.00075 12.27640
```

For the state of Hawaii, we now impute an even bigger number of freezing days, while the deleted, true value was 0. The impact on the fit is similar as it was with strategy 2).

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 66.57107    2.00466  33.208  < 2e-16 ***
Population   0.37502    0.12243   3.063 0.003771 **
Murder      -1.32308    0.19082  -6.934 1.60e-08 ***
Frost       -0.01595    0.04908  -0.325 0.746796
HS.Grad      6.10990    1.51291   4.039 0.000218 ***
---
Residual standard error: 0.7322 on 43 degrees of freedom
  (2 observations deleted due to missingness)
Multiple R-squared: 0.727,    Adjusted R-squared: 0.7016
F-statistic: 28.62 on 4 and 43 DF,  p-value: 1.256e-11
```

Also 3) does not yield any benefit over 1) for the imputation for Frost. Thus, we would proceed with 1) here and definitely leave the states with missing values out of the analysis. Moreover, we here state that things are similar with variable Murder, but leave producing the results as an exercise.

Synopsis

We conclude that it is not so simple regenerate and impute missing information. While there are cases where the mean or regression fill-in methods do provide an advantage, they are often useless or even make things worse. The success of the latter depends on the collinearity of the predictors – the imputed values will be more accurate the more collinear the predictors are. Also note that both fill-in

techniques will introduce a bias towards zero in the regression coefficients while tending to reduce the variance.

For situations, where a substantial proportion of the data is missing, strategies 1-3) usually do not work well. We recommend using more sophisticated approaches then, which are beyond the scope of this course.

11 Modeling Strategies

By now, we have learnt quite a bit of techniques for regression analysis, i.e. estimation, diagnostics, transformations and variable selection). What has been left dubious is in which order to apply these, and what is generally a good strategy to proceed.

There is no definite answer to this: regression analysis is the search for structure in the data and there are no hard-and-fast rules about how it should be done. Professional regression analysis can be seen as an art and definitely requires skill an expertise – one must be alert to unexpected structure in the data.

Thus far, no one has implemented a computer program for automagically conducting a complete regression analysis. Because of the difficulties in automating the assessment of regression graphics in an intelligent manner, we cannot expect that this will be accomplished soon. The human analyst is capable of assessing plots in the light of contextual information, and that is his/her big advantage.

11.1 Guideline for Regression Analysis

We here try to give some rough guidelines on the process of a regression analysis which is aimed at finding a model which accurately describes the data and does not show any systematic shortcomings.

0) Getting acquainted with the data and preprocessing

This includes learning to know the meaning of all variables in the dataset. It is usually very helpful to use short, but informative names. The data are checked for impossible or highly unlikely values, gross errors and outliers. If possible, these are corrected, in all other cases their values should be set to `NA`. Finally, we investigate the frequency of missing values, and whether they appear in systematic patterns. If yes, we should be very careful when drawing conclusions from our analysis results.

1) First-aid transformations

Some general statistical considerations as well as specific knowledge lead to transformations to bring all variables on a plausible scale – often a transformed one. If there are no specific reasons against, the first-aid transformations from 8.3 should always be applied.

2) Fitting a big model

The first model we try to fit is a big one which potentially contains too many predictors. In particular, we use:

- all predictors, if their number does not exceed $n/5$.
- if there are more than $n/5$ predictors, we may try to only include the ones for which we previously expect/know that they have some impact on the response.
- if b) still leads to a model with more than $n/5$ predictors, we would do some forward search with a non-restrictive p-value (e.g. 0.2) as a criterion.

If we have previous knowledge that lets us assume that there are interactions between the predictors, we would include these, too.

3) Model diagnostics

We would do the diagnostic plots to check for normality and outliers in the residuals, for checking constant variance and uncorrelated errors. If we see some anomalies, then these are some strategies which may mitigate the problems, or even fully correct them:

- in many cases, and for dealing with all of the problems from above, we can do a transformation of the response variable
- when all the other diagnostics look appealing but only the residuals are long-tailed (and symmetric), using robust regression is more efficient, i.e.. leads to more precise estimates.
- if we have previous knowledge that the variance is unequal (e.g. when the response is a mean), or if we just observe non-constant variance and cannot get rid of it by a transformation, we would choose to do weighted regression
- for correlated errors, we could try block building according to time or location of measurement and include this as a nominal predictor in the model. If this fails, we could use the generalized least squares approach to produce estimates that account for the error correlation

4) Non-linearities

We can plot residuals or partial residuals against the predictors. If some of these relations appear to be non-linear, we could try transformations, i.e. a different one than before, or add terms of higher order to the model. Another alternative would be to use more sophisticated techniques (Splines, GAM, …).

5) **Variable selection**

Preferably, we would now run an all subsets variable selection with either AIC or BIC as a criterion. This is only a viable option if there are not too many (say about 10) predictors. In other cases, we would work with a backward elimination, or better, a stepwise backward elimination scheme (e.g. the one implemented in R-function `step()`).

There are cases where it can be useful to do scatter plots of the residuals versus the eliminated predictors. Sometimes, this leads to insight about possible transformation – often though, when time is limited, one does without this.

6) **Interactions**

We can try whether (two-way) interactions terms between the predictors that remained in the model leads to a significant improve in goodness-of-fit. Interactions with predictors that are not in the current model are unwanted and rarely useful. If we decide to use one of these, the (non-significant) predictor should also be re-introduced into the model as a main effect.

7) **Influential data points**

We are now looking for data points which strongly attract the regression fit, i.e. influential data points, which can also be seen as multivariate outliers. If they exist, they often require deeper insight for deciding whether to consider or remove them. Comparing results with and without them can help, too.

8) **Do model and coefficients make sense?**

If there are predictors which are highly implausible, or if an estimated coefficient has the wrong sign, contrary to what we would expect from gut feeling or to what existing theory says, we would remove the predictors from the model, if there are no drastic changes to the fit.

If steps 4-8) substantially altered the model, then we should go back to 3) and repeat the diagnostics. It can also be helpful to perform 4-8) again.

This was a general strategy for analyses that were aimed at obtaining a good descriptive model. When the goal is testing a hypothesis, we would proceed similarly. However, we have to be careful with variable transformations, variable selection and collinearity. Often, the question dictates what can be done, and what cannot be done.

On the other hand, if the goal with a regression analysis is prediction, then proceeding according to the above guideline is still a good idea, though we can be a little more relaxed. We would usually not be too picky when doing variable selection, and also minor model violations would be tolerable, as long as the forecast is good. It may be advisable to check the generalization abilities of several competing models with out-of-sample data or cross validation,

11.2 Significance vs. Relevance

Statistical significance is not equivalent to practical significance. The larger the sample, the smaller your p-values for the same effect will be, so do not confuse p-values with an important predictor effect. With large datasets, one routinely produces statistically significant results even when the actual effects are practically unimportant. Would we for example really care that the test scores were 0.1% higher in one state than another, or that some medication reduced pain by 2%?

Because a model is usually only an approximation of the underlying reality, the exact meaning of the parameters is debatable at the very least. As a consequence, the precision of the statement such as $\beta_1 = 0$ is completely at odds with the approximate nature of the model. Moreover, it is highly unlikely that a predictor that one has taken the trouble to measure and analyze has exactly zero effect on the response. It may often be small, but it will not be zero.

This means in many cases, we know the point null hypothesis is false without even looking at the data. Furthermore, we know that the more data we have, the greater the power of our tests. Even small differences from zero will be detected with a large sample. Now if we fail to reject the null hypothesis, we might simply conclude that we did not have enough data to get a significant result. According to this view, the hypothesis test just becomes a test of sample size.

12 Extending the Linear Model

Linear models are central to the practice of statistics and can be seen as part of the core knowledge of any applied statistician. While they are very versatile, there are situations that cannot be handled within the standard framework. Here, we will take care of some of these.

12.1 What is the difference?

So far, the response Y was a continuous random variable whose range was (at least theoretically) reaching from minus to plus infinity. There are situations where this is clearly not the case – i.e. always, when we want to model and predict a response that is for example binary in $\{0,1\}$, or a proportion in $[0,1]$.

Then, applying the standard multiple regression framework will ultimately result in responses that are beyond the set of values which are foreseen in that problem. Thus, we need some additional techniques which can deal with these types of situations.

12.2 An Overview of the Framework

Depending on how exactly the response variable is, there are several different approaches. Here, we will give an overview of the most important ones.

Logistic Regression

In toxicological studies, one tries to infer whether a lab mouse survives when it is give a poisonous dose of a particular concentration. In human medicine, we are often interested in the contrary case: how much “dose” has an effect, i.e. clearly reduces pain or other symptoms. Here, the response variable is a binary variable in $\{0,1\}$. Our goal will be to study the outcome depending on one or several predictors.

A statistical model for this situation takes into account that for a given (intermediate) concentration, we will only have an effect on some of the subjects, but not on all of them. We are thus trying to model the relation between $P(Y_i = 1)$ and a number of predictors. Obviously, the simplest approach is

$$P(Y_i = 1) = \beta_0 + \beta_1 x_{i1} + ... + \beta x_{ip}\,.$$

However, this can lead to probabilities beyond the interval of $[0,1]$. To avoid this, we could transform the response variable to a scale that ranges from minus to plus infinity. The usual choice is the so-called logit transformation $p \mapsto \ln(p/(1-p))$. What we obtain is the logistic regression model:

$$\log\left(\frac{P(Y_i = 1)}{1 - P(Y_i = 1)}\right) = \beta_0 + \beta_1 x_{i1} + ... + \beta x_{ip}$$

Poisson Regression

What are predictors for the locations of starfish? For answering this question, we could analyze several areas for which some properties are known. The response Y_i is a count – the simplest model in this case is that it has a Poisson distribution. We then assume that the logged parameter λ_i at location i depends in a linear way on the covariates:

$$Y_i \sim Pois(\lambda_i) \text{ where } \log(\lambda_i) = \beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip}.$$

Log-linear Models

Another extension of the linear model is necessary for the case where we try to predict a nominal response variable. For example, we may be interested in giving probabilities for the favorite party of a person, depending on predictors such as education, age, etc. Such data can be summarized and displayed with contingency tables.

Generalized Linear Models

Log-linear models, logistic and Poisson regression all fit within the framework of generalized linear models (GLMs). They are based on the notion that the suitable transformed expected value of the response Y has a monotone relation to a linear combination of the predictors, i.e.:

$$g(E[Y_i]) = \beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip}.$$

Additionally, we require that the variance of the response Y is of the form $\phi \cdot v(E[Y])$, where ϕ is an additional parameter, and $v(\cdot)$ a specific function. Moreover, there are some restrictions on the density of Y, given the predictors x.

While this may seem awfully complicated, it can be shown that these conditions are fulfilled in many practically relevant situations. In particular, not only the extension to the linear model discussed above fall within this class of models, but also the multiple linear regression model with Gaussian errors.

Thus, this is what it is: a more general formulation of linear modeling. There are a number of theoretical results characterizing the properties of such models, which led to some general basic principles for estimation, inference, model diagnostics and variable selection.

However, we will here do without much discussion of this general framework and the corresponding results. We limit ourselves to the discussion of the specific cases sketched above. For readers who are interested in pursuing the theory on GLMs, we refer to the seminal work “Generalized Linear Models” by McCullagh and Nelder (Chapman and Hall, 1989).

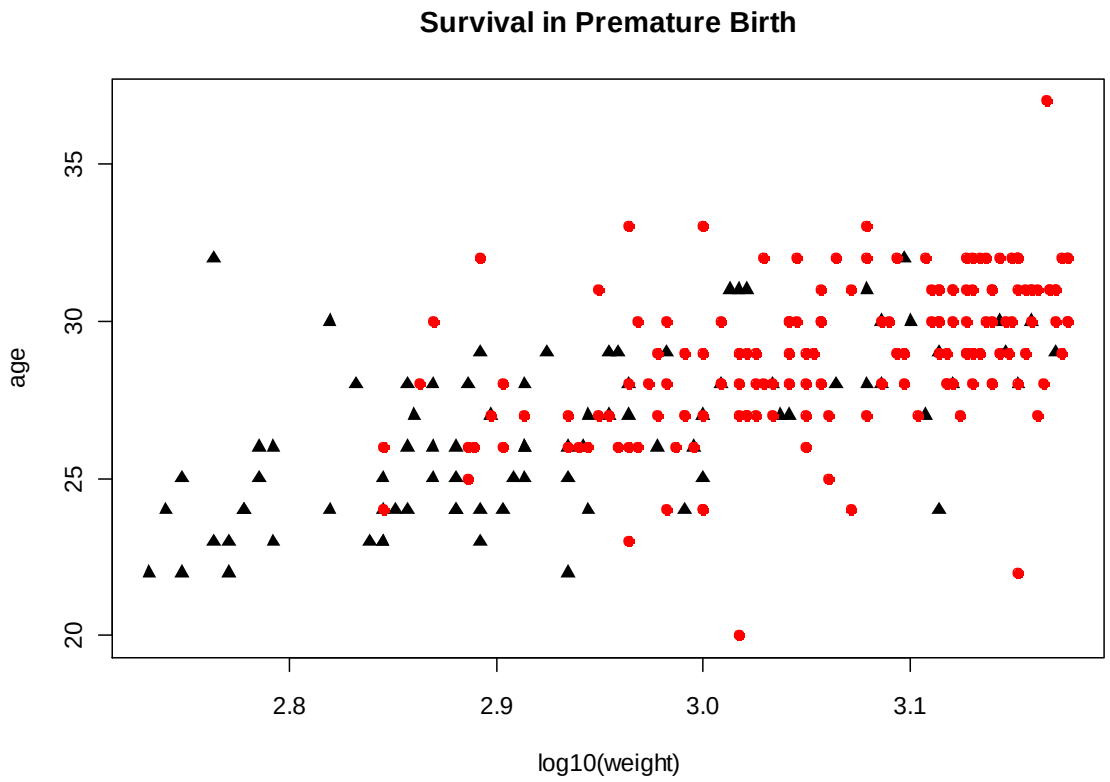
13 Binary Logistic Regression

We have explained above why binary response variables do not fit within the framework of multiple linear regression. Here, we will discuss the necessary extension. We will see that some of the previously pursued ideas will reappear, while some novel, other issues arise here. Our discussion includes:

- Formulation of the model
- Estimation
- Inference
- Model diagnostics
- Model choice

13.1 Example

In this section, we will discuss an example dealing with survival after premature birth. A study of Hubbard (1986) contains data of 247 early born babies. Predictors for their survival are birth weight (in grams), birth age (in weeks of pregnancy), the apgar scores (judging the vital functions one and five minutes after birth) and the pH-value of the babies' blood (providing further information on how well the baby breathes).



In the plot above, we display age versus the log10-transformed birth weight. Surviving babies are coded with red dots, while the ones that died are plotted by black triangles. It is apparent that the proportion of surviving babies depends on age and weight: the older and heavier a baby is born prematurely, the better the odds for surviving are. The goal with our logistic regression analysis will be the quantitatively model the odds for survival regarding the influence of the predictors.

13.2 Logistic Regression Model

In the premature birth example, the response variable Y_i is binary, taking values 0 (death) and 1 (survival). Thus, Y_i has a Bernoulli distribution, i.e. every Y_i has a different Bernoulli distribution and we denote the respective parameters, the success rates, by p_i. It is very important to note that

$$p_i = P(Y_i = 1 | x_1, ..., x_p) = E[Y_i | x_1, ..., x_p],$$

i.e. the parameter can also be seen as the conditional probability for survival given the predictors, or equally, as the expected value of the response variable Y_i, again given the predictors. As we have already concluded above, the simplest approach for linear modeling would be:

$$p_i = \beta_0 + \beta_1 x_{i1} + ... + \beta x_{ip}.$$

This is inappropriate here, because while the fitted values are probabilities, they can exceed the interval $[0,1]$. We can try to solve the problem by transforming the response to a real-valued scale. The most commonly used transformation which maps from $[0,1]$ to $(-\infty, +\infty)$ is the ***logit function***:

$$g(p_i) = \log\left(\frac{p_i}{1 - p_i}\right)$$

Its interpretation is as follows: probabilities are mapped to logged odds. Because these logged-odds are real-valued, they can be modeled using the desired linear combination of predictors. This is the logistic regression model:

$$\log\left(\frac{p_i}{1 - p_i}\right) = \beta_0 + \beta_1 x_{i1} + ... + \beta x_{ip}$$

Note that there is no error term ε_i. Why don't we need it here? Well, the left hand side of the equation is (a transformation of) a probability, which already accounts for the uncertainty in the response. Hence, the uncertainty in a babies' survival for a given combination of birth age and weight is already dealt with.

For estimating this model, we require the Y_i to be independent. For the predictors, however, there are no restrictions: they can be categorical (factors), higher-order polynomial terms or other transformations of the original predictors. Also interactions among predictors are allowed.

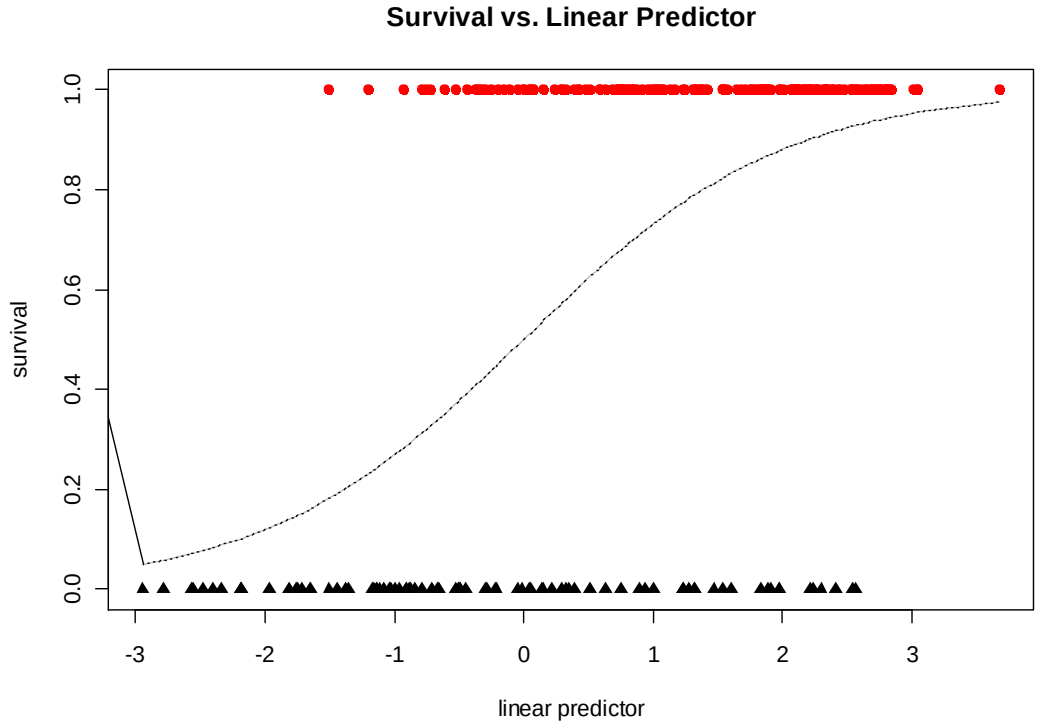
We introduce some additional notation. The right-hand side of the above equation is also called ***linear predictor***, denoted by η_i:

$$\eta_i = \beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip} .$$

The function $g(\cdot)$ that maps p_i, or better, the expected values $E[Y_i]$ to the values of the linear predictor is called the ***link function***. The logit function is the default choice with logistic regression, but there are other options. They only need to map

$$[0,1] \mapsto (-\infty, +\infty)$$

Another choice of such a function with some practical relevance is the inverse of the Gaussian cumulative distribution function. This is known as the probit link function.



Example

We fit the logistic regression model to the premature birth example. The result is:

```
> glm(survival ~ l10weight+age, data=baby, family="binomial")

Coefficients:
     (Intercept)  I(log10(weight))              age
        -33.9711           10.1685            0.1474
```

What is the interpretation of this output? By applying the inverse logit function, we obtain a probability of survival for every point in the above scatter plot:

$$P\left[Y = 1 \,|\, \log_{10}(weight), age\right] = g^{-1}\left(-33.97 + 10.17 \cdot \log_{10}(weight) + 0.14 \cdot age\right)$$

As we had expected before, this probability is the small in the lower left, and high in the top right corner. Note that even when we extrapolate beyond the range of x-values that are present for fitting, we would still obtain fitted values in $[0,1]$.

While displaying the result from the fit in the 2-dimensional scatter plot would require shading the area in the plot according to the fitted probability, things are much simpler if we plot the response (survival) versus the linear predictor η, see above. If we need to come up with a prediction for survival given a particular predictor configuration, we would forecast survival when $\hat{p}_i \geq 0.5$.

13.3 Estimation and Interpretation of Coefficients

While in multiple linear regression, estimation of the coefficients was based on minimizing the sum of squared residuals, this concept is not that straightforward to extend with logistic regression. We use a different approach here: maximum likelihood estimation (MLE), i.e. the regression coefficients β_j are determined such that the log-likelihood function is maximized.

$$l(\beta) = \sum_{i=1}^{n} \left(y_i \log(\hat{p}_i) + (1 - y_i) \log(1 - \hat{p}_i) \right)$$

The log-likelihood is a function of the regression parameter vector β, although they are not apparent on the right hand side of the definition. However, they are there, hidden within the fitted values $\hat{p}_i$. MLE requires maximizing $l(\beta)$ which is done by taking partial derivatives.

This leads to a non-linear equation system that is usually solved by an iterative approach. It is based on formulating a linear approximation in every step, which is then solved using weighted linear regression. When the changes in the fitted values are small enough, the process stops and the solution is found.

MLE may seem a totally different concept to the one we employed in multiple linear regression. However, this is not the case: for normally distributed errors, the least squares estimator coincides with the maximum likelihood estimator – which again neatly demonstrates, how both logistic and multiple linear regression fit within the framework of GLMs, where parameter estimation is always done using MLE.

Example

We now turn our attention to the interpretation of the coefficients β_j. As we had stated above, the log-odds for $Y_i = 1$ are a linear function of the predictors. Thus, if predictor x_j is increased by 1 unit, then the log-odds in favor of $Y = 1$ increase by β_j if all other predictors remain unchanged. We illustrate this with the premature birth example, where we consider an individual with $\log_{10}(weight) = 3$ and birth ages of 30 weeks. We have:

$$\eta = -33.9711 + 10.1685 \cdot 3.0 + 0.1474 \cdot 30 = 0.957 ,$$

which are the log-odds for survival. If we take $\exp(0.957) = 2.604$, we obtain the odds for survival. It is thus 2.604 times more likely to survive than die when born at this particular combination of age and weight. On the other hand, the probability for survival is:

$$g^{-1}(\eta) = \frac{\exp(\eta)}{1+\exp(\eta)} = \frac{\exp(0.957)}{1+\exp(0.957)} = 0.723$$

Now, if we compare to an individual with birth age 31 weeks (and the rest remaining as above), we obtain the odds as $\exp(1.104) = 3.017$. If we divide the two odds, we obtain the odds-ratio:

$$\frac{3.017}{2.604} = 1.159 = \exp(\hat{\beta}_2)$$

The odds for surviving increase by $\exp(\hat{\beta}_2)$ (i.e. about 15%) when a baby of the same weight is born one week later – this is a more illustrative way to see the parameter $\hat{\beta}_2$.

13.4 Inference

If we look at the summary output of a logistic regression model, things are similar, but not equal, as before:

```
> summary(fit)

Call:
glm(formula = survival ~ I(log10(weight)) + age,
    family  = "binomial", data = baby)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2983  -0.7451   0.4303   0.7557   1.8459

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)      -33.97108    4.98983  -6.808 9.89e-12 ***
I(log10(weight))  10.16846    1.88160   5.404 6.51e-08 ***
age                0.14742    0.07427   1.985   0.0472 *
---
(Dispersion parameter for binomial family taken to be 1)
    Null deviance: 319.28  on 246  degrees of freedom
Residual deviance: 235.94  on 244  degrees of freedom
AIC: 241.94

Number of Fisher Scoring iterations: 4
```

Besides the different call, an important difference is that we now have deviance residuals instead of the plain ones as before. There is no more “residual standard

error", but only some deviance measures. Also the global F-test is missing. We will explain these differences in the following section 13.5.

What might seem like a minor detail, i.e. the fact that we now have a z- instead of a t-value, also has some implications. In multiple linear regression, given Gaussian errors, it is quite easy to show that the estimated regression coefficients $\hat{\beta}_j$ are normally distributed. This is no longer true for logistic regression. However, one can show that $\hat{\beta}_j$ is at least approximately Gaussian with covariance matrix V. Thus, we are using

$$Z = \frac{\hat{\beta}_j - \beta_j}{\sqrt{\hat{V}_{jj}}}$$

as a test statistic. Because the covariance matrix can be derived directly from the coefficients and no error variance $\hat{\sigma}_\varepsilon^2$ is involved, we can replace the former Student distribution by the Gaussian. Hence, we have the z-value here.

13.5 Goodness-of-Fit

In multiple linear regression, we can use the sum of squared residuals as a goodness-of-fit measure. This is replaced by the so-called residual deviance in logistic regression. This is twice the difference between the log-likelihood for the saturated model, and the maximum of the log-likelihood for our current model. Since for the non-grouped binary data we have here, the log-likelihood for the saturated model is zero, we obtain:

$$D(y, \hat{p}) = -2\sum_{i=1}^{n}\left(y_i \log(\hat{p}_i) + (1-y_i)\log(1-\hat{p}_i)\right),$$

which is $-2l(\hat{\beta})$, minus twice the log-likelihood of our model. The deviance is especially useful for comparing nested models. If we have two models, where the smaller one $(S, \textit{with } q \textit{ parameters})$ is comprised within the bigger one $(B, \textit{with } p \textit{ parameters})$, we can use the likelihood ratio test to check whether the bigger one yields an improvement. Null hypothesis and test statistic are:

$$H_0 : \beta_{q+1} = \beta_{q+2} = ... = \beta_p = 0$$

$$2\left(l^{(B)} - l^{(S)}\right) = D\left(y, \hat{p}^{(S)}\right) - D\left(y, \hat{p}^{(B)}\right),$$

where $l^{(B)}$ and $l^{(S)}$ are the values of the log-likelihood function from the big and small models. The likelihood ratio test amounts to computing the difference in deviance which has an approximate chi-square distribution. The number of degrees of freedom corresponds to the difference in the number of parameters between the two models, i.e.

$$D^{(S)} - D^{(B)} \sim \chi^2_{p-q}$$

The above test is already implemented in the R-function `drop1()` for excluding predictors from a given model. We will try this and check the results on the premature birth dataset:

```
> drop1(fit, test="Chisq")
Single term deletions

Model:
survival ~ I(log10(weight)) + age
                 Df Deviance    AIC    LRT   Pr(Chi)
<none>                235.94 241.94
I(log10(weight))  1   270.19 274.19 34.247 4.855e-09 ***
age               1   239.89 243.89  3.948   0.04694 *
```

The decision is to keep both predictors in the model. Note that for continuous or binary predictors, the above output tests the same null hypothesis as the summary output. However, the test statistics are not equal: in some cases, the standard errors for the $\hat{\beta}_j$ can be overestimated and so the z-value is too small, and the significance of an effect could be missed. This is known as the Hauck-Donner effect, and it is the reason why the deviance based test outlined here is preferred.

The above output also provides some further information in column AIC. The definition of the criterion is:

$$AIC = D(y, \hat{p}) + 2 \cdot (\#\ of\ \ parameters),$$

i.e. the deviance penalized by twice the number of predictors used. This can conveniently be used as a means for comparing models of different size. Also for logistic regression, its application is popular especially when doing variable selection with stepwise procedures.

Null Deviance

The smallest model that we consider is the one where there are no predictors, but only an intercept. The fitted values will all be equal to $\hat{p}^{(0)}$. Our best fit (F) and the smallest model (0) are nested, thus we can perform a deviance test:

$$2\left(l^{(0)} - l^{(F)}\right) = D\left(y, \hat{p}^{(F)}\right) - D\left(y, \hat{p}^{(0)}\right).$$

The summary output in 13.4 tells us that:

```
    Null deviance: 319.28  on 246  degrees of freedom
Residual deviance: 235.94  on 244  degrees of freedom
```

The deviance difference is 83.34. Using the χ^2-distribution with 2 degrees of freedom (i.e. the difference among the two models), we obtain a very small p-value that is close to 0. The two predictors thus have are highly significant for predicting survival in premature birth.

13.6 Model Diagnostics

Model checking is just as important in logistic regression as it is in linear modeling. Although there is no error term in logistic regression, we will base the diagnostics again between the observed and fitted values. Unlike the case of linear models, we now have to make allowance for the fact that these differences have different variances. There are two types of residuals in common use:

Pearson Residuals

A very simple approach to the calculation of residuals is to take the difference between observed and fitted value and divide by an estimate of the standard deviation. The resulting residual has the form

$$R_i = \frac{y_i - \hat{p}_i}{\sqrt{\hat{p}_i(1-\hat{p}_i)}}.$$

This is called the Pearson residual, because R_i^2 is the contribution of the i^{th} observation to the (here not discussed) Pearson chi-square statistic for model comparison. It is important to note that usually Pearson residuals exceeding a value of two in absolute value warrant a closer look. In R, Pearson residuals can easily obtained using the command `resid(fit, type="pearson")`, when object `fit` contains the results of a logistic regression.

Deviance Residuals

An alternative residual is derived from the contribution of instance i to the deviance $D(y, \hat{p})$. In particular, this contribution is:

$$d_i = -2 \cdot \left(y_i \log\left(\hat{p}_i\right) + (1-y_i)\log\left(1-\hat{p}_i\right)\right).$$

For obtaining a residual that can be well interpreted, we take the square root and enhance it by the sign of the difference between true and fitted value:

$$D_i = sign(y_i - \hat{p}_i) \cdot \sqrt{d_i}$$

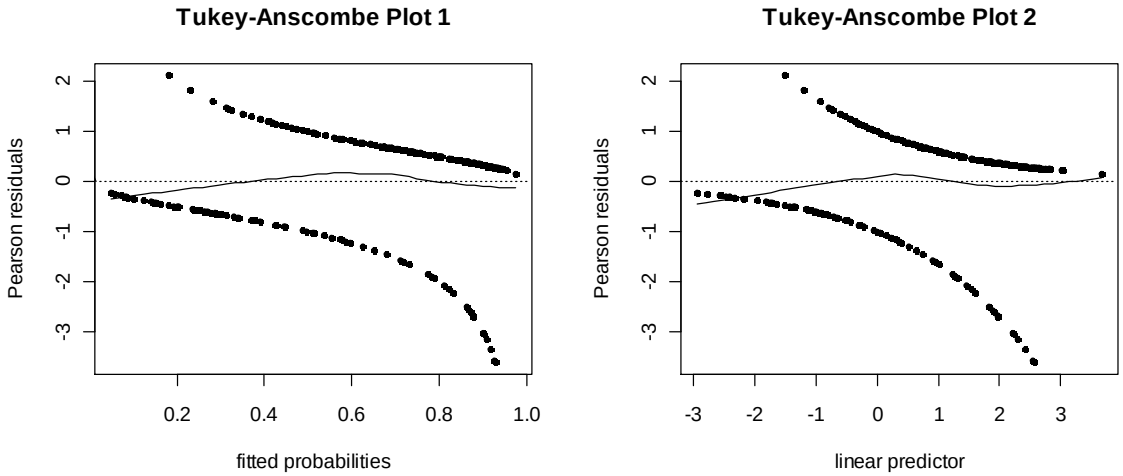
Also here, observations with a deviance residual in excess of two may indicate lack of fit. Again, in R, computation is simple: `resid(fit, type="deviance")`, when as before, object `fit` contains the results of a logistic regression.

Studentized Residuals

The residuals we defined so far take into account the fact that different observations have different variances, but they do not account for additional variation arising from estimation of the parameters in the way studentized residual in multiple linear regression models do. When doing diagnostics with R, studentized residuals will be computed and used. The transformation depends on the hat matrix and often, the differences are only small. Thus, we omit the details on studentizing the residuals here.

Tukey-Anscombe Plot

Also for logistic regression, the Tukey-Anscombe plot remains an important means for model diagnostics. For both the y- and the x-axis there are several alternatives: we can plot the Pearson or Deviance residuals versus either the values of the linear predictor, or versus the fitted probabilities, see below:



The interpretation of these plots is more difficult than in multiple linear regression. Because every observation may only take two values, either 0 or 1, the residuals lie on two monotonous curves.

Thus, we can only see an inadequacy of the model if we display a smoother: doing so basically amounts to comparing against a non-parametric model for p_i or η_i, respectively. It is important to use a non-robust smoother: else, high or low values for p_i with only few observations will be interpreted as outliers and are down-weighted. This is unwanted.

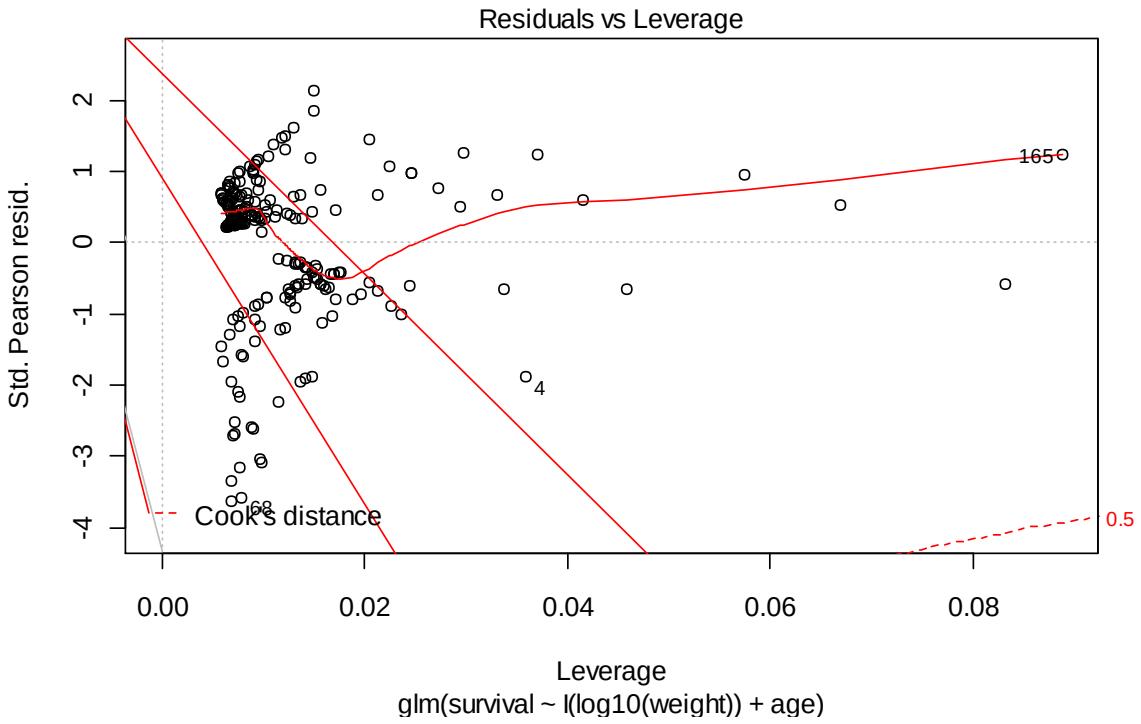
When applying `plot(fit)` on the output of a binary logistic regression, R uses a robust smoother. The fragment of code below helps in producing an adequate Tukey-Anscombe plot this type of models:

```
xx <- predict(fit, type="response")
yy <- residuals(fit, type="pearson")
scatter.smooth(xx, yy, family="gaussian", pch=20)
abline(h=0, lty=3)
```

The Normal plot and the Scale-Location plot are of no use in logistic regression with the non-grouped data we discussed so far. Checking for influential observations may be useful, though. This plot can easily be obtained with

```
plot(fit, which=5)
```

The results can be seen in the plot below. We see that there is some gap near the zero line. In this example here it is caused by the fact that the central observations which have small leverages have predicted probabilities around 0.5, and thus cannot have large residuals.


Residuals vs Leverage
Std. Pearson resid.
Leverage
glm(survival ~ I(log10(weight)) + age)
Cook's distance
0.5
165
4
68
0.00
0.02
0.04
0.06
0.08
2
1
0
-1
-2
-3
-4

14 Binomial Regression Models

In chapter 13 we discussed binary logistic regression for response variables with values being either 0 or 1. We will extend this to situations where we need to model proportions that are in the interval $[0,1]$. This can arise naturally with $0/1$-responses, i.e. always when we have a batch of individuals sharing their (predictor) properties. Then, we consider the proportion of $Y=1$ for each batch. The following examples illustrate the concept:

Example 1

Rotenone is an insecticide. It is not surprising that the effect depends on the dose which is applied. We made the following observations:

Concentration in $\log$ **of** mg/l	**Number of insects** n_i	**Number of killed insects** y_i
0.96	50	6
1.33	48	16
1.63	46	24
2.04	49	42
2.32	50	44

These data are grouped, and we are mainly interested in the proportion of insects that survive at a particular concentration. We will base this on the notion that for the number of killed insects, we have $Y_i \sim Bin(n_i, p_i)$.

Example 2

Sometimes, there is more than only one predictor. The following data come from a study on infant respiratory disease, namely the proportions of children developing bronchitis or pneumonia in their first year of life by type of feeding and sex:

Sex	**Feeding**	**Number of children** n_i	**Number of children with disease** y_i
Boy	Bottle	458	77
	Mixed	147	19
	Breast	494	47
Girl	Bottle	384	48
	Mixed	127	16
	Breast	464	31

Again, these data are grouped. For a given predictor setting (an instance i), there are multiple observations n_i. For the number of diseased children, we have $Y_i \sim Bin(n_i, p_i)$ as above. Again, the focus is on the proportions p_i.

14.1 Model and Estimation

In both of the above examples, we are after a relation between p_i and the linear predictor η_i, in general notation:

$$p_i = P(Y_i = 1 | x_1, ..., x_p) \text{ is related to } \eta_i = \beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip}$$

This very much resembles the situation in chapter 13, except that we now have grouped data, i.e. an instance i contains information on more than one single individual. We have already seen in chapter 13, $p_i = \eta_i$ is not appropriate because we require $p_i \in [0,1]$. We will again use the logit link function such that $\eta_i = g(p_i)$:

$$\log\left(\frac{p_i}{1-p_i}\right) = \beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip}$$

Note that here p_i is also the expected value $E[Y_i / n_i]$, and thus, also this model here fits within the GLM framework. Because $n_i p_i$ has a Binomial distribution, the log-likelihood function will be:

$$l(\beta) = \sum_{i=1}^{k} \left[\log\binom{n_i}{y_i} + n_i y_i \log(p_i) + n_i (1-y_i) \log(1-p_i) \right].$$

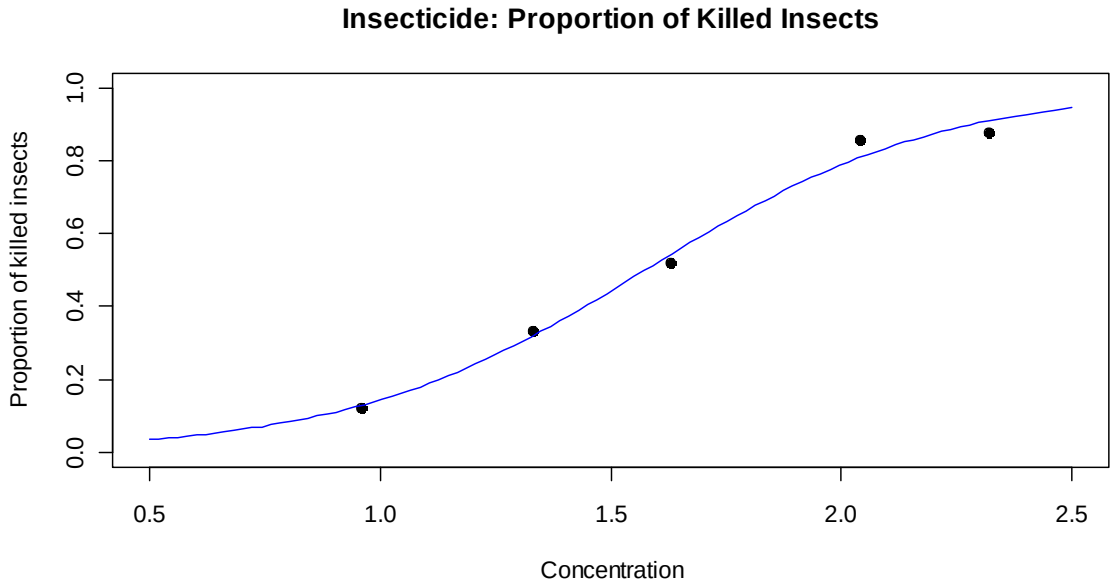
While the β cannot be found explicitly on the right hand side, they are there implicitly via the fitted values p_i. Parameter estimation is again done using MLE. Because we now have grouped data, the syntax for model fitting with R is slightly different:

```
> killsurv
     killed surviv
[1,]      6     44
[2,]     16     32
[3,]     24     22
[4,]     42      7
[5,]     44      6
> fit <- glm(killsurv~conc, family="binomial")
```

We need to generate a two-column matrix where the first contains the “successes” and the second contains the “failures”. The fit is then:

```
> summary(glm(killsurv ~ conc, family = "binomial")

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -4.8923     0.6426  -7.613 2.67e-14 ***
conc          3.1088     0.3879   8.015 1.11e-15 ***
---
    Null deviance: 96.6881  on 4  degrees of freedom
Residual deviance:  1.4542  on 3  degrees of freedom
AIC: 24.675
```

Because there is only a single predictor, the fit is easy to visualize, too:



The interpretation of the coefficients and the individual hypothesis tests are as before in chapter 13. Some new issues arise with the goodness-of-fit statistic.

14.2 Goodness-of-Fit Test

We are going to use the residual deviance as a goodness-of-fit measure. This is twice the difference between the log-likelihood for our current model and the saturated model. This is the same paradigm as before, but now, the saturated model no longer has zero deviance. It has as many parameters as there are batches, and thus fits the respective proportions perfectly, i.e. $\hat{p}_i = y_i / n_i$ for all batches $i = 1,...,k$. In such a case, the residual deviance becomes:

$$D(y,\hat{p}) = 2\sum_{i=1}^{k}\left[y_i \log\left(\frac{y_i}{\hat{y}_i}\right) + (n_i - y_i)\log\left(\frac{(n_i - y_i)}{(n_i - \hat{y}_i)}\right)\right]$$

Now because the saturated model fits as well as any model can fit, the deviance measures how close our model comes to perfection.

Provided that the Y_i are truly binomial and that the n_i are relatively large, the deviance is approximately χ^2 distributed where the degrees of freedom are $k - (\#\ of\ \ predictors) - 1$ if the model is correct. Thus, we can use the deviance to test whether the model provides an adequate fit. In R, we type

```
> pchisq(deviance(fit), df.residual(fit), lower=FALSE)
[1] 0.69287
```

Because this p-value is well in excess of 0.05, we may conclude that this model fits sufficiently well – of course we can never conjecture from such a result that the

model is correct. We can perform the same test for the null deviance, whose value and the degrees of freedom can be taken from the summary output:

```
> pchisq(96.6881, 4, lower=FALSE)
[1] 4.985178e-20
```

The p-value is very small, which provides evidence that we cannot ascribe the number of killed insects to simple variation not depending on the concentration of insecticide.

Please note that a random variable with a χ^2_d has mean d and standard deviation $\sqrt{d}$ so that it is often possible to quickly judge whether a deviance is large or small without explicitly computing the p-value. Or even more focused on the application: if the deviance is far in excess of the degrees of freedom, the model is not worth much.

However, note that the χ^2 is only an approximation which requires sufficiently large batches to work well – a rule of the thumb is that at least all $n_i \geq 5$. This clearly shows that the quick check for model accuracy is not suitable for non-grouped binary response as in chapter 13.

On the other hand, the comparison of two nested models is exactly the same for grouped and non-grouped data. Moreover, there are not many aspects with model diagnostics. But we will turn our attention to a topic which arises from the goodness-of-fit test.

14.3 Overdispersion

We have seen above that if the binomial regression model is correct, we expect that the residual deviance will be approximately equal to the degrees of freedom. However sometimes, we observe deviances that are much larger. Then, we need to determine what aspect of the model is faulty.

The predominant cause is that the model has a wrong structural form, i.e. it does not include the right predictors, or we failed to transform and combine them in the correct way. This should become apparent from the diagnostic plots. No matter how difficult this may be in practice, suppose now that we are able to exclude has the wrong structural form.

Another explanation for too large deviance with respect to the degrees of freedom is the presence of a small number of outliers. This is also easy to check with the diagnostic plots, fortunately. Having excluded also this possibility, there is a further one that remains: deficiencies in the random part of the model:

A binomial distribution arises when the probability of success is independent and identical for each trial within a batch. In this case, the variance of Y_i will be equal to $n_i p_i (1 - p_i)$. However, if the assumptions are violated, the variance is usually greater – this is called ***overdispersion***. The contrary case of lower variance can also happen and is called underdispersion. It is much rarer, though.

There are two causes for overdispersion. It can be due to non-constant p_i, i.e. violation of the “identical” assumption. It often happens if unrecorded variables have an impact, of if the population we sample from is not homogeneous, i.e. clusters exist. Overdispersion can also result from dependence between trials. If the response has a common cause, say a disease influenced by genetic factors, the responses will tend to be positively correlated. Or, subjects may be influenced by the other objects under study: e.g. if the food supply is limited, the survival odds of one individual may increase if others die.

Overdispersion can be dealt with: we can introduce an additional dispersion parameter ϕ that is estimated from the data – we divide the deviance by the degrees of freedom. The dispersion parameter $\hat{\phi}$ has an impact on the inference: it appears in the coefficient covariance matrix and (in case of overdispersion) leads to smaller confidence intervals and less significant test results for the individual hypothesis tests.

15 Poisson Regression for Count Data

This chapter deals with responses that are counts, i.e. positive integers. If the counts are upper bounded, then we are usually in the case of a binomial regression as in chapter 14. On the other hand, there are situations where the counts are unbounded, and all of them are sufficiently large with small relative variation, e.g. all instances lie in $[5000, 6000]$. Then the normal approximation usually works well, such that a multiple regression model may be used. However, there are also situations where the use of a Poisson regression is a must:

- when the size of the population is unknown and the counts are small
- when the size of the population is large and hard to come by, and the probability of "success", and thus the counts are small.

A typical example for the latter case is modeling the incidence of rare forms of cancer in a given geographical area. For illustrating the former case, we will consider the tortoise example.

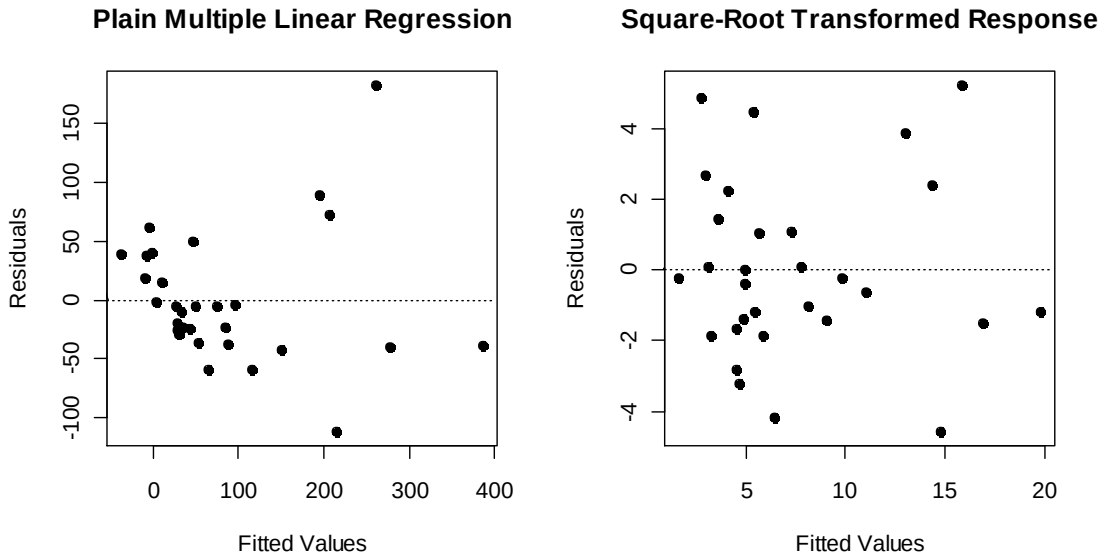
Example

For 30 of Galapagos Islands' we have the number of species of tortoise that live there, plus five geographic variables. These are the area of the island, the highest elevation, the distance to the nearest island, the distance to Santa Cruz island and the area of the adjacent island.

```
> head(gala)
             Species  Area Elevation Nearest Scruz Adjacent
Baltra            58 25.09       346     0.6   0.6     1.84
Bartolome         31  1.24       109     0.6  26.3   572.33
Caldwell           3  0.21       114     2.8  58.7     0.78
Champion          25  0.10        46     1.9  47.4     0.18
Coamano            2  0.05        77     1.9   1.9   903.82
Daphne.Major      18  0.34       119     8.0   8.0     1.84
```

We first fit a multiple linear regression model and analyze the Tukey-Anscombe plot in the left panel below. It does not surprise us that the variance is non-constant. Moreover, we have previously learnt that a square-root transformation is appropriate. Indeed, this clears up the variance issue. Because also the coefficient of determination is quite high ($R^2 = 0.78$), one might think that the fit is good and end the analysis here.

While the model with the square-root transformed response is adequate, there could be room for improvement. Especially the validity of the normal approximation is in question, because some of the counts are small. Thus, we use a Poisson regression model.



15.1 Model, Estimation and Inference

We have count responses Y_i for which we, given the predictors, assume a Poisson distribution with parameter λ_i, i.e. $Y_i \sim Pois(\lambda_i)$. Our goal is to relate the parameter to the predictors, and because λ_i can take positive values only, we will employ the log as a link function:

$$\log(\lambda_i) = \beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip}$$

Because $E[Y_i] = \lambda_i$, we again have the previously recognized situation that a link functions opens the door to modeling the expected value of the distribution of Y by the linear predictor. Here, the log-likelihood is:

$$l(\beta) = \sum_{i=1}^{n} \left(y_i \cdot \log(\lambda_i) - \lambda_i - \log(y_i!) \right)$$

Again, there is no closed form solution and we have to resort to the iteratively reweighted least squares approach for an approximation. We fit the Poisson regression model to the Galapagos data:

```
> summary(glm(Species ~ ., family = "poisson", data = gala)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.155e+00  5.175e-02  60.963  < 2e-16 ***
Area        -5.799e-04  2.627e-05 -22.074  < 2e-16 ***
Elevation    3.541e-03  8.741e-05  40.507  < 2e-16 ***
Nearest      8.826e-03  1.821e-03   4.846 1.26e-06 ***
Scruz       -5.709e-03  6.256e-04  -9.126  < 2e-16 ***
Adjacent    -6.630e-04  2.933e-05 -22.608  < 2e-16 ***
---
    Null deviance: 3510.73  on 29  degrees of freedom
Residual deviance:  716.85  on 24  degrees of freedom
```

As a goodness-of-fit measure we will again use the residual deviance. Here, it is:

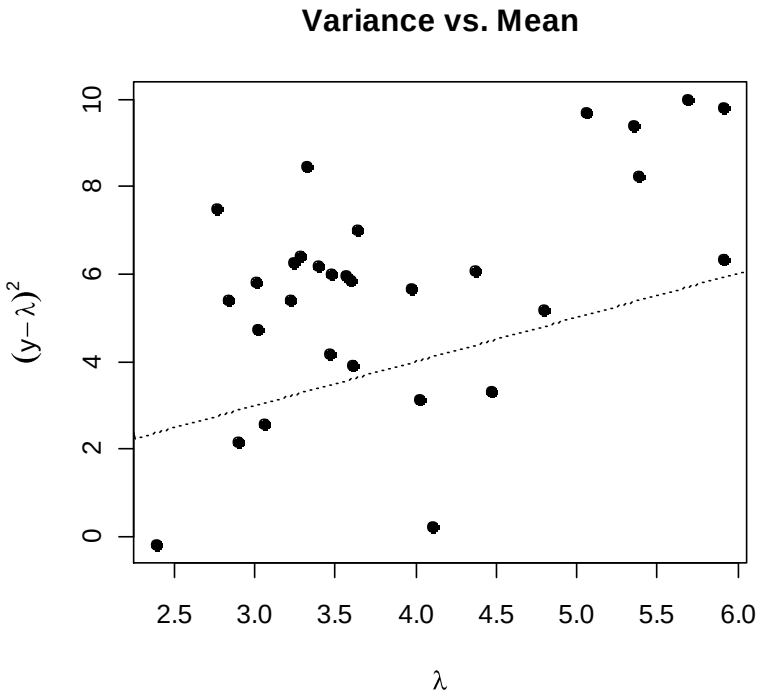
$$D = 2\sum_{i=1}^{n}\left[y_i \log\left(\frac{y_i}{\hat{\lambda}_i}\right) - (y_i - \hat{\lambda}_i)\right]$$

For judging the goodness-of-fit, we can again use the χ^2 asymptotics: the residual deviance should in the same range as the degrees of freedom, else we have under- or overdispersion, i.e. an inadequate model. As before, if we wish to compare two nested models, it can be done by the deviance difference.

Example

In our example, the residual deviance is 717 on just 24 degrees of freedom, indicating that we have an ill-fitting model if a Poisson distribution for the response is correct. Because it is neither outliers nor a deficiency in functional form which cause this, the reason must be elsewhere.

For a random variable with a Poisson distribution, the mean is equal to the variance. It is difficult to check this for a model under investigation, but plotting $(y - \hat{\lambda})$ versus $\hat{\lambda}$ serves as a crude approximation.



We observe that while the variance is roughly proportional to the mean, it is clearly bigger. Thus we are in a case where the variance assumption is violated, but the link function, the distribution and the choice of predictors are correct. Thus, the estimates for β_j will be consistent, but the standard errors are wrong. We cannot take any conclusions on which predictors are significant from the summary output above.

Again, we are in a case where we have overdispersion. Also in the context of Poisson regression, this can be cured by introducing an additional dispersion parameter. It can be estimated by:

$$\hat{\phi} = \frac{\sum (y_i - \hat{\lambda}_i) / \hat{\lambda}_i}{n - p}$$

This is the sum of squared Pearson residuals, divided by the degrees of freedom in this model. We can adjust the summary output by:

```
> disp <- sum(resid(fit02,type="pearson")^2)/fit02$df.res
> disp
[1] 31.74914
>
> summary(fit02, dispersion=disp)

Call: glm(Species ~ ., family = "poisson", data = gala)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.1548079  0.2915897  10.819  < 2e-16 ***
Area        -0.0005799  0.0001480  -3.918 8.95e-05 ***
Elevation    0.0035406  0.0004925   7.189 6.53e-13 ***
Nearest      0.0088256  0.0102621   0.860    0.390
Scruz       -0.0057094  0.0035251  -1.620    0.105
Adjacent    -0.0006630  0.0001653  -4.012 6.01e-05 ***
---
(Dispersion parameter taken to be 31.74914)

    Null deviance: 3510.73  on 29  degrees of freedom
Residual deviance:  716.85  on 24  degrees of freedom
AIC: 889.68
```

Note that the estimation of dispersion and regression parameters are independent, thus modifying the dispersion parameter does not change the coefficients. In our example, some of the predictors now turn out to be non-significant, indeed. There is some similarity in the variables which are picked out when compared to the multiple linear regression model.

16 Multinomial Data

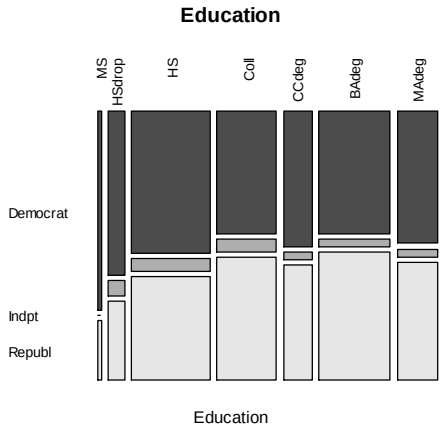
We are now considering data where the response is a categorical variable with more than two levels. This can be seen as an extension to binary logistic and binomial regression that were discussed in chapters 13 and 14. We will distinguish between ***nominal multinomial data*** where there is no natural order to the response categories, and ***ordinal multinomial data*** where this is the case.

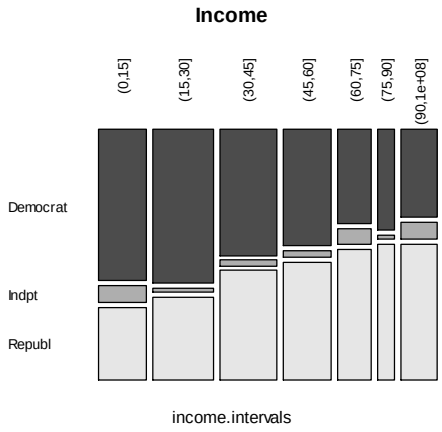
16.1 Multinomial Logit Model

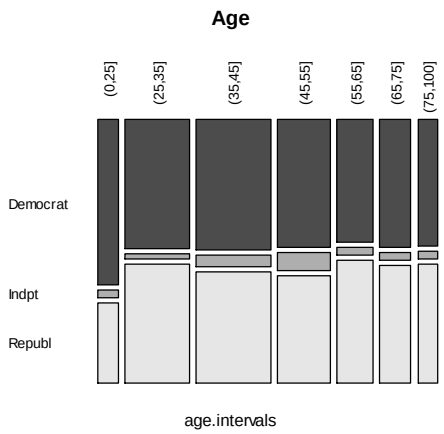
The multinomial logit model which will be discussed in this section is intended for nominal data. It can in principle also be used for ordinal data, but the information about the order will not be used. We will study an example:

Example

The data we analyze are a simplified subset of the 1996 American National Election Study. The target variable is the party identification: Democrat, Independent or Republican. As predictors, we here consider age, income as (pseudo-)continuous variables, and education level which is a factor with 7 levels.







The data are best visualized using mosaic plots. While for the categorical predictor education, this is straightforward, we need to categorize age and income first. We do so by generating 7 levels each.

```
mosaicplot(table(nes$educ,nes$party), color=TRUE,
           main="Education", xlab="Education", cex=.8, las=2)

income.int <- cut(nes$income, c(0,15,30,45,60,75,90,10^8))
mosaicplot(table(income.int, nes$party), color=TRUE,
           main="Income", cex=.8, las=2)

age.intervals <- cut(nes$age, c(0,25,35,45,55,65,75,100))
mosaicplot(table(age.intervals, nes$party), color=TRUE,
           main="Age", cex=.8, las=2)
```

The plots are shown above. We observe that the proportion of Democrats falls with educational status, reaching a plateau for the college educated. Also, as income increases, there is an increase in the proportion of Republicans. Finally, the relation between party and age is not clear. While the (few) very young seem to favor the Democrats, there is no clear trend for ages above 25.

Also note that this is cross-sectional rather than longitudinal data, so we cannot say anything about what will happen with an individual when it gets older or develops to a higher income. With such data, we will only be able to make conclusions about the relative probability of party affiliations for different individuals with varying age, income and education.

Model

From the marginal distributions above we conclude that there are relations between the predictors and the response. Our goal is now to include them in a multivariate regression type model to answer whether and which of them are statistically significant. This can be done with the ***multinomial logit model***. Let Y_i be a random variable coding the response categories with values $1, 2, ..., J$. Then,

$$p_{ij} = P(Y_i = j)$$

is the probability that the response of the i^{th} observation falls into the j^{th} category. As with binary data, where $J = 2$, we may encounter both non-grouped and grouped data.

Thus, let Y_{ij} be the number of observations falling into category j for group or individual i. Then, we define

$$n_i = \sum_j Y_{ij} ,$$

which is the number of individuals in group i. For non-grouped data, $n_i = 1$ for all observations i, and also, only one of the Y_{ij} will be equal to one, and the rest will be zero. The Y_{ij}, conditional on the total n_i, follow a multinomial distribution:

$$P(Y_{i1}=y_{i1},Y_{i2}=y_{i2},...,Y_{iJ}=y_{iJ})=\frac{n_i}{y_{i1}!\cdot...\cdot y_{iJ}!}p_{i1}^{y_{i1}}\cdot...\cdot p_{iJ}^{y_{iJ}}$$

As with binomial data, our goal will again be to find a relation between the probabilities p_{ij} and the predictors x_i, while ensuring that the probabilities are restricted to values between 0 and 1. We will apply a similar idea as before:

$$\log\left(\frac{P(Y_i=j)}{P(Y_i=1)}\right)=\log\left(\frac{p_{ij}}{p_{i1}}\right)=\eta_{ij}=\beta_{0j}+\beta_{1j}x_{i1}+...+\beta_{pj}x_{ip} \text{ for all } j=2,...,J$$

This is a logit model for probability quotients, where we compare each of the categories against the first one, which serves as the reference category. The use of such a baseline category is dictated by the constraint that $\sum_j p_{ij}=1$.

In principle, we are free in the choice of the baseline, but we here choose the first category because this is what R does by default. Also note that the above is an equation system with $J-1$ rows, where the coefficients are different in each row. The parameters will be estimated using a maximum likelihood approach. In R, this can be done using function `multinom()` from `library(nnet)`.

```
> library(nnet)
> fit <- multinom(party ~ age + income + educ, data=nes)
# weights:  30 (18 variable)
initial  value 1037.090
iter  10 value  783.325
iter  20 value  756.095
iter  30 value  755.807
final value     755.806
converged
> summary(fit)
Call:
multinom(formula = party ~ age + income + educ, data = nes)

Coefficients:
      (Intrcpt)      age   income   educ.L   educ.Q   educ.C
Indpt    -5.136    0.005    0.016    5.244   -6.341    4.693
Republ   -1.409    0.010    0.013    0.564   -0.720    0.017
         educ^4   educ^5   educ^6
Indpt    -2.552    1.291   -0.539
Republ    0.000   -0.103   -0.129

Std. Errors:
     (Intrcpt)      age   income   educ.L    educ.Q   educ.C
Indpt    0.643    0.011    0.005    0.461    0.396    0.473
Republ   0.275    0.004    0.002    0.432    0.393    0.328
          educ^4    educ^5    educ^6
Indpt  0.471    0.489    0.430
Republ 0.263    0.217    0.176

Residual Deviance: 1511.612
AIC: 1547.612
```

We observe that quite a number of parameters are estimated. Here in particular, it is 18 parameters, and the general formula is $p^* \cdot (J-1)$, where p^* is the number of columns in the design matrix, i.e. 1 for the intercept plus the number of predictors at their respective degrees of freedom. This shows us that for estimating multinomial logit models with many response categories we will quickly need a lot of observations. The rule of the thumb, saying that we need at least 5 observations per estimated parameters is still valid here.

Inference

For inferring whether the k^{th} predictor has a significant impact on the response, we cannot perform individual hypothesis tests anymore, although standard errors for the estimates are provided. The reason is that now all parameters $\beta_{k2},...,\beta_{kJ}$ need simultaneously be equal to zero. Thus, we have to resort to a comparison of nested models, which will as before be based on log-likelihood ratios, resp. deviance differences. As an example, we will compare against a model without education as a predictor variable:

```
> fit.age.inc <- multinom(party ~ age + income, data=nes)
> deviance(fit.age.inc)-deviance(fit)
[1] 13.70470
> pchisq(13.70470, fit$edf-fit.age.inc$edf, lower=FALSE)
[1] 0.3199618
```

We obtain a p-value of 0.32 and thus, there is no significant contribution of predictor `education`. This may come as a surprise regarding the mosaic plot shown above. However, the biggest differences in party affiliation are among the young people below 25 years of age, which represent only a very small fraction of the observations. Hence, we can do without `education` here.

Prediction

One of the predominant goals with multinomial logit models is to obtain predicted probabilities. We here show them for some arbitrary 10 instances out of the 944 that are present in total.

```
> round(predict(fit, type="probs"),3)[sample(1:944)[1:10],]
    Democrat Indpt Republ
743    0.339 0.058  0.603
239    0.524 0.018  0.457
659    0.515 0.036  0.449
174    0.513 0.024  0.462
903    0.282 0.042  0.676
863    0.345 0.037  0.618
96     0.624 0.035  0.340
162    0.625 0.035  0.340
923    0.393 0.048  0.559
795    0.410 0.033  0.557
```

From the model output above, we learned that the probability of being in favor of the Republican party increases with income and age. The same is true for the

Independent. However, because this party is only small, the intercept is smaller, and thus also the fitted probabilities. When we for a person need to predict which party he/she is going to vote for, we would just choose the one with the highest probability. This is easy to obtain from R:

```
> predict(fit, type="class")[sample(1:nrow(nes))[1:10]]
 [1] Republ   Democrat   Democrat   Democrat   Republ
Republ   Democrat   Democrat   Republ   Republ
```

Model Diagnostics

Up until today, there is no meaningful definition of what residuals are in the context of the multinomial logit model. There are some for each of the $J-1$ equations in the system, and they also depend on the choice of the baseline category. How these could be displayed in comprehensive form is unclear. Thus, we here remain without effective tools for model enhancement.

16.2 Ordinal Multinomial Response

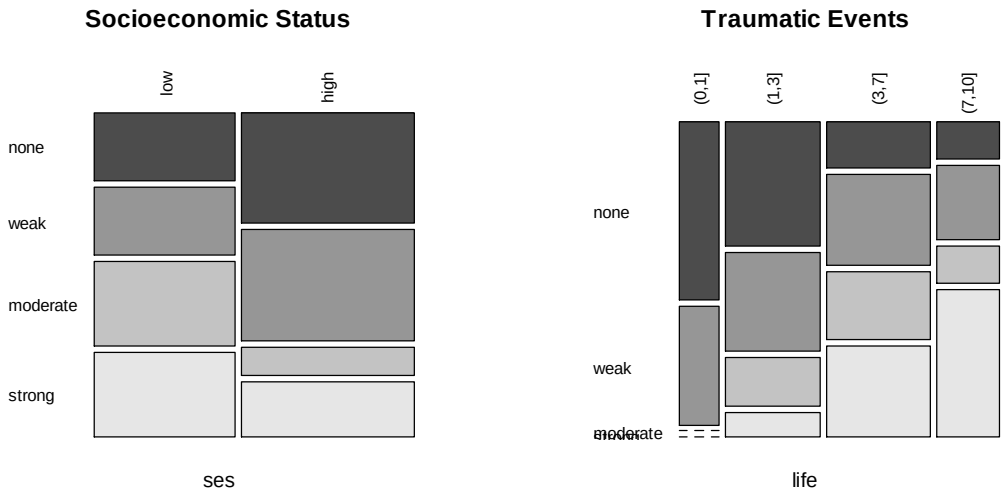
In this section, the response will still be categorical with $J \geq 2$, but now there is now a natural ordering among the categories. Here, we will discuss how the order can be incorporated when fitting a regression type model. We will consider the following example.

Example

In a study, the goal was to relate mental impairment to socioeconomic status and the frequency of potentially traumatic events in a persons' life, such as e.g. death of relatives, divorce, periods of unemployment, etc. The response variable `mental` was ordinal with levels "none", "weak", "moderate" and "strong", taken from the judgment by an expert. Predictor `ses` is binary with levels "low" and "high", coding for the respective socioeconomic status. Finally, behind `life` there is a count, i.e. the number of events. Thus, this predictor was square root transformed.

Again, the raw data are best visualized with mosaic plots below. It seems as if low socioeconomic status and increasing number of traumatic events contribute to a higher state of mental impairment. However, we want to go beyond this visualization of marginal distribution and use a regression model.

```
mosaicplot(table(impair$ses, impair$mental), color=TRUE,
           main="Socioeconomic Status", xlab="ses", cex=.8,
           las=2)

life.intervals <- cut(impair$life, c(0,1,3,7,10))
mosaicplot(table(life.intervals, impair$mental), color=TRUE,
           main="Traumatic Events", xlab="life", cex=.8,
           las=2)
```



Model

Suppose we have J ordered categories and for an individual i, with ordinal response Y_i, we again have:

$$p_{ij} = P(Y_i = j) \text{ for } j = 1,...,J .$$

With an ordered response, it is often easier and more powerful to work with cumulative probabilities, i.e.

$$\gamma_{ij} = P(Y_i \leq j) .$$

These are obviously increasing, and also invariant under the combination of adjacent categories. Moreover, $\gamma_{iJ} = 1$, so we need only to model $J-1$ probabilities. As usual, we must link the γs to a linear combination of the predictors. We will consider three possibilities which all take the form

$$g(\gamma_{ij}) = \alpha_j - x_i^T \beta .$$

For the link function $g(\cdot)$, we can either choose the usual logit, but also the probit and the complementary log-log are possible. Notice that we have explicitly specified the intercepts α_j, thus the predictor vector x_i does not include an intercept. Moreover, the regression coefficients β do not depend on the class j and thus, the predictors have some uniform effect on the response categories.

This model is much easier to comprehend if we use the notion of a latent variable Z_i. It may be thought of as the underlying continuous, but unobserved, response. In practice, we are limited to observing Y_i which are a discretized version of Z_i, and we have:

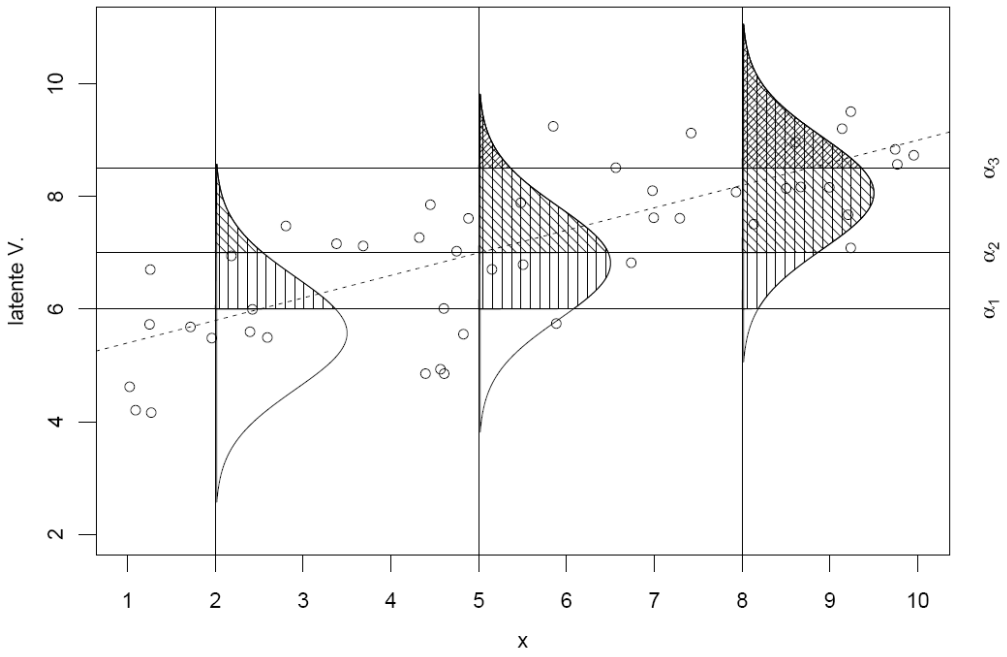
$$Y_i = j \text{, if } \alpha_{j-1} < Z_i \leq \alpha_j .$$

The thresholds α_j do not need to be equidistant and they are usually not known a priori, but estimated from the data. Furthermore, we suppose that the relation

between the latent variable Z_i and the predictors is given by some multiple linear regression model, i.e.

$$Z_i = x_i^T \beta + E_i$$

This notion of a latent continuous response can be visualized with the plot below. We see that Z is positively related to a single predictor x, with errors E having a specified distribution, usually the logistic distribution. The response Y_i has 4 levels that are separated by the thresholds α_j.



We are now considering the event $\{Y_i \leq j\}$, which is equivalent to $\{Z_i \leq \alpha_j\}$. With some algebra, we obtain:

$$\gamma_{ij} = P(Y_i \leq j) == P(Z_i \leq \alpha_j) = P(E_i \leq \alpha_j - x_i^T \beta) = F(\alpha_j - x_i^T \beta),$$

where $F(\cdot)$ is the cumulative distribution function of the E_i. The logistic distribution where $F(x) = e^x / (1+e^x)$ implies the logit link, whereas a Gaussian distribution leads to the probit link, and the extreme value distribution to the so-called complementary log-log link. We continue with the usual logistic distribution and obtain:

$$\gamma_{ij} = \frac{\exp(\alpha_j - x_i^T \beta)}{1+\exp(\alpha_j - x_i^T \beta)}.$$

We return to our mental impairment example and fit the above proportional-odds model using function `polr()` from `library(MASS)`.

```
library(MASS)
fit <- polr(mental ~ ses + life, data=impair)
```

The summary output looks as follows:

```
> summary(fit)

Re-fitting to get Hessian

Call:
polr(formula = mental ~ ses + life, data = impair)

Coefficients:
          Value Std. Error t value
seshigh -1.1112     0.6109  -1.819
life     0.3189     0.1210   2.635

Intercepts:
                Value   Std. Error t value
none|weak       -0.2819  0.6423    -0.4389
weak|moderate    1.2128  0.6607     1.8357
moderate|strong  2.2094  0.7210     3.0644

Residual Deviance: 99.0979
AIC: 109.0979
```

As we had expected from the mosaic plots, `ses` has a negative coefficient, meaning that higher socioeconomic status leads to less mental impairment. On the other hand, the more traumatic events a person has experienced, the bigger his/her potential mental impairment will be.

The summary output does not only contain the regression coefficients, but also the intercepts α_j which serve as categorization thresholds for the latent variable Z_i. They are obtained from a maximum likelihood optimization which is done simultaneously with regression parameter estimation.

Inference

Again, instead of performing single hypothesis tests, it is better to run deviance tests for nested models. We first try to exclude predictor `ses`:

```
> fit.life <- polr(mental ~ life, data=impair)
> deviance(fit.life)-deviance(fit)
[1] 3.429180
> pchisq(3.429180, fit$edf-fit.life$edf, lower=FALSE)
[1] 0.0640539
```

With a p-value of 0.064 there is no firm statistical evidence that the socioeconomic status should be included in the model. On the other hand, when we try to exclude the second predictor `life`:

```
> fit.ses <- polr(mental ~ ses, data=impair)
> deviance(fit.ses)-deviance(fit)
[1] 7.776457
> pchisq(7.776457, fit$edf-fit.ses$edf, lower=FALSE)
[1] 0.005293151
```

Now, we obtain a small p-value and thus, we should not remove the number of traumatic experiences from the full model. We will now compare the model where only predictor `life` is present against the null model with only an intercept:

```
> fit.empty <- polr(mental ~ 1, data=impair)
> deviance(fit.empty)-deviance(fit.life)
[1] 6.514977
> pchisq(6.514977, fit.life$edf-fit.empty$edf, lower=FALSE)
[1] 0.01069697
```

Again, this is clearly significant and we decide that it seems appropriate to model mental impairment by using `life` as a predictor.

Prediction

As above, R allows convenient prediction of either probabilities or class membership. We obtain:

```
> predict(fit.life, type="probs")
         none      weak   moderate     strong
1  0.49337624 0.3037364 0.11173924 0.09114810
2  0.08867378 0.1932184 0.21717188 0.50093592
3  0.29105068 0.3324785 0.18429073 0.19218007
4  0.35380472 0.3345600 0.16025764 0.15137767
5  0.42203463 0.3245379 0.13545441 0.11797305
6  0.56498863 0.2747487 0.09032363 0.06993902
```

for the probabilities of the first 6 instances. The class memberships are again determined by the class with the highest probability, i.e.:

```
> predict(fit.life, type="class")
 [1] none   strong weak   none   none   none
```

Model Diagnostics

As for unordered responses and the multinomial logit model, also here there is no useful definition of residuals that would allow for model enhancement through diagnostics.

17 Non-Parametric Regression

While we first dealt with multiple linear regression models of the form $Y = X\beta + \varepsilon$, we then generalized this concept and allowed responses following a distribution from an exponential family: these were called generalized linear models. In this chapter, we now switch our attention to the linear predictor $\eta = X\beta$ and want to make this relation more flexible.

There is a wide variety of methods available for this. We will first focus on non-parametric regression methods for the single predictor case. These are also known as ***smoothing techniques***. While they can be very useful, an application under the presence of many predictors quickly gets challenging due to the ***curse of dimensionality***. A way out are additive models: they allow for more flexibility than multiple linear regression, yet remain tractable because they impose some additional structure than simple smoothing techniques.

17.1 Introduction

As pointed out above, we here start with a simple regression problem, i.e. one, where only a single predictor is present. Given fixed predictor values $x_1, ..., x_n$, we observe responses $y_1, ..., y_n$ with the relation:

$$y_i = f(x_i) + \varepsilon_i, \text{ for all } i = 1, ..., n.$$

The errors ε_i are assumed to be iid with zero mean and unknown, but constant variance σ_ε^2. Another unknown is the functional relation $f(\cdot)$. So far, we considered parametric approaches, where we assumed $f(\cdot)$ to belong to a parametric family of functions, i.e. $f(\cdot)$ was specified up to a finite number of parameters $\beta_1, ..., \beta_p$. Some examples include:

$$f(x) = \beta_0 + \beta_1 x$$

$$f(x) = \beta_0 + \beta_1 x + \beta_2 x^2$$

$$f(x) = \beta_0 + \beta_1 x + \beta_2 x^{\beta_3}$$

This shows again that the parametric approach is very versatile, in that it is not restricted to linear predictors only. We can add many different model terms such as polynomials and other functions of the predictors to obtain flexible fits. Note that the third function above specifies a parametric, but non-linear regression model, which's fitting and properties were not discussed in this course.

However, no matter what finite parametric family we choose, its flexibility will always be limited and may exclude some plausible functional forms. Thus, there are regression problems where a non-parametric approach is appealing. It allows to choose $f(\cdot)$ from some smooth family of functions, which is generally larger than any parametric family. We do still need to make some assumptions on $f(\cdot)$: usually, these are continuity and some degree of smoothness.

17.2 Advantages and Disadvantages

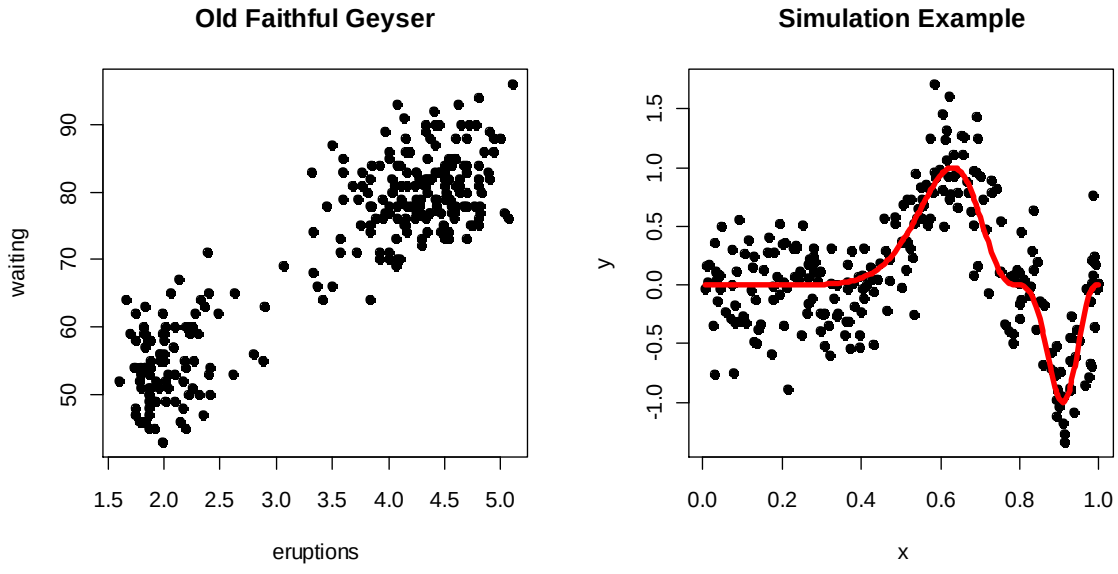
If you have good prior knowledge about an appropriate parametric model family or also if evidence shows that a linear relation between response and predictor(s) is enough, then this is the way to go. Some advantages include:

- If the parametric model is correct, it is more efficient, i.e. less data are necessary to obtain predictions of the same quality.
- There is a formula that describes the response-predictor relation, and the estimated parameters usually have a clear interpretation.
- Formal inference is possible, i.e. we can attribute the variation in the response to one or several predictors.
- Prediction/interpolation is much simpler to perform and in principle, even some (mild) extrapolation is possible, whereas non-parametric models will not yield good predictions in areas where little data were present in fitting.

When some non-linearity becomes apparent from diagnostic plots, we would usually first try some variable transformations. If these prove to be non-sufficient, trying a non-parametric approach can be useful. Some advantages include:

- It requires little to no prior knowledge and is more flexible. We can let a tool do the job of revealing a good functional form.
- When one chooses a wrong parametric model, this will result in a bias. Because the non-parametric approach assumes far less, it is less prone to make bad mistakes.

17.3 Examples



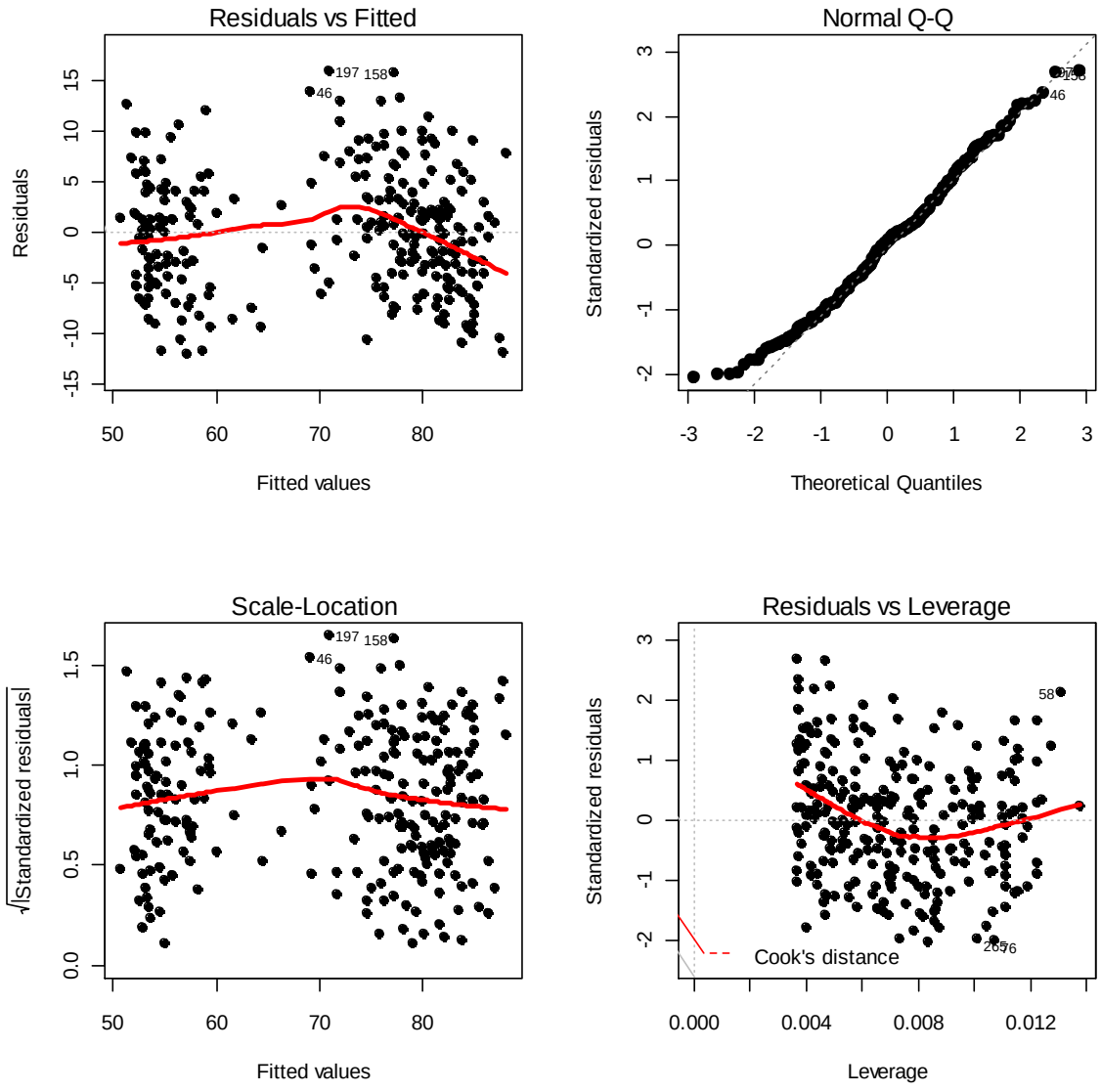
We will consider two examples here, which both originate from the book of Härdle (Smoothing Techniques with Implementations in S, 1991). One is based upon real

data from Old Faithful geyser in Yellowstone National Park. The aim is to relate the waiting time until the next eruption to the duration of the previous eruption. Second, we consider a simulated data set, where

$$f(x) = \sin^3(2\pi x^3).$$

The y-values are obtained by adding a Gaussian error with $\sigma_\varepsilon = 1/3$. This simulated data set has the advantage that we can check how well some smoothing techniques can reveal the true functional relation under the presence of some noise – something that is never possible with real data. We also note that the function $f(\cdot)$ cannot simply be modeled in parametric form.

While it may seem that the relation between eruption and waiting time with Old Faithful geyser is a linear one, the diagnostic plots clearly show that this simple relation is not adequate:



Thus, using some smoothing techniques may well be appropriate for both examples which are considered here.

17.4 Kernel Smoothers

The simplest form of smoothing would be to use some moving average estimator. This means that the y-values are averaged over a fixed-size window of x-values. We can generalize this simple approach by some weighted smoothing using a so-called kernel function $K(\cdot)$. Then, our estimate of $f(\cdot)$, which we will call $\hat{f}_\lambda(\cdot)$ is defined as:

$$\hat{f}_\lambda(x) = \frac{1}{n}\sum_{j=1}^{n} w_j Y_j \text{ , with weights } w_j = \frac{1}{\lambda} \cdot K\left(\frac{x - x_j}{\lambda}\right).$$

For the kernel, we require $\int K = 1$. For moving average smoothing, the kernel is a rectangular box. However in practice, one often prefers to give more weight to the observations that are adjacent to x, and thus e.g. chooses the Gaussian density function as a kernel. Also note that there is an additional parameter λ. It is called the ***bandwidth*** and control the smoothness of the fitted curve.

If the predictor values are spaced very unevenly, the above estimator can yield poor results. This problem can be mitigated somewhat by the ***Nadaraya-Watson kernel estimator***. It is defined as follows:

$$\hat{f}_\lambda(x) = \frac{\sum_{j=1}^{n} w_j Y_j}{\sum_{j=1}^{n} w_j}, \text{ with } w_j \text{ as above.}$$

This estimator is a modified version of the above one. Its advantage is that the weights for the fitted value at each observation x_i will sum up to one. It can be shown that the mean squared error

$$MSE(x) = E[(f(x) - \hat{f}_\lambda(x))^2]$$

with optimal choice of the smoothing parameter λ is of the order $n^{-4/5}$. This has to be compared with the MSE of a parametric model which better, and of order n^{-1}. However, this only holds if the parametric model is correct. Else, there will be a certain point after which the parametric model does no longer improve. Thus, smoothing is like the insurance business – we pay a premium that protects against "damage" and is lost in cases of good outcome, i.e. when the true relation could be described by a parametric model. However, we are (asymptotically) safe in situations when there is "damage", i.e. a parametric model is not appropriate.

17.5 Choosing the Kernel

For the application of a kernel estimator, we have to fix $K(\cdot)$ and λ. We will treat the choice of the former first. Some desirable properties of a kernel are smoothness and compactness. Smoothness is required such that the resulting fit, $\hat{f}_\lambda(\cdot)$ is smooth, which rules out the rectangular box kernel. Compactness ensures

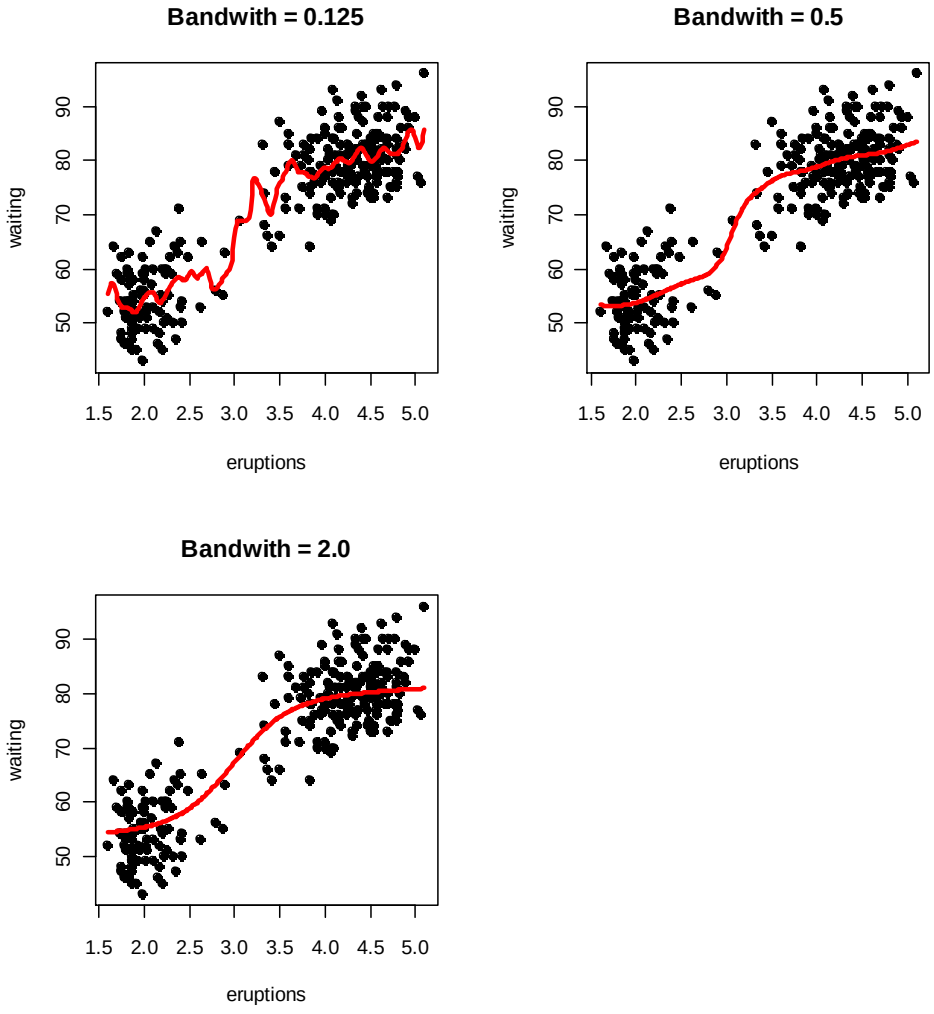
that only “local data” have impact on the fitted value $\hat{y}_i$. This is the reason why the Gaussian kernel is less common than the so-called ***Epanechnikov kernel***:

$$K(x) = \begin{cases} \frac{3}{4}(1-x^2), & if \ |x|<1 \\ 0 & else \end{cases}$$

It can be shown that this kernel is satisfactory regarding smoothness and compactness, and also allows for speedy computation. However, smoothing techniques usually are not too crucially dependent on kernel choice and many kernel functions will yield acceptable results.

17.6 Choice of the Bandwith

Far more important is the choice of the smoothing parameter λ. If we choose a too small λ, then we undersmooth and the fit will be very rough. On the other hand, if λ is too large, we oversmooth and important features will be lost. Thus, the goal is to find the right compromise between fitting noise and canceling out true structure. There are two basic approach to do so, a) by eyeballing, and b) using quantitative approaches such as (generalized) cross-validation.

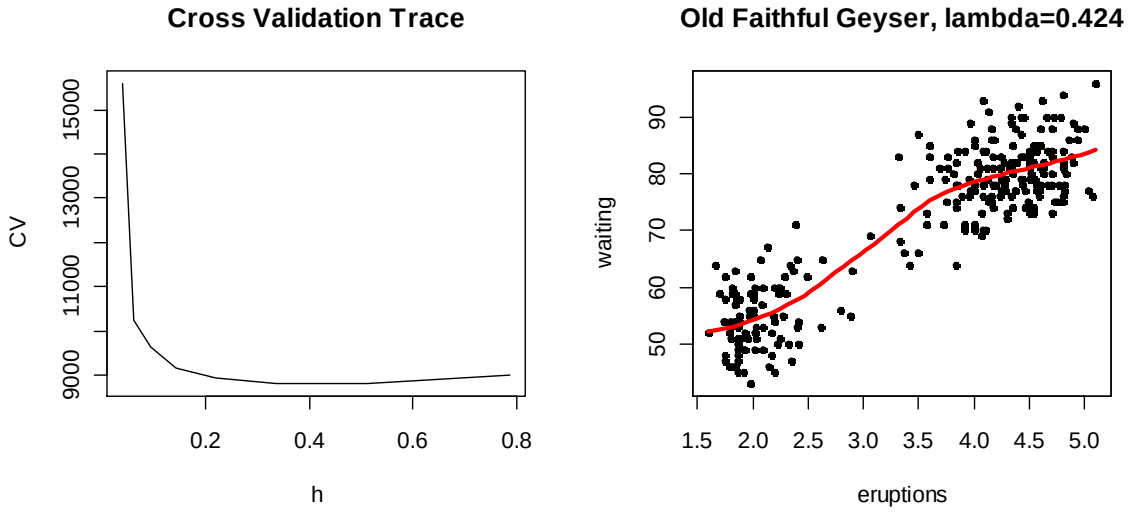


The Nadaraya-Watson estimator is implemented in function `ksmooth()`. We have to choose the kernel and can set the bandwidth. As can be seen above, we experiment with the default value of $\lambda = 0.5$, as well as with $\lambda = 0.125$ and $\lambda = 2$. Clearly, $\lambda = 0.125$ is too small, resulting in a fit which is too rough. The other two choices both seem reasonable.

Because it is a tough call between $\lambda = 0.5$ and $\lambda = 2$, and also because any subjectivity in such decisions is usually undesired, we will consider an automatic method for smoothing parameter selection. Cross validation is popular for this task. The criterion is

$$CV(\lambda) = \frac{1}{n}\sum_{j=1}^{n}(y_i - \hat{f}_{\lambda(j)}(x_j))^2 ,$$

where $\hat{f}_{\lambda(j)}(\cdot)$ is the fit that is obtained when the jth data point was omitted from the fitting process. Thus, we fit j smoothers and for each, we compute the discrepancy between the fit for x_j and the observed response y_j. Of course, this needs to be done for a set of candidate λ that may seem suitable according to some eyeballing.

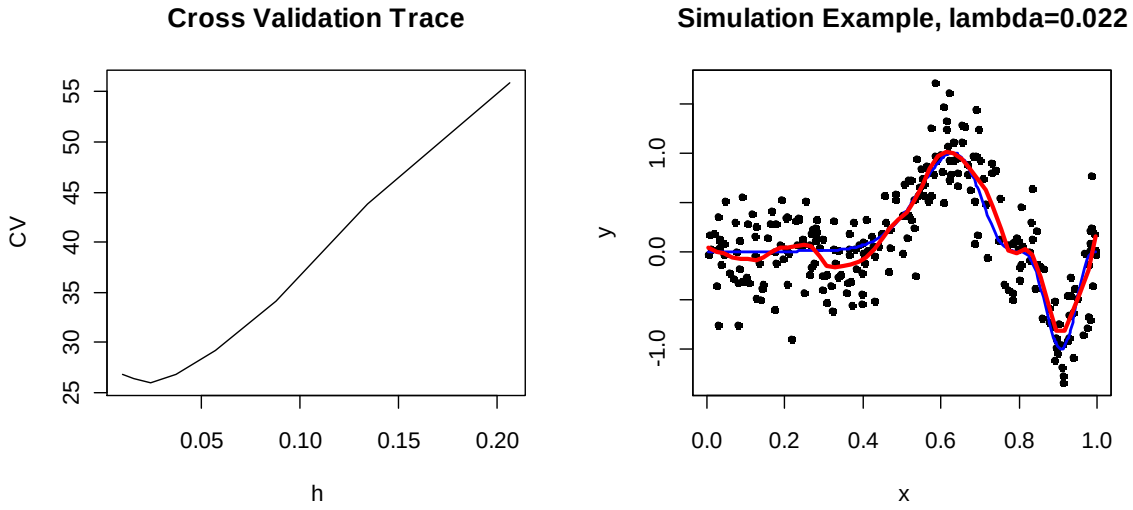


This is a computationally expensive process which is also quite cumbersome to implement. For both these drawbacks, there are solutions. Sometimes generalized cross validation (GCV) approaches are used which allow for closed form computation of some fitting criterion. On the other hand, function `hcv()` in `library(sm)` in R has a relatively quick implementation for choosing λ via cross validation:

```
library(sm)
lambda <- hcv(eruptions, waiting, display="lines")
sm.regression(eruptions, waiting, h=lambda)
```

The left panel above shows the cross validation criterion as a function of λ. In function `sm.regression()` a Nadaraya-Watson estimator with Gaussian kernel

is employed. The smoothing parameter λ then is the standard deviation of the kernel. We here also show the result with the simulation example:



The cross validation quite clearly votes for a small λ. In the fit, we observe some slight undersmoothing, but overall, the smoother is able to quite well reveal the true structure.

17.7 Smoothing Splines

The basic notion behind the non-parametric regression is that there is that the relation between predictor and response is:

$$Y_i = f(x_i) + \varepsilon_i$$

Now our goal typical goal is to minimize the errors ε_i. Without any restriction on the smoothness of $f(\cdot)$, this would naturally lead to a solution that connects all data points, is very rough and avoids any error. As we have seen above, this is not desirable, because we then treat random variation as true structure. To overcome these problems, we may choose the smoothing function such that it balances goodness-of-fit versus the smoothness:

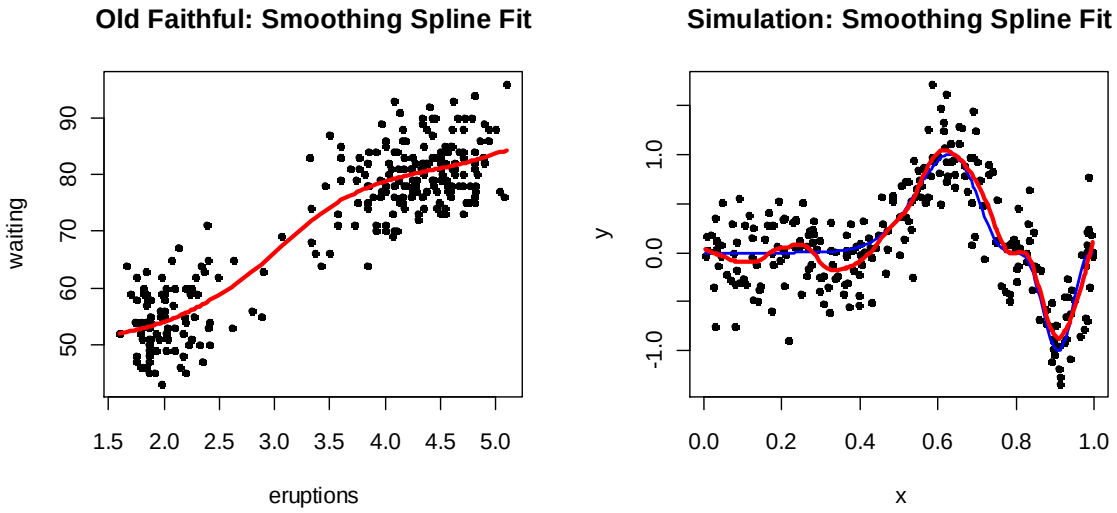
$$\frac{1}{n}\sum_{i=1}^{n}(Y_i - f(x_i))^2 + \lambda\int\left(f''(x)\right)^2 dx$$

This is known as the ***smoothing spline fit***. Some algebra shows that the solution $f(\cdot)$ is always going to be a piecewise cubic polynomial in each interval (x_i, x_{i+1}) (hereby w.l.o.g. assuming that the x_i are sorted). These have the property that $f(\cdot)$ and its first two derivatives are continuous.

Because we now know the general form of the solution and it is parametric, the task is reduced to estimating the coefficients of the cubic polynomials. In R, this is implemented in function `smooth.spline()`, which also contains an automatic selection of the smoothing parameter λ that is based on cross validation:

```
fit <- smooth.spline(eruptions, waiting)
plot(eruptions, waiting)
lines(fit)

fit <- smooth.spline(x, y)
plot(x, y)
lines(x, m)
lines(fit)
```

The results with smoothing splines are very similar to what we obtained from the kernel estimators, though the fitting paradigm is completely different here. Yet, we will still consider a third approach for non-parametric regression



17.8 Local Polynomials

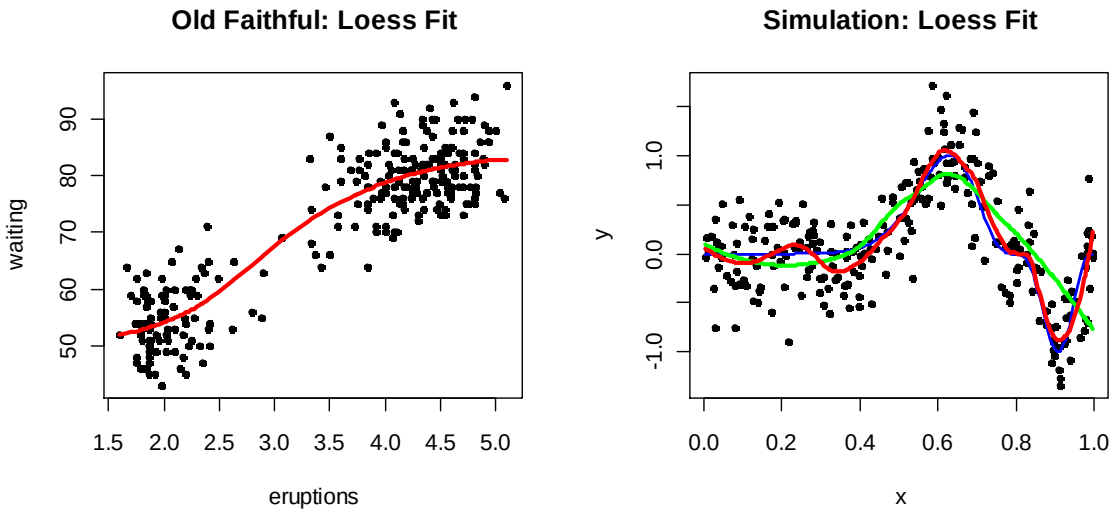
Both kernel and spline smoothing seem to do a good job on the two problems we supplied. However, both of them are relatively vulnerable to the presence of outliers. A way out is to use a smoothing procedure that is based on fitting local polynomials. This works as follows.

We must select a window. Then, a polynomial is fitted to the data within this window using some robust method. The predicted response at the window center is the fitted value. Then, the window is slid over the range of the data. In R, function `loess()` implements this, with polynomials of second order for the local fits. The smoothing parameter is the window with, which per default is set to `3/4`. The code for this fit is as follows:

```
fit  <- loess(waiting ~ eruptions, data=faithful)
plot(eruptions, waiting)
lines(fit$x[order(fit$x)], fit$fitted[order(fit$x)])
fit1 <- loess(y ~ x, data=exa)
fit2 <- loess(y ~ x, data=exa, span=0.22)
plot(x, y, pch=19, cex=0.7)
lines(fit1$x[order(fit1$x)], fit1$fitted[order(fit1$x)])
```

```
lines(fit2$x[order(fit2$x)], fit2$fitted[order(fit2$x)])
```

For the simulation example, we observe that the first fit with the default window size (green line in the plot below) results in oversmoothing, where important features in the data are canceled out. We thus correct the window size such that it matches the cross validation result obtained with the kernel estimator. Then, the result is again similar as before.



17.9 Comparison of Methods

We have now seen three different smoothing techniques which all resulted in very similar fits. As this has been a subject of tremendous interest in the statistical community for a fairly long time, many more approaches do exists. Saying which is the best smoother is impossible – this depends on the data, i.e. the task at hand and fact whether human intervention for bandwidth selection is feasible. Due to its robust nature, the loess smoother is much liked in practice. In cases where there are no outliers, smoothing splines yields very similar fits, but is computationally cheaper.

We will conclude this section with some general remarks: generally, if there was no noise, interpolating between the data points would be the method of choice. When some moderate amount of noise is present, non-parametric regression is often appealing: there is enough signal to justify a flexible fit, and also enough noise to make smoothing worthwhile. Finally, with larger amounts of noise or very sparse data, parametric methods become relatively more attractive, because nothing more than a simple model can be justified.

18 Additive Models

For problems with more than one predictor, the above smoothing techniques are no longer appropriate. We require a sufficient number of data points in the neighborhood where local fitting is performed. In higher dimensions, this is hard to come by with – a fact which is known as the curse of dimensionality. However, it may still be the case that a multiple linear regression model of the form

$$Y_i = \beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip} + \varepsilon_i$$

is not fully appropriate. There is a wide choice of (predictor) transformations which could be tried. But if there are a large number of explanatory variables, this quickly becomes very time consuming and it would be very convenient to have an automatic tool that assists. This is achieved by the additive model:

$$Y_i = \beta_0 + f_1(x_1) + ... + f_p(x_p) + \varepsilon_i ,$$

where the $f_j(\cdot)$ are some smooth, potentially non-parametric functions. For the errors, we again assume that they are i.i.d. with zero mean and constant variance. Such additive models are far more flexible than a linear model (without transformations), but can still be efficiently fitted and well interpreted, because the $f_j(\cdot)$ can be plotted to give an impression of the marginal relationships.

Additive modeling (and its fitting algorithms) is again very versatile: we can restrict some $f_j(\cdot)$ to be the identity times β_j. This is, using a predictor in a parametric, rather than a non-parametric form. This would also be the natural procedure if there are some categorical variables that serve as predictors. Moreover, also the presence of interactions between a (two-level) categorical and a continuous variable is possible, which means that (two) different non-parametric functions are fitted for that predictor.

18.1 Software for Fitting Additive Models

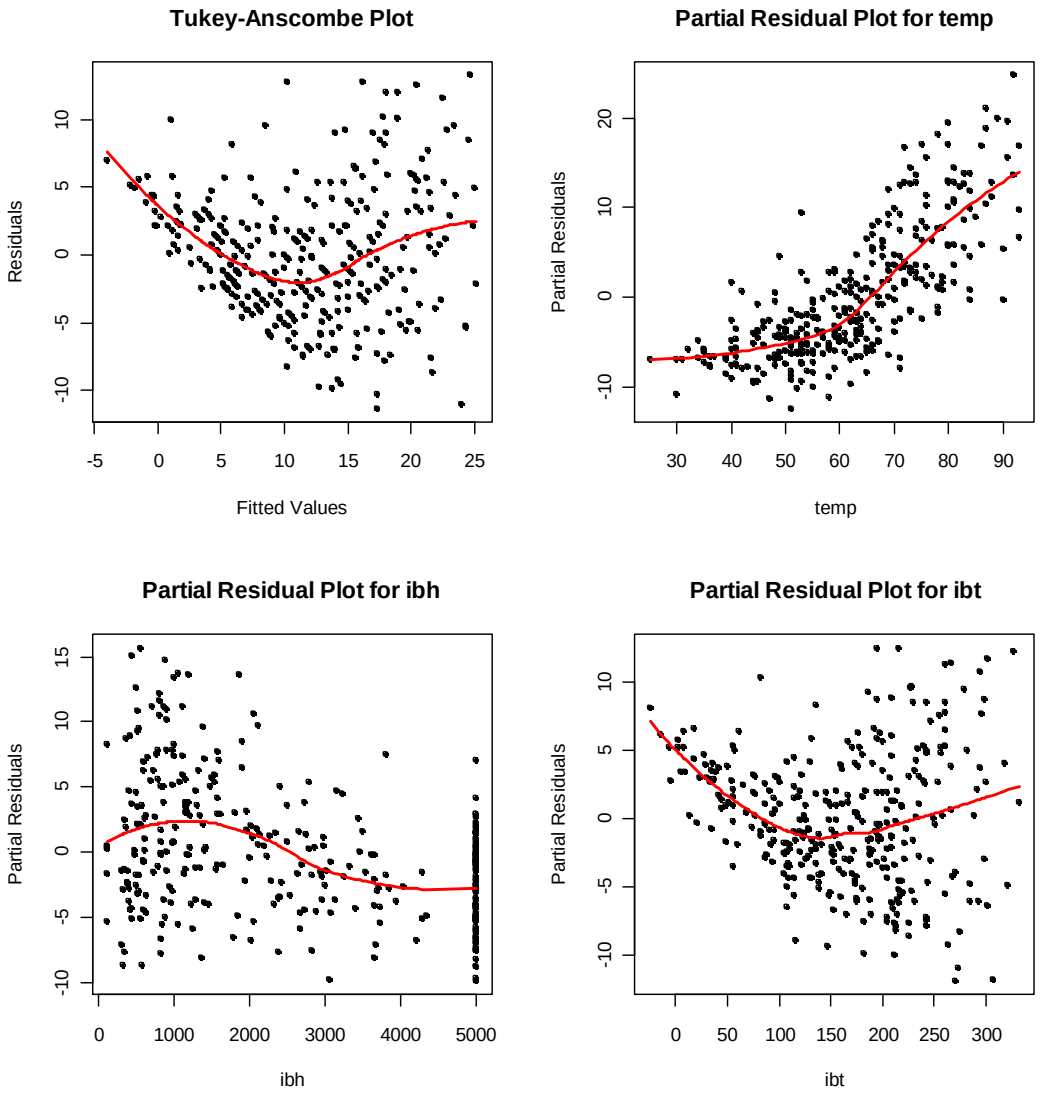
There are several packages with which additive models can be fit in R. The two most popular ones are `library(gam)` and `library(mgcv)`. The former allows more choice in the smoothers that are employed and is based on a backfitting algorithm. This is an iterative procedure that is based on univariate fits. These can principally be based on any non-parametric regression method, e.g. smoothing splines or loess. We can even use different smoothers on different predictors with differing amounts of smoothing.

On the other hand, `library(mgcv)` is based on a penalized smoothing spline approach. This means that the additive model is in fact re-expressed as a complex parametric model based on cubic polynomials. This is all done behind the scenes, and it also includes the choice of the smoothing parameters based on generalized cross validation.

18.2 Example

We here use data from a study of the relation between atmospheric ozone concentration and some meteorological predictors. The data originate from the Los Angeles basin and were recorded in 1976. We only consider three predictors: `temp`, the temperature measured at El Monte, `ibh`, the inversion base height at the LAX airport, and `ibt`, the inversion top temperature, again at LAX. We first fit a multiple linear regression model that will serve as a reference.

```
>  summary(lm(O3 ~ temp + ibh + ibt, data = ozone))
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -7.7279822  1.6216623  -4.765 2.84e-06 ***
temp         0.3804408  0.0401582   9.474  < 2e-16 ***
ibh         -0.0011862  0.0002567  -4.621 5.52e-06 ***
ibt         -0.0058215  0.0101793  -0.572    0.568
---
Residual standard error: 4.748 on 326 degrees of freedom
Multiple R-squared: 0.652,    Adjusted R-squared: 0.6488
F-statistic: 203.6 on 3 and 326 DF,   p-value: < 2.2e-16
```



We note that predictor `ibt` is not significant, whereas else, according to the summary output, the fit does seem reasonable. The next step is to perform some model diagnostics, see the plots on the previous page.

The Tukey-Anscombe plot shows a very clear violation of the model assumptions. We conjecture that the fitted model is not adequate. Moreover, the partial residuals (which contain the effect that can be attributed to a predictor, after the response has been corrected for the effect of all other predictors) of all three predictors show clear non-linearity. We could now try to improve this by searching for transformations. While this may be beneficial not only for improving the fit, but also for understanding the physical mechanisms behind, we here do without and rely on additive models instead. First, we employ function `gam()` and use a loess smoother on each predictor:

```
> summary(fit.gam)

Call: gam(O3 ~ lo(temp) + lo(ibh) + lo(ibt), data=ozone)

Deviance Residuals:
     Min       1Q   Median       3Q      Max
-13.1146  -2.3624  -0.2092   2.1732  12.4447

(Dispersion Parameter for gaussian family: 18.6638)

    Null Deviance: 21115.41 on 329 degrees of freedom
Residual Deviance: 5935.096 on 318.0005 degrees of freedom
AIC: 1916.049

Df for Terms and F-values for Nonparametric Effects

            Df Npar Df Npar F     Pr(F)
(Intercept)  1
lo(temp)     1     2.5 7.4550 0.0002456 ***
lo(ibh)      1     2.9 7.6205 8.243e-05 ***
lo(ibt)      1     2.7 7.8434 9.917e-05 ***
```

What can we learn from this? First, we see that there are deviances and a dispersion parameter. This shows to us, that function `gam()` can also deal with non-Gaussian structures. Indeed, there is an extension to ***Generalized Additive Models***, hence the name GAM. Second, we note that all three parameters are now significant. However, we spend more degrees of freedom here due to the use of the loess smoother; they vary between 2.5-2.9. Moreover, we can compute an approximate coefficient of determination:

```
> 1-5935.096/21115.41
[1] 0.7189211
```

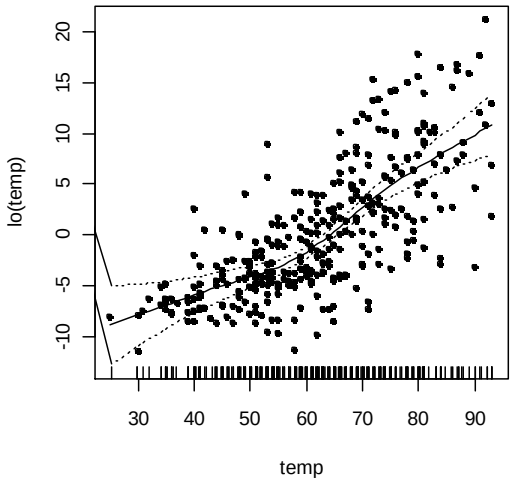
It improved from 0.652 with the multiple linear regression to 0.719 with the additive model. The latter, however, spends more degrees of freedom. Thus, the comparison is not a fair one. Also note that the individual hypothesis tests in the summary output above originate from a score test and should be seen as

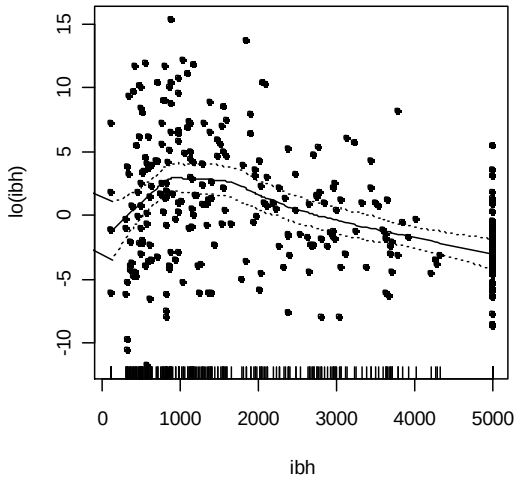
approximate at best. It is usually more reliable to perform a nested model comparison which has an approximate F-distribution:

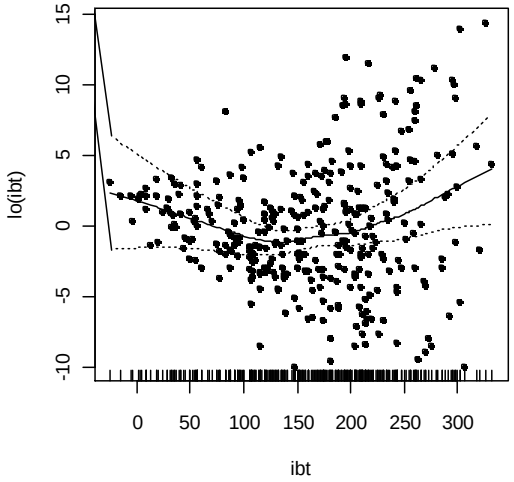
```
> fit.gam.small <- gam(O3 ~ lo(temp) + lo(ibh), data=ozone)
> anova(fit.gam.small, fit.gam, test="F")


Analysis of Deviance Table
Model 1: O3 ~ lo(temp) + lo(ibh)
Model 2: O3 ~ lo(temp) + lo(ibh) + lo(ibt)
  Resid. Df Resid. Dev     Df Deviance      F Pr(>F)
1    321.67     6044.6
2    318.00     5935.1 3.6648   109.47 1.6005  0.179
```

In contrast to the result from the summary output, we here obtain a p-value of 0.18 which means that predictor `ibt` is not significant. We can now do some model diagnostics:

```
> plot(fit.gam, residuals=TRUE, se=TRUE, pch=19, cex=0.7)
```







This shows the three partial residual plots with the smooth functions that were fitted, including some confidence bands. For `ibt`, a horizontal line would fit within the confidence bands, which yields some further evidence that we can do without this predictor. Moreover, outliers or leverage points may be detected in these plots, however here, they do not exist. However, if they did, then it is usually wise to work with `library(gam)` and loess smoothers.

For the sake of completeness, we here show the fitting process when working with `library(mgcv)`, too:

```
> fit.mgcv <- gam(O3 ~ s(temp) + s(ibh) + s(ibt), data=ozone)
> summary(fit.mgcv)

Family: gaussian
Link function: identity

Formula:
O3 ~ s(temp) + s(ibh) + s(ibt)

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  11.7758     0.2382   49.44   <2e-16 ***
---
Approximate significance of smooth terms:
          edf Ref.df      F  p-value
s(temp) 3.386  4.259 20.681 6.84e-16 ***
s(ibh)  4.174  5.076  7.338 1.36e-06 ***
s(ibt)  2.112  2.731  1.400    0.245
---
R-sq.(adj) =  0.708   Deviance explained = 71.7%
GCV score = 19.346  Scale est. = 18.72     n = 330
```

Here, we use splines as a smoother for the three predictors. In fact, we are forced to do so, because there is no alternative. The amount of smoothing is chosen internally by a GCV approach, while with `library(gam)`, this needs to be controlled by the user (note that we relied on the default values, which worked well in this example).

There is some more evidence into the direction that predictor `ibt` is not significant. Moreover, the R-squared is (i.e. Deviance explained) takes a similar value. Here, we also obtain an adjusted R-squared, which shows some improvements on the previous one from multiple linear regression. Finally, we can inspect the partial residual plots. We here without displaying them, because the resulting plots look almost identical to the ones obtained with `library(gam)`.