The **lm** function fits a linear model with normal errors and constant variance. This function is generally used for regression analysis using continuous xplanatory variables. For models including only categorical explanatory ariables or mix of categorical and continuous variables the **aov** function is enerally used. The syntax for the two functions is similar. A typical call to he, e.g., **lm** function is:

```
fit<-lm(y ~ x)
```

fter a model has been fitted using **lm** or **aov**, a number of *generic* functions xists that can be used to get information about the fitted model. Some useful eneric functions are:

- **summary(fit)**

The form of the output returned by the function depends on the class of ts argument. When the argument is an **lm** object, the function computes nd returns a list of summary statistics of the fitted linear model including arameter estimates and standard errors. If the argument is an **aov** object n analysis of variance table is returned. Possible to choose **summary.aov** or **ummary.lm** to get the alternative form of output.

- **update(fit)**

unction to use to modify and re-fit a model.

- **plot(fit)**

roduces diagnostic plots for model checking.

- **model.matrix(fit)**

xtracts the model matrix/design matrix from a model formula.

- **anova(fit)**

omputes analysis of variance (or deviance) tables for one or more fitted nodel objects. When given a single argument it returns a table which tests hether the model terms are significant but when given a sequence of objects he function tests the models against one another.

- **coef(fit)**

xtracts model coefficients from the estimated model.

- **fitted(fit)**

xtracts fitted values, predicted by the model for the values of the explanatory ariable(s) included.

- **predict(fit)**

roduces predicted values, standard errors of the predictions and confidence

- **residuals(fit)**

Extracts residuals from the model object.

- **rstandard(fit)**

Extracts standardized residuals from the model object.

- **rstudent(fit)**

Extracts studentized residuals from the model object.

- **hatvalues(fit)**

Returns the diagonal elements of the hat matrix.

Contrasts

Functions contrasts may be constructed by the functions, **contr.treatment**, **contr.helmert**, **contr.poly**, **contr.sum** with the number of levels as arguments. The default contrast in R is the treatment contrast which contrasts each level with the baseline level (specified by **base**). **cont.helmert** returns Helmert contrasts, **contr.poly** returns contrasts based on orthogonal polynomials and **contr.sum** uses "sum to zero contrasts." See also Example 3.9.

3.13 Problems

Please notice that some of the problems are designed for being solved by hand calculations while others require access to a computer.

Exercise 3.1
Consider the generalized loss function (deviance) defined in Definition 2.12. Show that for $(Y_1, Y_2, \ldots, Y_n)^T$ being a sequence of i.i.d. normally distributed variables the deviance becomes the classical sum of squared error loss function.

Exercise 3.2
Consider the regression model

$$Y_t = \beta x_t + \varepsilon_t$$

where $\mathrm{E}[\{\varepsilon_t\}] = 0$. Suppose that n observations are given.

Question 1 Assume that $\mathrm{Var}[\varepsilon_t] = \sigma^2 / x_t^2$ but that the elements of the sequence $\{\varepsilon_t\}$ are mutually uncorrelated. Consider the unweighted least squares estimator $(\widehat{\beta}^*)$.

- Is the estimator unbiased?
- Calculate the variance of the estimator.

Question 2 Calculate the variance of the weighted least squares estimator $(\widehat{\beta})$.

This assumption does not contradict the assumption of a multiplicative Poisson model (no interaction) for accident *counts*. The count of injured persons is compounded with the number of accidents in the sense that for each accident there is at least one injured person, but the number of injured persons may vary from 1 to some upper limit. Therefore, it is meaningful to assume that the number of injured persons may be described by a Poisson distributed random variable with some constant overdispersion σ^2.

To estimate σ^2, we use the residual deviance, $\mathrm{D}\left(\boldsymbol{y};\boldsymbol{\mu}(\widehat{\boldsymbol{\beta}})\right) = 15.74$ with 6 degrees of freedom, leading to $\widehat{\sigma}^2 = 2.6235$. For testing purposes, we shall use *scaled deviances* in analogy with (4.40).

Rescaling the residual deviances by division by $\widehat{\sigma}^2 = 2.6235$ we obtain the following table of Type III partitioning of the scaled deviances:

```
> OD <- 1/2.6235
> w <- rep(OD,12)

> glmacc2<-glm(formula = Numacc ~ Year + Quarter,
           weights = w, family=poisson(link=log),
           data=accdat2)

> anova(glmacc2,test='Chisq')

Analysis of Deviance Table
Model: poisson, link: log
Response: Numacc
Terms added sequentially (first to last)

        Df Deviance Resid. Df Resid. Dev P(>|Chi|)
NULL                       11    25.5392
Year     2   2.0123         9    23.5270    0.3656
Quarter  3  17.5269         6     6.0000    0.0006
```

.9 Generalized linear models in R

lodel fitting

eneralized linear models are analyzed by means of the function **glm**. Most of e features of a **lm** object are also present in a **glm** object. The first arguments the **glm** function are:

- A *linear model* specified by a model formulae.
- A **family** object specifying the distribution. Valid family names are **binomial**, **gaussian**, **Gamma**, **inverse.gaussian**, **poisson**, **quasi**, quasibinomial and quasipoisson.

- A link function given as an argument to the family.

The model formulae are specified the same way as in the **lm** function, see Section 3.12 on page 81. A typical call to the **glm** function is:

```
\item fit<-glm(y ~ x, family=poisson(link = log))
```

After a model has been fitted using **glm**, some of the same *generic* functions that are used for the general linear model can by used. A list of some useful functions is given in Section 3.12 on page 81.

The linear prediction and the fitted values are extracted from a **glm** object (e.g., fit) by using the method **predict.glm**. A simple way of extracting the linear prediction and the fitted values is to use the expressions **predict.glm**(fit) and **fitted**(fit), respectively.

Residuals

The residuals may be extracted from a **glm**-object by using the **residuals**() method with an argument type = specifying the type. The deviance residuals is the default option in the **residuals** method. They may also be extracted by specifying type='**deviance**'. The response residuals are produced by specifying type='**response**', the Pearson residuals by specifying type='pearson' and the the working residuals are produced from a **glm**-object by specifying type=' working' in the **residuals** extractor.

Dispersion parameter

The **summary** reports the value for the dispersion parameter σ^2. For **binomial** and Poisson families, the dispersion parameter is 1, and is not estimated in these cases. For the **Gamma** and Gaussian families, σ^2 is estimated by the Pearson X^2 statistic (4.66). The dispersion parameter is used in the computation of the reported standard errors and z-values for the individual coefficients. These defaults can be overridden by specifying the value for the dispersion parameter using the dispersion = argument to the **summary** method. Specification of dispersion = 0 will result in the Pearson estimate, irrespective of the family of the object.

In case of overdispersion, **family** = **quasi** can be used which allows for the choice of a quasi-distribution where the user specifies the variance function $V(\cdot)$. For quasibinomial and quasipoisson errors, **family** = quasibinomial or **family** = quasipoisson can be used, respectively.

4.10 Problems

Please notice that some of the problems are designed for being solved by hand calculations while others require access to a computer.