University of Economics, Prague

Faculty of Informatics and Statistics

and

Charles University in Prague

Faculty of Mathematics and Physics

Exercises for Non-life Insurance

Michal Pešta
Barbora Petrová
Tereza Smolárová
Pavel Zimmermann

Preface

abc

Keywords

insurance

Notation

$a.s.$	...	almost surely
$\mathcal{B}$	...	Brownian bridge
$\xrightarrow{a.s.}$	...	convergence almost surely
$\xrightarrow{\mathscr{D}}$	...	convergence in distribution
$\xrightarrow{\mathsf{P}}$	...	convergence in probability P
$\xrightarrow{\mathscr{D}[a,b]}$	...	convergence in the Skorokhod topology on $[a, b]$
O, o	...	deterministic Landau symbols, confer Appendix
E	...	expectation
iid	...	independent and identically distributed
$\mathcal{I}$	...	indicator function
$\mathbb{Z}$	...	integers, i.e., $\{\ldots, -2-1, 0, 1, 2, \ldots\}$
$\mathbb{N}$	...	natural numbers, i.e., $\{1, 2, \ldots\}$
$\mathbb{N}_0$	...	natural numbers with zero, i.e., $\{0, 1, 2, \ldots\}$
P	...	probability
$\mathbb{R}$	...	real numbers
sgn	...	signum function
$\mathscr{D}[a, b]$	...	Skorokhod space on interval $[a, b]$

$\mathcal{W}$	...	standard Wiener process
O_{P},o_{P}	...	stochastic Landau symbols, confer Appendix
$[\cdot]$	...	truncated number to zero decimal digits (rounding down the absolute value of the number while maintaining the sign)
Var	...	variance

Contents

Chapter 1

Application of generalized linear models in tariff analysis

Generalized linear models, often known by the acronym GLMs, represent an important class of regression models that have found extensive use in actuarial practice. The statistical study an actuary performs to obtain a tariff is called a *tariff analysis*. In the 1990's British actuaries introduced GLMs as a tool for tariff analysis and since then this has become the standard approach in many countries. The aims of this chapter are to present the basic theory of GLMs in the tariff analysis setting and to demonstrate three datasets thereon.

1.1 The basic theory of pricing with GLMs

The actuary uses the data to find a model which describes for example how the claim frequency or claim severity depends on the number of explanatory variables, i.e., rating factors. We next define some basic concepts used in connection with the tariff analysis.

A claim is an event reported by the policy holder, for which she/he demands economic compensation. *The duration of a policy* is the amount of time it is in force, usually measured in years, in which case one may also use the term policy years. The duration of a group of policies is obtained by adding the duration of the individual policies. *The claim frequency* is the number of claims divided by duration, i.e., the average number of claims per time period (usualy one year). *The claim severity* is the total claim amount divided by the number of claims, i.e., the average cost per claim. *The pure premium* is the total claim amount divided by the duration, i.e., average cost per policy year (or other time period). The pure premium is the product of claim frequency and claim severity.

1.1.1 The structure of generalized linear models

In our notation of GLMs, we assume that the data are in a list form with the n observations organized as a column vector $\mathbf{y} = (y_1, \ldots, y_n)^\top$, which is supposed to be a realization of a random vector $\mathbf{Y} = (Y_1, \ldots, Y_n)^\top$. The generalized linear models are based on the following three building blocks:

(i) *An exponential family of distributions:* The dependent variable Y_i (for the ith of n independently sampled observations) has a distribution from the exponential family with frequency function

$$f_{Y_i}(y_i; \theta_i, \phi) = \exp\left\{ \frac{y_i\theta_i - b(\theta_i)}{\phi/w_i} + c(y_i, \phi, w_i) \right\}, \tag{1.1}$$

where θ_i is an unknown parameter of our interest. The technical restriction, which relates to parameter θ_i is that the parametric space must be open, i.e., θ_i takes values in an open set. Another unknown parameter is called a *dispersion parameter* $\phi > 0$, which is the same for all i. A known prior weight $w_i \geqslant 0$ on the another hand as θ_i may vary between observations. A *cumulant function* $b(\theta_i)$ is twice continuously differentiable with an invertible second derivative. For every choice of such a function, we get a family of probability distributions, e.g., the normal, Poisson and gamma distributions. Given the choice of $b(\cdot)$, the distribution is completely specified by the parameters θ_i and ϕ. The known function $c(\cdot, \cdot, \cdot)$ is of little interest in GLMs theory as it does not depend on θ_i.

The mean and variance of the dependent variable Y_i with frequency function (1.1) are expressed as

$$\begin{aligned} \mathsf{E}(Y_i) &= \mu_i = b'(\theta_i), \\ \mathsf{Var}(Y_i) &= \phi b''(\theta_i)/w_i = \phi v(\mu_i)/w_i, \end{aligned} \tag{1.2}$$

where $v(\mu_i) = b''((b')^{-1}(\mu_i))$ is a *variance function*. The proof (1.2) can be found at Ohlsson and Johansson (2010) in Section 2.1.3.

Remark 1.1. It is important to keep in mind that the expression (1.1) is only valid for the y_i that are possible outcomes of Y_i. For other values of y_i we assume $f_{Y_i}(y_i; \theta_i, \phi) = 0$.

Remark 1.2. The dependent variable Y_i may be discrete, continous or a mixture. Thus, frequency function(1.1) may be interpreted as a probability density or a probability mass function, depending on the application. For example Y_i may be number of claims, claim frequency or claim severity.

Remark 1.3. *Weights of observations*: The weights, w in general, have several different interpretations in actuarial applications, e.g., duration when Y_i is claim frequency or number of claims when the Y_i is claim severity.

(ii) *A linear predictor:* A linear combination of explanatory variables is considered

$$\eta_i = \mathbf{x}_i^\top \boldsymbol{\beta} = \sum_{j=1}^{r} x_{ij}\beta_j; \qquad i = 1, \ldots, n, \tag{1.3}$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_r)^\top$is a r-dimensional vector of unknown parameters and x_{ij} is a given value of a covariate x_j for observation i. For $r \leqslant n$, we have a $n \times r$ known matrix $\mathbf{X}$ of explanatory variables.
Note that the expression (1.3) is called a *systematic component* of the model.

(iii) *A link function:* The fundamental object in the GLMs is called a *link function*, since it links the mean to the linear structure through

$$g(\mu_i) = \eta_i = \mathbf{x}_i^\top \boldsymbol{\beta} = \sum_{j=1}^{r} x_{ij}\beta_j; \qquad i = 1, \ldots, n. \tag{1.4}$$

The known link function $g(\cdot)$ must be strictly monotone and twice differentiable. The choice of the link function depends on character of the data and is somehow arbitrary. In non-life insurance pricing, a logarithmic link function is by far the most common one. An inverse of the link function, $g^{-1}(\eta_i) = \mu_i$, is a *mean function*.

Remark 1.4. *Offset*: Sometimes, part of the expected value is known beforehand. This situation can be handled by including an offset in the GLM analysis. This could be done on the linear scale because the offset is simply an additional explanatory variable, whose coefficient is fixed at 1

$$g(\mu_i) = \mathbf{x}_i^\top \boldsymbol{\beta} + \xi_i; \qquad i = 1, \ldots, n. \tag{1.5}$$

The offset is often interpreted as a measure of exposure. In actuarial context, the exposure might be duration.

1.1.2 GLM in the multiplicative model

A multiplicative model is often reasonable choice of model in non-life insurance pricing. Here for simplicity, we consider model with just two rating factors. Note that different notation is used as it was introduced in the GLM structure but we go back to the list form notation at the end.

We denote a tariff cell (k,l), where $k = 1,\dots,K$ and $l = 1,\dots,L$ indicate the class of the first and second rating factor respectively. The expected value of the dependent variable Y_{kl} in tariff cell (k,l) is a target of the tariff analysis and it is given as $\mathsf{E}Y_{kl} = \mu_{kl}$. Thus, the multiplicative model is express as

$$\mu_{kl} = \gamma_0\gamma_{1k}\gamma_{2l}. \tag{1.6}$$

The parameters γ_{1k} correspond to the different classes fot the first rating factor and γ_{2l} for the second. Further, we have to specify a reference cell, called a *base cell*, to make the parameters unique as the model (1.6) is over-parametrized. Say for simplicity that the base cell is $(1,1)$. Then we put $\gamma_{11} = \gamma_{21} = 1$. Now we can interprete γ_0 as a *base value*. The other parameters $\gamma_{..}$ are called *relativities*. They measure the relative difference in relation to the base cell. For example, if $\gamma_{12} = 0.8$ then the mean in cell $(2,1)$ is 20% lower than in the base cell $(1,1)$.

In general the overall level of premium is controlled by adjusting the base value γ_0, while the others parameters control how much to charge for a policy, given this base value. In practice, we first determine the relativities $\gamma_{..}$ and then the base value is set to give the required overall premium.

By taking the logarithms of both sides in (1.6), we get

$$\log\mu_{kl} = \log\gamma_0 + \log\gamma_{1k} + \log\gamma_{2l}. \tag{1.7}$$

Now, we can rewrite the model (1.7) to the list form notation by sorting the tariff cells in order $(1,1),(1,2),\dots,(1,K),(2,1),\dots,(2,L)$. We reindex the dependent variables, their expected values and rename the parameters as follows

$$\begin{array}{ccc}
Y_1 = Y_{11} & \mu_1 = \mu_{11} & \beta_1 = \log\gamma_0 \\
Y_2 = Y_{21} & \mu_2 = \mu_{21} & \beta_2 = \log\gamma_{12} \\
\vdots & \vdots & \vdots \\
Y_K = Y_{K1} & \mu_K = \mu_{K1} & \beta_K = \log\gamma_{1K} \\
Y_{K+1} = Y_{12} & \mu_{K+1} = \mu_{12} & \beta_{K+1} = \log\gamma_{22} \\
\vdots & \vdots & \vdots \\
Y_{K\cdot L} = Y_{KL} & \mu_{K\cdot L} = \mu_{KL} & \beta_{K+L-2} = \log\gamma_{2L}
\end{array}$$

By the aid of the dummy variables, the multiplicative model for the expected value

can be rewritten

$$\log \mu_i = \sum_{j=1}^{K+L-2} x_{ij}\beta_j; \qquad i = 1, \ldots, K \cdot L. \tag{1.8}$$

This is the same linear structure as in (1.4). The dependent variable Y_i has a distribution from the exponential family with frequency function (1.1) and for $\forall i$ they are mutualy independent random variables. The logarithmic function is strictly monotone and twice differentiable, therefore it is the link function.

In practice, it seems to have become a standard to use in non-life insurance pricing the GLM with a relative Poisson distribution for claim frequency and GLM with a relative gamma distribution for claim severity. In both cases the log link function is used; see Ohlsson and Johansson (2010) in Sections 2.1.1 and 2.1.2.

1.1.3 Parameter estimation

This section presents a maximum likelihood method for estimating the regression parameters $\boldsymbol{\beta}$ in (1.4). First, we recapitulate some relations between the parameters

$$\begin{aligned} &\mu_i = b'(\theta_i), \qquad \eta_i = g(\mu_i), \qquad \eta_i = \mathbf{x}_i^\top \boldsymbol{\beta} = \sum_{j=1}^{r} x_{ij}\beta_j, \\ &\theta_i = (b')^{-1}(\mu_i), \qquad \mu_i = g^{-1}(\eta_i). \end{aligned} \tag{1.9}$$

The log-likelihood function for each of the independent random variable Y_i with frequency function (1.1) is expressed as

$$\ell_i(y_i; \theta_i, \phi) = \frac{y_i\theta_i - b(\theta_i)}{\phi/w_i} + c(y_i, \phi, w_i).$$

We consider the log-likelihood $\ell_i(y_i; \theta_i, \phi)$ to be a function of parameters θ_i and ϕ with given y_i. Even though is the function of $\boldsymbol{\beta}$, rather than θ_i as $\theta_i = (b')^{-1}(g^{-1}(\mathbf{x}_i^\top \boldsymbol{\beta}))$; see (1.9). The log-likelihood function for random vector $\mathbf{Y}$ is then

$$\ell(\mathbf{y}; \boldsymbol{\theta}, \phi) = \sum_{i=1}^{n} \ell_i(y_i; \theta_i, \phi) = \frac{1}{\phi} \sum_{i=1}^{n} w_i(y_i\theta_i - b(\theta_i)) + \sum_{i=1}^{n} c(y_i, \phi, w_i).$$

The partial derivative of ℓ with respect to β_j, by the chain rule, equals to

$$\frac{\partial \ell}{\partial \beta_j} = \sum_{i=1}^{n} \frac{\partial \ell_i}{\partial \theta_i} \frac{\partial \theta_i}{\partial \mu_i} \frac{\partial \mu_i}{\partial \eta_i} \frac{\partial \eta_i}{\partial \beta_j} = \frac{1}{\phi} \sum_{i=1}^{n} w_i \frac{y_i - \mu_i}{v(\mu_i) g'(\mu_i)} x_{ij},$$

where we used the following calculations and relations from (1.9)

$$\begin{aligned}
\frac{\partial \ell_i}{\partial \theta_i} &= \frac{w_i(y_i - b'(\theta_i))}{\phi}, \\
\frac{\partial \theta_i}{\partial \mu_i} &= \frac{1}{\frac{\partial \mu_i}{\partial \theta_i}} = \frac{1}{\frac{\partial b'(\theta_i)}{\partial \theta_i}} = \frac{1}{b''(\theta_i)} = \frac{1}{v(\mu_i)}, \\
\frac{\partial \mu_i}{\partial \eta_i} &= \frac{1}{\frac{\partial \eta_i}{\partial \mu_i}} = \frac{1}{\frac{\partial g(\mu_i)}{\partial \mu_i}} = \frac{1}{g'(\mu_i)}, \\
\frac{\partial \eta_i}{\partial \beta_j} &= x_{ij}.
\end{aligned}$$

By setting all these r partial derivatives equal to zero and multiplying by ϕ, which does not have any effect of the maximization, we get the maximum likelihood equations

$$\sum_{i=1}^{n} w_i \frac{y_i - \mu_i}{v(\mu_i) g'(\mu_i)} x_{ij} = 0; \qquad j = 1, \ldots, r. \tag{1.10}$$

It might look as if the solution is simply $\mu_i = y_i$, but then we forget that $\mu_i = \mu_i(\boldsymbol{\beta})$ also has to satisfy the relation given by

$$\mu_i = g^{-1}(\eta_i) = g^{-1}\left(\sum_{j=1}^{r} x_{ij}\beta_j \right).$$

Except for a few special cases, e.g., in a saturated model the solution is $\hat{\mu}_i = y_i$ the maximum likelihood equations (1.10) must be solved numerically. Newton-Raphson's method and Fisher's scoring method are widely used numerical methods for solving (1.10). More about these methods and a question whether these equations give an unique maximum of the likelihood can be found in Ohlsson and Johansson (2010) in Sections 3.2.3 and 3.2.4. We define the saturated model in the following section.

In the multiplicative model (1.6), we are not interested in the estimates of parameters β but rather the relativities γ, which are the basic building blocks of the tariff. These are found by the relation $\hat{\gamma}_j = \exp \hat{\beta}_j$ for $j = 1, \ldots, r$.

1.1.4 Model selection

The actuary might want to investigate whether the chosen model fits the data well, whether to add additional rating factors or omit the included ones. The following statistical tests can be used to evaluate these aims.

Goodness of fit

The goodness of fit is measured by two different statistics, the deviance and the Pearson chi-square statistic. Note that in this part we assume that the dispersion parameter ϕ is fixed.

A model with the maximum number of parameters, i.e., there are as many parameters as observations ($r = n$) is called a *saturated model*. It is also referred as a *maximum* or a *full model*. In this case, we get a perfect fit by setting all $\hat{\mu_i} = y_i$. These are in fact the maximum likelihood estimates of expected values, since they satisfy the maximum likelihood equations (1.10). In this model the log-likelihood function achieves its maximum achievable value and is denoted by

$$\ell^*(\mathbf{y}; \hat{\boldsymbol{\theta}}^*, \phi) = \sum_{i=1}^{n} \frac{y_i \hat{\theta_i^*} - b(\hat{\theta_i^*})}{\phi / w_i} + \sum_{i=1}^{n} c(y_i, \phi, w_i),$$

where $\hat{\theta}_i^* = (b')^{-1}(y_i)$ for $i = 1, \ldots, n$. While this model is trivial and of no practical interest, it is often used as a benchmark in measuring the goodness of fit of other models, since it has a perfect fit.

A *scale deviance* of the fitted model D^* is defined as two times the difference between maximum log-likehood achievable and maximum log-likehood obtained by the fitted model, i.e.,

$$\begin{aligned} D^* &= 2[\ell(\mathbf{y}; \hat{\boldsymbol{\theta}}^*, \phi) - \ell(\mathbf{y}; \hat{\boldsymbol{\theta}}, \phi)] \\ &= \frac{2}{\phi} \sum_{i=1}^{n} w_i [y_i(\hat{\theta}_i^* - \hat{\theta}_i) - (b(\hat{\theta}_i^*) - b(\hat{\theta}_i))], \end{aligned} \tag{1.11}$$

where $\hat{\theta}_i = (b')^{-1}(g^{-1}(\mathbf{x}_i^\top \hat{\boldsymbol{\beta}}))$. The term *scaled* refers to the scalling by parameter ϕ. In some cases an unscaled deviance is preferable as it is not dependent on parameter ϕ. We get the deviance by multiplying the expression (1.11) by ϕ, i.e., $D = \phi D^*$.

Another classic and important statistic used for measuring the goodness of fit of the fitted model is the generalized Pearson chi-square statistic X^2 defined as

$$X^2 = \sum_{i=1}^{n} \frac{(y_i - \hat{\mu}_i)^2}{\mathsf{Var}(Y_i)} = \frac{1}{\phi} \sum_{i=1}^{n} w_i \frac{(y_i - \hat{\mu}_i)^2}{v(\hat{\mu}_i)}, \tag{1.12}$$

where $v(\hat{\mu}_i)$ is the estimated variance function for the given distribution. Pearson chi-square statistic X^2 is approximately χ^2 disributed with $n - r$ degrees of freedom, where r is the number of estimated parameters β. Note that as before, there is also an unscaled Pearson chi-square statistic ϕX^2.

Submodel testing

It is very important to test the significance of included rating factors, since model gives more accurate estimates when is not over-parametrized. As an aid in deciding whether or not to include a rating factor in the model, we may perform a likelihood ratio test (LRT) of the two models against each other, with and without the particular rating factor. We denote the model with p, $1 \leqslant p < r$, parameters β omitted as a submodel. This test is only for two nested models with a null hypothesis

$$H_0 = \textit{submodel holds} \text{ vs. alternative } H_1 = \textit{model holds}.$$

The LR statistic is define as

$$LR = 2 \log \left[\frac{L(model)}{L(submodel)} \right], \tag{1.13}$$

where L is maximized likelihood function under the model or submodel. The LR statistic is under null hypothesis approximately χ^2 distributed with p degrees of freedom. The null hypothesis is rejected when the large values of LR are observed.

The parameter ϕ is also included in the LR statistics distribution and therefore it has to be estimated. When in the both log-likelihoods the same estimator of parameter ϕ is used, we can rewrite the LR statistic into a difference of the deviances (1.11). Note that in this case R does not report the log-likelihood values but it reports deviances.

Akaike information criterion (AIC)

How do we compare models when they are not nested? One way is to use an Akaike information criterion

$$AIC = -2 \times \ell(\mathbf{y}; \hat{\boldsymbol{\theta}}, \hat{\phi}) + 2 \times (\textit{number of parameters}).$$

The term $2 \times (\textit{number of parameters})$ is a penalty for the complexity of the model. With this penalty, we cannot improve on a fit simply by introducing additional parameters. Note that we can always make the log-likelihood greater by introducing additional parameters. This statistic can be used when we are comparing several alternative models that are not necessarily nested. We pick the model that minimizes AIC. If the models under consideration have the same number of parameters, we choose the model that maximizes the log-likelihood.

1.1.5 Nuisance parameter estimation

In section 1.1.3 we have estimated the regression coefficients $\boldsymbol{\beta}$ using the maximum likelihood method. Parameter ϕ is also often uknown in practice. Our aim in this part is to present one of the possibilities how to find an approximately unbiased estimate $\hat{\phi}$ of parameter ϕ.

The expected value of Pearson chi-square statistic X^2 (1.12) is close to $n-r$ as X^2 is approximately χ^2_{n-r}-distributed, i.e.,

$$\mathsf{E}(X^2) = \mathsf{E}\left(\frac{1}{\phi}\sum_{i=1}^{n} w_i \frac{(y_i-\hat{\mu}_i)^2}{v(\hat{\mu}_i)}\right) \approx (n-r).$$

From the expression above, we conclude that the $\hat{\phi}_{X^2}$ is the approximately unbiased estimate of parameter ϕ

$$\hat{\phi}_{X^2} = \frac{1}{n-r}\sum_{i=1}^{n} w_i \frac{(y_i-\hat{\mu}_i)^2}{v(\hat{\mu}_i)}. \tag{1.14}$$

1.2 Examples of pricing with GLMs

Our main aim in the following three examples is to perform a standard GLM tariff analysis. We carry out a separate analysis for claim frequency and severity and then we obtain the relativities for the pure premium by multiplying the results from these two analyses. Whole source code can be found at the R files *AutoCollision_SC.R*, *busscase_SC.R* and *mccase_SC.R*.

1.2.1 Automobile UK Collision

Mildenhall (1999) considered 8,942 collision losses from private passenger United Kingdom automobile insurance policies. The data were derived from Nelder and McCullagh (1989, Section 8.4.1) but originated from Baxter, Coutts, and Ross (1980). Here, we consider a sample of $n = 32$ of Mildenhall data.

Exercise 1.1. Perform a GLM tariff analysis for the Automobile UK Collision data.

We obtain a dataset *AutoCollision* from an insurance data package. It is a collection of insurance datasets, which are often used in claims severity and claims frequency modelling. More information about this package can be found on a web site
https://cran.r-project.org/web/packages/insuranceData/index.html

```
install.packages("insuranceData")
library("insuranceData")
```

```
data(AutoCollision)
auto <- AutoCollision
```

The drivers are divided into cells on the basis of the following two rating factors:

Age The age group of driver, in the span A-H. The youngest drivers belong to age group A, the oldest to age group H.

Vehicle_ Use Purpose of the vehicle use:

1 - Business

2 - DriveShort, which means drive to work but less than 10 miles

3 - DriveLong, which means drive to work but more than 10 miles

4 - Pleasure

For each of $8 \cdot 4 = 32$ cells, the following totals are known:

Severity Average amount of claims (severity) in pounds sterling adjusted for inflation.

Claim_ Count Number of claims.

We now have a closer look at our data and some descriptive statistics.

```
auto
summary(auto)
```

```
      Age        Vehicle_Use    Severity      Claim_Count
 A      :4   Business  :8    Min.   :153.6   Min.   :  5.0
 B      :4   DriveLong :8    1st Qu.:212.4   1st Qu.:116.2
 C      :4   DriveShort:8    Median :250.5   Median :208.0
 D      :4   Pleasure  :8    Mean   :276.4   Mean   :279.4
 E      :4                   3rd Qu.:298.2   3rd Qu.:366.0
 F      :4                   Max.   :797.8   Max.   :970.0
 (Other):8
```

We have only one observation for each combination of each level of rating factors. From the summary output above, we see that all average amounts of claims are higher than zero so we won't have any constraints in claim severity modelling.

Claim frequency

The number of claims in each cell is the quantity of our interest in this part. Therefore, we will consider a variable *Claim_ Count* as a dependent variable.

Following plots 1.1 show the conditional histograms separated out by drivers age and vehicle use. It can be seen that with increasing age also increases the number of claims. For vehicle use, those who drive to work less than 10 miles had a various numbers of claims.

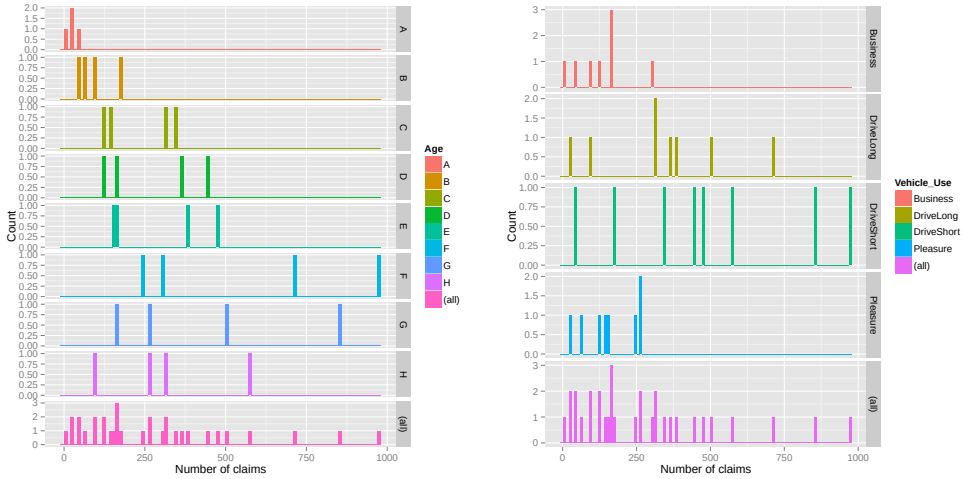

Figure 1.1: Conditional histograms.

We fit a Poisson distribution with a logarithmic link function and two rating factors *Age* and *Vehicle_ Use*. We store it in an object *model.frequency_ p* and get a summary at the same time.

```
summary(model.frequency_p <-glm(Claim_Count ~Age + Vehicle_Use,
data=auto, family=poisson))
```

Table 1.1 shows the estimates of Poisson regression coefficients for each variable, along with their standard errors, Wald z-statistics (z-values) and the associated p-values. We tested null hypothesis $H_0 : \beta_j = 0$ vs. alternative $H_1 : \beta_j \neq 0$ for $j = 1, \ldots, 11$.

	Estimate	Std. Error	z-value	p-value
(Intercept)	2.370	0.110	21.588	< 0.001
AgeB	1.425	0.118	12.069	< 0.001
AgeC	2.347	0.111	21.148	< 0.001
AgeD	2.515	0.110	22.825	< 0.001
AgeE	2.582	0.120	23.488	< 0.001
AgeF	3.225	0.108	29.834	< 0.001
AgeG	3.002	0.109	27.641	< 0.001
AgeH	2.639	0.110	24.053	< 0.001
Vehicle_UseDriveLong	0.925	0.036	25.652	< 0.001
Vehicle_UseDriveShort	1.286	0.034	37.307	< 0.001
Vehicle_UsePleasure	0.166	0.041	4.002	< 0.001

Table 1.1: Summary of model.frequncy_p.

The indicator variable, shown as *AgeB* is the expected difference in log count between

age class B and the reference age class A when all the other variables are held constant. The expected log count for age class B is 1.425 higher than the expected log count for age class A. The same interpretation of the estimates of regression coefficients applies to variable *Vehicle_ Use* where the reference class is business. We won't be considering any submodel testing due to the fact that all p-values are much more lower than 0.05.

We would like to find out whether our model *model.frequncy_ p* fits the data reasonably. We can use a residual deviance to perform a goodness of fit test.

```
with(model.frequency_p, cbind(res.deviance = deviance, df =
df.residual, p = pchisq(deviance, df.residual, lower.tail = FALSE)))
```

P-value $3.574 \cdot 10^{-28}$ strongly suggests that the data does not fit the model well. It is possible that over-dispersion is present and a negative binomial regression would be more appropriate for modelling claim frequency. We perform the negative binomial regression with logarithmic link using glm.nb function from a MASS package. We fit the model and store it in an object *model.frequency_ nb*. We get a summary 1.2 at the same time.

```
library("MASS")
summary(model.frequency_nb <-glm.nb(Claim_Count ~Age + Vehicle_Use,
data=auto))
```

	Estimate	Std. Error	z-value	p-value
(Intercept)	2.362	0.150	15.790	< 0.001
AgeB	1.430	0.166	8.618	< 0.001
AgeC	2.383	0.160	14.880	< 0.001
AgeD	2.531	0.160	15.855	< 0.001
AgeE	2.597	0.159	16.290	< 0.001
AgeF	3.213	0.158	20.342	< 0.001
AgeG	2.958	0.158	18.665	< 0.001
AgeH	2.631	0.159	16.514	< 0.001
Vehicle_UseDriveLong	0.915	0.090	10.170	< 0.001
Vehicle_UseDriveShort	1.291	0.089	14.490	< 0.001
Vehicle_UsePleasure	0.225	0.093	2.432	0.015

Table 1.2: Summary of model.frequncy_nb.

The interpretation of the estimates of regression coefficients is the same as in Poisson regression model. Again we won't be considering any submodel testing due to the fact that all p-values are lower than 0.05.

We would like to compare Poisson model with negative binomial model (Poisson model is nested in the negative binomial model) and check the negative binomials assumption that stays the conditional mean is not equal to the conditional variance. To do so we can use a likelihood ratio test.

```
pchisq(2*(logLik(model.frequency_nb)-logLik(model.frequency_p)),
df=1, lower.tail= FALSE)
```

P-value $3.946 \cdot 10^{-22}$ strongly suggests that the negative binomial model is more appropriate for our data than the Poisson model. To see how well negative binomial model fits the data, we will perform a goodness of fit chi-squared test.

```
with(model.frequency_nb, cbind(res.deviance = deviance, df =
df.residual, p = pchisq(deviance, df.residual, lower.tail = FALSE)))
```

From the results of the above tests, we have decided to choose the negative binomial with logarithmic link function model *model.frequency_ nb* as a final claim frequency model even though the goodness of fit test is statistically significant (p-value equals to 0.031).

Claim severity

The average amount of claims in each cell is the quantity of interest in this part. First, we create various plots for a better visualisation of the dependence between the severity and the rating factors.

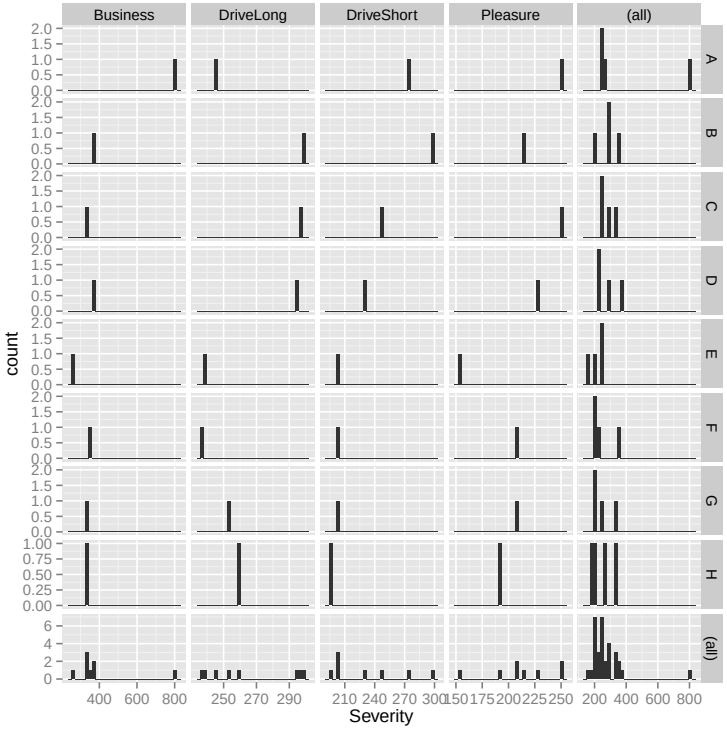

Figure 1.2: Conditional histograms.

Conditional histograms separated out by age and vehicle use classes are shown above at

Figure 1.2 and the violin plots at Figure 1.3. The highest average amount of claims was observed for the youngest drivers who drive for business. On the another hand the lowest was observed for the middle aged drivers, i.e., age class E, which drives for pleasure.

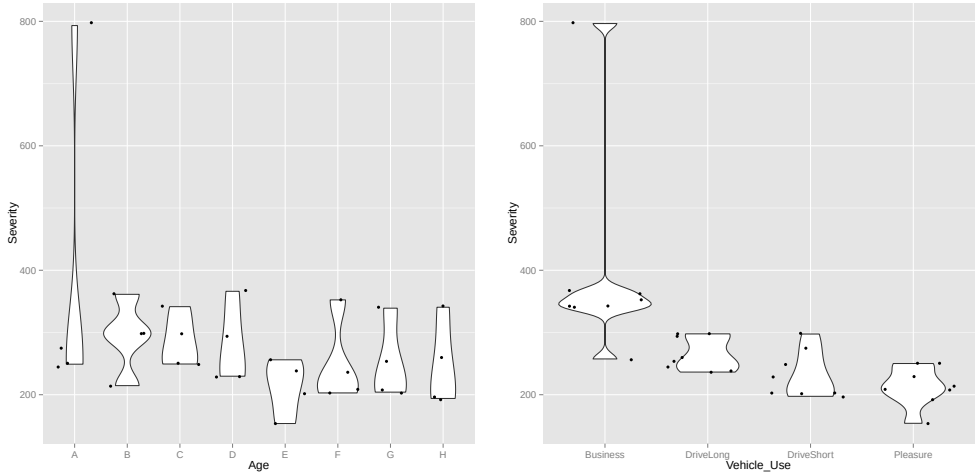

Figure 1.3: Violin plots.

We fit a gamma distribution with a logarithmic link function and two rating factors *Age* and *Vehicle_ Use*. We store it in an object *model.severity_ g* and get a summary 1.3 at the same time.

```
summary(model.severity_g <- glm(Severity ~ Age + Vehicle_Use,
data = auto, family = Gamma("log")))
```

	Estimate	Std. Error	t-value	p-value
(Intercept)	6.241	0.101	61.500	< 0.001
AgeB	-0.208	0.122	-1.699	0.104
AgeC	-0.230	0.122	-1.881	0.074
AgeD	-0.263	0.122	-2.149	0.043
AgeE	-0.531	0.122	-4.339	< 0.001
AgeF	-0.382	0.122	-3.121	0.005
AgeG	-0.374	0.122	-3.057	0.006
AgeH	-0.394	0.122	-3.218	0.004
Vehicle_UseDriveLong	-0.357	0.087	-4.128	< 0.001
Vehicle_UseDriveShort	-0.505	0.087	-5.839	< 0.001
Vehicle_UsePleasure	-0.589	0.087	-6.801	< 0.001

Table 1.3: Summary of model.severity_g.

From the summary output 1.3, we see that two p-values are higher than 0.05. We will

test the overall effect of variable *Age* by comparing the deviance of the full model with the deviance of the model excluding drivers age.

```
model.severity_g1 <- update(model.severity_g, . ~ . - Age)
anova(model.severity_g1, model.severity_g, test = "Chisq")
```

The seven degree of freedom chi-square test indicates that *Age*, taken together, is a statistically significant predictor of *Severity* and we will continue working with an initial full model *model.severity_ g*.

We have also fitted gamma and normal distributions with several different link functions. We used the residual deviances to perform a goodness of fit tests for our models to find out how well these models fit the data. The deviances together with p-values from goodness of fit testing are shown in following Table 1.4:

Model	Null deviance	Res. deviance	p-value
Gamma with log link	3.206	0.581	1
Gamma with inverse link	3.206	0.426	1
Gamma with inverse squared link	3.206	0.318	1
Gamma with identity link	3.206	0.716	1
Normal with identity link	378148	136500	0
Normal with log link	378148	91618	0
Normal with inverse link	378148	40461	0

Table 1.4: Deviance table.

From the p-values at Table 1.4, we see that the normal regression models do not fit nearly as well as the gamma models. As a final severity model, we have chosen gamma with logarithmic link function model *model.severity_ g* even though it does not have the smallest value of residual deviance. The reason is better interpretability among the models.

Pure premium: Combining the models

We have chosen age class A and vehicle use class business as a base tariff cell. The choice was based only on our own decision. Note that any other combination can be chosen.

```
rel <- data.frame(rating.factor =
c(rep("Age", nlevels(auto$Age)), rep("Vehicle use",
nlevels(auto$Vehicle_Use))),
class = c(levels(auto$Age),levels(auto$Vehicle_Use)),
stringsAsFactors = FALSE)
print(rel)
```

We determine the relativities for claim frequency and severity separately by using GLMs.

```
rels <- coef( model.frequency_nb)
```

```
rels <- exp( rels[1] + rels[-1] ) / exp( rels[1] )
rel$rels.frequency <- c(c(1, rels[1:7]), c(1, rels[8:10]))

rels <- coef(model.severity_g)
rels <- exp(rels[1] + rels[-1])/exp(rels[1])
rel$rels.severity <- c(c(1, rels[1:7]), c(1, rels[8:10]))
```

Then we multiply these results to get relativities for pure premium. All relativities are shown at Table 1.5.

```
rel$rels.pure.premium <- with(rel, rels.frequency * rels.severity)
print(rel, digits = 2)
```

rating factor	class	rel. frequency	rel. severity	rel. pure premium
Age	A	1.0	1.00	1.0
Age	B	4.2	0.81	3.4
Age	C	10.8	0.79	8.6
Age	D	12.6	0.77	9.7
Age	E	13.4	0.59	7.9
Age	F	24.9	0.68	17.0
Age	G	19.3	0.69	13.2
Age	H	13.9	0.67	9.4
Vehicle use	Business	1.0	1.00	1.0
Vehicle use	DriveLong	2.5	0.70	1.7
Vehicle use	DriveShort	3.6	0.60	2.2
Vehicle use	Pleasure	1.3	0.56	0.7

Table 1.5: Relativities from the negative binomial GLM for claim frequency and the log gamma GLM for claim severity.

1.2.2 Bus Insurance

Transportation companies own one or more buses which are insured for a shorter or longer period. Dataset *busscase* carries information about 670 companies that were policyholders at the former Swedish insurance company Wasa during the years $1990-1998$ and it contains following variables in Swedish acronyms:

zon Geographic zone numered from 1 to 7, in a standard classification of all Swedish parishes:

1 - central and semi-central parts of Sweden's three largest cities

2 - suburbs and middle-sized towns

3 - lesser towns, except those in 5 or 7

4 - small towns and countryside, except $5-7$

5 - northern towns

6 - northern countryside

7 - Gotland (Sweden's largest island)

bussald The age class of the bus, in the span $0-4$.

kundnr An ID number for the company, re-coded here for confidentially reasons.

antavt Number of observations for the company in a given tariff cell based on zone and age class. There may be more than one observation per bus, since each renewal is counted as a new observation.

dur Duration measured in days and aggregated over all observations in the tariff cell.

antskad The corresponding number of claims.

skadkost The corresponding total claim cost.

Dataset is available on a web site http://staff.math.su.se/esbj/GLMbook

Exercise 1.2. Perform a GLM tariff analysis for the Bus Insurance data.

We start with loading the data into R and looking at some desriptive statistics.

```
bus<-read.table("http://staff.math.su.se/esbj/GLMbook/busscase.txt")
summary(bus)
names(bus) <- c("zon", "bussald", "kundnr", "antavt", "dur", "antskad",
"skadkost")
summary(bus)
```

```
      zon           bussald          kundnr          antavt
 Min.   :1.000   Min.   :0.000   Min.   :  1.0   Min.   :  1.00
 1st Qu.:3.000   1st Qu.:1.000   1st Qu.:179.0   1st Qu.:  2.00
 Median :4.000   Median :3.000   Median :328.5   Median :  4.00
 Mean   :3.967   Mean   :2.615   Mean   :335.2   Mean   : 12.87
 3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:506.0   3rd Qu.:  9.00
 Max.   :7.000   Max.   :4.000   Max.   :670.0   Max.   :392.00

      dur           antskad           skadkost
 Min.   :    1   Min.   :  0.000   .      :926
 1st Qu.:  365   1st Qu.:  0.000   0      :221
 Median :  725   Median :  0.000   -1820  : 20
 Mean   : 2284   Mean   :  1.953   -1815  :  9
```

```
 3rd Qu.: 1876   3rd Qu.:  1.000   -1785  :  4
 Max.   :66327   Max.   :402.000   -1810  :  4
                                   (Other):358
```

From the summary output above, we see that some changes need to be made before we start with the GLM tariff alanysis. Variable *skadkost* is a factor and in 926 observations a dot is recorded. When we have a closer look at our data we can see that variable *antskad* is in these observations equal to zero. This implies zero claim cost. Due to this fact, we create a new variable *skadkostN*, where are the dots replaced by 0. The original variable *skadkost* remains unchanged.

```
bus$skadkostN <- bus$skadkost
bus$skadkostN[bus$skadkostN== NA] <- 0
```

Another thing, which we have to consider and change in original data are the variable types. We would like to convert variable *skadkost* from factor to numeric and variables *zon* and *bussald* from numeric to factor. To convert factor *skadkost* to numeric (continous), we first convert it to character due to the fact that converting factors directly to numeric data can lead to unwanted outcomes.

```
class(bus$skadkostN)
bus$skadkostN <- as.character(bus$skadkostN)
bus$skadkostN <- as.numeric(bus$skadkostN)
```

There is no problem with converting in opposite way.

```
bus <- within(bus, {
       zon <- factor(zon)
       bussald <- factor(bussald)
})
summary(bus)
```

```
 zon     bussald     kundnr          antavt            dur
 1: 79   0:208   Min.   :  1.0   Min.   :  1.00   Min.   :    1
 2:143   1:208   1st Qu.:179.0   1st Qu.:  2.00   1st Qu.:  365
 3:198   2:219   Median :328.5   Median :  4.00   Median :  725
 4:766   3:242   Mean   :335.2   Mean   : 12.87   Mean   : 2284
 5: 70   4:665   3rd Qu.:506.0   3rd Qu.:  9.00   3rd Qu.: 1876
 6:258           Max.   :670.0   Max.   :392.00   Max.   :66327
 7: 28
    antskad           skadkost     skadkostN
 Min.   :  0.000   .      :926   Min.   : -17318
 1st Qu.:  0.000   0      :221   1st Qu.:      0
 Median :  0.000   -1820  : 20   Median :   3525
 Mean   :  1.953   -1815  :  9   Mean   :  52871
 3rd Qu.:  1.000   -1785  :  4   3rd Qu.:  39897
```

```
 Max.   :402.000   -1810  :  4   Max.   :1330610
                   (Other):358   NA's   :926
```

Claim frequency

For modelling a number of claims divided by duration, i.e., an average number of claims per year, is reasonable to assume a Poisson or a negative binomial regression with an offset term.

Following two plots 1.4 show the conditional histograms separated out by zone and age classes. It can be seen that there is not a visible difference among zones or even among age classes. We can observe very similar shape of all the conditional histograms with the highest value at zero.

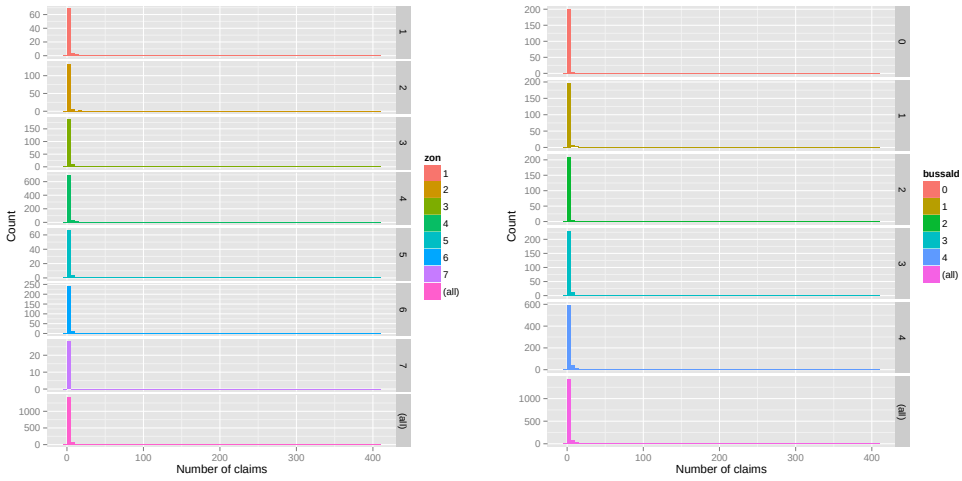

Figure 1.4: Conditional histograms.

We fit the Poisson distribution with a logarithmic link, two rating factors *zon* and *bussald* and *log(duration)* as offset because our link function is logarithm. We store it in an object *model.frequency_p* and get a summary at the same time.

```
summary(model.frequency_p <-glm(antskad~zon+bussald+offset(log(dur)),
data=bus, family=poisson))
```

Table 1.6 shows the estimates of Poisson regression coefficients for each varible, along with their standard errors, Wald z-statistics (z-values) and the associated p-values. We tested null hypothesis $H_0 : \beta_j = 0$ vs. alternative $H_1 : \beta_j \neq 0$ for $j = 1, \ldots, 11$.

The indicator variable *zon2* compares between zone 2 and zone 1. The expected log count for zone 2 decreases by 1.604. The indicator variable *bussald1* is the expected difference in log count (approx. 0.134) between age class one and reference age class 0. We can also test

	Estimate	Std. Error	z-value	p-value
(Intercept)	−4.342	0.050	−86.672	< 0.001
zon2	−1.604	0.072	−22.379	< 0.001
zon3	−2.147	0.080	−26.944	< 0.001
zon4	−1.920	0.044	−43.182	< 0.001
zon5	−1.787	0.127	−14.014	< 0.001
zon6	−1.809	0.070	−25.997	< 0.001
zon7	−2.570	0.270	−9.535	< 0.001
bussald1	−0.134	0.058	−2.289	0.022
bussald2	−1.125	0.075	−15.043	< 0.001
bussald3	−1.396	0.080	−17.525	< 0.001
bussald4	−1.470	0.050	−29.136	< 0.001

Table 1.6: Summary of model.frequncy_p.

the overall effect of the rating factor *bussald* by comparing the deviance of the full model with the deviance of the model excluding bus age.

```
model.frequency_p1 <-update(model.frequency_p,.~.-bussald)
anova(model.frequency_p1,model.frequency_p,test="Chisq")
```

The four degrees of freedom chi-square test indicates that *bussald*, taken together, is a statistically significant predictor of claim frequency. Due to this fact, we will continue working with an initial full model *model.frequency_p*.

The information on deviance is also provided in R summary output. We will use the residual deviance to perform a goodness of fit test to find out whether our model fits our data reasonably.

```
with(model.frequency_p, cbind(res.deviance = deviance, df =
df.residual, p = pchisq(deviance, df.residual, lower.tail = FALSE)))
```

P-value $3.344 \cdot 10^{-248}$ strongly suggests that the data does not fit the model well. It is possible that over-dispersion is present and a negative binomial regression would be more appropriate for modelling claim frequency. We perform a negative binomial regression with logarithmic link using glm.nb function from a MASS package. We fit the model and store it in an object *model.frequency_nb*. We summarize it in Table 1.7.

```
library("MASS")
summary(model.frequency_nb <-glm.nb(antskad~zon+bussald+
offset(log(dur)), data=bus))
```

The interpretation of the estimates of regression coefficients is the same as in the Poisson regression model. From the summary output 1.7, we see that one p-value is higher than 0.05. We will test if the rating factor *bussald* influence our outcome.

	Estimate	Std. Error	z-value	p-value
(Intercept)	−5.517	0.161	−34.341	< 0.001
zon2	−0.964	0.177	−5.454	< 0.001
zon3	−1.384	0.175	−7.904	< 0.001
zon4	−1.319	0.143	−9.216	< 0.001
zon5	−1.072	0.230	−4.670	< 0.001
zon6	−1.158	0.164	−7.077	< 0.001
zon7	−1.827	0.367	−4.975	< 0.001
bussald1	−0.265	0.145	−1.825	0.068
bussald2	−0.483	0.151	−3.212	0.001
bussald3	−0.456	0.145	−3.137	0.002
bussald4	−0.870	0.116	−7.538	< 0.001

Table 1.7: Summary of model.frequncy_nb.

```
model.frequency_nb1 <-update(model.frequency_nb,.~.-bussald)
anova(model.frequency_nb1,model.frequency_nb,test="Chisq")
```

Based on a p-value, we reject the submodel and we will continue working with an initial full model *model.frequency_ nb*.

We would like to compare Poisson model with negative binomial model (Poisson model is nested in the negative binomial model) and check the negative binomials assumption that stays the conditional mean is not equal to the conditional variance. To do so we can use a likelihood ratio test.

```
pchisq(2*(logLik(model.frequency_nb)-logLik(model.frequency_p)),
df=1, lower.tail= FALSE)
```

P-value 0 strongly suggests that the negative binomial model is more appropriate for our data than the Poisson model. To see how well negative binomial model fits the data, we will perform a goodness of fit chi-squared test.

```
with(model.frequency_nb, cbind(res.deviance = deviance, df =
df.residual, p = pchisq(deviance, df.residual, lower.tail = FALSE)))
```

We conclude that the model fits reasonably well because the test is not statistically significant. Therefore, we have decided to choose the negative binomial with logarithmic link function model *model.frequency_ nb* as a final claim frequency model.

Claim severity

When we are performing claim severity analysis we use total claim cost divided by number of claims, i.e., the average cost per claim as a dependent variable.

Conditional histograms separated out by zone and age classes are shown at Figure 1.5. We can see that in row, which coresponds to the island Gotland (zone 7) are some empty cells. This could be caused by poor bus service in island.

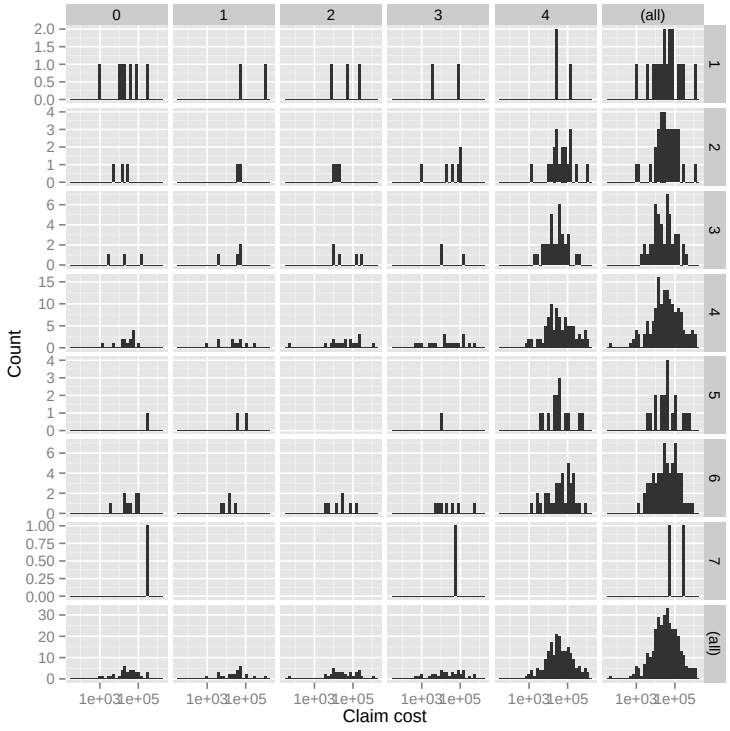

Figure 1.5: Conditional histograms.

Two graphs 1.6 are the violin plots. The most claims costs were observed in the small towns and countryside (zone 4) and only two in Gotland (zone 7). The black points (lines) at the bottom represent zero and negative values of claim cost.

We fit a gamma distribution with a logarithmic link, two rating factors *zon* and *bussald* and number of claims as the weights. We store it in an object *model.severity_g* and get a summary 1.8 at the same time. Because we are using gamma distribution we need to remove the zero values from our data.

```
summary(model.severity_g <- glm(skadkostN ~ zon + bussald,
data =bus[bus$skadkostN > 0, ], family = Gamma("log"),
weights = antskad))
```

From the summary output 1.8, we see that two p-values are higher than 0.05. We will test separately if rating factors *zon* and *bussald* influence our outcome.

```
model.severity_g1 <- update(model.severity_g, . ~ . - zon)
```

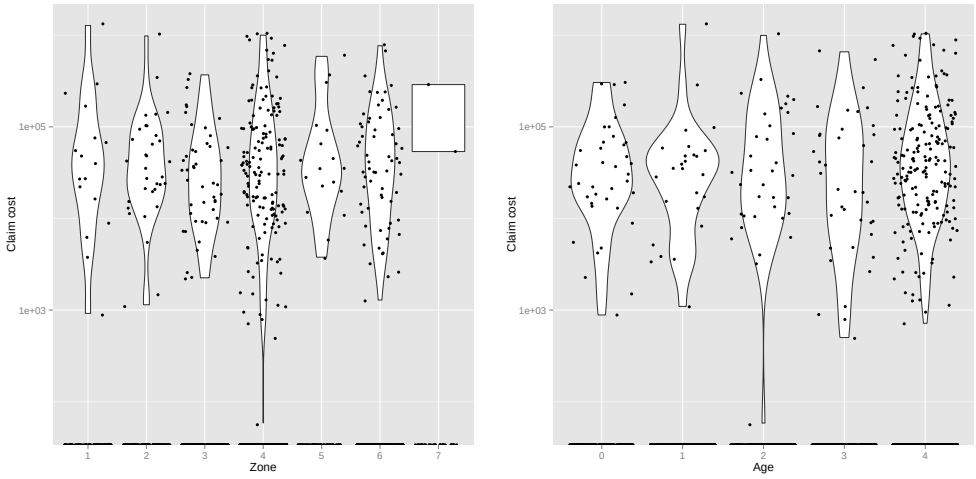

Figure 1.6: Violin plots.

	Estimate	Std. Error	t-value	p-value
(Intercept)	12.539	0.145	86.709	< 0.001
zon2	-1.053	0.302	-3.481	< 0.001
zon3	-2.111	0.307	-6.874	< 0.001
zon4	-1.851	0.220	-8.409	< 0.001
zon5	-1.736	0.474	-3.661	< 0.001
zon6	-1.722	0.280	-6.159	< 0.001
zon7	-0.582	2.190	-0.266	0.791
bussald1	1.419	0.197	7.190	< 0.001
bussald2	0.074	0.295	0.250	0.803
bussald3	0.870	0.352	2.469	0.014
bussald4	1.339	0.231	5.786	< 0.001

Table 1.8: Summary of model.severity_g.

```
anova(model.severity_g1, model.severity_g, test = "Chisq")
model.severity_g2 <- update(model.severity_g, . ~ . - bussald)
anova(model.severity_g2, model.severity_g, test = "Chisq")
```

Based on a p-values, we reject the submodels and we will continue working with an initial full model *model.frequency_g*.

We have also fitted gamma and normal distributions with several different link functions. We used the residual deviances to perform a goodness of fit tests for our models to find out how well these models fit the data. The deviances together with p-values from goodness of fit testing are shown in following Table 1.9.

Model	Null dev.	Res. dev	p-value
Gamma with log link	5372.4	3201.4	0
Gamma with inverse link	5372.4	3203.8	0
Gamma with identity link	5372.4	3334.2	0
Normal with identity link	$5.3816 \cdot 10^{14}$	$1.5622 \cdot 10^{14}$	0
Normal with log link	$5.3816 \cdot 10^{14}$	$1.1218 \cdot 10^{14}$	0
Normal with inverse link	$5.3816 \cdot 10^{14}$	$1.3489 \cdot 10^{14}$	0

Table 1.9: Deviance table.

As a final severity model, we have decided to choose gamma with logarithmic link function model *model.severity_ g* even though the goodness of fit test is statistically significant. The reasons for our decision are the smallest value of residual deviance and better interpretability among the models.

Pure premium: Combining the models

A cell $(4, 4)$ is a base tariff cell as we have chosen the class with the highest duration to be the base tariff class.

```
rel <- data.frame(rating.factor =
c(rep("Zone", nlevels(bus$zon)), rep("Age Class",
nlevels(bus$bussald))),
class = c(levels(bus$zon),levels(bus$bussald)),
stringsAsFactors = FALSE)
```

We calculate the duration and number of claims for each level of each rating factors. We set the contrasts so the baseline for the models is the level with the highest duration. The foreach package is used here to execute the loop both for its side-effect (setting the contrasts) and to accumulate the sums.

```
install.packages("foreach")
library("foreach")

new.cols <-
   foreach (rating.factor = c("zon", "bussald"),
   .combine = rbind) %do%
{
   nclaims <- tapply(bus$antskad, bus[[rating.factor]], sum)
   sums <- tapply(bus$dur, bus[[rating.factor]], sum)
   n.levels <- nlevels(bus[[rating.factor]])
   contrasts(bus[[rating.factor]]) <-
     contr.treatment(n.levels)[rank(-sums, ties.method = "first"), ]
   data.frame(duration = sums, n.claims = nclaims)
```

```
}
rel <- cbind(rel, new.cols)
rm(new.cols)
print(rel)
```

We determine the relativities for claim frequency and severity separately by using GLMs.

```
rels <- coef( model.frequency_nb)
rels <- exp( rels[1] + rels[-1] ) / exp( rels[1] )
rel$rels.frequency <-
c(c(1, rels[1:6])[rank(-rel$dur[1:7], ties.method = "first")],
c(1, rels[7:10])[rank(-rel$dur[8:12], ties.method = "first")])

rels <- coef(model.severity_g)
rels <- exp(rels[1] + rels[-1])/exp(rels[1])
rel$rels.severity <- c(c(1, rels[1:6])[rank(-rel$duration[1:7],
ties.method = "first")],
c(1, rels[7:10])[rank(-rel$duration[8:12],
ties.method = "first")])
```

Then we multiply these results to get relativities for pure premium. All relativities are shown at Table 1.10 together with the duration and number of claims for each level of rating factor.

```
rel$rels.pure.premium <- with(rel, rels.frequency * rels.severity)
print(rel, digits = 2)
```

rating factor	class	duration	number of claims	rel. frequency	rel. severity	rel. pure premium
zone	1	177591	915	0.34	0.18	0.06
zone	2	273386	252	0.27	0.16	0.04
zone	3	383127	194	0.25	0.12	0.03
zone	4	2122797	1292	1.00	1.00	1.00
zone	5	103949	67	0.31	0.18	0.06
zone	6	417412	278	0.38	0.35	0.13
zone	7	43379	14	0.16	0.56	0.09
age class	0	210060	571	0.42	3.81	1.60
age class	1	225348	613	0.63	2.39	1.51
age class	2	258798	262	0.62	1.08	0.66
age class	3	278969	219	0.77	4.13	3.17
age class	4	2548466	1347	1.00	1.00	1.00

Table 1.10: Relativities from the negative binomial GLM for claim frequency and the log gamma GLM for claim severity.

1.2.3 Motorcycle Insurance

Under a headline "case study", we will present larger example using authentic insurance data from the former Swedish insurance company WASA concerns partial casco insurance for motorcycles during years $1994 - 1998$. Partial casco covers theft and some other causes of loss, like fire. Note that this concept of insurance is not used in all countries. Dataset *mccase* contains the following variables in Swedish acronyms:

agarald The owners age, between 0 and 99.

kon The owners gender:

M - male

K - female

zon Geographic zone numered from 1 to 7, in a standard classification of all Swedish parishes:

1 - central and semi-central parts of Sweden's three largest cities

2 - suburbs and middle-sized towns

3 - lesser towns, except those in 5 or7

4 - small towns and countryside, except $5 - 7$

5 - northern towns

6 - northern countryside

7 - Gotland (Sweden's largest island)

mcklass MC class, a classification by the so called EV ratio, defined as (Engine power in kW $\cdot 100$) / (Vehicle weight in kg $+75$), rounded to the nearest lower integer. The 75 kg represent the average driver weight. The EV ratios are divided into seven classes:

1 - EV ratio -5

2 - EV ratio $6 - 8$

3 - EV ratio $9 - 12$

4 - EV ratio $13 - 15$

5 - EV ratio $16 - 19$

6 - EV ratio $20 - 24$

7 - EV ratio $25-$

fordald Vehicle age, between 0 and 99.

bonuskl Bonus class, taking values from 1 to 7. A new driver starts with bonus class 1; for each claim-free year the bonus class is increased by 1. After the first claim the bonus is decreased by 2; the driver can not return to class 7 with less than 6 consecutive claim free years.

duration The number of policy years.

antskad The number of claims.

skadkost The claim cost.

Case study 1.1. Construct a pricing plan for the Motorcycle Insurance data.

We start with loading the data into R and looking at some desriptive statistics.

```
mccase<-read.fwf("http://staff.math.su.se/esbj/GLMbook/mccase.txt",
widths=c(2,1,1,1,2,1,8,4,8),
col.names=c("agarald","kon","zon","mcklass","fordald",
"bonuskl","duration","antskad","skadkost"))
summary(mccase)
```

```
    agarald      kon            zon           mcklass       fordald
 Min.   : 0.00   K: 9853   Min.   :1.000   Min.   :1.0   Min.   : 0.00
 1st Qu.:31.00   M:54695   1st Qu.:2.000   1st Qu.:3.0   1st Qu.: 5.00
 Median :44.00             Median :3.000   Median :4.0   Median :12.00
 Mean   :42.42             Mean   :3.213   Mean   :3.7   Mean   :12.54
 3rd Qu.:52.00             3rd Qu.:4.000   3rd Qu.:5.0   3rd Qu.:16.00
 Max.   :92.00             Max.   :7.000   Max.   :7.0   Max.   :99.00

    bonuskl         duration          antskad          skadkost
 Min.   :1.000   Min.   : 0.0000   Min.   :0.0000   Min.   :     0
 1st Qu.:2.000   1st Qu.: 0.4630   1st Qu.:0.0000   1st Qu.:     0
 Median :4.000   Median : 0.8274   Median :0.0000   Median :     0
 Mean   :4.025   Mean   : 1.0107   Mean   :0.0108   Mean   :   264
 3rd Qu.:7.000   3rd Qu.: 1.0000   3rd Qu.:0.0000   3rd Qu.:     0
 Max.   :7.000   Max.   :31.3397   Max.   :2.0000   Max.   :365347
```

From the summary output above, we see that we need to change a variable type of variables *agarald*, *zon*, *mcklass*, *fordald* and *bonuskl*. We would like to convert them from integers to factors. Note that we will further consider all factor variables as the rating factors.

```
mccase <- within(mccase, {
    zon <- factor(zon)
    mcklass <- factor(mcklass)
    bonuskl <- factor(bonuskl)
})
```

For every motorcycle, the data contains the exact owners and vehicle age, but our choice was to have just three age classes and three vehicle age classes. Our decision was based on the first and on the third quartile values. We divide values of variables *agarald* and *fordald* into these classes. Obviously a large number of alternative groupings are possible. We create two new factor variables and original variables remain unchanged.

```
mccase$agarald2 <- cut(mccase$agarald, breaks = c(0, 30, 50, 99),
labels = as.character(1:3), include.lowest = TRUE, ordered_result
= TRUE)
mccase$fordald2 <- cut(mccase$fordald, breaks = c(0, 5, 15, 99),
labels = as.character(1:3), include.lowest = TRUE, ordered_result
= TRUE)
summary(mccase)
```

```
    agarald      kon       zon       mcklass      fordald      bonuskl
 Min.   : 0.00   K: 9853   1: 8582   1: 7032   Min.   : 0.00   1:14498
 1st Qu.:31.00   M:54695   2:11794   2: 5204   1st Qu.: 5.00   2: 8926
 Median :44.00             3:12722   3:18905   Median :12.00   3: 6726
 Mean   :42.42             4:24816   4:12378   Mean   :12.54   4: 5995
 3rd Qu.:52.00             5: 2377   5:11816   3rd Qu.:16.00   5: 5101
 Max.   :92.00             6: 3884   6: 8407   Max.   :99.00   6: 5349
                           7:  373   7:  806                   7:17953
duration          antskad          skadkost        agarald2  fordald2
Min.   : 0.0000   Min.   :0.0000   Min.   :     0   1:15662   1:16705
1st Qu.: 0.4630   1st Qu.:0.0000   1st Qu.:     0   2:30219   2:27843
Median : 0.8274   Median :0.0000   Median :     0   3:18667   3:20000
Mean   : 1.0107   Mean   :0.0108   Mean   :   264
3rd Qu.: 1.0000   3rd Qu.:0.0000   3rd Qu.:     0
Max.   :31.3397   Max.   :2.0000   Max.   :365347
```

Further, we aggregate the data according to the rating factors so that each line of the dataset would correspond to one particular combination of the rating factors, i.e., one tariff class. For the aggregation we use plyr package, which contains tools for splitting, applying and combining data.

```
install.packages("plyr")
library(plyr)
mccaseA=ddply(mccase, .(kon,zon,mcklass,bonuskl,agarald2,fordald2),
summarize,antskad=sum(antskad),skadkost=sum(skadkost),duration=
sum(duration))
summary(mccaseA)
dim(mccaseA)
```

```
 kon      zon     mcklass bonuskl agarald2 fordald2    antskad
```

```
 K:1505   1:641   1:561   1:645   1:1320   1:1287   Min.   : 0.0000
 M:2414   2:682   2:543   2:565   2:1414   2:1439   1st Qu.: 0.0000
          3:704   3:750   3:521   3:1185   3:1193   Median : 0.0000
          4:779   4:652   4:510                     Mean   : 0.1779
          5:441   5:648   5:494                     3rd Qu.: 0.0000
          6:523   6:549   6:500                     Max.   :10.0000
          7:149   7:216   7:684
skadkost          duration
Min.   :     0   Min.   :   0.000
1st Qu.:     0   1st Qu.:   1.237
Median :     0   Median :   3.984
Mean   :  4348   Mean   :  16.646
3rd Qu.:     0   3rd Qu.:  13.101
Max.   :546139   Max.   :1024.912

[1] 3919    9
```

After these changes have been made, our final dataset *mccaseA* contains 3,919 valid observations.

Claim frequency

The average annual number of claims per policy in each cell is the quantity of our interest in this part. We want to relate this annual claim frequency to the given rating factors to establish or analyze a tariff. We will try to find a well fitting model for the claim frequency in terms of these rating factors. Before we do so we create various conditional histograms separated out by all rating factors individually for a better visualisation of the dependence between claim frequency and the rating factors. They are shown at Figure 1.7.

Using the aggregate data *mccaseA*, a logarithmic link function with Poisson distrubution was fit using all rating factors and duration as the weights. We store the model in an object *model.frequency_pw* and get a summary at the same time.

```
summary(model.frequency_pw <-glm(antskad/duration ~ kon + zon
+ mcklass + agarald2 + fordald2 + bonuskl, data=mccaseA
[mccaseA$duration > 0,], family=poisson, weights=duration))
```

Table 1.11 shows the estimates of Poisson regression coefficients for each variable, along with their standard errors, Wald z-statistics (z-values) and the associated p-values. We tested null hypothesis $H_0 : \beta_j = 0$ vs. alternative $H_1 : \beta_j \neq 0$ for $j = 1, \ldots, 24$.

From the summary output 1.11, it is easy to determine how many claims on average a owner with the arbitrary rating factor levels produces, compared with owner in a base class.

	Estimate	Std. Error	z-value	p-value
(Intercept)	−4.092	0.216	−18.951	< 0.001
konM	0.335	0.135	2.485	0.013
zon2	−0.526	0.108	−4.884	< 0.001
zon3	−1.006	0.118	−8.549	< 0.001
zon4	−1.452	0.104	−13.914	< 0.001
zon5	−1.667	0.342	−4.873	< 0.001
zon6	−1.355	0.248	−5.458	< 0.001
zon7	−1.842	1.003	−1.837	0.066
mcklass2	0.238	0.199	1.195	0.232
mcklass3	−0.378	0.169	−2.233	0.026
mcklass4	−0.261	0.181	−1.444	0.149
mcklass5	0.106	0.172	0.619	0.536
mcklass6	0.565	0.172	3.278	0.001
mcklass7	0.220	0.437	0.504	0.614
agarald2.L	−1.125	0.081	−13.866	< 0.001
agarald2.Q	0.454	0.071	6.425	< 0.001
fordald2.L	−0.936	0.091	−10.286	< 0.001
fordald2.Q	0.001	0.072	0.007	0.995
bonuskl2	−0.028	0.146	−0.190	0.849
bonuskl3	0.013	0.159	0.080	0.936
bonuskl4	0.232	0.153	1.510	0.131
bonuskl5	0.005	0.174	0.030	0.976
bonuskl6	−0.040	0.178	−0.225	0.822
bonuskl7	0.159	0.113	1.406	0.160

Table 1.11: Summary of model.frequncy_pw.

The base class is formed by the young female owners whose own the new cars with the first mc class in the cities and belong to the first bonus class. The corresponding average number of claims for example for old male owners whose own old cars with the sixth mc class in the suburbs and belong to the fourth bonus class equals to $\exp\{-4.092 + 0.335 - 0.526 + 0.565 + 0.454 + 0.001 + 0.232\} = 0.048$, that is, one claim each 21 years on average.

The warning are given in R output because averages are used, which are not integers in most cases, therefore not Poisson distributed. This does not present any problems with estimating the regression coefficients, but it prohibits the glm function from computing the Akaike information criterion (AIC). We can use quasipoisson family to stop glm complaining about nonintegral response.

```
summary(model.frequency_qp <-glm(antskad/duration ~ kon + zon
+ mcklass + agarald2 + fordald2 + bonuskl, data=mccaseA
```

```
[mccaseA$duration > 0,], family=quasipoisson, weights=duration))
```

An eqvivalent way how to estimate this GLM is by using an offset; see Remark 1.4. So, using the standard setup for a Poisson regression with a logarithm link, we have:

$$\log \mathsf{E} Y_i = \mathbf{x}_i^\top \boldsymbol{\beta} + \log w_i.$$

We fit the Poisson distribution with a logarithmic link, all rating factors and *log(duration)* as the offset because our link function is logarithm. We store it in an object *model.frequency_p* and get a summary at the same time.

```
summary(model.frequency_p <- glm(antskad ~ kon + zon + mcklass
+ agarald2 + fordald2 + bonuskl +offset(log(duration)),
family = poisson, data = mccaseA[mccaseA$duration > 0,]))
```

The summary output for model *model.frequency_p* is exactly the same as for the model *model.frequency_pw*. Therefore, we will use summary Table 1.11 for both models. Further, we see that all bonus classes p-values are higher than 0.05. We would like to test the overall effect of rating factor *bonuskl* by comparing the deviance of the full model with the deviance of the model excluding bonus class *model.frequency_p1*.

```
model.frequency_p1 <-update(model.frequency_p,.~.-bonuskl)
anova(model.frequency_p1,model.frequency_p,test="Chisq")
```

The six degree of freedom chi-squared test indicates (p-value equals to 0.484) that bonus class, taken together, is a statistically insignificant predictor of our outcome as we do not reject the submodel *model.frequency_gi1*. For this reason is bonus class meaningful rating factor and we will omit it from full initial model *model.frequency_p*.

The information on deviance is also provided in R summary output. We can use the residual deviance to perform a goodness of fit test for the our model. The residual deviance is the difference between the deviance of the current model and the maximum deviance of the ideal model where the predicted values are identical to the observed. Therefore, if the residual difference is small enough, the goodness of fit test will not be significant, indicating that the model fits the data.

```
with(model.frequency_p1, cbind(res.deviance = deviance, df =
df.residual, p = pchisq(deviance, df.residual, lower.tail = FALSE)))
```

We conclude that the model fits reasonably well because the goodness of fit chi-squared test is not statistically significant (p-value equals to one). Furthermore, we used a likelihood ratio test to compare Poisson model with the negative binomial model. The test result confirmed the suitability of the Poisson with logarithmic link function model *model.frequency_p1* therefore we have decided to choose it as a final claim frequency model.

Claim severity

The claim cost divided by number of claims, i.e., the average cost per claim in each cell is the quantity of our interest in this part. We will try to find a well fitting model for the claim severity in terms of given rating factors. Before we do so we create violin plots separated out by all rating factors individually for a better visualization of the dependence between claim severity and the rating factors. They are shown at Figure 1.8 and note that the black points (lines) at the bottom represent zero values of claim cost.

We fitted gamma and normal distributions with several different link functions using all rating factors and the number of claims as the weights. We used the residual deviances to perform a goodness of fit tests for our models to find out how well these models fit the data. The deviances together with p-values from goodness of fit testing are shown in following Table 1.12:

Model	Null deviance	Res. deviance	p-value
Gamma with log link	1520.6	1030.9	$1.162 \cdot 10^{-51}$
Gamma with inverse link	1520.6	1025.3	$6.015 \cdot 10^{-51}$
Normal with identity link	$5.989 \cdot 10^{12}$	$4.143 \cdot 10^{12}$	0
Normal with log link	$5.989 \cdot 10^{12}$	$3.200 \cdot 10^{12}$	0

Table 1.12: Deviance table.

From the residual deviances, we see that normal regression models do not fit nearly as well as the gamma models. Even thought all the p-values are lower than 0.05, all goodness of fit tests are statistically significant, we have chosen the gamma with inverse link model *model.severity_ gi* as our initial model for claim severity due to the fact it has the smallest residual deviance value among the models.

```
summary(model.severity_gi <- glm(skadkost ~ kon + zon + mcklass
+ agarald2 + fordald2 + bonuskl, data = mccaseA[mccaseA$skadkost > 0, ],
family = Gamma, weights = antskad))
```

A summary Table 1.13 shows the estimates of coefficients for each variable, along with their standard errors, t-statistics (t-values) and the associated p-values. We tested null hypothesis $H_0 : \beta_j = 0$ vs. alternative $H_1 : \beta_j \neq 0$ for $j = 1, \ldots, 24$.

From the summary output 1.13 we see that all bonus classes p-values are higher than 0.05. We would like to test the overall efect of rating factor *bonuskl* by comparing the deviance of the full model with the deviance of the model excluding bonus class *model.frequency_ gi1*.

```
model.severity_gi1 <- update(model.severity_gi, . ~ . - bonuskl)
anova(model.severity_gi1, model.severity_gi, test = "Chisq")
```

The six degree of freedom chi-squared test indicates (p-value equals to 0.308) that bonus class, taken together, is a statistically insignificant predictor of our outcome as we do not

	Estimate	Std. Error	t-value	p-value
(Intercept)	$7.572 \cdot 10^{-05}$	$1.146 \cdot 10^{-05}$	6.608	< 0.001
konM	$-2.203 \cdot 10^{-05}$	$7.150 \cdot 10^{-06}$	-3.081	0.002
zon2	$3.290 \cdot 10^{-09}$	$1.551 \cdot 10^{-06}$	0.002	0.998
zon3	$9.241 \cdot 10^{-06}$	$3.468 \cdot 10^{-06}$	2.665	0.008
zon4	$2.158 \cdot 10^{-06}$	$1.747 \cdot 10^{-06}$	1.236	0.217
zon5	$5.217 \cdot 10^{-05}$	$3.722 \cdot 10^{-05}$	1.402	0.162
zon6	$3.781 \cdot 10^{-05}$	$2.022 \cdot 10^{-05}$	1.870	0.062
zon7	$1.535 \cdot 10^{-03}$	$2.240 \cdot 10^{-03}$	0.686	0.493
mcklass2	$4.867 \cdot 10^{-07}$	$1.179 \cdot 10^{-05}$	0.041	0.967
mcklass3	$-2.351 \cdot 10^{-05}$	$8.251 \cdot 10^{-06}$	-2.849	0.006
mcklass4	$-1.871 \cdot 10^{-05}$	$8.491 \cdot 10^{-06}$	-2.204	0.028
mcklass5	$-1.755 \cdot 10^{-05}$	$8.528 \cdot 10^{-06}$	-2.058	0.040
mcklass6	$-2.339 \cdot 10^{-05}$	$8.277 \cdot 10^{-06}$	-2.825	0.005
mcklass7	$-1.034 \cdot 10^{-05}$	$2.367 \cdot 10^{-05}$	-0.437	0.662
agarald2.L	$5.836 \cdot 10^{-06}$	$2.328 \cdot 10^{-06}$	2.507	0.013
agarald2.Q	$3.680 \cdot 10^{-06}$	$1.552 \cdot 10^{-06}$	2.371	0.018
fordald2.L	$3.705 \cdot 10^{-05}$	$8.152 \cdot 10^{-06}$	4.545	< 0.001
fordald2.Q	$1.159 \cdot 10^{-05}$	$5.006 \cdot 10^{-06}$	2.316	0.021
bonuskl2	$-8.268 \cdot 10^{-07}$	$2.962 \cdot 10^{-06}$	-0.279	0.780
bonuskl3	$2.265 \cdot 10^{-06}$	$4.137 \cdot 10^{-06}$	0.547	0.584
bonuskl4	$1.220 \cdot 10^{-06}$	$3.151 \cdot 10^{-06}$	0.387	0.699
bonuskl5	$2.572 \cdot 10^{-06}$	$4.285 \cdot 10^{-06}$	0.600	0.549
bonuskl6	$4.675 \cdot 10^{-06}$	$5.535 \cdot 10^{-06}$	0.845	0.399
bonuskl7	$-2.619 \cdot 10^{-06}$	$2.125 \cdot 10^{-06}$	-1.233	0.218

Table 1.13: Summary of model.frequncy_gi.

reject the submodel *model.frequency_gi1*. Therefore is for our analysis meaningful rating factor and we will omit it from full initial model *model.frequency_gi*.

Pure premium: Combining the models

A cell $(2, 4, 3, 2, 2)$ is a base tariff cell as we have chosen the class with the highest duration to be the base tariff class.

```
rel <- data.frame(rating.factor =
c(rep("Gender", nlevels(mccaseA$kon)), rep("Zone", nlevels(mccaseA$zon)),
rep("MC class", nlevels(mccaseA$mcklass)),rep("Age", nlevels(
    mccaseA$agarald2)),
rep("Vehicle age", nlevels(mccaseA$fordald2))),
```

```
class = c(levels(mccaseA$kon),levels(mccaseA$zon),levels(mccaseA$mcklass)
    ,
levels(mccaseA$agarald2),levels(mccaseA$fordald2)),
stringsAsFactors = FALSE)
print(rel)
```

We calculate the duration and number of claims for each level of each rating factors. We set the contrasts so the baseline for the models is the level with the highest duration. The foreach package is used here to execute the loop both for its side-effect (setting the contrasts) and to accumulate the sums.

```
install.packages("foreach")
library("foreach")
new.cols <-
foreach(rating.factor=c("kon", "zon","mcklass","agarald2","fordald2"),
   .combine = rbind) %do%
{
   nclaims <- tapply(mccaseA$antskad, mccaseA[[rating.factor]], sum)
   sums <- tapply(mccaseA$duration, mccaseA[[rating.factor]], sum)
   n.levels <- nlevels(mccaseA[[rating.factor]])
   contrasts(mccaseA[[rating.factor]]) <-
     contr.treatment(n.levels)[rank(-sums, ties.method = "first"), ]
   data.frame(duration = sums, n.claims = nclaims)
}
rel <- cbind(rel, new.cols)
rm(new.cols)
print(rel)
```

We determine the relativities for claim frequency and severity separately by using GLMs.

```
rels <- coef( model.frequency_p1)
rels <- exp( rels[1] + rels[-1] ) / exp( rels[1])
rel$rels.frequency <-
c(c(1, rels[1])[rank(-rel$duration[1:2], ties.method = "first")],
  c(1, rels[2:7])[rank(-rel$duration[3:9], ties.method = "first")],
  c(1, rels[8:13])[rank(-rel$duration[10:16], ties.method = "first")],
  c(1, rels[14:15])[rank(-rel$duration[17:19], ties.method = "first")],
  c(1, rels[16:17])[rank(-rel$duration[20:22], ties.method = "first")])

rels <- coef(model.severity_gi1)
rels <- rels[1]/(rels[1]+ rels[-1])
rel$rels.severity <-
c(c(1, rels[1])[rank(-rel$duration[1:2], ties.method = "first")],
  c(1, rels[2:7])[rank(-rel$duration[3:9], ties.method = "first")],
  c(1, rels[8:13])[rank(-rel$duration[10:16], ties.method = "first")],
```

```
  c(1, rels[14:15])[rank(-rel$duration[17:19], ties.method = "first")],
  c(1, rels[16:17])[rank(-rel$duration[20:22], ties.method = "first")])
```

Then we multiply these results to get the relativities for pure premium. All relativities are shown at Table 1.14 together with the duration and number of claims for each level of rating factor.

```
rel$rels.pure.premium <- with(rel, rels.frequency * rels.severity)
print(rel, digits = 2)
```

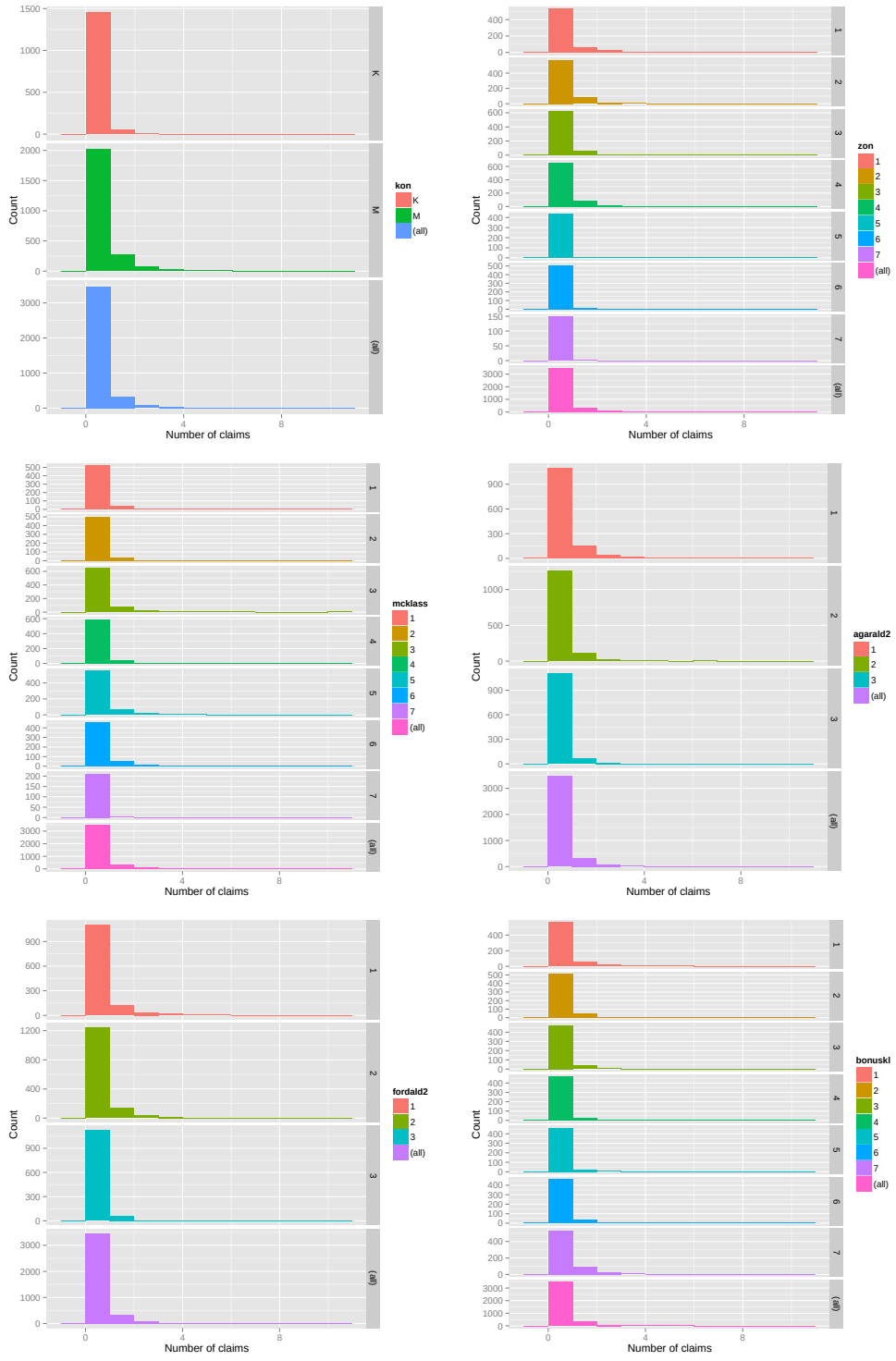

Figure 1.7: Conditional histograms.

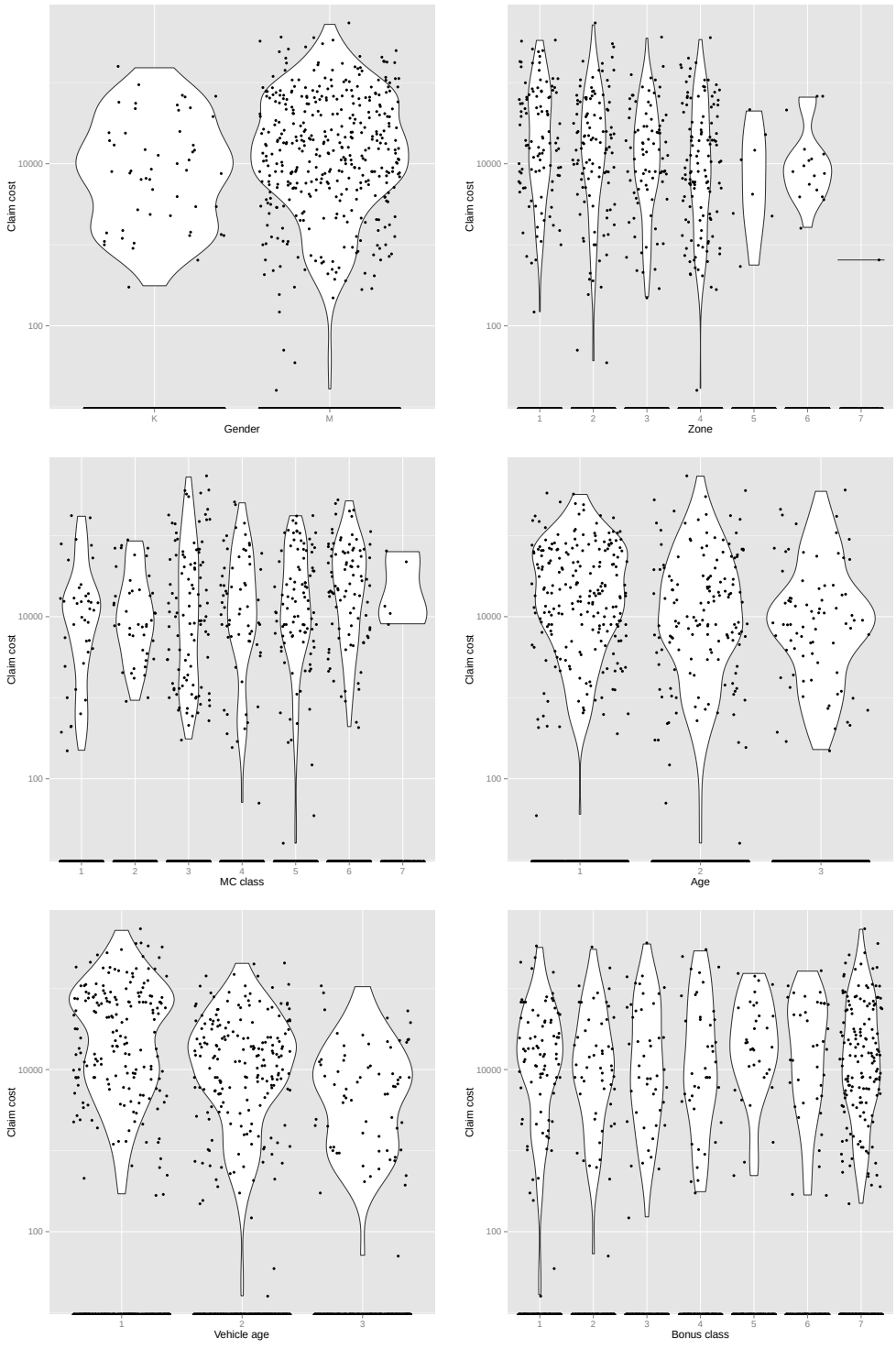

Figure 1.8: Violin plots.

rating factor	class	duration	number of claims	rel. frequency	rel. severity	rel. pure premium
Gender	K	7126	61	1.40	1.42	1.99
Gender	M	58111	636	1.00	1.00	1.00
Zone	1	6205	183	0.23	0.98	0.23
Zone	2	10103	167	0.36	0.89	0.32
Zone	3	11677	123	0.59	1.00	0.59
Zone	4	32628	196	1.00	1.00	1.00
Zone	5	1582	9	0.25	0.67	0.17
Zone	6	2800	18	0.19	0.60	0.11
Zone	7	241	1	0.16	0.05	0.08
MC class	1	5190	46	1.16	1.30	1.50
MC class	2	3990	57	1.86	1.47	2.73
MC class	3	21666	166	1.00	1.00	1.00
MC class	4	11740	98	0.69	1.46	1.02
MC class	5	13440	149	1.27	1.00	1.27
MC class	6	8880	175	0.79	1.33	1.05
MC class	7	331	6	1.31	1.16	1.52
Age	1	11225	353	1.55	0.94	1.47
Age	2	33676	235	1.00	1.00	1.00
Age	3	20336	109	0.33	0.94	0.31
Vehicle age	1	17228	314	1.01	0.87	0.88
Vehicle age	2	27160	304	1.00	1.00	1.00
Vehicle age	3	20849	79	0.39	0.69	0.26

Table 1.14: Relativities from the Poisson GLM for claim frequency and the inverse gamma GLM for claim severity.

Chapter 2

Extreme value theory

2.1 Introduction

The greatest part of the indemnities paid by the insurance companies is represented by a few large claims rising from an insurance portfolio. Therefore, it is of prime interest to calculate their precise estimates. Moreover, frequency and size of the extreme events often creates the base of pricing models for reinsurance agreements as excess-of-loss reinsurance contract which guarantees that the reinsurer reimburses all expenditure associated with a claim as soon as it exceeds a specific threshold. Portfolio managers might be further interested in estimation of high quantiles or the probable maximum loss.

The chapter discusses a universal procedure for description of statistical behaviour of the extreme claims. To this end, we present a basic features of extreme value theory, in particular we pay special attention to threshold models that are intended to identify distribution of exceedances over a high threshold. From a statistical perspective, the threshold can be loosely defined such that the population tail can be well approximated by an extreme value model, obtaining a balance between the bias due to the asymptotic tail approximation and parameter estimation uncertainty due to the sparsity of threshold excess data. We present traditional approaches for threshold selection including mean residual life plot, threshold choice plot, L-moments plot and dispersion index plot. Once the optimal threshold is identified, an asymptotic extreme value model can be fit. Maximum likelihood has emerged as a flexible and powerful modeling tool in such applications, but its performance with small samples has been shown to be poor relative to an alternative fitting procedure based on probability weighted moments. To incorporate an extra information provided by the probability weighted moments model in a likelihood-based analysis, we utilize a penalized maximum likelihood estimator that retains the modeling flexibility and large sample optimality of the maximum likelihood estimator, but improves on its small-sample properties. In this chapter

we apply all three methods on different samples and provide their comparison. As the next step, the fitted model has to be verified using a graphical or statistical assessment. The present chapter considers the medical insurance claims from the SOA Group Medical Insurance Large Claims Database. A model is fitted to the amounts of the 1997, 1998 and 1999 group claims.

The chapter is organized as follows. In Section 2.2 we provide an overview of the classical extreme value theory, mainly we focus on formulation of so-called block maxima and state the theorem which is considered to be the cornerstone of the theory. In Section 2.3 we present threshold models that analyze statistical behaviour of claim exceedances over a high threshold. After a general theoretical background we introduce methods for threshold selection followed by several inference approaches accompanied by their comparison. The rest of the section is devoted to model validation, where we present the graphical as well as the statistical verification. The remainder of the chapter demonstrates practical implementation of the threshold models on SOA Group Medical Insurance Large Claims data (Section 2.4). The database description together with a brief descriptive statistics can be found in Subsection 2.4.1. Subsection 2.4.2 is devoted to data analysis, in other words we demonstrate the approach based on the extreme value theory to model behaviour of large claims exceeding a high threshold. The aim of the chapter is to find the optimal threshold level and estimate parameters of the extremal distribution for particular claim years and prove the adequacy of the fitted models. In Subsection 2.4.3 we suggest two applications of the model that can be broadly applied in insurance strategy. Finally, conclusions are stated in Section 2.5.

2.2 Classical extreme value theory

This chapter is dedicated to provide a mathematical formulation of problems that involve application of extreme value theory. The first part deals with the model formulation and general statements about so called block maxima which can be considered as the cornerstone of the theory. In the next chapter we will focus on the different approach to extremal behavior, known as threshold models.

2.2.1 Asymptotic models

Let $X_1, \ldots, X_n$ be a sequence of independent random variables with common distribution F. In practice, the serie usually represents values of a process measured on a time scale (for instance, daily measurements of rainfall) or values of a random process recorded over a predefined time period (for instance, year observations of insurance claims). The theory focuses on the statistical behaviour of block maxima defined as:

$$M_n = \max\{X_1, \ldots, X_n\}.$$

It can be easily shown that the exact distribution function of M_n has the form of F^n for all values of n. However, it is not possible to model the distribution function according this formula immediately, since F is an unknown function. One could argue that F could be estimated from observed data and subsequently used for the distribution function of the block maxima. Nevertheless, this is not an appropriate approach. Even small discrepancies in the estimation of F can cause significant discrepancies for F^n.

Exercise 2.1. Show that block maxima are mutually independent random variables.

Let $X_1, \ldots, X_{mn}$ be a sequence of independent random variables and define the block maxima as follows: $M_{jn} = \max\left\{X_{(j-1)n+1}, \ldots, X_{jn}\right\}$ for $j = 1, \ldots, m$. Then for $j_1, j_2 \in \{1, \ldots, m\}$ such that $j_1 \neq j_2$ and any $m_1, m_2 > 0$ we have:

$$\begin{aligned} F_{M_{j_1 n}, M_{j_2 n}}(m_1, m_2) &= \mathbb{P}\left[M_{j_1 n} \leqslant m_1, M_{j_2 n} \leqslant m_2\right] \\ &= \mathbb{P}\left[\max\left\{X_{(j_1-1)n+1}, \ldots, X_{j_1 n}\right\}, \max\left\{X_{(j_2-1)n+1}, \ldots, X_{j_2 n}\right\}\right] \\ &= \mathbb{P}\left[\max\left\{X_{(j_1-1)n+1}, \ldots, X_{j_1 n}\right\}\right] \mathbb{P}\left[\max\left\{X_{(j_2-1)n+1}, \ldots, X_{j_2 n}\right\}\right] \\ &= \mathbb{P}\left[M_{j_1 n} \leqslant m_1\right] \mathbb{P}\left[M_{j_2 n} \leqslant m_2\right] = F_{M_{j_1 n}}(m_1) F_{M_{j_2 n}}(m_2) \end{aligned}$$

and thus the block maxima are independent random variables.

The extreme value theory assumes the function F to be unknown and seeks for an appropriate distribution families to model F^n directly. The *Extremal Types Theorem* defines the range of possible limit distributions for normalized block maxima. For convenience we state a modified version of the theorem, the exact formulation and an outline of the proof can be found in Coles (2001).

Theorem 2.1. *If there exist sequence of constants* $\{a_n > 0\}$ *and* $\{b_n\}$ *such that*

$$\mathbb{P}\left[\frac{M_n - b_n}{a_n} \leqslant z\right] \longrightarrow G(z), \qquad n \longrightarrow \infty \tag{2.1}$$

for a non-degenerate distribution function G*, then* G *is a member of the generalized extreme value family*

$$G(z) = \exp\left\{-\left(1 + \xi\left(\frac{z-\mu}{\sigma}\right)\right)_+^{-1/\xi}\right\}, \tag{2.2}$$

where (μ, σ, ξ) *are the location, scale and shape parameters respectively,* $\sigma > 0$ *and* $z_+ = max\,(z, 0)$.

Remark 2.1. The fact that the normalizing constants are unknown in practice is irrelevant.

Statement (2.1) can be equivalently rewritten as:

$$\mathbb{P}\left[M_n \leqslant z\right] \longrightarrow G\left(\frac{z-b}{a}\right) = G^*(z), \qquad n \longrightarrow \infty$$

where $b_n \to b$ and $a_n \to a$ as $n \to \infty$. G^* is just another member of the generalized extreme value family. Thus the estimation of the parameters of the functions G^* and G involves the same procedure.

The theorem implies that the normalized block maxima converge in distribution to a variable having the distribution function G, commonly termed as the *Generalized Extreme Value* (GEV) distribution. Notable feature of the previous statement is that G is the only possible limit regardless of the original distribution function F.

The generalized extreme value family includes three classes of distribution known as the Gumbel, Fréchet and negative Weibull families respectively. Each type can be obtained by a particular choice of the shape parameter ξ. The Fréchet and negative Weibull classes correspond respectively to the case when $\xi > 0$ and $\xi < 0$. The Gumbel class is defined by continuity when $\xi \to 0$. It follows that in practice these three types give quite different representations of extreme value behaviour corresponding to distinct forms of tail behaviour for the distribution function F of the original data. Consider the upper end-point z_+ of the limit distribution G, i.e., z_+ is the smallest value of z such that $G(z) = 1$. Then for the Fréchet and Gumbel distribution z_+ is infinite, whereas in the case of Weibul distribution it is finite.

Remainder of this section is devoted to the formulation of extreme quantiles estimates. Assume a serie of independent identically distributed random variables $X_1, X_2, \ldots$. Further let us split the sequence into blocks of length n, for some large n, and generate a serie of maxima corresponding to each block, to which the generalized extreme value distribution can be fitted. Then by inverting Equation (2.1) we arrive to the following expression for extreme quantiles:

$$q_p = \begin{cases} \mu - \frac{\sigma}{\xi}\left(1 - (-\log(1-p))^{-\xi}\right), & \text{for } \xi \neq 0, \\ \mu - \sigma \log\left(-\log(1-p)\right), & \text{for } \xi = 0, \end{cases} \tag{2.3}$$

where $G(q_p) = 1 - p$. q_p from the previous equation is commonly termed as the *return level* associated with the *return period* $1/p$. The quantile q_p can be interpreted as a value that is exceeded by the maximum in any relative period with probability p. Quantile analysis enables to express stochastic models on the scale of observed values, therefore it provides illustrative interpretation. In particular, if q_p is plotted against $y_p = -\log(1-p)$ on a logarithmic scale (q_p is plotted against $\log y_p$), the plot is linear in the case $\xi = 0$, convex with asymptotic limit for $\xi < 0$ and concave without any finite bound for $\xi > 0$. The graph is called a *return level plot* and indicates whether the fitted distribution resembles rather

Gambel, Fréchet or negative Weibul.

2.2.2 Inference procedure

Behaviour of block maxima can be described by distribution in the spirit of Theorem 2.1. Its application involves blocking the data into sequences of equal length, determining the maximum for each block and fitting the generalized extreme value distribution to such specified values. It has been shown that determining the block size can be the crucial issue. Large blocks generate few maxima, which leads to large variance in parameters estimation and subsequently the accuracy of fitted distribution. On the other hand if the blocks are too small, the set of block maxima includes values that are rather usual than extreme. Then the limit distribution stated in Theorem 2.1 is likely to be poor. Thus the choice of the length for block is the trade-off between bias and variance. Although some statistical methods can be derived, in practice the length corresponds to a reasonable time unit, for instance annual maxima are often applied.

Remark 2.2. Any extreme value analysis suffers from limited amount of data for model estimation. Extremes are scarce, which involves large variation of model estimates. Modeling only block maxima is thus inaccurate if other data on extremes are available. This issue has motivated the search of description of extremal behaviour incorporating extra information. There are basically two well-known general approaches. One is base on exceedances of a high threshold (discussed in Chapter 2.3) and the other one is based on the behaviour of the r largest order statistics within a block, for small values of r. There are numerous published application of the r largest order statistic model, for instance see Coles (2001).

2.3 Threshold models

The threshold models are very effective methods of modeling extreme values. A threshold value, or generally set of threshold values, is used to distinguish ranges of values where the behaviour predicted by the model varies in some important way. Some of the commonly used graphical diagnostics and related statistics are described in Subsection 2.3.2.

The major benefit of threshold models is avoiding the procedure of blocking which can be very inefficient and inaccurate if one block happens to contain more extreme events than another one. The drawback with the threshold models is that once the threshold value has been identified it is treated as fixed, thus the associated uncertainty is ignored in further inference. Moreover, there are often more than one suitable threshold value stemming from various estimation procedures, thus different tail behaviour will be ignored as well when fixing the threshold. As suggested in Scarrott and MacDonald (2012) an informal approach to overcoming these problems is to evaluate the sensitivity of the inferences (e.g. parameters

or quantiles) to different threshold choices.

Direct comparison of the goodness of the fit for different thresholds is complicated due to the varying sample sizes. Recently, various extreme value mixture models have been developed to overcome this problem. These mixture models typically approximate the entire distribution function, thus the sample size for each threshold is invariant. For more information about these models see Scarrott and MacDonald (2012).

2.3.1 Asymptotic models

Let $X_1, X_2, \ldots$ be a sequence of independent and identically distributed random variables with common distribution function F. Moreover let u be some high threshold value. Then we regard as extreme events those random variables X_i that exceed the threshold u. The stochastic behaviour of extreme event X_i can be described by the following formula:

$$\mathbb{P}\left[X_i > y \mid X_i > u\right] = \frac{1 - F(y)}{1 - F(u)}, \qquad y > u.$$

Apparently if the distribution function F was known, the distribution of threshold exceedances would satisfy the previous formula. In practice this is not the case and some approximation needs to be used.

Exercise 2.2. Show that excesses over a high threshold are mutually independent random variables.

Let $X_1, \ldots, X_{mn}$ be a sequence of independent random variables and define the excesses over threshold u as follows: $Z_j = X_i - u | X_i > u$ for $j = 1, \ldots, k$. Then for $j_1, j_2 \in \{1, \ldots, k\}$ such that $j_1 \neq j_2$ and any $z_1, z_2 > 0$ we have:

$$\begin{aligned} F_{Z_{j_1}, Z_{j_2}}(z_1, z_2) &= \mathbb{P}\left[Z_{j_1} \leqslant z_1, Z_{j_2} \leqslant z_2\right] \\ &= \mathbb{P}\left[X_{i_1} - u \leqslant z_1, X_{i_2} - u \leqslant z_2 | X_{i_1} > u, X_{i_2} > u\right] \\ &= \mathbb{P}\left[X_{i_1} \leqslant z_1 + u | X_{i_1} > u\right] \mathbb{P}\left[X_{i_2} \leqslant z_2 + u | X_{i_2} > u\right] \\ &= \mathbb{P}\left[Z_{j_1} \leqslant z_1\right] \mathbb{P}\left[Z_{j_2} \leqslant z_2\right] = F_{Z_{j_1}}(z_1) F_{Z_{j_2}}(z_2) \end{aligned}$$

and thus the excesses over a high threshold are independent random variables.

The extreme value theory assumes the function F to be unknown and seeks for an appropriate distribution families to model F^n directly. The *Extremal Types Theorem* defines the range of possible limit distributions for normalized block maxima. For convenience we state a modified version of the theorem, the exact formulation and an outline of the proof can be found in Coles (2001).

The following statement extends Theorem 2.1 and provides us the sought approach for description of the threshold behaviour.

Theorem 2.2. *Let $X_1, X_2, \ldots$ be a sequence of independent random variables with common distribution function F, and let*

$$M_n = max\{X_1, \ldots, X_n\}.$$

Denote an arbitrary term in the X_i sequence by X, and suppose that F satisfies Theorem 2.1, so that for large n,

$$\mathbb{P}[M_n \leqslant z] \longrightarrow G(z), \qquad n \longrightarrow \infty$$

for a non-degenerate distribution function G given as:

$$G(z) = \exp\left\{-\left(1 + \xi\left(\frac{z-\mu}{\sigma}\right)\right)_+^{-1/\xi}\right\},$$

for some $\mu, \sigma > 0$ and ξ. Then, for large enough u, the distribution function of X, conditional on $X > u$, can be approximated as:

$$\mathbb{P}[X \leqslant y \mid X > u] \longrightarrow H(y), \qquad u \longrightarrow \infty, \tag{2.4}$$

where

$$H(y) = 1 - \left(1 + \xi\,\frac{y-u}{\sigma_u}\right)_+^{-1/\xi}, \qquad y > u, \tag{2.5}$$

where $z_+ = max\,(z, 0)$ and we adopt the conversion of using $\sigma_u = \sigma + \xi(u - \mu)$ to denote the scale parameter corresponding to excess of the threshold u.

Remark 2.3. The previous theorem can be rephrased in the spirit of behaviour of $X - u$, i.e., if the assumptions of the statement are satisfied, then, for large enough u, the distribution function of $X - u$, conditional on $X > u$, can be approximated as:

$$\mathbb{P}[X - u \leqslant y \mid X > u] \longrightarrow H^{(u)}(y), \qquad u \longrightarrow \infty, \tag{2.6}$$

where

$$H^{(u)}(y) = 1 - \left(1 + \xi\,\frac{y}{\sigma_u}\right)_+^{-1/\xi}, \qquad y > 0, \tag{2.7}$$

where again $z_+ = \max(z, 0)$ and $\sigma_u = \sigma + \xi(u - \mu)$.

The family of distributions given by Equation (2.5) is known as *Generalized Pareto Distribution* (GPD). The previous theorem put in relation the statistical behaviour of block maxima and threshold excesses, i.e., if block maxima can be approximated by distribution

function G, then threshold excesses have a corresponding approximate distribution within the generalized Pareto family. Notice that the parameters of the generalized Pareto family are uniquely determined by those associated with corresponding generalized extreme value distribution. Mainly, the shape parameter ξ is invariant to the threshold selection and block size n.

The shape parameter ξ is dominant in determining the qualitative behaviour of the generalized Pareto distribution. Similarly as it is for the generalized extreme value family, the generalized Pareto family includes three classes of distribution linked to a particular choice of parameter ξ: exponential, Pareto and beta. The Pareto and beta classes correspond respectively to the case when $\xi > 0$ and $\xi < 0$. The exponential distribution is defined by continuity when $\xi \to 0$.

Similarly as demonstrated in Section 2.2, the return level associated with the return period $1/p$ can be derived by inverting Equation (2.7). Thus, the expression for the extreme quantiles is given as:

$$q_p = \begin{cases} -\frac{\sigma_u}{\xi}\left(1 - p^{-\xi}\right), & \text{for } \xi \neq 0, \\ -\sigma_u \log p, & \text{for } \xi = 0, \end{cases} \tag{2.8}$$

where $H^{(u)}(q_p) = 1 - p$. We recall that the quantile q_p can be interpreted as a value that is exceeded by the maximum in any relative period with probability p.

2.3.2 Graphical threshold selection

The classical fixed threshold modeling approach uses graphical diagnostics, essentially assessing aspects of the model fit, to make an a priori threshold choice. An advantage of this approach is that it requires a graphical inspection of the data, comprehending their features and assessing the model fit. A key drawback is that they can require substantial expertise and can be rather subjective, as can be seen in the Subsection 2.4.2.

Consider a sequence of raw data $x_1, \ldots, x_n$ as realizations of independent and identically distributed random variables $X_1, \ldots, X_n$. Further let us define a high threshold u, for which the exceedances $\{x_i;\ x_i > u\}$ are identified as extreme events. In further text we refer to these excesses as y_j, i.e., $y_j = x_i$ for those indices $i \in \{1, \ldots, n\}$ for which $x_i > u$. In the spirit of Theorem 2.2, sequence $y_1, \ldots, y_k$ can be regarded as independent realizations of a random variable Y whose distribution function can be approximated by a member of the generalized Pareto family defined by Formula (2.5).

Unlike the method of block maxima which takes the highest event within a block as extremal and does not consider whether the event is extremal comparing the rest of observations, the threshold method identifies an event as extremal if it exceeds a high threshold. Moreover this threshold is estimated on the basis an entire data set. Nevertheless the issue of threshold selection and choice of bloc size is analogous, thus the trade-off between bias

and variance arises again. If the threshold is too high only few observations are generated and the estimated limit distribution is more likely to be poor, i.e., high variance of estimated parameters is involved. On the other hand if the threshold is too low the sequence of exceedances contains values that are rather usual than extreme and as the consequence the estimated limit distribution is likely to violate the asymptotic framework of the model. In practical application as low a threshold as possible is accepted on condition that such an adopted threshold value provides reasonable approximation for the limit distribution.

In the following subsection we will discuss several concrete procedures for the threshold selection. A brief overview of the methods stated above can be found in Ribatet (2011).

Mean residual life plot

The first method introduced by Davison and Smith (2012), the *mean residual life plot*, is based on the theoretical mean of the generalized Pareto family. In more detail, let Y be a random variable with distribution function $GPD\ (u_0, \sigma_{u_0}, \xi)$ and let u_1 be a threshold such that $u_1 > u_0$. Then random variable $Y - u_1 | Y > u_1$ is also generalized Pareto distribution with updated parameters $u_1, \sigma_{u_1} = \sigma_{u_0} + \xi u_1$ and $\xi_1 = \xi$. Assume an arbitrary $u_1 > u_0$ and $y > 0$. The claim then stems from the following inference:

$$\begin{aligned}
\mathbb{P}\left(Y - u_1 > y | Y > u_1\right) &= \frac{1 - H^{(u_0)}(y + u_1)}{1 - H^{(u_0)}(u_1)} \\
&= \frac{\left(1 + \xi\ \frac{y+u_1}{\sigma_{u_0}}\right)_+^{-1/\xi}}{\left(1 + \xi\ \frac{u_1}{\sigma_{u_0}}\right)_+^{-1/\xi}} \\
&= \left(1 + \xi\ \frac{y}{\sigma_{u_0} + \xi u_1}\right)_+^{-1/\xi}.
\end{aligned}$$

If a random variable Y has a generalized Pareto distribution with parameters μ, σ and ξ, then

$$\mathbb{E}\left[Y\right] = \mu + \frac{\sigma}{1 - \xi}, \qquad \text{for } \xi < 1.$$

When $\xi \geqslant 1$, the mean is infinite. In practice, if Y represents excess over a threshold u_0, and if the approximation by a generalized Pareto distribution is sufficiently accurate, we have:

$$\mathbb{E}\left[Y - u_0 | Y > u_0\right] = \frac{\sigma_{u_0}}{1 - \xi}.$$

Now, if the generalized Pareto distribution is valid as a model for excesses of the threshold u_0, it has to be valid for all thresholds $u_1 > u_0$, provided a change of scale parameter as

derived above. Thus, we have:

$$\mathbb{E}\left[Y - u_1 | Y > u_1\right] = \frac{\sigma_{u_1}}{1-\xi} = \frac{\sigma_{u_0} + \xi u_1}{1-\xi}. \tag{2.9}$$

We observe that for $u > u_0$ expression $\mathbb{E}\left[Y - u | Y > u_0\right]$ is a linear function of variable u. Furthermore, $\mathbb{E}\left[Y - u | Y > u\right]$ is obviously the mean of the excesses of the threshold u, which can easily be estimated using the empirical mean. By virtue of Equation (2.9), these estimated are expected to change linearly with rising values of threshold u, for which the generalized Pareto model is valid. The mean residual life plot consists of points

$$\left\{ \left(u, \frac{1}{k} \sum_{i=1}^{k} (y_i - u) \right) : u \leqslant x_{\max} \right\}, \tag{2.10}$$

where k denotes the number of observations that exceed threshold u, and $x_{\max}$ is the maximum of all observations x_i, $i = 1 \ldots, n$. Confidence intervals can be added to the plot since the empirical mean can be approximated by normal distribution (in the spirit of the Central Limit Theorem). However, this approximation is rather poor for high values of thresholds as the are less excesses. Moreover, by construction, the plot always converges to the point $(x_{\max}, 0)$.

Threshold choice plot

The *threshold choice plot* provides a complementary technique to the mean residual life plot for identification of a proper threshold. The method is based on fitting the generalized Pareto distribution at a range of thresholds, and to observe stability of parameter estimates. The argument is as follows. Let Y be a random variable with distribution function $GPD\ (u_0, \sigma_{u_0}, \xi)$ and let u_1 be a higher threshold. Then random variable $Y|Y > u_1$ has also generalized Pareto distribution with updated parameters $u_1, \sigma_{u_1} = \sigma_{u_0} + \xi(u_1 - u_0)$ and $\xi_1 = \xi$. Assume an arbitrary $u_1 > u_0$ and $y > u_1$. The derivation proceeds as follows:

$$\begin{aligned}
\mathbb{P}\left(Y > y | Y > u_1\right) &= \frac{1 - H(y)}{1 - H(u_1)} \\
&= \frac{\left(1 + \xi \, \frac{y - u_0}{\sigma_{u_0}}\right)_+^{-1/\xi}}{\left(1 + \xi \, \frac{u_1 - u_0}{\sigma_{u_0}}\right)_+^{-1/\xi}} \\
&= \left(1 + \xi \, \frac{y - u_1}{\sigma_{u_0} + \xi(u_1 - u_0)}\right)_+^{-1/\xi}.
\end{aligned}$$

Hence the scale parameter depends on value of a threshold u (unless $\xi = 0$). This difficulty can be remedied by defining the scale parameter as

$$\sigma^* = \sigma_u - \xi u = \sigma_{u_0} - \xi u_0. \tag{2.11}$$

With this new parametrization, σ^* is independent of u as it results from the last equality. Thus, estimates of σ^* and ξ are expected to be consistent for all $u > u_0$ if u_0 is suitable threshold for the asymptotic approximation. The threshold choice plots represents the points

$$\{(u, \sigma^*) : u \leqslant x_{\max}\} \qquad \text{and} \qquad \{(u, \xi) : u \leqslant x_{\max}\}, \tag{2.12}$$

where $x_{\max}$ is the maximum of the observations x_i, $i = 1 \ldots, n$.

L-moments plot

In statistics, a probability density or an observed data set are often summarized by its moments or cumulants. It is also common, when fitting a parametric distribution, to estimate the parameters by equating the sample moments with those of the fitted distribution. However, the moments-based method is not always satisfactory. Particularly, when the sample is too small, the numerical values of sample moments can be markedly different from those of the probability distribution from which the sample was drawn. Hosking (1990) described in detailed the alternative approach based on quantities called *L-moments*. These are analogous to the conventional moments but can be estimated by linear combinations of order statistics (thus "L" in "L-moments" emphasized that the quantities are linear functions). They have the advantage of being more robust to the presence of outliers in the data and experience also shows that they are more accurate in small samples than even the maximum likelihood estimates. Moreover, they are able to characterize wider range of distribution functions.

In order to explain how the L-moments-based method is utilized, when selecting the threshold value, it is worthwhile to provide their exact definition.

Definition 2.1. Let X be a real-valued random variable with cumulative distribution function F, and let $X_{1:n} \leqslant X_{2:n} \leqslant \ldots \leqslant X_{n:n}$ be the order statistics of a random sample of size n drawn from the distribution of X. *L-moments* of X are quantities defined as:

$$\lambda_r \equiv r^{-1} \sum_{k=0}^{r-1} (-1)^k \binom{r-1}{k} \mathbb{E}X_{r-k:r}, \qquad r = 1, 2, \ldots, n.$$

Remark 2.4. L-moment λ_2 is a measure of the scale or dispersion of the random variable X (see Hosking (1990)). Higher L-moments λ_r,$r \geqslant 3$ are for convenience often standardized, so that they are independent of the units of measurement of X. Let us, therefore, introduce

the *L-moment ratios* of X defined as quantities:

$$\tau_r = \frac{\lambda_r}{\lambda_2}, \qquad r = 3, 4, \ldots, n.$$

The L-moments $\lambda_1, \ldots, \lambda_r$ together with the L-moment ratios are useful tools for describing distribution. The L-moments can be regarded as an analogy to (conventional) central moments and the L-moment ratios are analogous to moment ratios. In particular, $\lambda_1, \lambda_2, \tau_3$ and τ_4 can be regarded as measures of location, scale, skewness and kurtosis respectively.

As derived in Hosking (1990), the L-kurtosis and L-skewness for generalized Pareto distribution are given by formulas:

$$\tau_3 = \frac{1+\xi}{3-\xi},$$
$$\tau_4 = \frac{(1+\xi)(2+\xi)}{(3-\xi)(4-\xi)},$$

thus parameter τ_4 can be expressed in terms of τ_3 as:

$$\tau_4 = \tau_3 \ \frac{1+5\tau_3}{5+\tau_3}. \tag{2.13}$$

The previous equation creates the headstone for the *L-moment plot* which is represented by points defined by:

$$\{(\hat{\tau}_{3,u}, \hat{\tau}_{4,u}) : u \leqslant x_{\max}\}, \tag{2.14}$$

where $\hat{\tau}_{3,u}$ and $\hat{\tau}_{4,u}$ are estimations of the L-kurtosis and L-skewness based on excess over threshold u and $x_{\max}$ is the maximum of the observations x_i, $i = 1 \ldots, n$. In practical applications, the theoretical curve defined by Equation (2.13) is traced as a guideline and the choice of the accurate threshold value is assessed according to position of the estimated points with respect to this line.

Dispersion index plot

The last graphical technique for threshold selection presented in this chapter, the *dispersion index plot*, is particularly useful when analysing time series. According to the Extreme value theory, the occurrence of excesses over a high threshold in a fixed period, generally this is a year, must be distributed as Poisson process. As for a random variable having Poisson distribution, the ratio of the variance and the mean is equal to 1. Cunnane (1979) introduced a dispersion index statistics defined as:

$$DI = \frac{s^2}{\lambda}, \tag{2.15}$$

where s^2 is the intensity of the a Poisson process and λ the mean number of events in a block, both corresponding to a high threshold value. Equation (2.15) creates the basis for the dispersion index plot which can be represented by points defined as:

$$\{(u, DI) : u \leqslant x_{\max}\}, \tag{2.16}$$

where $x_{\max}$ is the maximum of the observations x_i, $i = 1 \ldots, n$. The plot enables to test if the ratio DI differs from 1 for various levels of threshold. In practice, confidence levels for DI are appended to the graph (DI can be considered to have χ^2 distribution with $M - 1$ degree of freedom, M being the total number of fixed periods).

2.3.3 Inference procedure

By almost universal consent, a starting point for modeling the extreme values of a process is based on distributional models derived from asymptotic theory. Of less common agreement is the method of inference by which such asymptotic behaviour is approximated. In this section, we aim to get the reader acquainted with the most commonly used ones: the maximum likelihood, the probability weighted moments and penalized maximum likelihood. In order to learn about remaining approaches we refer to the following papers: Pickands (1975) introduces the whole problematics of fitting the generalized Pareto distribution, Hosking and Wallis (1987) provides an overview of the *moments based method*, Juaréz and Schucany (2004) broadly discuss so called *minimum density power divergence estimator*, Peng and Welsh (2001) deal with the *median* approach, and the *maximum goodness-of-fit estimator* is summarized by Luceno (2006). Recently, Zvang (2007) discussed the *likelihood moment* method.

As a general framework for extreme value modeling, the maximum likelihood based estimators have many advantages. As Coles and Dixon (1999) pointed out, likelihood functions can be constructed for complex modeling situations enabling, for example, non-stationarity, covariate effects and regression modeling. This combination of asymptotic optimality, ready-solved inference properties and modeling flexibility represents a comprehensive package for alternative methods to compete against. Moreover, approximate inference, such as confidence intervals, are straightforward. An argument that is addressed against the maximum likelihood method is its small sample properties. Hosking et al. (1985) showed that dealing with small sample, probability weighted moments estimator performs better in terms of bias and mean square error. Despite many advantages of maximum likelihood, this fact remains a source criticism, since it is common in practice to make inference with very few data. We aim to utilize both of the methods in the computational part and provide their comparison. In light of Coles and Dixon (1999) who suggested a modification of the maximum likelihood estimator using a penalty function for fitting generalized extreme value distribution, we demonstrate the same approach for inference for the generalized Pareto distribution.

As they showed, the penalized maximum likelihood estimator retains all of the advantages of the maximum likelihood and additionally improves small sample properties which are comparable with those of the probability weighted moments estimator.

Maximum likelihood

Let us recall that the values $y_1, \ldots, y_k$ are the k extreme events excessing threshold u, i.e., $y_j = x_i$ for those indices $i \in \{1, \ldots, n\}$ for which $x_i > u$. For convenience, we denote the excesses of threshold u as z_j, i.e., $z_j = y_j - u$ for $j = 1, \ldots, k$. Considering distribution function of the form given by Formula (2.7), the likelihood function can be then derived from the following relation (for more detailed description see a standard asymptotic likelihood theory summarized by Cox and Hinkley (1974), for example):

$$L(\sigma_u, \xi) = \prod_{j=1}^{k} \frac{\mathrm{d}H^{(u)}(z_j; \sigma_u, \xi)}{\mathrm{d}z_j} \tag{2.17}$$

Thus for $\xi \neq 0$ the log-likelihood function can be expressed as:

$$l(\sigma_u, \xi) = -k \log \sigma_u - \left(1 + \frac{1}{\xi}\right) \sum_{j=1}^{k} \log\left(1 + \xi \frac{z_j}{\sigma_u}\right), \tag{2.18}$$

provided $(1 + \xi z_j/\sigma_u) > 0$ for $i = j, \ldots, k$, otherwise, $l(\sigma_u, \xi) = -\infty$. In the case $\xi = 0$, the log-likelihood has the following form:

$$l(\sigma_u) = -k \log \sigma_u - \frac{1}{\sigma_u} \sum_{j=1}^{k} z_j. \tag{2.19}$$

The log-likelihood function defined above can be made arbitrarily large by taking $\xi < -1$ and ratio σ_u/ξ close enough to $\max\{z_j; j = 1, \ldots, k\}$. Thus the maximum likelihood estimators are taken to be the values ξ and σ_u, which yield a local maximum of the function. The analytical maximization of the log-likelihood is not possible, thus to find the local maximum requires using of numerical methods. In the practical application, we adopted a procedure based on a quasi-Newton method (also known as a variable metric algorithm). Quasi-Newton methods are based on Newton's method to find the stationary point of a function, where the gradient is 0. Newton's method assumes that the function can be locally approximated as quadratic in the region around the optimum, and uses the first and second derivatives to find the stationary point. In higher dimensions, Newton's method uses the gradient and the Hessian matrix of second derivatives of the function to be minimized. In quasi-Newton methods the Hessian matrix does not need to be computed. The Hessian is updated by analyzing successive gradient vectors instead.

Remark 2.5. A potential complication with the use of maximum likelihood methods con-

cerns the regularity conditions that are required for the usual asymptotic properties. Such conditions are not satisfied because the end-points of the generalized Pareto distribution are functions of the parameter values: $-\sigma_u/\xi$ is an upper end-point of the distribution when $\xi > 0$, or a lower end-point when $\xi < 0$, respectively. This issue was studied by Smith (1985) who proved that maximum likelihood estimators are regular (in the sense of having the usual asymptotic properties) whenever $\xi > -0.5$, for $-1 < \xi < -0.5$ the estimators are generally obtainable, but do not have the standard asymptotic properties, and the estimators are unlikely to be obtainable whenever $\xi < 1$.

Probability weighted moments

The probability weighted moments of a continuous random variable X with distribution function F are the quantities

$$M_{p,r,s} = \mathbb{E}\left[X^p (F(X))^r (1 - F(X))^s\right],$$

where p, r and s are real parameters. As suggested in Hosking and Wallis (1987), for the generalized Pareto distribution it is convenient to set the parameters as follows: $p = 1$ and $r = 0$. Then the explicit formula for the probability weighted moments exists and has the form:

$$\alpha_s \equiv M_{1,0,s} = \frac{\sigma_u}{(s+1)(s+1-\xi)},$$

which exists provided that $\xi < 1$. Consequently, the previous equation for $s = 0, 1$ enables us to derive the analytical expressions for parameters σ_u and ξ, those are then given by:

$$\sigma_u = \frac{2\alpha_0\alpha_1}{\alpha_0 - 2\alpha_1}, \tag{2.20}$$

$$\xi = 2 - \frac{\alpha_0}{\alpha_0 - 2\alpha_1}. \tag{2.21}$$

The probability weighted moments estimator $\hat{\sigma}_u$ and $\hat{\xi}$ are obtained by replacing α_0 and α_1 in Equation (2.21) by estimators based on the ordered sample $z_{1:k}, \ldots, z_{k:k}$. The statistic

$$a_s = \frac{1}{k} \sum_{j=1}^{k} \frac{(k-j)(k-j-1)\cdots(k-j-s+1)}{(k-1)(k-1)\cdots(k-s)}\, z_{j:k} \tag{2.22}$$

is an unbiased estimator of α_s. Probability weighted moment α_s can be estimated as:

$$\tilde{a}_s = \frac{1}{k} \sum_{j=1}^{k} (1 - p_{j:k})^r z_{j:k}, \tag{2.23}$$

where $p_{j:k}$ is a so called plotting position, i.e., it is an empirical estimate of the considered distribution function based on the sample $z_1, \ldots, z_k$. Reasonable choice for the plotting position is $p_{j:k} = (j+\gamma)/(n+\delta)$, where γ and δ are suitable constants. For detailed discussion on deriving of the unbiased and consistent estimators we refer to Landwehr et al. (1979). Whichever variant is used, the estimators of α_r, σ_u, and ξ are asymptotically equivalent. In the computational part, we aim to use the biased estimator given by Formula (2.23) with parameters $\gamma = -0.35$ and $\delta = 0$. These values were recommended by Landwehr et al. (1979) for the Wakeby distribution, of which the generalized Pareto distribution is a special case.

Penalized maximum likelihood

Penalized maximum likelihood is a simple method how to incorporate into an inference information that is supplementary to that provided by data. Many of improvements have been suggested so far, in order to balance the appropriate likelihood function. The most commonly applied one is non-parametric smoothing, that penalizes roughness of the likelihood function. The presented modification is taken from Coles and Dixon (1999); they use a penalty function to provide the likelihood function with the information that the value of ξ is smaller than 1, and that values close to 1 are less likely than smaller values. The penalty function has the explicit form:

$$P(\xi) = \begin{cases} 1 & \text{if } \xi \leqslant 0 \\ \exp\left\{-\lambda\left(\frac{1}{1-\xi} - 1\right)^{\alpha}\right\} & \text{if } 0 < \xi < 1 \\ 0 & \text{if } \xi \geqslant 1 \end{cases} \tag{2.24}$$

for a range of non-negative values of parameters α and λ. With declining value of parameter α the penalty function raises for values of ξ which are large, but less than 1, while the penalty function descends for values of ξ close to 0. The shape of the penalty function corresponding to different values of λ is straightforward; declining values of λ lead to more severe relative penalty for ξ close to 1. The penalty function for various values of α and λ is shown in Figure 2.1. After numerous experimentation, Coles and Dixon (1999) found that the combination $\alpha = \lambda = 1$ leads to a reasonable performance across a range of values for ξ and sample sizes. Thus our results are reported with respect to this choice.

The corresponding penalized likelihood function is then:

$$L_p(\sigma_u, \xi) = L(\sigma_u, \xi) \cdot P(\xi), \tag{2.25}$$

where L is the likelihood function defined by Formula (2.17). Apparently, the values of σ_u and ξ which maximize the penalized likelihood function given by Formula (2.25) are the

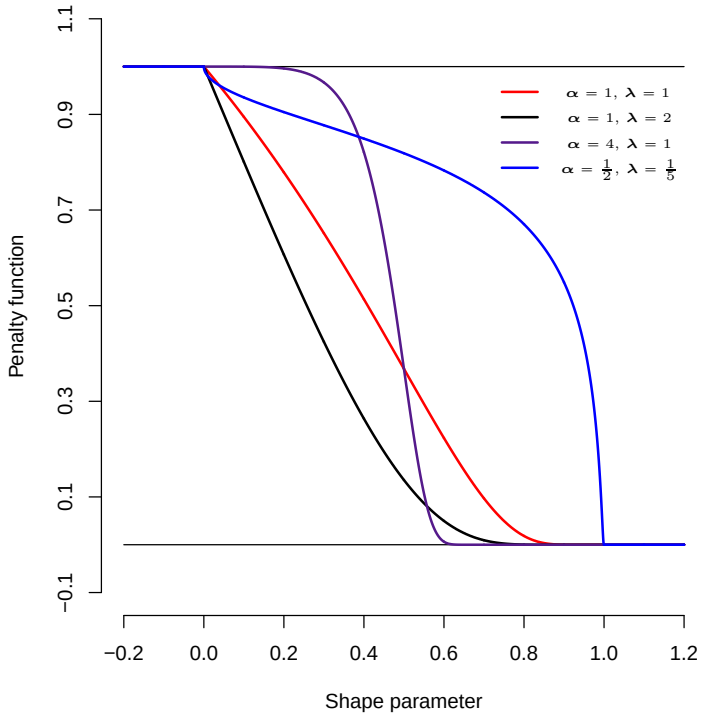

Figure 2.1: Penalty function for various values of α and λ plotted against shape parameter ξ.

maximum penalized likelihood estimator.

Remark 2.6. The performance of different estimators is often judged in terms of the accuracy of estimation of extreme quantiles, since these quantities are the principal requirement of an extreme value analysis. In practice, estimates of quantiles g_p are obtained by substituting estimates of σ_u and ξ into Formula (2.8), i.e.,

$$\hat{q}_p = \begin{cases} -\frac{\hat{\sigma}_u}{\hat{\xi}} \left(1 - p^{-\hat{\xi}}\right), & \text{for } \hat{\xi} \neq 0, \\ -\hat{\sigma}_u \log p, & \text{for } \hat{\xi} = 0. \end{cases} \tag{2.26}$$

Comparison of methods

Coles and Dixon (1999) carried out multiple simulations in order to assess the accuracy of the maximum likelihood method and the method of probability weighted moments. They simulated thousands of samples of different sizes from the generalized extreme value distribution with various shape parameter keeping the other parameter (μ and σ) fixed. Although

in their computations they used the method of block maxima for fitting the generalized extreme value distribution, Hosking and Wallis (1987) showed that the same conclusions can made when applying the threshold methods.

The authors investigated the frequency of convergence failure of the Newton-Raphson iteration algorithm used for maximization of the likelihood function. Hosking and Wallis (1987) concluded that the vast majority of failures of the algorithms were caused by the non-existence of a local maximum of the likelihood function rather than by failure of the algorithm itself. Failure of the algorithm to converge occurred exclusively for samples for which the other estimation methods gave large negative estimates of ξ.

Both studies showed that for small sample sizes ($n \leqslant 15$), in terms of bias and mean square error, maximum likelihood appeared to be a poor competitor to the probability weighted moments estimator, especially for the estimate of a particularly extreme quantile q_p, when ξ is positive. This poor relative performance of the maximum likelihood estimator was explained by examining the sampling behaviour of the original parameters of the considered distribution (see Coles and Dixon (1999)). It was revealed that main difference between the methods arose in the distribution of the shape parameter estimate, which, in the case of maximum likelihood, was positively skewed. Moreover, the expression for quantiles is a non-linear function of ξ, thus a small positive bias for ξ translated to a substantial bias for q_p. For more detailed discussion on bias and mean square error of parameter estimates, we refer to Hosking and Wallis (1987).

Coles and Dixon (1999) emphasize that the relative poor estimation of ξ by the maximum likelihood method can be attributed to the different assumptions. In particular, the probability weighted moments estimator a priori assumes that $\xi < 1$. Thus, the estimated parameter space of $(-\infty, \infty)$ is reduced to $(\infty, 1)$. A consequence of this fact is a reduction in the sampling variation of the estimate of ξ. Moreover, increasing value of ξ leads to higher negative bias in the estimate of ξ. Taking into account the non-linearity in Equation (2.8) as a function of ξ, underestimation of ξ is penalized much less heavily, in terms of mean square error, than overestimation. Otherwise specified, the method of the probability weighted moments provides a certain possibility of trade-off between bias and variance in the estimate of ξ. As a consequence, the distribution of extreme quantiles q_p, which are computed on the basis of the probability weighted moments estimates, avoids the heavy upper tail of the distribution provided by the maximum likelihood estimates. Obviously, the restriction $\xi < 1$ viewed as prior information should be available also to the likelihood-based analysis when comparing the methods fairly. One possibility is to adopt a suitable penalty function in order to incorporate the required structural information about ξ. Thus, we arrive to the method of penalized maximum likelihood estimator.

2.3.4 Model checking

In this section we introduce a statistical approach as well as a graphical approach to check the validity of an extrapolation based on the fitting generalized Pareto distribution.

Graphical verification of the fitted model

Although the graphical approach is not a sufficient justification, it can provide a reasonable prerequisite. We discuss four methods of goodness-of-fit, namely probability plots, quantile plots, return level plots and density plots.

A *probability plot* is a comparison of the empirical distribution and the fitted distribution. Assume $z_{1:k} \leqslant z_{2:k} \leqslant \cdots \leqslant z_{k:k}$ to be the ordered sample of the k extreme excesses over a high threshold $z_1, \ldots, z_k$, i.e. $z_j = x_j - u$ for those indices $i \in \{1, \ldots, n\}$ for which $x_i > u$ and $x_1, \ldots, x_n$ is sequence of the original data. Then the empirical distribution function evaluated at $z_{j:k}$ can be computed as:

$$\tilde{H}^{(u)}(z_{j:k}) = \frac{1}{k} \sum_{i=1}^{k} \mathbb{I}(z_{j:i} \leqslant z_{j:k}) = \frac{j}{k}. \tag{2.27}$$

By substituting of estimated parameters (obtained by application of arbitrary estimating procedure presented in Section 2.3.3 into Equation (2.7), the corresponding model-based estimates of the distribution function evaluated at $z_{j:k}$ are:

$$\hat{H}^{(u)}(z_{j:k}) = 1 - \left(1 + \hat{\xi}\, \frac{z_{j:k}}{\hat{\sigma}_u}\right)^{-1/\hat{\xi}}. \tag{2.28}$$

If the model performs well, then $\tilde{H}^{(u)}(z_{j:k})$ is approximately equal to $\hat{H}^{(u)}(z_{j:k})$ for each j, thus a probability plot, consisting of the points

$$\left\{ \left(\tilde{H}^{(u)}(z_{j:k}), \hat{H}^{(u)}(z_{j:k}) \right); j = 1, \ldots, k \right\},$$

should lie close to the unit diagonal. Any substantial departures from the guideline indicate some failure in the model. A disadvantage of the probability plot is that both estimates of distribution function, $\tilde{H}^{(u)}(z_{j:k})$ as well as $\hat{H}^{(u)}(z_{j:k})$, approach 1 as j increases. It is the accuracy of the model for large values that is of the greatest interest, in other words, the probability plot provides the least information in the region of most concern.

A *quantile plot* represents an alternative that overcomes the deficiency described in the last paragraph. The quantile plot consists of the points

$$\left\{ \left((\hat{H}^{(u)})^{-1} \left(\frac{j}{k} \right), z_{j:k} \right); j = 1, \ldots, k \right\},$$

where $(\hat{H}^{(u)})^{-1}$ denotes the estimate of the quantile, i.e.,

$$(\hat{H}^{(u)})^{-1}\left(\frac{j}{k}\right) = -\frac{\hat{\sigma}_u}{\hat{\xi}}\left(1-\left(1-\frac{j}{k}\right)^{-\hat{\xi}}\right).$$

Similarly as in the previous case, any departures from the unit diagonal indicate model failure.

A *return level plot* is particularly convenient for interpreting extreme value models. The tail of the distribution is compressed, therefore the return level estimates for long return periods are displayed. The plot consists of the locus of points

$$\{(\log y_p, \hat{q}_p)\,; 0 < p < 1\},$$

where $\hat{q}_p$ is the estimate of quantile q_p given by Formula (2.26) and $y_p = -\log(1-p)$ is the return level function (see Section 2.2.1). If the model is suitable for data, the model-based curve and empirical estimates should be in reasonable agreement. Any substantial or systematic departures from guideline suggests an inadequacy of the model. Moreover, the guideline is a straightforward indicator of the type of the generalized Pareto distribution. If the guideline is linear ($\xi = 0$) the exponential distribution is the most likely distribution for the excesses over a high threshold, convex guideline suggest that the excesses have beta distribution and concave guideline corresponds to Pareto distribution.

All methods of goodness-of-fit presented so far are derived from comparison of model-based and empirical estimates of the distribution function. The last graphical check is an equivalent diagnostic, based on the density function though. A *density plot* is a comparison of the probability density function of a fitted model with a histogram of the data. This is generally less informative than the previous plots though. The reason is that the histogram can vary substantially with the choice of grouping intervals.

Statistical verification of the fitted model

In order to check the validity of an extrapolation based on the fitting generalized Pareto distribution, empirical distribution function statistics is employed. This method compares the hypothesized distribution function $\hat{H}^{(u)}$ and the empirical distribution function $\tilde{H}^{(u)}$. We present several test statistics including Kolmogorov-Smirnov (D), Cramer-von Mises (W^2), and also Anderson-Darling statistics (A^2).

Assume $z_{1:k} \leqslant z_{2:k} \leqslant \cdots \leqslant z_{k:k}$ to be the ordered sample of the k extreme excesses over a high threshold $z_1, \ldots, z_k$, as it was defined above. Then the hypothesized distribution function $\hat{H}^{(u)}$ and the empirical distribution function $\tilde{H}^{(u)}$ have the form given by Formulas (2.27) and (2.28). The *Kolmogorov-Smirnov statistic* measures the maximum deviation

of the empirical distribution function and the fitted distribution function, it is defined as:

$$\begin{aligned} D^+ &= \max_{1\leqslant j\leqslant k}\left\{\frac{j}{k}-\hat{H}^{(u)}\left(z_{j:k}\right)\right\} \\ D^- &= \max_{1\leqslant j\leqslant k}\left\{\hat{H}^{(u)}\left(z_{j:k}\right)-\frac{j-1}{k}\right\} \\ D &= \max\left\{D^+,D^-\right\}. \end{aligned} \tag{2.29}$$

The hypothesis regarding the distributional form is rejected if the test statistic, D, is greater than the critical value obtained from a table. There are several variations of these tables in the literature distinguishing by different scalings for the test statistic and critical regions. These alternatives are provided by software programs that perform the test.

The *Cramér–von Mises test* is based on a statistic of the type:

$$W^2=\frac{1}{12k}+\sum_{j=1}^{k}\left(\hat{H}^{(u)}\left(z_{j:k}\right)-\frac{2j-1}{2k}\right)^2. \tag{2.30}$$

As it was in the case of Kolmogorov-Smirnov statistic, the Cramer-von Mises statistic measures in some way the discrepancy between the empirical distribution function and the theoretical function F. Also in this case, the critical values do not depend on the specific distribution being tested, thus they can be obtained from a table.

The last test statistic used in the computational part in oder to assess goodness of the fit is the *Anderson-Darling statistic*. This test gives more weight to the tails than the Kolmogorov-Smirnov test. It has the following form:

$$A^2=-\frac{1}{k}\sum_{j=1}^{k}\left(2j-1\right)\left(\log\hat{H}^{(u)}\left(z_{j:k}\right)+\log\hat{H}^{(u)}\left(z_{k+1-j:k}\right)\right)-k. \tag{2.31}$$

Some of the tests, for example Kolmogorov-Smirnov and Cramer-von Mises, are distribution free in the sense that the critical values do not depend on the specific distribution being tested. The Anderson-Darling test makes use of the specific distribution in calculating critical values. This has the advantage of allowing a more sensitive test and the disadvantage that critical values must be calculated for each distribution. Currently, tables of critical values are available for several distribution families, including generalized Pareto distribution.

It has been proven that the Cramer-von Mises test statistic and the Anderson-Darling statistics performs better than the Kolmogorov-Smirnov statistics, and the Anderson-Darling test gives more weight to the tails of the distribution than the Cramer-von Mises. Historically, the Kolmogorov-Smirnov test statistic has been the most used empirical distribution function statistic, but it tends to be the least powerful, overall. In practice, it is often recommended to use the Anderson-Darling test at the first place and the Cramer-von Mises test as the second choice. For more detailed discussion on advantages and disadvantages of

the test statistics see Stephens (1974).

2.4 Case study

Case study 2.1. Consider the SOA Group Medical Insurance Large Claims Database (available online at web pages of the Society of actuaries (2004)) recording all the claim information over the period 1997, 1998 and 1999. Carry out the analysis of the tail distribution based on the theory explained in Section 2.3. Proceed in the following steps.

(i) Provide basic data descriptive statistics and fit ordinary distribution to data using maximum likelihood approach.

(ii) Employing the graphical threshold selection find candidates for the optimal level of the threshold.

(iii) Conduct the GPD inference for all candidates selected in the previous step and based on statistical testing choose the optimal threshold.

(iv) Compare the approximation of large claims modeled by the ordinary distribution fitted in the first step with generalized Pareto distribution derived in the previous step.

(v) Construct the quantile-quantile plot for fitted gerenalized Pareto distrbution for selected optimal threshold and assess the goodness of the fit. Moreover, estimation the probable maximum loss.

2.4.1 Data description

In the last decades the Society of Actuaries (SOA) has been leading a series of projects exploring and analysing medical insurance large claims. As a part of the research, a number of insurers were requested to collect diverse information about the claims of each claimant. SOA has released dozens of reports and papers discussing medical insurance large claims. The research by Grazier (2004), initiated in October 1998, provides analysis of factors affecting claims incidence rates and distribution of claim sizes. The study considers claims from years 1997, 1998 and 1999. Although the published monograph is rather descriptive, it creates a relevant source of information to our work. Namely we processed the same database as the authors did, therefore in later data description we will refer to their article. Likewise our work the paper by Cebrián et al. (2003) focuses on a statistical modeling of large claims based on the extreme value theory with particular emphasis on the threshold methods of fitting generalized Pareto distribution. A significant part of the article is devoted

to a practical implementation of the method carried out on 1991 and 1992 group medical claims database maintained by the SOA. The study considered censored data, i.e., claimants with paid charges exceeding $\$25,000$.

We consider the SOA Group Medical Insurance Large Claims Database (available online at web pages of the Society of actuaries (2004)), which records all the claim information over the period 1997, 1998 and 1999. Since each of the records for specific year contains sufficiently high number of observations we will carry out our practical implementation on 1997 data. Those related to years 1998 and 1999 will serve as an assessment for predicting the future claim sizes based on the extreme value theory and classical forecasting based on the fitting of standard distributions, respectively. Each record of row of the file represents one particular claim and comprises of 27 fields, which can be divided into three subgroups. First 5 columns provide a general information about a claimant such as his/her date of birth or sex. Other 12 columns quantify various types of medical charges and expenses. In our computational experiments we consider only the total paid charges (column 17). The last 10 fields summarize details connected to the diagnosis. For more detailed description of the data we refer to Grazier (2004).

Remark 2.7. The whole analysis is carried out in program R. This section is devoted to commend on the course of analysis and obtained graphs and tables, relevant inputs and outputs to program R are inserted in the text. The complete code can be find at the end of the chapter in Appendix 2.6.

Detailed descriptive statistic is presented only for the claim year 1997, though we provide a brief comment on behavior of the claims in the course of the whole period at the end of the paragraph. The data set consists of approximately 1.2 million observations of $2 billion in total paid charges. The size of the claims range from $0.01 to $1225908.30 with the sample mean equal to $1613.58. Comparing the quantiles of the distribution, which are stated in Table 2.1 together with overall descriptive statistics, we can conclude that the observations are mainly concentrated in lower values.

mean	1 613.58	minimum	0.01
1st quantile	101.00	maximum	1 225 908.30
median	293.70	standard deviation	7 892.97
3rd quantile	993.60	skewness	34.82

Table 2.1: Descriptive statistics of the claims (1997).

The descriptive statistics can be computed in program R as follows:

```
procedure_descriptive_stat <- function (data){

  c (length (data), sum (data), summary (data))
```

```
}

sample_length <- 10000
set.seed (123)
log_claims <- log (sample (claims, sample_length, replace = FALSE,
                           prob = NULL))

procedure_descriptive_stat (claims)
```

The output of the program gives the following results (which are summarized also in Tabel 2.1):

```
Length          Sum             Min.            1st Qu.
1.241438e+06    2.003162e+09    1.000000e-02    1.010000e+02

Median          Mean            3rd Qu.         Max.
2.937000e+02    1.614000e+03    9.936000e+02    1.226000e+06
```

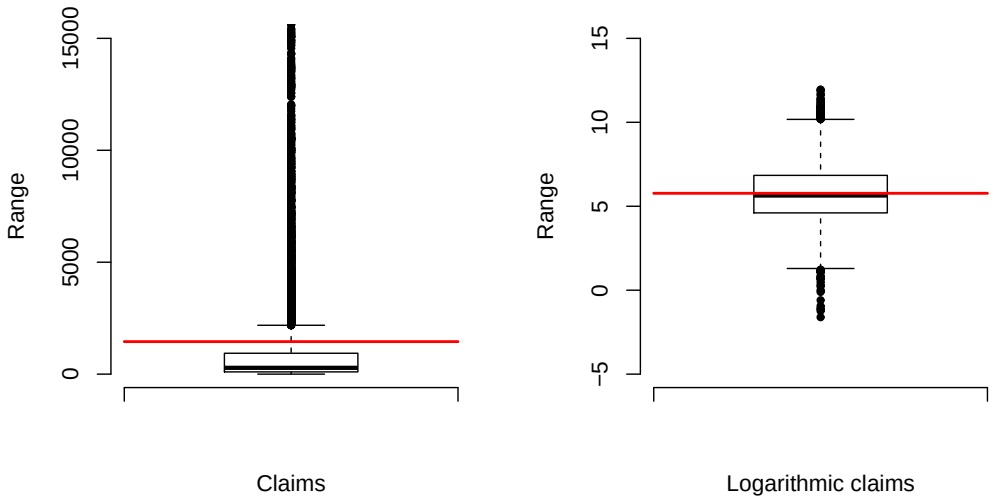

Figure 2.2: Box plots of the claims 1997 (left) and the logarithmic claims 1997 (right).

Figure 2.2 displays box plots for the claims and the logarithmic claims. The red line in both pictures express the mean. The left box plot in Figure 2.2 clearly shows that the mass is concentrated in lower values. We note that only claims smaller than 15000 are displayed on the left picture in order to distinguish at a glance different values of quantiles, the median and the mean. The right graph demonstrates shifting of the mass when applying

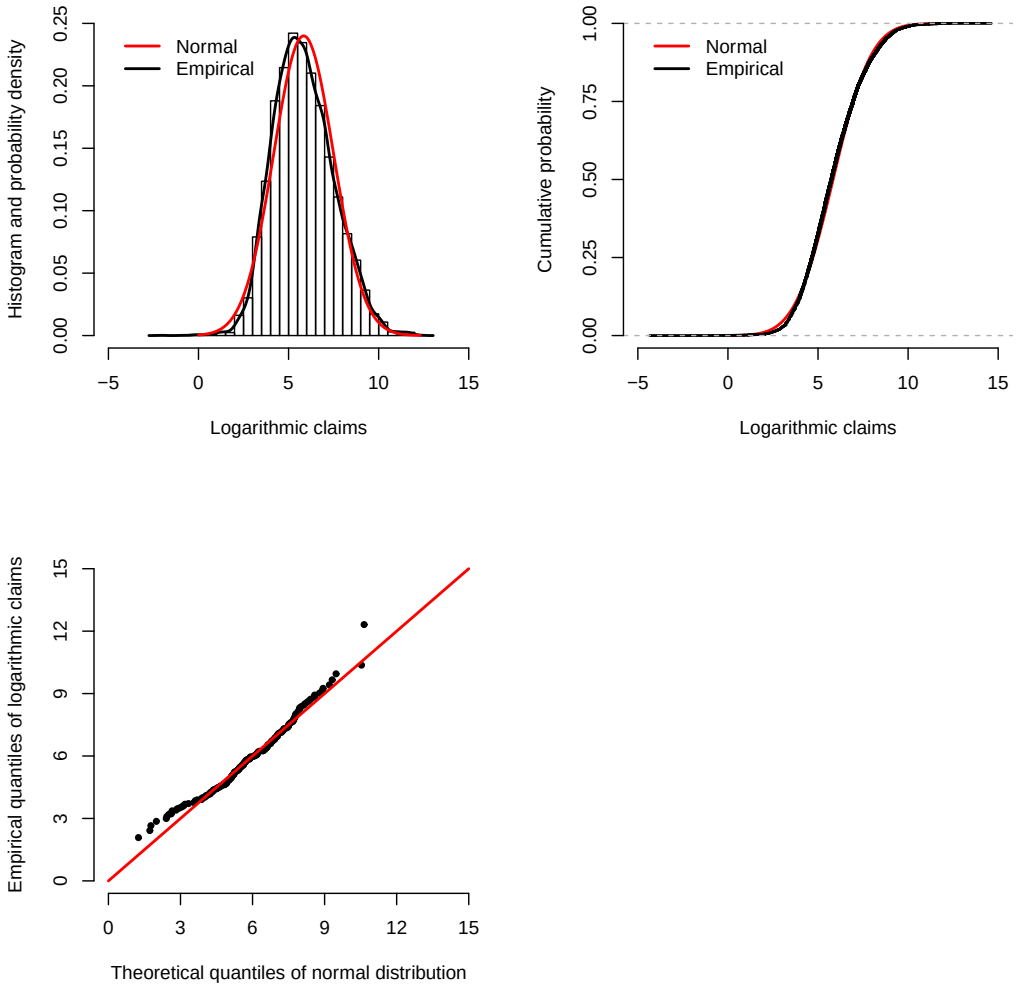

Figure 2.3: The empirical density of the logarithmic claims (1997) approximated by normal distribution (top left), the empirical distribution of the logarithmic claims (1997) approximated by normal distribution (top right), the quantile-quantile plot comparing the empirical quantiles of the logarithmic claims (1997) and the theoretical quantiles of normal distribution (bottom left).

the logarithmic transformation. We observe that the data are still slightly right-skewed even on the log-scale since the mean is higher than the median (the skewness coefficient equals to 34.82). This indicates the long-tailed behavior of the underlying data.

We depicted the box plot using R as follows:

```
procedure_plot_box_plot <- function (data){

  log_data <- log (data)

  # box plot
  par (mfrow = c (1,2))

  # box plot of claims

  boxplot (data, axes=FALSE, pch=20, ylim=c(0,15000), xlab="Claims",
           ylab="Range")
  lines (c (0.5, 1.5), c (mean (data), mean (data)), col="red", lw=2)
  axis (1, at=seq (0.5, 1.5, by=1), labels=c ("",""))
  axis (2, at=seq (0, 15000, by=5000))

  # box plot of logarithmic claims

  boxplot (log_data, axes=FALSE, pch=20, ylim=c(-5,15),
           xlab="Logarithmic claims", ylab="Range")
  lines (c (0.5, 1.5), c (mean (log_data), mean (log_data)), col="red",
         lw=2)
  axis (1, at=seq(0.5, 1.5, by=1), labels=c ("",""))
  axis (2, at=seq (-5, 15, by=5))
}

procedure_plot_box_plot (claims)
```

In order to compare the goodness of prediction of future large claims based on standard statistical methods and the method based on the fitting of generalized Pareto distribution, we are interested in finding a distribution which traces behaviour of the claims the best. With respect to the number of observations, we carried out the computation on a random sample of 10000. Figure 2.3 displays the empirical density and the empirical distribution of the logarithmic claims approximated by the fitted normal distribution. Considering both upper graphs we can conclude that the estimated normal distribution describes well the logarithmic claim amounts although the empirical data appear to be slightly skewed. The quantile-quantile plot in Figure 2.3 provides us with another evidence of goodness of the fit. The graph displays theoretical quantiles of the fitted normal distribution plotted against empirical quantiles of logarithmic values of the claims. The axis of the quadrant represented by the red line approximates well the points, thus we can conclude that the empirical quantiles can be treated as quantiles of normal distribution.

The accuracy of the fitted distribution was verified by the Shapiro-Wilk test on the confidence level 0.05. We run the test ten times on different random samples consisting of

100 observations. Based on the p-values (the confidence level was exceeded in nine cases out of ten) we have not rejected the null hypothesis and thus we can conclude that logarithmic claims are normally distributed or the claims are log-normally distributed, respectively [3].

	$\hat{\mu}$	$\hat{\sigma}$
1997	5.820	1.666
1998	5.835	1.687
1999	5.800	1.708

Table 2.2: Mean and standard deviation of the fitted normal distribution to the logarithmic data (claim years 1997, 1998 and 1999).

Distribution fitting is implemented in program R via procedure `procedure_fitting_distribution`, namely we use R built function `fitdistr`. The goodness of the fit is assessed by Shapiro-Wilk test `shapiro.test` run 10 times on different samples. For graphical illustration of the fit we use procedure `procedure_plot_descriptive_stat`. The code is as follows:

```
procedure_fitting_distribution <- function (data, only_est){

  # function fitdistr uses maximum likelihood estimator for fitting
  # standard distributions
  fit <- fitdistr (data, "normal")

  log_mean_est <<- fit$estimate[1]
  log_sd_est <<- fit$estimate[2]

  # if parameter only_est is set to be 1 (TRUE) then procedure is
  # dedicated only to estimate distribution function parameters and
  # breaks at the following line
  if (only_est==1){}
  else {
    # testing accuracy of lognormal disribution (n = number of
    # repetitions of Shapiro-Wilk test)
    n <- 10
    p_value_sw <- rep (0, n)

    for (k in 1:n){
      data_test <- sample (data, 50, replace=FALSE, prob=NULL)
```

[3] When considering several hypotheses in the same test the problem of multiplicity arises. Application of the Holm–Bonferroni method is one of the many ways to address this issue. It modifies the rejection criteria in order to control the overall probability of witnessing one or more type I errors at a given level.

```
      test_sw <- shapiro.test (data_test)
      p_value_sw [k] <- test_sw$p.value
    }

    c (log_mean_est, log_sd_est, p_value_sw)
  }
}

procedure_plot_descriptive_stat <- function (data, log_mean_est,
                                             log_sd_est){

  log_data <- log (data)

  # goodness of fit plots
  par (mfrow = c (2,2))

  n <- 200
  log_data_fit <- seq (0, max (log_data), length = n)

  # histogram accompanied by theoretical and empirical densities
  histogram <- hist (log_data, breaks=25, plot=FALSE)
  histogram$counts <- histogram$counts / (diff (histogram$mids [1:2])*
                      length (log_data))

  data_norm <- dnorm (log_data_fit, mean = log_mean_est, sd = log_sd_est)

  plot (histogram, xlab="Logarithmic claims",
        ylab="Histogram and probability density", main="", axes=FALSE,
        xlim=c(-5,15), ylim=c(0,0.25))
  lines (density (log_data), col="black", lwd=2)
  lines (log_data_fit, data_norm, col="red", lwd=2)
  axis(1, at=seq (-5, 15, by=5))
  axis(2, at=seq (0, 0.25, by=0.05))
  legend (-5, 0.25, inset=.1, bty = "n", c("Normal","Empirical"),
          lwd=c(2,2) , col=c("red","black"), horiz=FALSE)

  # theoretical and empirical distribution functions
  plot (log_data_fit, pnorm (log_data_fit, mean = log_mean_est,
        sd = log_sd_est), type="l", col="red", lwd=2,
        xlab="Logarithmic claims", ylab="Cumulative probability",main="",
        axes=FALSE, xlim=c(-5,15), ylim=c(0,1))
  plot (ecdf (log_data), cex=0.01, col="black", lwd=2, add=TRUE)
  axis (1, at=seq (-5, 15, by=5))
```

```
  axis (2, at=seq (0, 1, by=0.25))
  legend (-5, 1, inset=.1, bty = "n", c("Normal","Empirical"),lwd=c(2,2),
          col=c("red","black"), horiz=FALSE)

  # quantile-quantile plot
  qqplot (rnorm (n, mean = log_mean_est, sd = log_sd_est), pch=20,
          log_data, xlab="Theoretical quantiles of normal distribution",
          ylab="Empirical quantiles of logarithmic claims", main="",
          axes=FALSE, xlim=c(0,15), ylim=c(0,15))
  lines (c (0, 15), c (0,15), lwd=2, col="red")
  axis (1, at=seq (0, 15, by=3))
  axis (2, at=seq (0, 15, by=3))
}

year <- 97
claims <- get (paste ("claims_", year, sep=""))

result_claims <- procedure_fitting_distribution (log_claims, 0)
result_claims

procedure_plot_descriptive_stat (claims, result_claims [1],
                                 result_claims [2])
```

Appart from the graphical results depicted in Figure 2.3, the above stated code has another output, `results_claims`, which summarizes estimated parameters of the fitted distribution and p-values of 10 Shapiro-Wilk tests:

```
mean           sd             p-value 1      p-value 2      p-value 3
5.824289294    1.653981713        0.94778916     0.94904064
    0.70782137

p-value 4      p-value 5      p-value 6      p-value 7      p-value 8
0.83808489     0.71463362     0.42941229     0.82198443     0.10121996

p-value 9      p-value 10
0.15263836     0.07965254
```

Analysis of the remaining claim years leads to similar conclusions as drawn above, i.e., we cannot reject the hypothesis of the normal distribution of the logarithmic claims for any of the sample. The parameters of the estimated distributions can be found in Table 2.2, where $\hat{\mu}$ and $\hat{\sigma}$ denote the parameters of the fitted normal distribution. One can observe that the sample mean and standard deviation do not differ significantly regarding the claim year.

2.4.2 Data analysis

Lognormal, log-gamma, gamma, as well as other parametric models have been often applied to model claim sizes in both life and non-life insurance (see for instance Klugman et al. (1998)). However, if the main interest is in the tail behaviour of loss distribution, it is essential to have a special model for largest claims. Distribution providing a good overall fit can be inadequate at modeling tails. Extreme value theory together with generalized Pareto distribution focus on the tails and are supported by strong theoretical arguments introduced in Chapters 2.2 and 2.3. The following chapter is devoted to reexamination of SOA Group Medical Insurance Large Claims Database using the threshold approach. To check the improvement achieved by using the generalized Pareto distribution instead of a classical parametric model, i.e., the traditional approach of fitting standard distributions based on the analysis performed in Subsection 2.4.1, we provide a brief discussion on these two methods as a summary of this chapter.

Application of graphical threshold selection

The first section focuses on identifying the optimal threshold level in order to construct the further inference. We attempt to apply graphical methods presented in Subsection 2.3.2 for each claim year 1997, 1998 and 1999. We provide a detailed analysis and description of respective plots for the claim year 1997 together with discussion on the optimal threshold choice plot. The results based on the same procedure for the remaining claim years 1998 and 1999 are summarized in Table 2.4 at the end of this section.

The mean residual life plot displays empirical estimates of the sample mean excesses plotted against a range of the thresholds, along with confidence interval estimates. Confidence intervals can be added to the plot based on the approximate normal distribution of sample mean stemming from the Central limit theorem. The threshold is chosen to be the lowest level where all the higher threshold based sample mean excesses are consistent with a straight line. Coles (2001) acknowledges that the interpretation of the mean residual life plot is not always simple in practice. The subsequent analysis shows that his attitude is reasonable.

Figure 2.4 shows the mean residual life plot with approximate 95% confidence intervals for the claim year 1997. In our analysis we proceeded censored data, i.e., claim ammounts exceeding \$100 000 over the related period. The left censorship is not a problem since we are interested in the extreme behavior, thus there is no truncation due to benefit maxima. The lower bound of \$100 000 for the threshold selection seems to be reasonable considering descriptive statistics of the claim database (see Table 2.1) and shape of the mean residual life plot for lower threshold than 100 000. The increasing variance of the mean residual life for high thresholds leads to wide confidence intervals, which must be taken into account when assessing the threshold choice. Once the confidence intervals are taken into account, the graph appears to ascend from $u = 100\,000$ to $u \approx 400\,000$, within it is approximately linear.

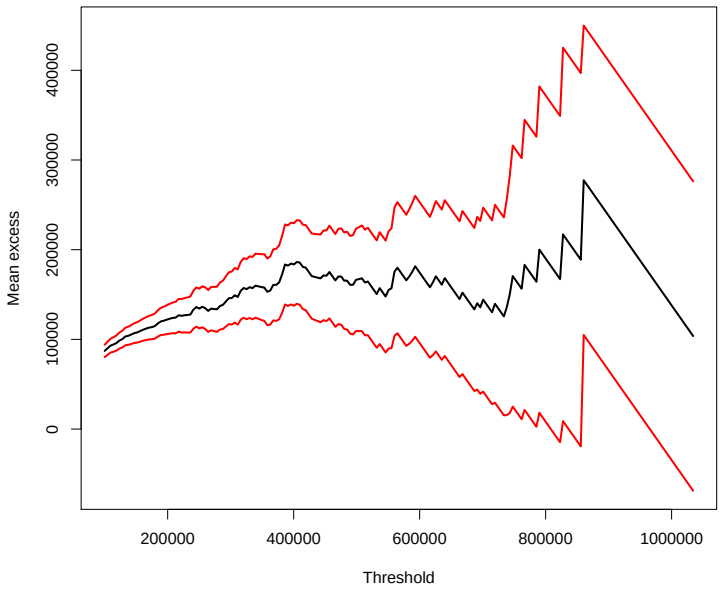

Figure 2.4: The threshold selection using the mean residual life plot (claim year 1997).

Upon $u \approx 700\ 000$ it decays slightly. It is tempting to conclude that there is no stability until $u \approx 700\ 000$, after which there is an evidence of approximate linearity. This conclusion suggests to take the threshold equal to 700000. However, there are just 10 observations above this level, too few to make meaningful inference. Moreover, the estimated mean residual life for large values of u is unreliable due to the limited sample of data on which the estimate and confidence interval are based. Hence, it is preferable to assume that there is some evidence of linearity above $u = 400\ 000$ and work with lower threshold level, which provides us with 50 observations.

As already stated, the threshold is selected to be the lowest level where all the higher threshold based sample mean excesses are consistent with a straight line, i.e., for any higher $u > u_0$ the mean residual life is linear in u with gradient $\xi/(1-\xi)$ and intercept $\sigma_{u_0}/(1-\xi)$. In further consideration we were particularly interested in estimation of the shape parameter ξ based on fitting a linear model to sample mean excesses above threshold choices stemming from the previous discussion (400000, 600000 and 700000). A simple linear regression estimates of parameters for three threshold values provide three fitted mean residual life straight lines in Figure 2.5. In the computation we applied the weighted least squares method for fitting the linear model, where the weights were chosen as to correspond to distribution of the claim mass. One way to achieve the desired effect is to assign to each mean excess

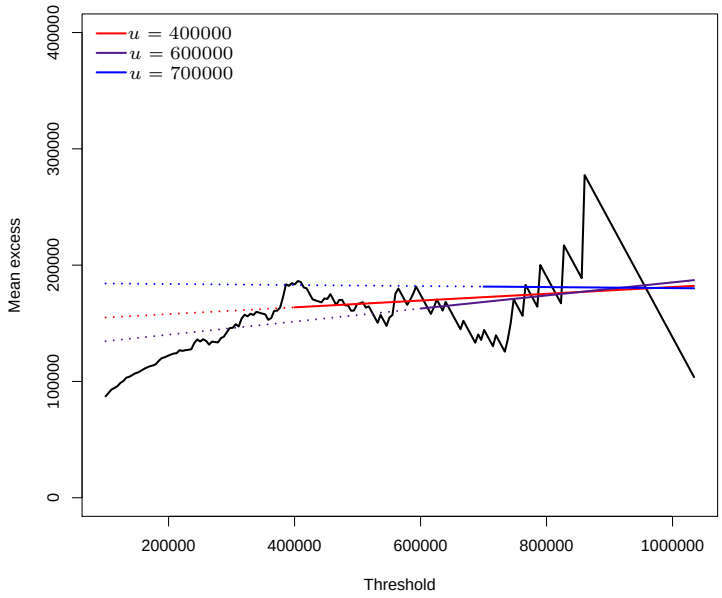

Figure 2.5: Mean residual life plot (claim year 1997) supplemented by the mean residual life estimates for thresholds $u = 400\,000$, 600 000 and 700 000.

the normalized number of observations in the original sample that exceed considered value. Table 2.3 displays obtained estimates of the shape parameter for three threshold choices.

u	ξ
400000	0.0283
600000	0.0532
700000	-0.0044

Table 2.3: Estimates of the shape parameter ξ based on the linear regression model for the mean residual life (claim year 1997).

The residual life plot and its alternative accompanied by linear fit can be obtained as program R outputs by running the beneath stated code. Note that the standard residual life plot was displayed by R built function `mrlplot`, whereas the other plot was implemented by hand in order to add the fitted regression lines.

```
procedure_mrlplot <- function (data, limit, conf_level){

  par (mfrow = c (1, 1))
```

```
  # built function (package evd) for the construcion of the empirical
  # mean residual life plot
  mrlplot (data, u.range=c (limit, quantile (data, probs=conf_level)),
           xlab="Threshold", ylab="Mean excess", main="", lwd=2, lty = 1,
           col = c ("red","black","red"), nt=200)
}

# fit linear regression model (x ~ threshold, y ~ mean access) in order
# to estiate shape parameter
procedure_linefit <- function (data, seq_x, seq_y, threshold_value){

  n <- length (seq_x)
  w <- rep (0, n)

  for (i in 1:n){
    w [i] <- length (which (data > seq_x [i]))
  }

  position <- which (seq_x > threshold_value) [1]
  seq_x_fit <- seq_x [position : n]
  seq_y_fit <- seq_y [position : n]
  w_fit <- w [position : n]
  w_fit <- w_fit / sum (w_fit)

  # fit linear regression model using built function lm (package stats)
  par_est <- lm (seq_y_fit ~ seq_x_fit, weights=w_fit)

  par_est
}

procedure_mrlplot_linefit <- function (data, min, max, n, threshold_vec){

  u_range <- seq (min, max, by=(max-min)/(n-1))
  mean_excess <- rep (0, length (u_range))

  for (i in 1:length (u_range)){
    data <- data [data > u_range [i]]
    data_excess <- data - u_range [i]
    mean_excess [i] <- sum (data_excess)/length (data_excess)
  }

  n_threshold <- length (threshold_vec)
```

```
  par_est_matrix <- matrix (0, ncol=2, nrow=n_threshold)

  for (i in 1:n_threshold){
    # fit linear regression model
    par_est <- procedure_linefit (data, u_range, mean_excess,
                                  threshold_vec [i])
    # parameters of estimated linear regression model
    par_est_matrix [i, 1] <- par_est$coefficients [1]
    par_est_matrix [i, 2] <- par_est$coefficients [2]
  }

  # regression estimate of shape parameter
  est_ksi <- par_est_matrix [, 2]/(1+par_est_matrix [, 2])

  lin_function <- function (x, i){par_est_matrix [i, 1] +
                                  par_est_matrix [i, 2]*x}

  col_vec <- c ("red", "purple4", "blue", "green", "navy")
  name_vec <- paste ("u = ", threshold_vec, sep="")
  par (mfrow=c (1, 1))

  plot (u_range, mean_excess, type="l", lwd=2, col="black",
        xlab="Threshold", ylab="Mean excess", ylim=c(0,400000))
  for (i in 1:n_threshold){
    lines (c (min, threshold_vec[i]), c (lin_function (min, i),
           lin_function (threshold_vec[i], i)), col=col_vec [i],
           lw=2, lty=3)
    lines (c (threshold_vec[i],max), c(lin_function(threshold_vec[i],i),
           lin_function (max, i)) , col=col_vec [i], lw=2)
  }
  legend ("topleft", bty = "n", name_vec, lwd=rep (2, n_threshold),
          col=col_vec [1:n_threshold], horiz=FALSE)

  est_ksi
}

claims_excess <- claims [claims > limit]
threshold <- get (paste ("threshold_", year, sep=""))

options("scipen" = 20)

# the empirical mean residual life plot
procedure_mrlplot (claims_excess, limit, 0.999)
```

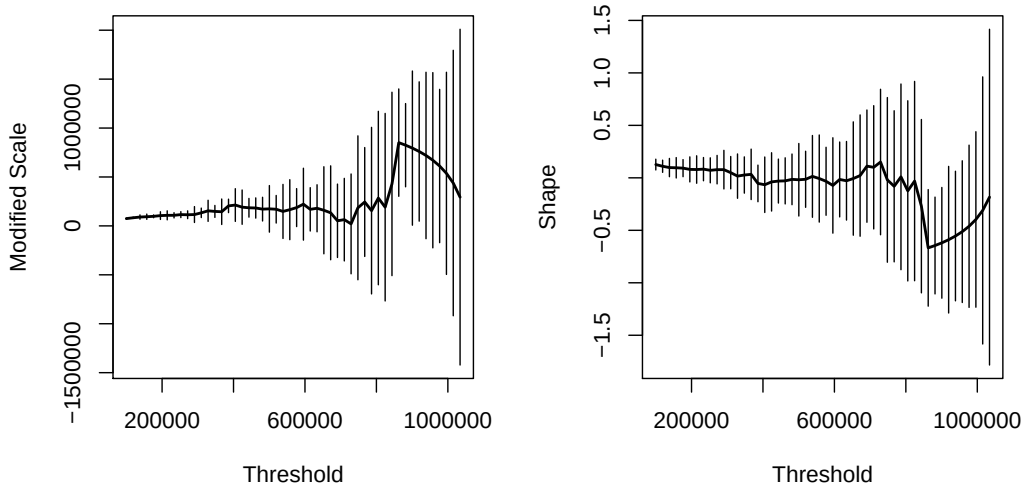

Figure 2.6: The threshold selection using the threshold choice plot (claim year 1997).

```
# linear regression estimates of parameters for thresholds values (given
# by variable threshold) acompanied by graphical illustration
est_ksi <- procedure_mrlplot_linefit (claims_excess, limit, quantile
# (claims_excess, probs=0.999), 200, threshold)
est_ksi
```

Program R also provides the estimates of parameter ξ for different choices of the threshold:

```
     400000        600000        700000
[1]  0.028278651   0.053213943   -0.004416776
```

The threshold choice plot displays empirical estimates of the modified scale parameter and shape both plotted against a range of the thresholds, along with confidence interval estimates. Confidence intervals can be added to the plot based on the so called delta method, as suggested in Coles (2001), or using Fisher information, as described in Ribatet (2011). The threshold is chosen to be the lowest level where all the higher threshold based modified scale and shape parameters are constant. However, in practice decisions are not so clear-cut and threshold selection is not always simple.

Figure 2.6 shows the threshold choice plot with approximate 95% confidence intervals for the claim year 1997. In order to reduce time consumption of the computation, we again considered claim amount exceeding \$100 000 over the related period. As in the case of the

mean residual life plot, we observe the increasing variance of the modified scale as well as the shape parameter for high thresholds leading to wide confidence intervals, which must be taken into consideration when assessing the threshold choice. Both graphs appear to fluctuate around a horizontal line from $u = 100\ 000$ to $u \approx 800\ 000$. The change in pattern for the threshold of $u \approx 800\ 000$ can be characterized by a sharp jump, whereupon the curve of modified scale decays significanty, or the curve of shape increases respectively. This observation suggests that there is no stability until u reaches 850000. Moreover, exceeding this level there is no evidence of approximate constancy for either modified scale or shape parameter. On the other hand, there are just 3 observations above this level, too few to make any further inference. Moreover, the accuracy of the related estimates is disputed due to the limited sample of data. Given these circumstances, it is probably better to choose a lower. In order to have a reasonably wide sample of data, a threshold lower then 600 000 is preferred. However it is difficult to make any decision about the exact value since the modified scale curve and the scale curve can be both well approximated by a horizontal line. In this case, we adopt conclusion arising from mean residual life plot or L-moments plot.

The threshold choice plot is obtained in program R using built function `tcplot`:

```
procedure_tcplot <- function (data, limit, conf_level){

  par (mfrow=c (1,2))

  # built function (package evd) for the construcion of threshold choice
  # plot
  tcplot (data, u.range = c (limit, quantile (data, probs=conf_level)),
          type="l", lwd=2, nt=50)
}

procedure_tcplot (claims_excess, limit, 0.999)
```

The L-moments plot displays empirical estimates of L-skewness plotted against empirical estimates of L-kurtosis, along with the theoretical curve defined by Equation (2.13) which is traced as a guideline. Decision about choice of the optimal threshold value is made based on position of estimated points with respect to this theoretical line. Unfortunately, it has been shown by Ribatet (2011) that the graphic has often poor performance on real data.

Figure 2.7 depicts the L-moments plot for the claims year 1997. The red line is associated with the theoretical curve which reflects general relation between L-skewness and L-kurtosis described by Formula 2.13. The black line traces their empirical estimates. As in the previous graphical illustrations we use lower bound for threshold limit equal to \$100 000 since we have already shown that a lower threshold value is rather unlikely. The empirical estimates of L-skewness and L-kurtosis for this lowest admissible threshold are represented by the inner endpoint of the black curve in the graph. As the threshold limit rises, the points corresponding to pairs of empirical estimates of L-moments scroll along the guideline towards

the beginning of the coordinate system. The theoretical relation between L-skewness and L-kurtosis given by Equation (2.13) seems to be accurate for sample of data generated by a threshold up to 500 000. Once the threshold reaches 500 000, the estimated points lie on the outer curve and converge to a point on the horizontal axis as the threshold approaches the maximal value of the sample data. Thus the particular shape of L-moments plot is tempting to select a threshold value that does not exceed 500 000.

L-moments plot is implemented in program R as follows:

```
# threshold selection: L-moments plot (dipicts L-kurtosis against
# L-skewness)
procedure_lmomplot_est <- function (data, min, max, n){

  u_range <- seq (min, max, by=(max-min)/(n-1))

  # L-skewness as a function of L-curtosis
  fc_tau_4 <- function (x){x*(1+5*x)/(5+x)}

  # x and y represent theoretical L-kurtosis and L-skewness
  x <- seq (0, 1, by=0.01)
  y <- fc_tau_4 (x)

  # vec_tau_3 and vec_tau_4 represent empirical L-kurtosis and L-skewness
  vec_tau_3 <- rep (0, length (u_range))
  vec_tau_4 <- rep (0, length (u_range))

  for (i in 1:length (u_range)){
    u <- u_range [i]
    data <- data [data >= u]

    # bouilt function lmoms (package lmomco) computes the sample
    # L-moments
    lmom_est <- lmoms (data, nmom=4) $ratios
    vec_tau_3 [i] <- lmom_est [3]
    vec_tau_4 [i] <- lmom_est [4]
  }

  par (mfrow=c (1,1))

  label_1 <- expression (tau[3])
  label_2 <- expression (tau[4])
  plot (x, y, type="l", lwd=2, col="red", xlab=label_1, ylab=label_2)
  points (vec_tau_3, vec_tau_4, type="l", lwd=2)
}
```

```
procedure_lmomplot_est (claims_excess, limit, quantile (claims_excess,
                        probs=0.997), 200)
```

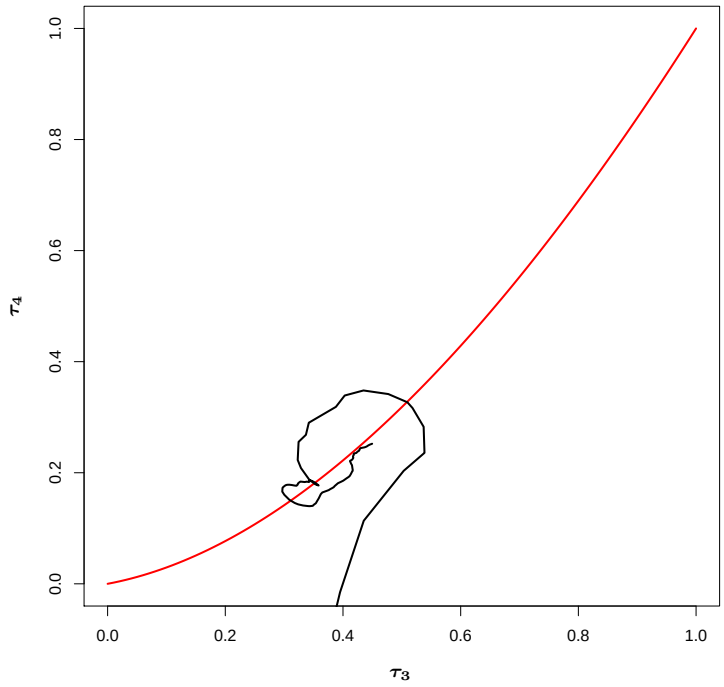

Figure 2.7: The threshold selection using the L-moments plot (claim year 1997).

Remark 2.8. In order to assess the optimal threshold level based on dispersion index plot, one needs to have two time series at disposal. The first one tracks the absolute values of the examined process, the other one records moments on a time scale when the events occur. Since the second information was not available for our claim database, utilization of this method was unfeasible.

We carried out a similar procedure for determining the optimal threshold also for the remaining claim years 1998 and 1999. The analysis did not result in any surprising conclusions that should be commented with special emphasis here. Conversely, analogous discussion on the graphical methods for threshold selection could be provided. However, instead of length interpretation of the graphical results, we only state valid threshold choices together with the estimates of the shape parameter ξ based on the linear regression model for the mean residual life in Table 2.4.

1997		1998		1999	
u	ξ	u	ξ	u	ξ
400000	0.0283	300000	0.1708	300000	0.1259
600000	0.0532	600000	0.0939	700000	-0.3350
700000	-0.0044	700000	-0.1267	1000000	-1.2795

Table 2.4: Estimates of the shape parameter ξ based on the linear regression model for the mean residual life (claim years 1997, 1998 and 1999).

To make conclusion about the threshold choice that will be used in further analysis, we take into account all results obtained from the previous graphical analysis. In the spirit of the mean residual life plot the threshold of either 400 000 or 700 000 should be adopted. Whereas the value of 700 000 provides better asymptotic linearity of the mean residual life corresponding to higher thresholds, the value of 400 000 generates large sample of data that is convenient for further distribution estimation. As suggested in paragraph discussing the threshold choice plot, a threshold lower than 600 000 is preferred in order to derive a meaningful inference for generalized Pareto distribution. The L-moments plot is tempting to select a threshold lower than 500 000 as the relation between empirical L-skewness and L-kurtosis is rather valid. Considering all these arguments we suggest to use the threshold value equal to 400 000 in further analysis.

Fitting the generalized Pareto distribution

The previous section was devoted to selecting the optimal threshold level in order to identify the extremal events that are supposed to have the generalized Pareto distribution. Considering the threshold choices from the previous part we sought for adequate shape and scale parameters of the generalized Pareto family for approximating the tail behavior of our data set. We utilize all three approaches for fitting distribution, the maximum likelihood, the probability weighted moments and the penalized maximum likelihood, that are presented in Section 2.3.3 pointing out their differences reflecting in estimated parameters and their confidence intervals. Let us recall that the analytical maximization of the log-likelihood is not possible, thus to find the local maximum requires using of numerical methods. In the implementation, we exploit a procedure based on a quasi-Newton method. In the case of the penalized maximum likelihood we adopted a proposal of Coles and Dixon (1999) to set the parameters α and λ both equal to 1. Finally, the constants γ and δ of the plotting position in the probability weighted moments method was set as follows: $\gamma = -0.35$ and $\delta = 0$.

Once the parameters are estimated we check the model using statistical tests (see Subsection 2.3.4) accompanied by graphical verification (see Subsection 2.3.4). We carried out the computation for each claim year separately. The whole procedure is presented in detail for the claim year 1997 whereas the results for the claim years 1998 and 1999 are briefly

	Threshold u		
	400000	600000	700000
`mle`	183913	174424	145182
	$(113914, 253911)$	$(150547, 198301)$	$(14828, 275535)$
`pmle`	183913	174424	145182
	$(113914, 253911)$	$(57283, 291565)$	$(121910, 168454)$
`pwm`	214691	207588	116705
	$(124034, 305348)$	$(52591, 362586)$	$(3560, 229851)$

Table 2.5: Estimates and confidence intervals of the scale parameter σ_u for different choices of the threshold (claim year 1997) applying the maximum likelihood method (`mle`), the penalized maximum likelihood method (`pmle`) and the probability weighted moments method (`pwm`).

	Threshold u		
	400000	600000	700000
`mle`	-0.0572	-0.0607	0.0493
	$(-0.3186, 0.2041)$	$(-0.3918, 0.2704)$	$(-0.6011, 0.6996)$
`pmle`	-0.0572	-0.0607	0.0000
	$(-0.3186, 0.2041)$	$(-0.5210, 0.3995)$	$(-0.0859, 0.0859)$
`pwm`	-0.1674	-0.1901	0.1961
	$(-0.5076, 0.1729)$	$(-0.7991, 0.4188)$	$(-0.5512, 0.9434)$

Table 2.6: Estimates and confidence intervals of the shape parameter ξ for different choices of the threshold (claim year 1997) applying the maximum likelihood method (`mle`), the penalized maximum likelihood method (`pmle`) and the probability weighted moments method (`pwm`).

summarized at the end of the section. Although we suggested to carry out further computation with the lowest value of threshold (for the claim year 1997 it was 400 000), we present the results obtained for all threshold levels that were identified as reasonable choices in the previous section.

Table 2.5 and Table 2.6 report the estimates and confidence intervals of scale and shape parameters obtained for different values of threshold levels applying the maximum likelihood, the penalized maximum likelihood and the probability weighted moments methods. First let us compare the maximum likelihood and probability weighted moments estimator. We observe difference in the scale and shape estimates for all choices of threshold level. For thresholds equal to 400 000 and 600 000 the maximum likelihood based scale parameter is smaller than the probability weighted moments based whereas for threshold equal to 700 000

it is greater. In terms of the shape parameter, it is the other way around, the maximum likelihood based shape parameter is bigger than the probability weighted moments based for threshold choices 400 000 and 600 000 and smaller for threshold equal to 700 000. However the sign of the shape parameter remains consistent with varying the estimation procedure. The exceedances over thresholds equal to 400 000 and 600 000 have beta distribution while those over 700 000 can be approximated by Pareto distribution. Note that the statement holds true when applying the penalized maximum likelihood estimator, except the case of exceedences over 700 000 which seem to be exponentially distributed. We revise that for small sample sizes ($n = 15$), in terms of bias and mean square error, the probability weighted moments estimator performs better than the maximum likelihood estimator. The relatively better estimation of ξ by the probability weighted moments method can be attributed to the a priori assumption that $\xi < 1$. The difference becomes even more significant for the estimate of a particularly extreme quantile q_p, when ξ is positive. Threshold 400 000 generates 50 observations thus the maximum likelihood estimates can be considered sufficiently accurate. For higher threshold only few observation are available (for 600 000 there are 16 exceedances, for 700 000 there are only 10). Therefore in these cases one should rely on the probability weighted moments estimator.

The motivation for using the penalized maximum likelihood method is to incorporate the extra information that $\xi < 1$ to the likelihood-based analysis in order to assess fairly the maximum likelihood and probability weighted moments estimator. However, as it follows from Formula (2.24), for negative value of parameter ξ the penalty function equals to 1 and the estimates based on the maximum likelihood method and the penalized maximum likelihood method are equal. Therefore setting the threshold level as 400 000 or 600 000 leads to the same estimates of the scale and shape parameters (see Table 2.5 and Table 2.6). Note that the confidence intervals vary due to different standard errors of estimated parameters. Both methods should generate different estimates for threshold 700 000 where ξ is positive. However we observe that the scale parameter remains consistent and the estimates of the shape parameter do not differ significantly. These results can be attributed to the fact that the value of the penalty function is close to 1 for positive values of ξ close enough to 0. From the behaviour of the penalty function, i.e. for all selected threshold levels its values are equal or very close to 1, it is not surprising that the estimates based on the penalized maximum likelihood method correspond rather those based on the maximum likelihood method than those based on the probability weighted moments method.

Once the parameters of the generalized Pareto distribution are specified, goodness of the fit has to be checked. Table 2.7 contains p-values of statistical tests presented earlier in Subsection 2.3.4, namely the Anderson-Darling test, Kolmogorov-Smirnov test and Cramer-von Mises test. We stated that the Anderson-Darling statistics performs better than the remaining two, particularly when assessing the tail distribution, thus we make conclusion predominantly on the basis of results of the Anderson-Darling test. We observe that p-

values of the test statistic exceed considerably the confidence level of 0.05 for all threshold choices and all estimating approaches. The same conclusion can be drawn when considering the Kolmogorov-Smirnov test or Cramer-von Mises test. Hence we statistically confirmed suitability of the fitted generalized Pareto distribution for all choices of threshold level.

	Threshold u		
	400000	600000	700000
`mle`	0.7799	0.7801	0.8581
	0.8183, 0.1871	0.7316, 0.8657	0.9445, 0.5794
`pmle`	0.7799	0.7801	0.8750
	0.8183, 0.9870	0.7316, 0.3842	0.9596, 0.5794
`pwm`	0.9733	0.8522	0.9117
	0.9671, 0.6521	0.9513, 0.9516	0.8953, 0.5794

Table 2.7: P-values of Anderson-Darling (top), Kolmogorov-Smirnov (bottom left) and Cramer-von Mises (bottom right) tests (claim year 1997).

For completeness, we attach the graphical assessment of goodness of the fit for threshold equal to 400 000 (for theoretical description of the graphical diagnostic see Subsection 2.3.4). Figure 2.8 and Figure 2.9 depict the probability plot, the quantile-quantile plot, the density plot and the return level plot for maximum likelihood and probability weighted moments approach. Note that we do not display the results for the penalized maximum likelihood estimator since for threshold 400 000 the scale and the shape parameters do not differ from those maximum likelihood based. Same parameters generate the same distribution and the whole graphical diagnostic is thus identical. The probability plot displays the empirical distribution function evaluated at the exceedances over threshold 400 000 against the model based distribution function evaluated at the same points. The fit of the distribution is accurate when the points are situated close enough to the quadrant axis. In both figures the quadrant axis (the red guideline) can be considered as a satisfactory approximation for plotted points. The quantile-quantile plot provides similar description of the fitted model as the probability plot, the plotted points correspond to combinations of the empirical quantiles and the model based quantiles. Hence the quadrant axis again expresses the perfect fit. The quantile-quantile plots for our data set confirm that the estimated model is adequate. Similar conclusion can be made by assessing the density plot which compares the model based density and the empirical density accompanied by the histogram. The last graphical diagnostic used was the return level plot which depicts empirical quantiles and the model based quantiles both against $\log y_p$ where $y_p = -log(1-p)$ and p is the reciprocal value of the return level. In Figures 2.8 and 2.9 the model based quantiles are related to the red curve whereas the empirical quantiles to the black points. We can claim that the red guideline is an adequate approximation for the dependence of the empirical quantiles on $\log y_p$. Moreover,

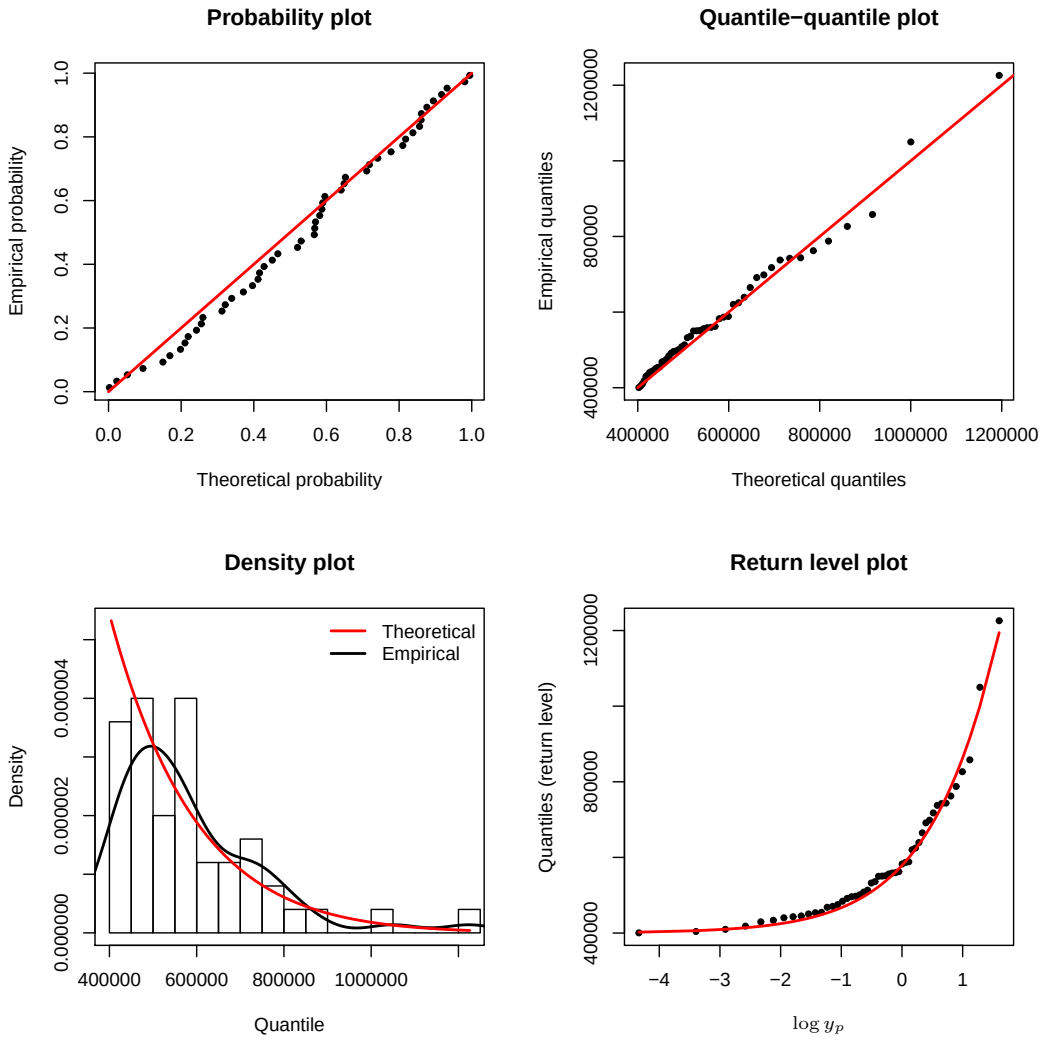

Figure 2.8: Graphical diagnostic for a fitted generalized Pareto model using the maximum likelihood model or the penalized maximum likelihood model equivalently (claim year 1997, threshold 400 000).

from the return level plot a particular type of the generalized Pareto distribution can be recognized. The convex guideline suggests that the excesses have beta distribution which corresponds to negative value of the shape parameter estimated for threshold 400 000.

The data analysis showed that the modeling of the tail behaviour by generalized Pareto distribution is appropriate. Although the goodness of the fit of the distribution was justified by statistical tests for all threshold choices and estimating procedure, we suggest to utilize

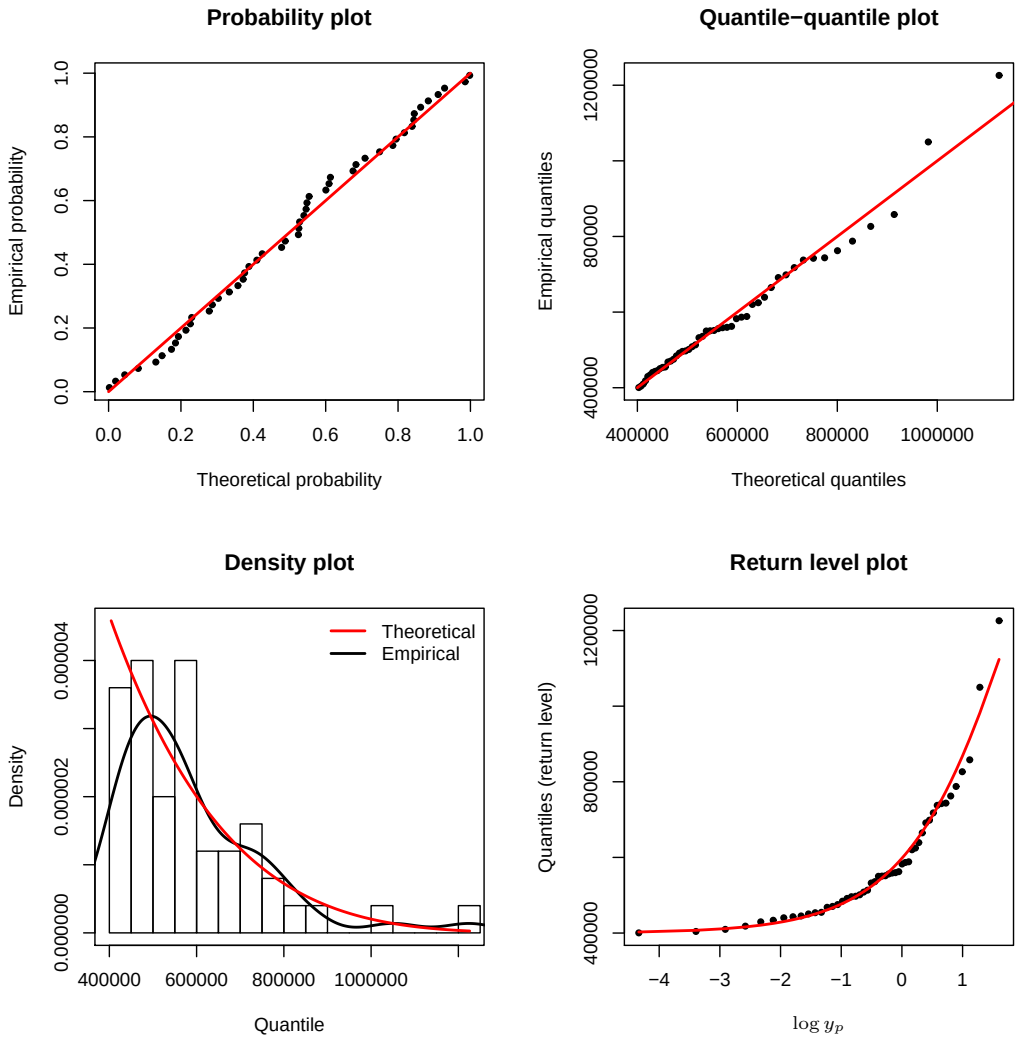

Figure 2.9: Graphical diagnostic for a fitted generalized Pareto model using the probability weighted moments model (claim year 1997, threshold 400 000).

shape and scale parameters computed on the basis of the probability weighted moments estimator for higher values of threshold level, i.e. for 600 000 and 700 000. As it was noted above, for small sample sizes the probability weighted moments estimator performs better in terms of bias and mean square error than the maximum likelihood estimator, in particular the estimate of an extreme quantile q_p can be significantly biased, especially when ξ is positive. For threshold equal to 400 000 one can decide for particular estimating procedure arbitrarily. In conclusion, if one particular choice of the generalized Pareto distribution has to

be made, we recommend to adopt the threshold level of 400 000. We believe that the number of exceedances over this threshold is adequate to construct reasonable fit. Higher thresholds generate few observation and thus the estimated model can suffer from inaccuracies.

Fitting the generalized Pareto distribution and assessment of goodness of the fit are implemented in program R via procedures
`procedure_gpd_inference`. This procedure provides estimation of the parameters of the distribution together with their onfidence intervals, to achieve this target we utilize built function `fitgpd`. Statistically, the goodness of the fit is assessed with three tests, Anderson-Darling (`ad.test`), Kolmogorov-Smirnov (`ks.test`) and Cramer-von Mises (`cvmts.test`). The graphical illustration of the goodness of the fit is provided through procedures

- `procedure_pgpd_plot`,
- `procedure_dgpd_plot`,
- `procedure_qqgpd_plot`,
- `procedure_retlevgpd_plot`.

```
# plot distribution function of generalized Pareto distribution
procedure_pgpd_plot <- function (data, u, scale_est, shape_est){

  data_cens <- data [data > u]
  data_cens <- sort (data_cens)
  n <- length (data_cens)

  prob_vec <- (seq (1, n, by=1)-0.35)/n
  p_vec <- pgpd (data_cens-u, scale=scale_est, shape=shape_est)

  plot (p_vec, prob_vec, pch=20,
        xlab="Theoretical probability", ylab="Empirical probability",
        main="Probability plot")
  lines (c (0, 1), c (0, 1), lwd=2, col="red")
}

# plot density function of generalized Pareto distribution accompanied
# by histogram
procedure_dgpd_plot <- function (data, u, scale_est, shape_est){

  k <- 200
  data_cens <- data [data > u]
  n <- length (data_cens)

  data_cens_fit <- seq (u, max (data_cens), length = k+1) - u
```

```
  data_cens_fit <- data_cens_fit [2:length (data_cens_fit)]

  par_breaks <- 0
  if (n < 25){par_breaks <- n} else {par_breaks <- 25}

  histogram <- hist (data_cens, breaks=par_breaks, plot=FALSE, freq=TRUE)
  histogram$counts <- histogram$counts / (diff (histogram$mids [1:2])*n)
  dens_vec <- dgpd (data_cens_fit, scale=scale_est, shape=shape_est)

  plot (histogram, xlab="Quantile", ylab="Density", main="Density plot",
        xlim=c (u, max (data_cens)), ylim=c (0, max (dens_vec)))
  lines (density (data_cens), col="black", lwd=2)
  lines (data_cens_fit+u, dens_vec, col="red", lwd=2)
  box(lty="solid")
  legend ("topright", bty = "n", c("Theoretical","Empirical"),lwd=c(2,2),
          col=c("red","black"), horiz=FALSE)
}

# plot quantile-quantile plot of generalized Pareto distribution
procedure_qqgpd_plot <- function (data, u, scale_est, shape_est, name){

  data_cens <- data [data > u]
  n <- length (data_cens)

  prob_vec <- (seq (1, n, by=1)-0.35)/n
  qt_vec <- qgpd (prob_vec, scale=scale_est, shape=shape_est)+u

  plot (qt_vec, sort(data_cens), pch=20, xlab="Theoretical quantiles",
        ylab="Empirical quantiles", main=name)
  lines (c (u, max (data_cens, qt_vec)), c (u, max (data_cens, qt_vec)),
         lwd=2, col="red")
}

# depict return level plot of generalized Pareto distribution
procedure_retlevgpd_plot <- function (data, u, scale_est, shape_est){

  data_cens <- data [data > u]
  n <- length (data_cens)

  prob_vec <- (seq (1, n, by=1)-0.35)/n
  ret_per <- log (-log (1-prob_vec))
  ret_lev <- qgpd (prob_vec, scale=scale_est, shape=shape_est)+u
```

```
  label <- expression (paste ("log ", y[p]))
  plot (ret_per, sort (data_cens), pch=20, xlab=label,
        ylab="Quantiles (return level)", main="Return level plot")
  lines (ret_per, ret_lev, col="red", lwd=2)
}

# fit generalized Pareto distribution and access goodness of the fit
procedure_gpd_inference <- function (data, u, method, only_est){

  # fitting generalized pareto distribution
  gpd_fit <- fitgpd (data, threshold=u, method)
  scale_est <<- gpd_fit$param [1]
  shape_est <<- gpd_fit$param [2]

  # if parameter only_est is set to be 1 (TRUE) then procedure is
  # dedicated only to estimate distribution function parameters and
  # breaks at the following line
  if (only_est==1){}
  else {
    # confidence intervals for fitted parameters
    conf_int_shape <- gpd.fishape (gpd_fit, 0.95)
    conf_int_scale <- gpd.fiscale (gpd_fit, 0.95)

    # checking the model (statistical approach: Anderson-Darling,
    # Kolmogorov-Smirnov, Cramer-von Mises tests)
    data_cens <- data [data>u]
    data_cens_adj <- data_cens-u
    n <- length (data_cens)
    data_gener <- rgpd (n, scale=scale_est, shape=shape_est)

    ad_test_p <- ad.test (data_cens_adj, pgpd, shape=shape_est,
                          scale=scale_est)$p.value
    ks_test_p <- ks.test (data_cens_adj, pgpd, shape=shape_est,
                          scale=scale_est)$p.value
    cvm_statistics <- cvmts.test (data_gener, data_cens_adj)
    cvm_test_p <- cvmts.pval (cvm_statistics, n, n)

    # checking the model (graphical approach)
    par (mfrow = c (2,2))
    options("scipen" = 20)

    procedure_pgpd_plot (data, u, scale_est, shape_est)
    procedure_qqgpd_plot (data, u, scale_est, shape_est,
```

```
                          "Quantile-quantile plot")
    procedure_dgpd_plot (data, u, scale_est, shape_est)
    procedure_retlevgpd_plot (data, u, scale_est, shape_est)

    # results (vector of length 8)
    c (scale_est [[1]], conf_int_scale [[1]], conf_int_scale [[2]],
    shape_est [[1]], conf_int_shape [[1]], conf_int_shape [[2]],
    ad_test_p [[1]], ks_test_p, cvm_test_p)
  }
}

# find number of observation above specified threshold (threshold set
# as vector)
procedure_data_length <- function (data, threshold_vec){

  data_length <- rep (0, length (threshold_vec))

  for (i in 1:length (threshold_vec)){
    data_length [i] <- length (data [data >= threshold_vec [i]])
  }

  data_length
}

method <- c ("mle", "pwmb", "mple")

# generation column and row names for result matrix
col_names <- c ("scale", "scale_inf", "scale_sup", "shape", "shape_inf",
                "shape_sup", "ad_test", "ks_test", "cvm_test")
row_names <- c (sapply (method, function (y) sapply (threshold,
                function (x) paste (y, "_", x, sep=""))))
result_matrix_claims <- matrix (0, nrow=length (row_names),
                                ncol=length (col_names),
                                dimnames=list (row_names, col_names))

# execute GPD inference for all choices of thresholds and all calculation
# methods
for (i in 1:length (method)){
  for (j in 1: length (threshold)){
    result_matrix_claims [3*(i-1)+j,] <- procedure_gpd_inference (claims,
                                         threshold [j], method [i], 0)
  }
}
```

```
# result matrix
result_matrix_claims <- round (result_matrix_claims, digit=4)
result_matrix_claims

# find number of observation above specified thresholds
claims_length <- procedure_data_length (claims, threshold)
claims_length
```

The estimates of the parameters of the fitted distribution accompanied by their confidence intervals and p-values of Kolmogorov-Smirnov test, Anderson-Darling test and Cramer-von Mises test are stated in table `results_matrix_claims`, where the matrix rows correspond to different choices of threshold level and inference procedure:

```
               scale      scale_inf  scale_sup  shape       shape_inf
mle_400000     183913     113914     253911     -0.0572     -0.3186
mle_600000     174424     150547     198301     -0.0607     -0.3918
mle_700000     145182      14828     275535      0.0493     -0.6011
pwmb_400000    183913     113914     253911     -0.0572     -0.3186
pwmb_600000    174424      57283     291565     -0.0607     -0.5210
pwmb_700000    145182     121910     168454      0.0000     -0.0859
mple_400000    214691     124034     305348     -0.1674     -0.5076
mple_600000    207588      52591     362586     -0.1901     -0.7991
mple_700000    116705       3560     229851      0.1961     -0.5512

               shape_sup  ad_test    ks_test    cvm_test
mle_400000     0.2041     0.7799     0.8183     0.1871
mle_600000     0.2704     0.7801     0.7316     0.8657
mle_700000     0.6996     0.8581     0.9445     0.5794
pwmb_400000    0.2041     0.7799     0.8183     0.9870
pwmb_600000    0.3995     0.7801     0.7316     0.3842
pwmb_700000    0.0859     0.8750     0.9596     0.5794
mple_400000    0.1729     0.9733     0.9671     0.6521
mple_600000    0.4188     0.8522     0.9513     0.9516
mple_700000    0.9434     0.9117     0.8953     0.5794
```

We repeated the same computational procedure for determining the parameters for the generalized Pareto distribution also for the remaining claim years 1998 and 1999. As a starting point we adopted the threshold level suggested earlier in part 2.4.2, i.e. for the claim year it is 300 000. However for the claim year 1999 the hypothesis that the exceedances over 300 000 have generalized Pareto distribution was rejected, thus we adjusted the threshold level to be in accordance with the generalized Pareto distribution. Table 2.8 summarized the estimated parameters of the distribution for all claim years and particular choices of threshold levels. We present the results obtained by applying the probability weighted moments

estimator only since two higher thresholds generate few observation for each claim year, thus the probability weighted moments based estimates are considered to be more accurate. For estimating the parameters for the lowest threshold choice an arbitrary estimating procedure could be used.

u	σ_u	ξ
400000	214691	-0.1674
600000	207588	-0.1901
700000	116705	0.1961

Claim year 1997

u	σ_u	ξ
300000	106967	0.1606
600000	103532	0.3566
700000	141152	0.3299

Claim year 1998

u	σ_u	ξ
500000	192410	0.3459
700000	221663	0.4577
1000000	973970	-0.3248

Claim year 1999

Table 2.8: Parameters of the generalized Pareto distribution for the claim years 1997, 1998 and 1999 estimated using the probability weighted moments approach.

Comparison with standard parametric fit

To check the improvement achieved by using the generalized Pareto distribution instead of a classical parametric model, one can adopt the methods introduced in Subsection 2.3.4. In this section we provide a comparison based on the quantile-quantile plot and Kolmogorov-Smirnov test. First let us recall the results from Subsection 2.4.1, i.e. the logarithmic data can be approximated by normal distribution. In other words, we assume that the data are lognormally distributed with specific parameters. Further we assume that the distribution of exceedances over a high threshold can be well described by generalized Pareto distribution with specific parameters. We aim to compare goodness-of-fit of lognormal approximation and generalized Pareto approximation taking into account a tail fraction given by different choices of the threshold level.

We provide a brief summary of an analysis carried out on the claim year 1997. Computational results from previous parts are exploited in order to model the tail behaviour of the data, namely we assumed that the logarithmic data are normally distributed with mean equal to 5.82 and standard deviation 1.666 (see Table 2.2). Further we considered three threshold levels (400 000, 600 000 and 700 000) and assumed that exceedances over these thresholds have generalized Pareto distribution with parameters as stated in Table 2.9.

	Threshold u		
	400000	600000	700000
σ_u	214691	207588	116705
ξ	-0.1674	-0.1901	0.1961

Table 2.9: Estimates of the scale and shape parameters for different choices of the threshold using the probability weighted moments approach (claim year 1997).

Figure 2.10 depicts quantile-quantile plots for the fitted lognormal model and generalized Pareto model for different threshold choices. Comparing the graphs corresponding to a particular threshold level we can state that the lognormal approximation is a poor competitor to the generalized Pareto approximation. Kolmogorov-Smirnov tests confirms conclusions made from the graphical approach. As indicated in Table 2.10 p-values of the tests assess-

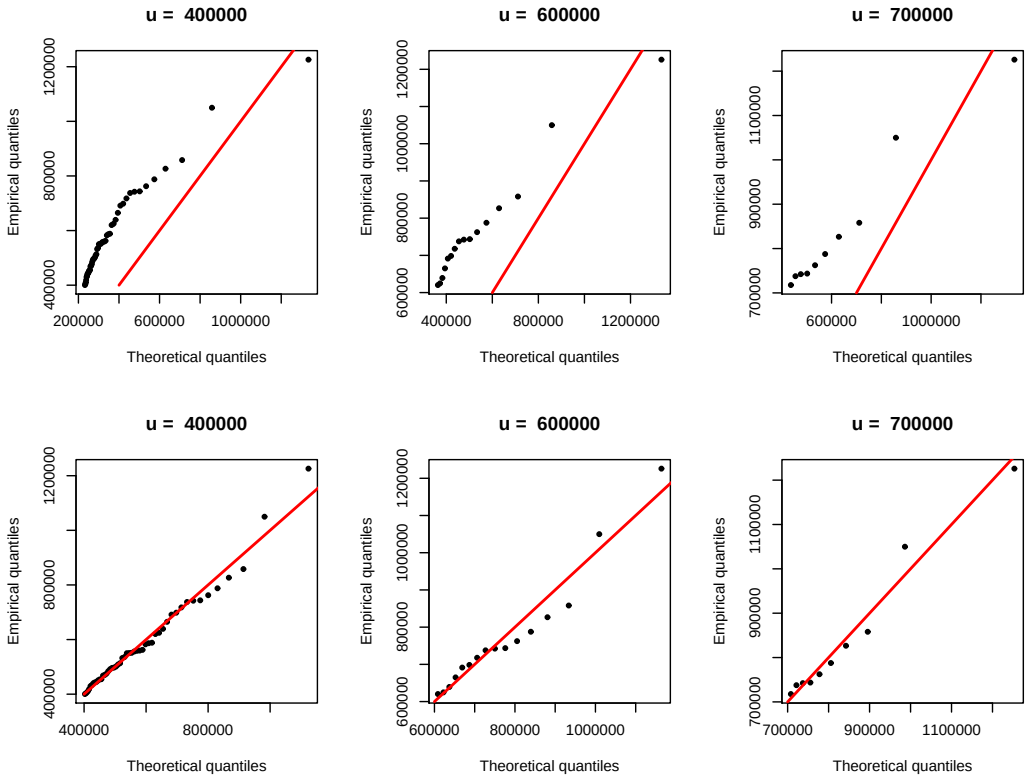

Figure 2.10: Quantile-quantile plots for a fitted lognormal model (top) and generalized Pareto model (bottom) for different threshold levels (claim year 1997).

ing adequacy of the lognormal distribution were considerably close to 0 whereas p-values of testing the generalized Pareto distribution exceeded the confidence level 0.05 in all cases.

	Threshold u		
	400000	600000	700000
`lnorm`	0.0000	0.0000	0.0000
`gpd`	0.9671	0.9513	0.8953

Table 2.10: P-values of Kolmogorov-Smirnov test for a fitted lognormal model and generalized Pareto model for different threshold levels (claim year 1997).

The above stated analysis can be implemented using program R as follows:

```
# plot quantile-quantile plot of log-normal distribution
procedure_qqlnorm_plot <- function (data, u, log_mean_est, log_sd_est,
                                    name){

  data_cens <- data [data > u]
  n <- length (data_cens)

  prob_vec <- (seq (length (data)-n+1, length (data), by=1)-0.35)/
              length (data)
  qt_vec <- qnorm (prob_vec, mean=log_mean_est, sd=log_sd_est)
  qt_vec <- exp (qt_vec)

  plot (qt_vec, sort(data_cens), pch=20, xlab="Theoretical quantiles",
        ylab="Empirical quantiles", main=name)
  lines (c (u, max (data_cens, qt_vec)), c (u, max (data_cens, qt_vec)),
         lwd=2, col="red")
}

# procedure comparing goodnaess of the fit of lognaormal and generalized
# Pareto distribution
procedure_compare_models <- function (data, threshold_vec, method){

  par (mfrow = c (2,3))
  ks_matrix <- matrix (0, nrow = 2, ncol=3)

  # fitting lognormal distribution
  sample_length <- 10000
  log_data <- log (sample (data, sample_length, replace = FALSE,
                   prob = NULL))
```

```
  procedure_fitting_distribution (log_data, 1)

  # quantile-quantile plot corresponding to fitted log-normal
  # distribution
  for (i in 1:length (threshold_vec)){
    procedure_qqlnorm_plot (data, threshold_vec [i], log_mean_est,
                            log_sd_est, paste ("u = ",threshold_vec[i]))
    data_cens <- data [data > threshold_vec [i]]
    data_cens_adj <- log (data_cens)

    ks_matrix [1, i] <- ks.test (data_cens_adj, pnorm, mean=log_mean_est,
                                 sd=log_sd_est)$p.value
  }

  for (i in 1:length (threshold_vec)){
    # fitting gpd distribution
    procedure_gpd_inference (data, threshold_vec [i], method, 1)

    # quantile-quantile plot corresponding to fitted generalized Pareto
    # distribution
    procedure_qqgpd_plot (data, threshold_vec [i], scale_est, shape_est,
                          paste ("u = ", threshold_vec [i]))

    data_cens <- data [data>threshold_vec [i]]
    data_cens_adj <- data_cens-threshold_vec [i]
    ks_matrix [2, i] <- ks.test (data_cens_adj, pgpd, shape=shape_est,
                                 scale=scale_est)$p.value
  }

  ks_matrix
}

# compare goodness of the fit of log-normal and generalized Pareto
# distribution
ks_matrix <- procedure_compare_models (claims, threshold, "pwmb")
ks_matrix <- round (ks_matrix, digit=4)
ks_matrix
```

The resulting statistical tests, `ks_matrix`, are printed in the following form (rows correspond to generalized Pareto distribution and lognormal distribution, whereas columns correspond to different choices of threshold):

```
                400000     600000     700000
lnorm   0.0000     0.0000     0.0000
```

```
gpd     0.9671     0.9513     0.8953
```

From demonstrated results, we can conclude that no satisfactory fit for the extremal events is obtained using a classical parametric model. Contrary to the generalized Pareto distribution, the fit does not significantly improve when increasing the threshold. Therefore the results highly support the use of the generalized Pareto models instead of the traditional parametric models and demonstrate the interest of extreme value theory when the concern lies in the right tail.

2.4.3 Applications of the model

In this chapter, we outline the practical utilization of the generalized Pareto distribution to model the behaviour of the exceedances over high thresholds for solving some actuarial issues. We focus on two applications, point estimation of high quantiles and prediction of probable maximum loss.

Point estimation of high quantiles

The high quantiles of the distribution of the claim amounts is a measure that provides useful information for insurance companies. In other fields of statistics quantiles are usually estimated by their empirical counterparts (see, for instance, Table 2.1 in Subsection 2.4.1). However when one is interested in extremal quantiles, this approach is not longer valid since estimation based on a low number of large observations tempts to be strongly imprecise. Thus, in such cases usage of the generalized Pareto model is preferred.

Let us denote as $F^{(u)}$ the common cumulative distribution function of $X-u$, conditional on $X > u$. From Formula (2.6), we see that a potential estimator for the distribution $F^{(u)}$ is $H^{(u)}$, provided u is sufficiently large. In other words, $H^{(u)}$ approximates the conditional distribution of the losses, given that they exceed the threshold u. Thus quantile estimators derived from this function (given by Equation (2.26)) are conditional quantile estimators that indicate the scale of losses that could be experienced if the threshold u was to be exceeded. To estimate the high unconditional quantiles, we aim relate the unconditional cumulative distribution function F to the conditional cumulative distribution function $F^{(u)}$. Denoting $\bar{F}(y) = 1 - F(y)$, we obtain:

$$\bar{F}^{(u)}(y) = \mathbb{P}\left[X - u \geqslant y \mid X > u\right] = \frac{\bar{F}(u+y)}{\bar{F}(u)}, \qquad y > 0.$$

Applying the generalized Pareto approximation for the distribution of the exceedances, we have:

$$\bar{F}(u+y) \approx \bar{F}(u)\left(1 - H^{(u)}(y)\right), \qquad y > 0.$$

Provided we have a large enough sample, we can estimate $\bar{F}(u)$ by its empirical counterpart,

i.e., $\bar{F}(u) = N_u/n$ where N_u and n are the number of claims above the threshold u and the total number of claims, respectively. To sum up, we can estimate the probability that the claim amount is larger than y, for any amount $y > u$ by

$$\hat{\bar{F}}(y) = \frac{N_u}{n}\left(1 - \hat{H}^{(u)}(y-u)\right) = \frac{N_u}{n}\left(1 + \hat{\xi}\,\frac{y-u}{\hat{\sigma}_u}\right)^{-1/\hat{\xi}}.$$

Substituting $F(q_p) = 1 - p$ to the latter equation, we obtain the following estimator of the quantile:

$$\hat{q}_p = u - \frac{\hat{\sigma}_u}{\hat{\xi}}\left(1 - \left(\frac{pn}{N_u}\right)^{-\hat{\xi}}\right). \tag{2.32}$$

Figure 2.11 displays the generalized Pareto quantiles based on the shape and scale parameters estimated for threshold 400 000 against the empirical quantiles. According to the graph these quantiles rather agree. Note that the total number of observation n exceeds 1 200 000 whereas threshold level 400 000 generates only 50 observation, thus the ratio n/N_u is considerably high and we have to set probability p proportionally low in order to obtain reasonable quantiles.

Probable maximum loss

Another useful measure in the field of insurance is the probable maximum loss which can be interpreted as the worst loss likely to happen. Generally, the probable maximum loss can be obtained by solving equation

$$\mathbb{P}\left[M_n \leqslant PML_p\right] = 1 - p$$

for some small p and PML_p denotes the probable maximum loss. This means that the probable maximum loss is a high quantile of the maximum of a random sample of size n thus the solution is given as:

$$PML_p = F^{-1}_{M_n}(1-p),$$

i.e., the probable maximum loss is computed as the $1-p$ quantile of the distribution of the maximum loss. We will use an approach based on the generalized Pareto approximation of the exceedances. Under the condition that the exceedances over threshold u can be approximated by the generalized Pareto distribution, the number N_u of exceedances over that threshold is roughly Poisson (for more details we refer to Cebrián et al. (2003)). In that situation, it can be proved that the distribution of the maximum above the threshold M_{N_u} of these N_u exceedances can be approached by a generalized extreme value distribution.

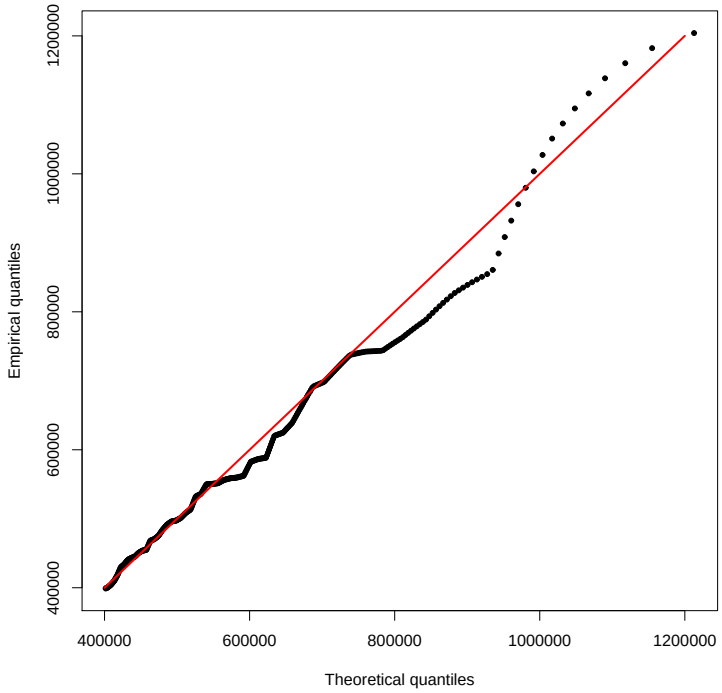

Figure 2.11: Generalized Pareto quantiles against their empirical analogues (claim year 1997, threshold 400000).

More precisely, consider a random variable N_u distributed according the Poisson law with mean λ, and let $X_1 - u, \ldots, X_{N_u} - u$ be a sequence of N_u independent and identically distributed random variables with common cumulative distribution function $H^{(u)}$,. Then for $M_{N_u} = \max\{X_1, \ldots, X_{N_u}\} - u$ the cummulative dstribution function is given as:

$$\mathbb{P}\left[M_{N_u} \leqslant x\right] \approx G(x)$$

with the following modification of location and scale parameters:

$$\begin{aligned} \mu &= \frac{\sigma_u}{\xi}\left(\lambda^{\xi} - 1\right), \\ \sigma &= \sigma_u \lambda^{\xi}. \end{aligned}$$

Using this distribution and Formula (2.3), the previous probable maximum loss definition results in:

$$PML_p = u + \frac{\sigma_u}{\xi}\left(\left(-\frac{\lambda}{\log(1-p)}\right)^{\xi} - 1\right).$$

Thus when estimating the probable maximum loss for particular p we can use the following formula:

$$PML_p = u + \frac{\hat{\sigma}_u}{\hat{\xi}}\left(\left(-\frac{\hat{\lambda}}{\log(1-p)}\right)^{\hat{\xi}} - 1\right),$$

where $\hat{\lambda} = N_u/n$.

Figure 2.12 shows the probable maximum loss for different probability levels. Similarly as in the previous subsection, p has to be chosen sufficiently low to balance the estimate of λ.

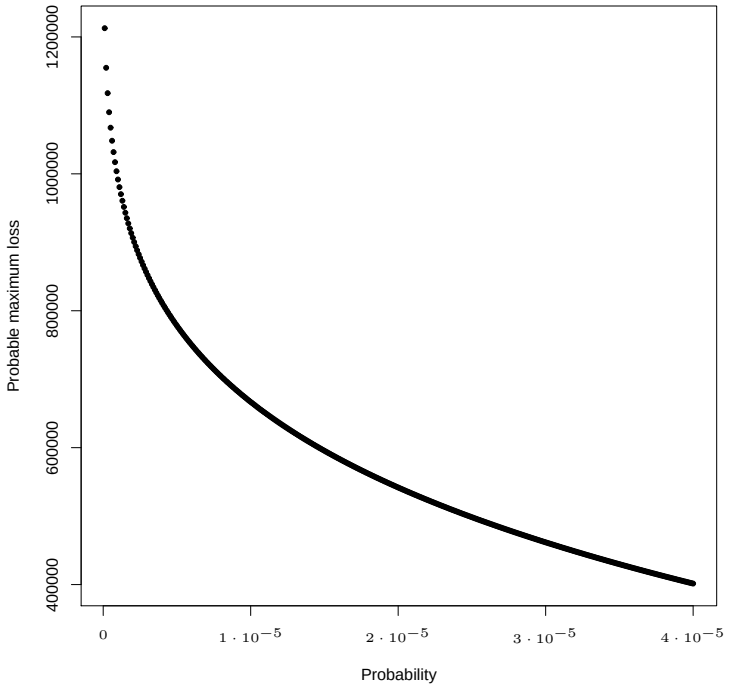

Figure 2.12: Probable maximum loss (claim year 1997).

The above application of the generalized Pareto distribution fit can be obtained by executing the following R code.

```
# plot quantile-quantile plot of generalized Pareto distribution with
# detailed emphasis on the tail
procedure_qqgpd_plot_mod <- function (data, u, scale_est, shape_est){

  data_cens <- data [data > u]
  n_cens <- length (data_cens)
  n <- length (data)

  prob_vec <- seq (0.0000001, 0.00004, by=0.0000001)
  prob_vec_modif <- 1-prob_vec
  qt_vec <- (((n/n_cens)*prob_vec)^(-shape_est)-1)*
            (scale_est/shape_est) + u
  qt_vec_emp <- quantile (data, probs=prob_vec_modif)

  plot (qt_vec, qt_vec_emp, pch=20, xlab="Theoretical quantiles",
        ylab="Empirical quantiles", main="")
  lines (c (400000, 1200000), c (400000, 1200000), lwd=2, col="red")
}

# plot probable maximum loss of generalized Pareto distribution
procedure_pml_plot <- function (data, u, scale_est, shape_est){

  data_cens <- data [data > u]
  n_cens <- length (data_cens)
  n <- length (data)
  lambda <- n_cens/n

  prob_vec <- seq (0.0000001, 0.00004, by=0.0000001)
  prob_vec_modif <- 1-prob_vec
  pml <- u + (scale_est/shape_est)*((-lambda/
         log (prob_vec_modif))^shape_est-1)

  plot (prob_vec, pml, pch=20, xlab="Probability",
        ylab="Probable maximum loss", main="")
}

# application of generalized Pareto distribution
procedure_gpd_application <- function (data, u, method){

  # fitting generalized pareto distribution
```

```
  procedure_gpd_inference (data, u, method, 1)

  par (mfrow=c (1, 1))
  # plot quantile-quantile plot for fitted generalized Pareto
  #distribution
  procedure_qqgpd_plot_mod (data, u, scale_est, shape_est)
  # plot probable maximum loss of generalized Pareto distribution
  procedure_pml_plot (data, u, scale_est, shape_est)
}

method <- "pwmb"

# print results
for (j in 1: length (threshold)){
  procedure_gpd_application (claims, threshold [j], method)
}
```

2.5 Conclusion

The presented study has shown the usefulness of extreme value theory for analysis of an insurance portfolio. Indeed, this theory undoubtedly allows to determine distribution of the large losses, high quantiles, and probable maximum loss. However, the generalized Pareto distribution is valid only above very high thresholds, thus the estimation of the optimal threshold level seems to be a crucial issue. We introduced the graphical threshold selection including the mean residual life plot, the threshold choice plot and the L-moments plot. We found out that the graphical approach is ambiguous since each of the plots might prefer a different threshold choice. In other words, we were not able to choose a single threshold level for either of the data sample. In the end we prioritized the lowest threshold suggested by the mean residual life plot since it ensured generating a reasonable amount of exceedances available for further inference. To estimate the shape and scale parameters of the generalized Pareto distribution we applied the maximum-likelihood, penalized maximum-likelihood and probability weighted moments methods. We observed slight difference in the estimated parameters for each of the methods. However the sign of the shape parameter remained consistent with varying the estimation procedure, thus the specific shape of the generalized Pareto distribution was independent of the applied method. We emphasize that for small sample sizes, in terms of bias and mean square error, the probability weighted moments estimator performs better than the maximum likelihood estimator. The lowest threshold levels generated sufficiently enough observations to consider the maximum likelihood estimates to be accurate. For higher thresholds only few observation were available, therefore in these cases one should rely on the probability weighted moments estimator. The estimates of the

shape and scale parameters based on the penalized maximum likelihood method were very close to those estimated by the maximum likelihood methods. This fact can be attributed to the specific behaviour of the penalty function, for all selected threshold levels its values were equal or very close to 1 and thus the penalized likelihood function was approximately the same as the likelihood function and their maximization led to similar estimates. In the next step of the estimation procedure we checked goodness of the fit. Although the Kolmogorov-Smirnov and Cramer-von Mises tests provided satisfactory results, our conclusions were derived mainly from p-values of the Anderson-Darling which seems to perform better than the previously mentioned ones. We found out that the generalized Pareto distribution with corresponding parameters can be considered as a valid model for all predetermined threshold levels and all estimating methods. At the end of the computational part we showed that no satisfactory fit for the extreme events can be obtained using a classical parametric model, and thus the generalized Pareto distribution performs better when modeling large claims.

2.6 Source code

```
# clear global environment
rm (list = ls (all=TRUE))

#rarchive<-"https://cran.r-project.org/src/contrib/Archive/"
#lP<-"POT/POT_1.1-3.tar.gz"
#linkP<-paste0(rarchive,lP)
#install.packages(linkP, repos=NULL, method="libcurl")
#lC<-"CvM2SL2Test/CvM2SL2Test_2.0-1.tar.gz"
#linkC<-paste0(rarchive,lP)
#install.packages(linkC, repos=NULL, method="libcurl")
#install.packages(c("MASS","stats","lmomco","ADGofTest"))
#install.packages(c("xtable","e1071"))

library (POT)
library (CvM2SL2Test)
library (MASS)
library (stats)
library (lmomco)
library (ADGofTest)
library (xtable)
library (e1071)


##### declaration of procedures
```

```
# read database
procedure_read_database <- function (dir, name){

  setwd (dir)
  database_inic <- read.table (file = name, header = TRUE,
                               sep = ",", dec = ".")
  claims <- database_inic [ ,17]

  claims
}

# fit ordinary distribution to data using maximum likelihood approach
# (fit normal distribution to logarithmic data)
procedure_fitting_distribution <- function (data, only_est){

  # function fitdistr uses maximum likelihood estimator for fitting
  # standard distributions
  fit <- fitdistr (data, "normal")

  log_mean_est <<- fit$estimate[1]
  log_sd_est <<- fit$estimate[2]

  # if parameter only_est is set to be 1 (TRUE) then procedure is
  # dedicated only to estimate distribution function parameters and
  # breaks at the following line
  if (only_est==1){}
  else {
    # testing accuracy of lognormal disribution (n = number
    # of repetitions of Shapiro-Wilk test)

    n <- 10
    p_value_sw <- rep (0, n)

    for (k in 1:n){
      data_test <- sample (data, 50, replace=FALSE, prob=NULL)
      test_sw <- shapiro.test (data_test)
      p_value_sw [k] <- test_sw$p.value
    }

    c (log_mean_est, log_sd_est, p_value_sw)
  }
}
```

```
# calculate descriptive statistics (number of observations (data length),
# sum of claims, min. claim, 1st quantile, median, mean, 3rd quantile,
# max. claim)
procedure_descriptive_stat <- function (data){

  c (length (data), sum (data), summary (data))
}

# graphical illustration of descriptive statistics (box plot of claims
# and logarithmic claims), graphical goodness of fit (theoretical and
# empirical density, distribution and quantile-quantile plot)
procedure_plot_descriptive_stat <- function (data, log_mean_est,

                    log_sd_est){

  log_data <- log (data)

  # box plot
  par (mfrow = c (1,2))

  # box plot of claims
  boxplot (data, axes=FALSE, pch=20, ylim=c(0,15000), xlab="Claims",
           ylab="Range")
  lines (c (0.5, 1.5), c (mean (data), mean (data)), col="red", lw=2)
  axis (1, at=seq (0.5, 1.5, by=1), labels=c ("",""))
  axis (2, at=seq (0, 15000, by=5000))

  # box plot of logarithmic claims
  boxplot (log_data, axes=FALSE, pch=20, ylim=c(-5,15),
           xlab="Logarithmic claims", ylab="Range")
  lines (c (0.5, 1.5), c (mean (log_data), mean (log_data)), col="red",
         lw=2)
  axis (1, at=seq(0.5, 1.5, by=1), labels=c ("",""))
  axis (2, at=seq (-5, 15, by=5))

  # goodness of fit plots
  par(mfrow = c (2,2))

  n <- 200
  log_data_fit <- seq(0, max (log_data), length = n)

  # histogram accompanied by theoretical and empirical densities
  histogram <- hist (log_data, breaks=25, plot=FALSE)
```

```
  histogram$counts <- histogram$counts / (diff (histogram$mids [1:2])*
                      length (log_data))

  data_norm <- dnorm (log_data_fit, mean = log_mean_est, sd = log_sd_est)
  plot (histogram, xlab="Logarithmic claims", ylab="Histogram and
        probability density", main="", axes=FALSE, xlim=c(-5,15),
        ylim=c(0,0.25))
  lines (density (log_data), col="black", lwd=2)
  lines (log_data_fit, data_norm, col="red", lwd=2)
  axis(1, at=seq (-5, 15, by=5))
  axis(2, at=seq (0, 0.25, by=0.05))
  legend (-5, 0.25, inset=.1, bty = "n", c("Normal","Empirical"),
          lwd=c(2,2) , col=c("red","black"), horiz=FALSE)

  # theoretical and empirical distribution functions
  plot (log_data_fit, pnorm (log_data_fit, mean = log_mean_est,
        sd = log_sd_est), type="l", col="red", lwd=2,
        xlab="Logarithmic claims", ylab="Cumulative probability",
        main="", axes=FALSE, xlim=c(-5,15), ylim=c(0,1))
  plot (ecdf (log_data), cex=0.01, col="black", lwd=2, add=TRUE)
  axis (1, at=seq (-5, 15, by=5))
  axis (2, at=seq (0, 1, by=0.25))
  legend (-5, 1, inset=.1, bty = "n", c("Normal","Empirical"),lwd=c(2,2),
          col=c("red","black"), horiz=FALSE)

  # quantile-quantile plot
  qqplot (rnorm (n, mean = log_mean_est, sd = log_sd_est), pch=20,
          log_data, xlab="Theoretical quantiles of normal distribution",
          ylab="Empirical quantiles of logarithmic claims", main="",
          axes=FALSE, xlim=c(0,15), ylim=c(0,15))
  lines (c (0, 15), c (0,15), lwd=2, col="red")
  axis (1, at=seq (0, 15, by=3))
  axis (2, at=seq (0, 15, by=3))
}

# plot quantile-quantile plot of log-normal distribution
procedure_qqlnorm_plot <- function (data, u, log_mean_est, log_sd_est,
                                    name){

  data_cens <- data [data > u]
  n <- length (data_cens)

  prob_vec <- (seq (length (data)-n+1, length (data), by=1)-0.35)/
```

```
              length (data)
  qt_vec <- qnorm (prob_vec, mean=log_mean_est, sd=log_sd_est)
  qt_vec <- exp (qt_vec)

  plot (qt_vec, sort(data_cens), pch=20, xlab="Theoretical quantiles",
        ylab="Empirical quantiles", main=name)
  lines (c (u, max (data_cens, qt_vec)), c (u, max (data_cens, qt_vec)),
         lwd=2, col="red")
}

# threshold selection: the empirical mean residual life plot
procedure_mrlplot <- function (data, limit, conf_level){

  par (mfrow = c (1, 1))

  # built function (package evd) for the construcion of the empirical
  # mean residual life plot
  mrlplot (data, u.range=c (limit, quantile (data, probs=conf_level)),
           xlab="Threshold", ylab="Mean excess", main="", lwd=2, lty = 1,
           col = c ("red","black","red"), nt=200)
}

# threshold selection: threshold choice plot (dependence of parameter
# estimates at various thresholds)
procedure_tcplot <- function (data, limit, conf_level){

  par (mfrow=c (1,2))

  # built function (package evd) for the construcion of threshold choice
  # plot
  tcplot (data, u.range = c (limit, quantile (data, probs=conf_level)),
          type="l", lwd=2, nt=50)
}

# threshold selection: L-moments plot (dipicts L-kurtosis against
# L-skewness)
procedure_lmomplot_est <- function (data, min, max, n){

  u_range <- seq (min, max, by=(max-min)/(n-1))

  # L-skewness as a function of L-curtosis
  fc_tau_4 <- function (x){x*(1+5*x)/(5+x)}
```

```
  # x and y represent theoretical L-kurtosis and L-skewness
  x <- seq (0, 1, by=0.01)
  y <- fc_tau_4 (x)

  # vec_tau_3 and vec_tau_4 represent empirical L-kurtosis and L-skewness
  vec_tau_3 <- rep (0, length (u_range))
  vec_tau_4 <- rep (0, length (u_range))

  for (i in 1:length (u_range)){
    u <- u_range [i]
    data <- data [data >= u]

    # bouilt function lmoms (package lmomco) computes the sample
    # L-moments
    lmom_est <- lmoms (data, nmom=4) $ratios

    vec_tau_3 [i] <- lmom_est [3]
    vec_tau_4 [i] <- lmom_est [4]
  }

  par (mfrow=c (1,1))

  label_1 <- expression (tau[3])
  label_2 <- expression (tau[4])
  plot (x, y, type="l", lwd=2, col="red", xlab=label_1, ylab=label_2)
  points (vec_tau_3, vec_tau_4, type="l", lwd=2)
}

# fit linear regression model (x ~ threshold, y ~ mean access) in order
# to estiate shape parameter
procedure_linefit <- function (data, seq_x, seq_y, threshold_value){

  n <- length (seq_x)
  w <- rep (0, n)

  for (i in 1:n){
    w [i] <- length (which (data > seq_x [i]))
  }

  position <- which (seq_x > threshold_value) [1]
  seq_x_fit <- seq_x [position : n]
  seq_y_fit <- seq_y [position : n]
  w_fit <- w [position : n]
```

```
  w_fit <- w_fit / sum (w_fit)

  # fit linear regression model using built function lm (package stats)
  par_est <- lm (seq_y_fit ~ seq_x_fit, weights=w_fit)

  par_est
}

# calculate linear regression estimates of parameters for thresholds
# values (given by variable threshold) and provide graphical illustration
# using mean residual life plot
procedure_mrlplot_linefit <- function (data, min, max, n, threshold_vec){

  u_range <- seq (min, max, by=(max-min)/(n-1))
  mean_excess <- rep (0, length (u_range))

  for (i in 1:length (u_range)){
    data <- data [data > u_range [i]]
    data_excess <- data - u_range [i]
    mean_excess [i] <- sum (data_excess)/length (data_excess)
  }

  n_threshold <- length (threshold_vec)
  par_est_matrix <- matrix (0, ncol=2, nrow=n_threshold)

  for (i in 1:n_threshold){
    # fit linear regression model
    par_est <- procedure_linefit (data, u_range, mean_excess,
                                  threshold_vec [i])
    # parameters of estimated linear regression model
    par_est_matrix [i, 1] <- par_est$coefficients [1]
    par_est_matrix [i, 2] <- par_est$coefficients [2]
  }

  # regression estimate of shape parameter
  est_ksi <- par_est_matrix [, 2]/(1+par_est_matrix [, 2])

  lin_function <- function (x, i){par_est_matrix [i, 1] +
                                  par_est_matrix [i, 2]*x}

  col_vec <- c ("red", "purple4", "blue", "green", "navy")
  name_vec <- paste ("u = ", threshold_vec, sep="")
  par (mfrow=c (1, 1))
```

```
  plot (u_range, mean_excess, type="l", lwd=2, col="black",
        xlab="Threshold", ylab="Mean excess", ylim=c(0,400000))
  for (i in 1:n_threshold){
    lines (c (min,threshold_vec[i]), c (lin_function (min, i),
           lin_function (threshold_vec[i], i)), col=col_vec [i], lw=2,
           lty=3)
    lines (c (threshold_vec[i],max), c(lin_function (threshold_vec[i],i),
           lin_function (max, i)) , col=col_vec [i], lw=2)
  }

  legend ("topleft", bty = "n", name_vec, lwd=rep (2, n_threshold),
          col=col_vec [1:n_threshold], horiz=FALSE)

  est_ksi
}

# plot distribution function of generalized Pareto distribution
procedure_pgpd_plot <- function (data, u, scale_est, shape_est){

  data_cens <- data [data > u]
  data_cens <- sort (data_cens)
  n <- length (data_cens)

  prob_vec <- (seq (1, n, by=1)-0.35)/n
  p_vec <- pgpd (data_cens-u, scale=scale_est, shape=shape_est)

  plot (p_vec, prob_vec, pch=20, xlab="Theoretical probability",
        ylab="Empirical probability", main="Probability plot")
  lines (c (0, 1), c (0, 1), lwd=2, col="red")
}

# plot density function of generalized Pareto distribution accompanied
# by histogram
procedure_dgpd_plot <- function (data, u, scale_est, shape_est){

  k <- 200
  data_cens <- data [data > u]
  n <- length (data_cens)

  data_cens_fit <- seq (u, max (data_cens), length = k+1) - u
  data_cens_fit <- data_cens_fit [2:length (data_cens_fit)]
```

```
  par_breaks <- 0
  if (n < 25){par_breaks <- n} else {par_breaks <- 25}

  histogram <- hist (data_cens, breaks=par_breaks, plot=FALSE, freq=TRUE)
  histogram$counts <- histogram$counts / (diff (histogram$mids [1:2])*n)
  dens_vec <- dgpd (data_cens_fit, scale=scale_est, shape=shape_est)

  plot (histogram, xlab="Quantile", ylab="Density", main="Density plot",
        xlim=c (u, max (data_cens)), ylim=c (0, max (dens_vec)))
  lines (density (data_cens), col="black", lwd=2)
  lines (data_cens_fit+u, dens_vec, col="red", lwd=2)
  box (lty="solid")
  legend ("topright", bty = "n", c("Theoretical","Empirical"),lwd=c(2,2),
          col=c("red","black"), horiz=FALSE)
}

# plot quantile-quantile plot of generalized Pareto distribution
procedure_qqgpd_plot <- function (data, u, scale_est, shape_est, name){

  data_cens <- data [data > u]
  n <- length (data_cens)

  prob_vec <- (seq (1, n, by=1)-0.35)/n
  qt_vec <- qgpd (prob_vec, scale=scale_est, shape=shape_est) + u

  plot (qt_vec, sort(data_cens), pch=20, xlab="Theoretical quantiles",
        ylab="Empirical quantiles", main=name)
  lines (c (u, max (data_cens, qt_vec)), c (u, max (data_cens, qt_vec)),
         lwd=2, col="red")
}

# depict return level plot of generalized Pareto distribution
procedure_retlevgpd_plot <- function (data, u, scale_est, shape_est){

  data_cens <- data [data > u]
  n <- length (data_cens)

  prob_vec <- (seq (1, n, by=1)-0.35)/n
  ret_per <- log (-log (1-prob_vec))
  ret_lev <- qgpd (prob_vec, scale=scale_est, shape=shape_est)+u

  label <- expression (paste ("log ", y[p]))
  plot (ret_per, sort (data_cens), pch=20,
```

```
        xlab=label, ylab="Quantiles (return level)",
        main="Return level plot")
  lines (ret_per, ret_lev, col="red", lwd=2)
}

# fit generalized Pareto distribution and access goodness of the fit
procedure_gpd_inference <- function (data, u, method, only_est){

  # fitting generalized pareto distribution

  gpd_fit <- fitgpd (data, threshold=u, method)
  scale_est <<- gpd_fit$param [1]
  shape_est <<- gpd_fit$param [2]

  # if parameter only_est is set to be 1 (TRUE) then procedure is
  # dedicated only to estimate distribution function parameters and
  # breaks at the following line
  if (only_est==1){}
  else {
    # confidence intervals for fitted parameters
    conf_int_shape <- gpd.fishape (gpd_fit, 0.95)
    conf_int_scale <- gpd.fiscale (gpd_fit, 0.95)

    # checking the model (statistical approach: Anderson-Darling,
    # Kolmogorov-Smirnov, Cramer-von Mises tests)
    data_cens <- data [data>u]
    data_cens_adj <- data_cens-u
    n <- length (data_cens)
    data_gener <- rgpd (n, scale=scale_est, shape=shape_est)

    ad_test_p <- ad.test (data_cens_adj, pgpd, shape=shape_est,
                          scale=scale_est)$p.value
    ks_test_p <- ks.test (data_cens_adj, pgpd, shape=shape_est,
                          scale=scale_est)$p.value
    cvm_statistics <- cvmts.test (data_gener, data_cens_adj)
    cvm_test_p <- cvmts.pval (cvm_statistics, n, n)

    # checking the model (graphical approach)
    par (mfrow = c (2,2))
    options("scipen" = 20)

    procedure_pgpd_plot (data, u, scale_est, shape_est)
    procedure_qqgpd_plot (data, u, scale_est, shape_est,
```

```
                          "Quantile-quantile plot")
    procedure_dgpd_plot (data, u, scale_est, shape_est)
    procedure_retlevgpd_plot (data, u, scale_est, shape_est)

    # results (vector of length 8)
    c (scale_est [[1]], conf_int_scale [[1]], conf_int_scale [[2]],
       shape_est [[1]], conf_int_shape [[1]], conf_int_shape [[2]],
       ad_test_p [[1]], ks_test_p, cvm_test_p)
  }
}

# find number of observation above specified threshold (threshold set
# as vector)
procedure_data_length <- function (data, threshold_vec){

  data_length <- rep (0, length (threshold_vec))

  for (i in 1:length (threshold_vec)){
    data_length [i] <- length (data [data >= threshold_vec [i]])
  }

  data_length
}

# procedure comparing goodnaess of the fit of lognaormal and generalized
# Pareto distribution
procedure_compare_models <- function (data, threshold_vec, method){

  par (mfrow = c (2,3))
  ks_matrix <- matrix (0, nrow = 2, ncol=3)

  # fitting lognormal distribution
  sample_length <- 10000
  log_data <- log (sample (data, sample_length, replace = FALSE,
                           prob = NULL))

  procedure_fitting_distribution (log_data, 1)

  # quantile-quantile plot corresponding to fitted log-normal
  # distribution
  for (i in 1:length (threshold_vec)){
    procedure_qqlnorm_plot (data, threshold_vec [i], log_mean_est,
                            log_sd_est, paste ("u = ",threshold_vec [i]))
```

```
    data_cens <- data [data > threshold_vec [i]]
    data_cens_adj <- log (data_cens)
    ks_matrix [1, i] <- ks.test (data_cens_adj, pnorm, mean=log_mean_est,
                                 sd=log_sd_est)$p.value
  }

  for (i in 1:length (threshold_vec)){
    # fitting gpd distribution
    procedure_gpd_inference (data, threshold_vec [i], method, 1)

    # quantile-quantile plot corresponding to fitted generalized Pareto
    # distribution
    procedure_qqgpd_plot (data, threshold_vec [i], scale_est, shape_est,
                          paste ("u = ", threshold_vec [i]))
    data_cens <- data [data>threshold_vec [i]]
    data_cens_adj <- data_cens-threshold_vec [i]
    ks_matrix [2, i] <- ks.test (data_cens_adj, pgpd, shape=shape_est,
                                 scale=scale_est)$p.value
  }

  ks_matrix
}

# plot quantile-quantile plot of generalized Pareto distribution with
# detailed emphasis on the tail
procedure_qqgpd_plot_mod <- function (data, u, scale_est, shape_est){

  data_cens <- data [data > u]
  n_cens <- length (data_cens)
  n <- length (data)

  prob_vec <- seq (0.0000001, 0.00004, by=0.0000001)
  prob_vec_modif <- 1-prob_vec
  qt_vec <- (((n/n_cens)*prob_vec)^(-shape_est)-1)*
            (scale_est/shape_est) + u
  qt_vec_emp <- quantile (data, probs=prob_vec_modif)

  plot (qt_vec, qt_vec_emp, pch=20, xlab="Theoretical quantiles",
        ylab="Empirical quantiles", main="")
  lines (c (400000, 1200000), c (400000, 1200000), lwd=2, col="red")
}

# plot probable maximum loss of generalized Pareto distribution
```

```
procedure_pml_plot <- function (data, u, scale_est, shape_est){

  data_cens <- data [data > u]
  n_cens <- length (data_cens)
  n <- length (data)
  lambda <- n_cens/n

  prob_vec <- seq (0.0000001, 0.00004, by=0.0000001)
  prob_vec_modif <- 1-prob_vec
  pml <- u + (scale_est/shape_est)*((-lambda/
         log (prob_vec_modif))^shape_est-1)

  plot (prob_vec, pml, pch=20, xlab="Probability",
        ylab="Probable maximum loss", main="")
}

# application of generalized Pareto distribution
procedure_gpd_application <- function (data, u, method){

  # fitting generalized pareto distribution

  procedure_gpd_inference (data, u, method, 1)

  par (mfrow=c (1, 1))
  # plot quantile-quantile plot for fitted generalized Pareto
  # distribution
  procedure_qqgpd_plot_mod (data, u, scale_est, shape_est)
  # plot probable maximum loss of generalized Pareto distribution
  procedure_pml_plot (data, u, scale_est, shape_est)
}


##### program

# define directory of database file (dir_read) and directory dedicated
dir_read <- ""

# read database
claims_97 <- procedure_read_database (dir_read, "claim97.txt")
claims_98 <- procedure_read_database (dir_read, "claim98.txt")
claims_99 <- procedure_read_database (dir_read, "claim99.txt")

# set special choice for threshold selection (stems from statistical
```

```
# analysis)
threshold_97 <- c (400000, 600000, 700000)
threshold_98 <- c (300000, 600000, 700000)
threshold_99 <- c (300000, 700000, 1000000)

# set claim year and limit for lowest excesses
year <- 97
limit <- 100000

# select corresponding claims, excesses and thrshold choices
claims <- get (paste ("claims_", year, sep=""))
claims_excess <- claims [claims > limit]
threshold <- get (paste ("threshold_", year, sep=""))


### PART 1: descriptive statistics
### (provides basic data descriptive statistics, fits ordinary
# distribution to data using maximum likelihood approach)

sample_length <- 10000

set.seed (123)
log_claims <- log (sample (claims, sample_length, replace = FALSE,
                           prob = NULL))
result_claims <- procedure_fitting_distribution (log_claims, 0)

# print results

# descriptive statistics: number of observations (length), sum of claims,
# min. claim, 1st quantile, median, mean, 3rd quantile, max. claim
procedure_descriptive_stat (claims)

# estimation of mean and standard deviation of normal distribution
# approximating logarithmic claims, p-value of Shapiro-Wilk test
# (executed 10 times on different samples)
result_claims

# box plot of claims and logarithmic claims, graphical goodness of
# the fit (theoretical and empirical density, distribution and
# quantile-quantile plot)
procedure_plot_descriptive_stat (claims, result_claims [1],
                                 result_claims [2])
```

```
### PART 2: threshold selection

options("scipen" = 20)

# print results (graphical threshold selection)

# the empirical mean residual life plot
procedure_mrlplot (claims_excess, limit, 0.999)

# threshold choice plot (dependence of parameter estimates at various
# thresholds)
procedure_tcplot (claims_excess, limit, 0.999)

# L-moments plot (dipicts L-kurtosis against L-skewness)
procedure_lmomplot_est (claims_excess, limit, quantile (claims_excess,
                        probs=0.997), 200)

# linear regression estimates of parameters for thresholds values (given
# by variable threshold) acompanied by graphical illustration
est_ksi <- procedure_mrlplot_linefit (claims_excess, limit,
           quantile (claims_excess, probs=0.999), 200, threshold)
est_ksi


### PART 3: GPD inference

method <- c ("mle", "pwmb", "mple")

# generation column and row names for result matrix
col_names <- c ("scale", "scale_inf", "scale_sup", "shape", "shape_inf",
                "shape_sup", "ad_test", "ks_test", "cvm_test")
row_names <- c (sapply (method, function (y) sapply (threshold,
                        function (x) paste (y, "_", x, sep=""))))
result_matrix_claims <- matrix (0, nrow=length (row_names),
                                ncol=length (col_names),
                                dimnames=list (row_names, col_names))

# execute GPD inference for all choices of thresholds and all calculation
# methods
for (i in 1:length (method)){
  for (j in 1: length (threshold)){
    result_matrix_claims [3*(i-1)+j,] <- procedure_gpd_inference (claims,
```

```
                                        threshold [j], method [i], 0)
  }
}

# print results

# round result matrix
result_matrix_claims <- round (result_matrix_claims, digits=4)
result_matrix_claims

# find number of observation above specified thresholds
claims_length <- procedure_data_length (claims, threshold)
claims_length


# PART 4: model comparison (goodness of the fit of log-normal and
# generalized Pareto distribution)

# print results:

# compare goodness of the fit of log-normal and generalized Pareto
# distribution
ks_matrix <- procedure_compare_models (claims, threshold, "pwmb")

# round Kolmogorov-Smirnov test matrix
ks_matrix <- round (ks_matrix, digits=4)
ks_matrix


# PART 5: model application (quantile-quantile plot for fitted
# gerenalized
# Pareto distrbution for selected threshold, estimation of probable
# maximum loss)

method <- "pwmb"

# print results

for (j in 1: length (threshold)){
  procedure_gpd_application (claims, threshold [j], method)
}
```

Chapter 3

Survival Data Analysis

Survival data analysis (SDA) typically focuses on *time to event data*. In the most general sense, it consists of techniques for positive-valued random variables, such as

- time to death,
- time to onset (or relapse) of a disease,
- length of a contract,
- duration of a policy,
- money paid by health insurance,
- viral load measurements,
- time to finishing a master thesis.

Typically, survival data are *not fully observed*, but rather are *censored*.

3.1 Theoretical background of SDA

The consequent theoretical summary provides just a necessary framework for solving some real problems postulated later on. For further reading, we refer to books by Fleming and Harrington (2005), Kalbfleisch and Prentice (2002), Collett (2014), and Kleinbaum and Klein (2012).

Failure time random variables are always non-negative. That is, if we denote the failure time by T, then $T \geqslant 0$. A random variable X is called a *censored failure time* random variable if $X = \min(T, U)$, where U is a non-negative *censoring* variable. In order to *define a failure time random variable*, we need:

(i) an unambiguous *time origin* (e.g., randomization to clinical trial, purchase of car),

(ii) a *time scale* (e.g. real time (days, years), mileage of a car),

(iii) definition of the *event* (e.g., death, need a new car transmission).

The illustration of survival data in Figure 3.1 shows several features which are typically encountered in analysis of survival data:

- individuals do not all enter the study at the same time,
- when the study ends, some individuals ($i = n$) still haven't had the event yet,
- other individuals drop out or get lost in the middle of the study, and all we know about them is the last time they were still 'free' of the event ($i = 2$),
- for the remaining individuals, the event happened ($i = 1$).

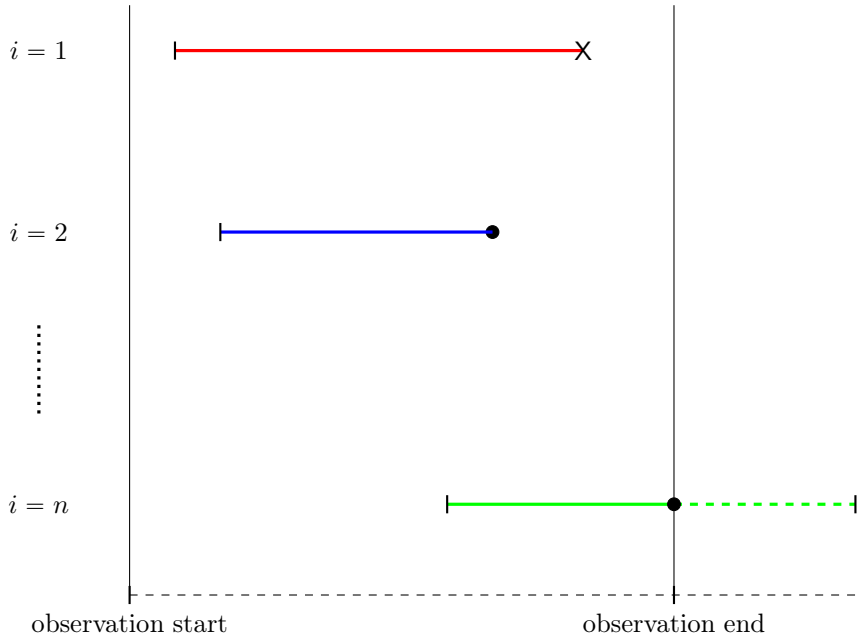

Figure 3.1: Illustration of survival data with right censoring, where • represents a censored observation (e.g., alive) and X stands for an event (e.g., died).

3.1.1 Types of censoring

Right-censoring

Only the random variable $X_i = \min(T_i, U_i)$ is observed due to:

- loss to follow-up,
- drop-out,
- study termination.

We call this right-censoring because the true unobserved event is to the right of our censoring time; i.e., all we know is that the event has not happened at the end of follow-up.

In addition to observing X_i, we also get to see the *failure indicator*

$$\delta_i = \begin{cases} 1 \text{ if } T_i \leqslant U_i \\ 0 \text{ if } T_i > U_i \end{cases}$$

Some software packages instead assume we have a *ensoring indicator*

$$c_i = \begin{cases} 0 \text{ if } T_i \leqslant U_i \\ 1 \text{ if } T_i > U_i \end{cases}$$

Right-censoring is the most common type of censoring assumption we will deal with this type in the forthcoming text. For the sake of completeness, we also describe the remaining types of censoring.

Left-censoring

One can only observe $Y_i = \max(T_i, U_i)$ and the *failure indicators*

$$\delta_i = \begin{cases} 1 \text{ if } U_i \leqslant T_i \\ 0 \text{ if } U_i > T_i \end{cases}$$

e.g., age at which children learn a task. Some already knew (left-censored), some learned during study (exact), some had not yet learned by end of study (right-censored).

Interval censoring

Observe (L_i, R_i) where $T_i \in (L_i, R_i)$.

3.1.2 Definitions and notation

There are several equivalent ways to characterize the probability distribution of a survival random variable. Some of these are familiar; others are special to survival analysis. We will focus on the following terms:

- The density function $f(t)$,

- The survivor function $S(t)$,
- The hazard function $\lambda(t)$,
- The cumulative hazard function $\Lambda(t)$.

Density function

- *discrete*; Suppose that T takes values in $a_1, a_2, \ldots$.

$$f(t) = \mathsf{P}[T = t] = \begin{cases} f_j \text{ if } t = a_j,\ j = 1, 2, \ldots \\ 0 \text{ otherwise} \end{cases}$$

- *continuous*

$$f(t) = \lim_{\Delta t \to 0_+} \frac{\mathsf{P}[t \leqslant T \leqslant t + \Delta t]}{\Delta t}$$

Survivorship function

Survivorship (or survivor or survival) function $S(t) = \mathsf{P}[T \geqslant t]$.

In other settings, the cumulative distribution function, $F(t) = \mathsf{P}[T \leqslant t]$, is of interest. In survival analysis, our interest tends to focus on the *survival function*, $S(t)$.

- *discrete*

$$S(t) = \sum_{a_j \geqslant t} f_j$$

- *continuous*

$$S(t) = \int_t^\infty f(u)\mathrm{d}u$$

Remarks:

- From the definition of $S(t)$ for a continuous variable, $S(t) = 1 - F(t)$ as long as $f(t)$ is absolutely continuous.
- For a discrete variable, we have to decide what to do if an event occurs exactly at time t; i.e., does that become part of $F(t)$ or $S(t)$?
- To get around this problem, several books define $S(t) = \mathsf{P}[T > t]$, or else define $F(t) = \mathsf{P}[T < t]$ (Collett, 2014).

Hazard function

Sometimes called an *instantaneous failure rate*, the *force of mortality*, or the *age-specific failure rate*.

- *discrete*

$$\lambda_j = \mathsf{P}[T = a_j | T \geqslant a_j] = \frac{f_j}{\sum_{k:\, a_k \geqslant a_j} f_k}$$

- *continuous*

$$\lambda(t) = \lim_{\Delta t \to 0_+} \frac{\mathsf{P}[t \leqslant T \leqslant t + \Delta t | T \geqslant t]}{\Delta t} = \frac{f(t)}{S(t)}$$

Cumulative hazard function

- *discrete*

$$\Lambda_j = \sum_{j:\, a_j \leqslant t} \lambda_j$$

- *continuous*

$$\Lambda(t) = \int_0^t \lambda(u) \mathrm{d}u$$

3.1.3 Several relationships

$S(t)$ and $\lambda(t)$ for a *continuous* r. v.

$$\lambda(t) = -\frac{\mathrm{d}}{\mathrm{d}t}[\log S(t)].$$

$S(t)$ and $\Lambda(t)$ for a *continuous* r. v.

$$S(t) = \exp\{-\Lambda(t)\}.$$

$S(t)$ and $\Lambda(t)$ for a *discrete* r. v. (suppose that $a_j < t \leqslant a_{j+1}$)

$$S(t) = \prod_{j:\, a_j < t} (1 - \lambda_j).$$

Sometimes, one defines $\Lambda(t) = \sum_{j:\, a_j < t} \log(1 - \lambda_j)$ so that $S(t) = \exp\{-\Lambda(t)\}$ in the discrete case, as well.

3.1.4 Measuring central tendency in survival

Mean survival—call this μ

- *discrete*

$$\mu = \sum_j a_j f_j$$

- *continuous*

$$\mu = \int_0^\infty u f(u) \mathrm{d}u$$

Median survival—call this τ, is defined by

$$S(\tau) = 0.5$$

Similarly, any other percentile could be defined.

In practice, we don't usually hit the median survival at exactly one of the failure times. In this case, the estimated median survival is the *smallest* time τ such that $S(\tau) \leqslant 0.5$

3.1.5 Estimating the survival or hazard function

We can estimate the survival (or hazard) function in two ways:

- by specifying a parametric model for $\lambda(t)$ based on a particular density function $f(t)$,
- developing an empirical estimate of the survival function (i.e., non-parametric estimation).

If no censoring

The empirical estimate of the survival function, $\widehat{S}(t)$, is the proportion of individuals with event times greater than t.

With censoring

If there are censored observations, then $\widehat{S}(t)$ is not a good estimate of the true $S(t)$, so other non-parametric methods must be used to account for censoring (life-table methods, Kaplan-Meier estimator)

3.1.6 Preview of coming attractions

Next we will discuss the most famous non-parametric approach for estimating the survival distribution, called the *Kaplan-Meier (KM) estimator*.

To motivate the derivation of this estimator, we will first consider a set of survival times where there is *no censoring*. The following are times to relapse (weeks) for 21 leukemia patients receiving control treatment:

```
> 1,1,2,2,3,4,4,5,5,8,8,8,8,11,11,12,12,15,17,22,23
```

How would we estimate $S(10)$, the probability that an individual survives to time 10 or later? What about $\widehat{S}(8)$? Is it 12/21 or 8/21?

3.1.7 Empirical survival function

When there is *no censoring*, the general formula is:

$$\widehat{S}(t) = \frac{\#\text{ of individuals with } T \geqslant t}{\text{total sample size}}$$

In most software packages, the survival function is evaluated just after time t, i.e., at t^+. In this case, we only count the individuals with $T > t$.

3.1.8 Kaplan-Meier estimator

What if there *is censoring*? [Note: times with + are right censored.]

```
> 6+,6,6,6,7,9+,10+,10,11+,13,16,17+,19+,20+,
 22,23,25+,32+,32+,34+,35+
```

We know $\widehat{S}(6) = 21/21$, because everyone survived at least until time 6 or greater. But, we can't say $\widehat{S}(7) = 17/21$, because we don't know the status of the person who was censored at time 6.

In a 1958, Kaplan and Meier proposed a way to nonparametrically estimate $S(t)$, even in the presence of censoring. The method is based on the ideas of *conditional probability*. Suppose $a_k < t \leqslant a_{k+1}$. Then

$$S(t) = \mathsf{P}[T \geqslant a_{k+1}] = \mathsf{P}[T \geqslant a_1, \ldots, T \geqslant a_{k+1}] = \prod_{j=1}^{k}\{1-\mathsf{P}[T = a_j | T \geqslant a_j]\} = \prod_{j=1}^{k}\{1-\lambda_j\}.$$

So

$$\widehat{S}(t) = \prod_{j}^{a_j<t}\left(1 - \frac{d_j}{r_j}\right),$$

where d_j is the *number of deaths* at a_j and r_j is the *number at risk* at a_j.

Intuition behind the Kaplan-Meier estimator

Think of dividing the observed timespan of the study into a series of fine intervals so that there is a separate interval for each time of death or censoring:

```
> _ , _ , D , _ , C , _ , C , D , D , D
```

Using the law of conditional probability,

$$\mathsf{P}[T \geqslant t] = \prod_j \mathsf{P}[\text{survive } j \text{ th interval } I_j | \text{ survived to start of } I_j],$$

where the product is taken over all the intervals including or preceding time t. There are 4 possibilities for each interval:

(i) *No events (death or censoring)* – conditional probability of surviving the interval is 1

(ii) *Censoring* – assume they survive to the end of the interval, so that the conditional probability of surviving the interval is 1

(iii) *Death, but no censoring* – conditional probability of not surviving the interval is # deaths (d) divided by # 'at risk' (r) at the beginning of the interval. So the conditional probability of surviving the interval is $1 - (d/r)$

(iv) *Tied deaths and censoring* – assume censorings last to the end of the interval, so that conditional probability of surviving the interval is still $1 - (d/r)$.

General Formula for j-th interval

It turns out we can write a general formula for the conditional probability of surviving the j-th interval that holds for all 4 cases. We could use the same approach by grouping the event times into intervals (say, one interval for each month), and then counting up the number of deaths (events) in each to estimate the probability of surviving the interval (this is called the *lifetable estimate*). However, the assumption that those censored last until the end of the interval wouldn't be quite accurate, so we would end up with a cruder approximation. As the intervals get finer and finer, the approximations made in estimating the probabilities of getting through each interval become smaller and smaller, so that the estimator converges to the true $S(t)$. This intuition clarifies why an alternative name for the KM is the *product limit estimator*.

KM estimator

$$\widehat{S}(t) = \prod_j^{\tau_j < t} \left(1 - \frac{d_j}{r_j}\right),$$

where

- $\tau_1, \ldots, \tau_K$ is the set of K distinct death times observed in the sample,
- d_j is the number of deaths at τ_j,
- r_j is the number of individuals 'at risk' right before the j-th death time (everyone dead or censored at or after that time),
- c_j is the number of censored observations between the j-th and $(j+1)$-st death times. Censorings tied at τ_j are included in c_j.

Remarks:

- $r_j = r_{j-1} - d_{j-1} - c_{j-1}$,
- $r_j = \sum_{l \geqslant j}(c_l + d_l)$,
- $\widehat{S}(t^+)$ changes only at the death (failure) times,
- $\widehat{S}(t^+)$ is 1 up to the first death time,
- $\widehat{S}(t^+)$ only goes to 0 if the last event is a death.

3.1.9 Confidence intervals

Greenwood's formula

Variance of KM estimator is

$$\mathsf{Var}\widehat{S}(t) = \left[\widehat{S}(t)\right]^2 \sum_j^{\tau_j<t} \frac{d_j}{r_j(d_j - r_j)}.$$

For a 95% confidence interval, we could use

$$\widehat{S}(t) \pm z_{1-\alpha/2}\mathsf{se}\left(\widehat{S}(t)\right),$$

where $\mathsf{se}\left(\widehat{S}(t)\right)$ is calculated using Greenwood's formula.

Log-log approach

Problem: This approach can yield values > 1 or < 0.

Better approach: Get a 95% confidence interval for $L(t) = \log\{-\log(S(t))\}$, i.e., *log-log approach*. Since this quantity is unrestricted, the confidence interval will be in the proper range when we transform back. Applying the *delta method*, we get:

$$\mathsf{Var}\widehat{L}(t) = \frac{1}{\left[\log \widehat{S}(t)\right]^2} \sum_j^{\tau_j<t} \frac{d_j}{r_j(d_j - r_j)}$$

We take the square root of the above to get $\mathsf{se}\left(\widehat{L}(t)\right)$, and then form the confidence intervals as:

$$\widehat{S}(t)^{\exp\{\pm 1.96\mathsf{se}(\widehat{L}(t))\}}$$

R gives an option to calculate these bounds (use `conf.type='log-log'` in `survfit`).

Log stabilizing transformation

R (by default) uses a *log transformation* to stabilize the variance and allow for non-symmetric confidence intervals. This is what is normally done for the confidence interval of an estimated odds ratio

$$\mathsf{Var}\left(\log \widehat{L}(t)\right) = \sum_{j}^{\tau_j<t} \frac{d_j}{r_j(d_j - r_j)}.$$

3.1.10 Lifetable or actuarial estimator

What to do when the *data are grouped*?

Our goal is still to estimate the survival function, hazard, and density function, but this is complicated by the fact that we don't know exactly when during each time interval an event occurs.

Notation:

- the j-th time interval is $[t_{j-1}, t_j)$,
- c_j ... the number of censorings in the j-th interval,
- d_j ... the number of failures in the j-th interval,
- r_j ... the number *entering* the interval.

We could apply the KM formula directly. However, this approach is unsatisfactory for grouped data ... it treats the problem as though it were in discrete time, with events happening only at 1 yr, 2 yr, etc. In fact, what we are trying to calculate here is the conditional probability of dying *within the interval*, given survival to the beginning of it.

We can assume that censorings occur on average *halfway* through the interval:

$$r'_j = r_j - c_j/2$$

The assumption yields the *actuarial estimator*. It is appropriate if censorings occur uniformly throughout the interval.

Remarks:

- Because the intervals are defined as $[t_{j-1}, t_j)$, the first interval typically starts with $t_0 = 0$.
- R and SAS estimate the survival function at the left-hand endpoint, $S(t_{j-1})$. *Stata* at the right-hand one.
- The implication in R and SAS is that $\widehat{S}(t_0) = 1$.

Quantities estimated

- Midpoint $t_{mj} = (t_j + t_{j-1})/2$.
- Width $b_j = t_j - t_{j-1}$.
- Conditional probability of dying $\widehat{q}_j = d_j/r'_j$.
- Conditional probability of surviving $\widehat{p}_j = 1 - \widehat{q}_j$.
- Cumulative probability of surviving at t_j: $\widehat{S}(t) = \prod_{l \leqslant j} \widehat{p}_l$.
- Hazard in the j-th interval

$$\widehat{\lambda}(t_{mj}) = \frac{\widehat{q}_j}{b_j(1 - \widehat{q}_j/2)}$$

  the number of deaths in the interval divided by the average number of survivors at the midpoint.

- Density at the midpoint of the j-th interval

$$\widehat{f}(t_{mj}) = \frac{\widehat{S}(t_{j-1})\widehat{q}_j}{b_j}.$$

3.1.11 Nelson-Aalen estimator

Estimating the *cumulative hazard* $\Lambda(t)$ can then be approximated by a sum

$$\widehat{\Lambda}(t) = \sum_j \lambda_j \Delta,$$

where the sum is over intervals, λ_j is the value of the hazard in the j-th interval and Δ is the width of each interval. Since $\lambda_j \Delta$ is approximately the probability of dying in the interval, we can further approximate by

$$\widehat{\Lambda}(t) = \sum_j \frac{d_j}{r_j}.$$

It follows that $\Lambda(t)$ will change only at death times, and hence we write the *Nelson-Aalen estimator* as

$$\widehat{\Lambda}_{NA}(t) = \sum_{j}^{\tau_j<t} \frac{d_j}{r_j}.$$

3.1.12 Fleming-Harrington estimator

Once we have $\widehat{\Lambda}_{NA}(t)$, we can also find another estimator of $S(t)$

$$\widehat{S}_{FM}(t) = \exp\{-\widehat{\Lambda}_{NA}(t)\},$$

which is called *Fleming-Harrington* estimator of the survival (an alternative to Kaplan-Meier).

3.1.13 Comparison of survival curves

Two survival curves

To test

$$H_0 : S_1(t) = S_2(t), \forall t$$
$$H_1 : \exists t : S_1(t) \neq S_2(t)$$

Cochran-Mantel-Haenszel Logrank test

The logrank test is the most well known and widely used. It also has an intuitive appeal, building on standard methods for binary data. (Later we will see that it can also be obtained as the score test from a partial likelihood from the Cox Proportional Hazards model.)

The logrank test is obtained by constructing a 2x2 table at each distinct death time, and comparing the death rates between the two groups, conditional on the number at risk in the groups. The tables are then combined using the Cochran-Mantel-Haenszel test. Note that the logrank is sometimes called the *Cox-Mantel* test.

- The logrank statistic depends on *ranks* of event times only.
- It does not matter which group you choose.
- Analogous to the CMH test for a series of tables at different levels of a confounder, the logrank test is most powerful when 'odds ratios' are constant over time intervals. That is, it is most powerful for *proportional hazards*.

Peto-Peto logrank test

The logrank test can be derived by assigning scores to the ranks of the death times. This is called the Peto-Peto logrank test, or the linear rank logrank test.

Comparing the CMH-type logrank and Peto-Peto logrank:

- *CMH-type logrank*: We motivated the logrank test through the CMH statistic for testing $H_0 : OR = 1$ over K tables, where K is the number of distinct death times.
- *Peto-Peto (linear rank) logrank*: The linear rank version of the logrank test is based on adding up 'scores' for one of the two treatment groups. The particular scores that gave us the same logrank statistic were based on the Nelson-Aalen estimator.
- If there are no tied event times, then the two versions of the test will yield identical results. The more ties we have, the more it matters which version we use.
- The Peto-Peto (linear rank) logrank test is *not available* in R. Use the CMH logrank test instead.

Peto-Peto-Prentice-Gehan-Wilcoxon test

This is the Prentice (1978) modification of the Peto & Peto (1972) modification of the Gehan (1965) modification of the Wilcoxon (1945) rank test.

Which test should we use?

- *Logrank test: CMH or Linear Rank?* If there are not too many ties, then it doesn't really matter.
- *Logrank test or (PPP)Gehan-Wilcoxon?* This is a more important choice.
- Both tests have the correct level (Type I error) for testing the *null hypothesis* of equal survival, $H_0 : S_1(t) = S_2(t)$.
- The choice of which test to use depends on the *alternative hypothesis*. This drives the *power* of the test.
- The Gehan-Wilcoxon test is sensitive to *early differences* in survival between groups.
- The Logrank test is sensitive to *later differences*.
- The logrank is most powerful under the assumption of *proportional hazards*, $\lambda_1(t) = \alpha\lambda_2(t)$, which implies an alternative in terms of the survival functions of $H_1 : S_1(t) = [S_2(t)]^\alpha$.

- The Wilcoxon has high power when the *failure times are lognormally distributed*, with equal variance in both groups but a different mean. It will turn out that this is the assumption of an accelerated failure time model.
- Both tests will lack power if the survival curves (or hazards) 'cross' (not proportional hazards). However, that does not necessarily make them invalid!

Three or more survival curves – P-sample logrank

There are more than two groups, and the question of interest is whether the groups differ from each other.

Stratified Logrank

Sometimes, even though we are interested in comparing two groups (or maybe P) groups, we know there are other factors that also affect the outcome. It would be useful to adjust for these other factors in some way.

A stratified logrank allows one to *compare groups*, but allows the *shapes of the hazards of the different groups to differ across strata*. It makes the assumption that the group 1 vs group 2 hazard ratio is constant across strata. In other words: $\lambda_{1s}(t)/\lambda_{2s}(t) = \theta$ where θ is constant over the strata ($s = 1, \ldots, S$).

3.2 Proportional Hazards

We will explore the relationship between survival and explanatory variables by modeling. In this class, we consider a broad class of regression models: *Proportional Hazards (PH) models*

$$\log \lambda(t; \mathbf{Z}) = \log \lambda_0(t) + \boldsymbol{\beta}\mathbf{Z}.$$

Suppose $Z = 1$ for treated subjects and $Z = 0$ for untreated subjects. Then this model says that the hazard is increased by a factor of e^{β} for treated subjects versus untreated subjects (e^{β} might be < 1).

This group of PH models divides further into:

- *Parameteric* PH models: Assume parametric form for $\lambda_0(t)$, e.g., Weibull distribution.
- *Semi-parametric* PH models: No assumptions on $\lambda_0(t)$: *Cox PH model*.

3.2.1 Cox Proportional Hazards model

Why do we call this proportional hazards?

$$\lambda(t; \mathbf{Z}) = \lambda_0(t) \exp\{\boldsymbol{\beta}\mathbf{Z}\}$$

Think of the first example, where $Z = 1$ for treated and $Z = 0$ for control. Then if we think of $\lambda_1(t)$ as the hazard rate for the treated group, and $\lambda_0(t)$ as the hazard for control, then we can write:

$$\lambda_1(t) = \lambda(t; Z = 1) = \lambda_0(t) \exp\{\beta Z\} = \lambda_0(t) \exp\{\beta\}$$

This implies that the ratio of the two hazards is a constant, ϕ, which does NOT depend on time, t. In other words, the hazards of the two groups remain proportional over time.

$$\phi = \frac{\lambda_1(t)}{\lambda_0(t)} = e^{\beta}$$

ϕ is referred to as the *hazard ratio.*

3.2.2 Baseline Hazard Function

In the example of comparing two treatment groups, $\lambda_0(t)$ is the hazard rate for the control group. In general, $\lambda_0(t)$ is called the *baseline hazard function*, and reflects the underlying hazard for subjects with all covariates $Z_1, \ldots, Z_p$ equal to 0 (i.e., the 'reference group').

The general form is

$$\lambda(t; \mathbf{Z}) = \lambda_0(t) \exp\{\beta_1 Z_1 + \ldots + \beta_p Z_p\}.$$

So when we substitute all of the Z_j equal to 0, we get $\lambda(t; \mathbf{Z}) = \lambda_0(t)$.

In the general case, we think of the i-th individual having a set of covariates $\mathbf{Z}_i = (Z_{1i}, Z_{2i}, \ldots, Z_{pi})$, and we model their hazard rate as some multiple of the baseline hazard rate

$$\lambda(t; \mathbf{Z}_i) = \lambda_0(t) \exp\{\beta_1 Z_{1i} + \ldots + \beta_p Z_{pi}\}.$$

This means we can write the log of the hazard ratio for the i-th individual to the reference group as

$$\log \frac{\lambda(t; \mathbf{Z}_i)}{\lambda_0(t)} = \beta_1 Z_{1i} + \ldots + \beta_p Z_{pi}.$$

The Cox Proportional Hazards model is a linear model for the log of the hazard ratio.

One of the biggest advantages of the framework of the Cox PH model is that we can estimate the parameters $\boldsymbol{\beta}$ which reflect the effects of treatment and other covariates without having to make any assumptions about the form of $\lambda_0(t)$. In other words, we don't have to assume that $\lambda_0(t)$ follows an exponential model, or a Weibull model, or any other particular parametric model. That's what makes the model *semi-parametric*.

Questions:

(i) Why don't we just model the hazard ratio, $\phi = \frac{\lambda_i(t)}{\lambda_0(t)}$, directly as a linear function of the covariates $\mathbf{Z}$?

(ii) Why doesn't the model have an intercept?

Parameters of interest, $\boldsymbol{\beta}$, are estimated via *partial likelihood*. $\lambda_0(\cdot)$ is treated as a nuisance parameter.

Adjustments for ties

The proportional hazards model assumes a continuous hazard—ties are not possible. There are four proposed modifications to the likelihood to adjust for ties:

(i) Cox's modification,

(ii) Efron's method (default in *R*),

(iii) Breslow's method,

(iv) exact method.

Bottom Line: Implications of Ties

- *When there are no ties*, all options give exactly the same results.
- *When there are only a few ties*, it won't make much difference which method is used. However, since the exact methods won't take much extra computing time, you might as well use one of them.
- *When there are many ties* (relative to the number at risk), the Breslow option (default) performs poorly (Farewell & Prentice, 1980; Hsieh, 1995). Both of the approximate methods, Breslow and Efron, yield coefficients that are attenuated (biased toward 0).
- *The choice of which exact method to use* should be based on substantive grounds—are the tied event times truly tied? ... or are they the result of imprecise measurement?
- *Computing time of exact methods* is much longer than that of the approximate methods. However, in most cases it will still be less than 30 seconds even for the exact methods.

- *Best approximate method* – the Efron approximation nearly always works better than the Breslow method, with no increase in computing time, so use this option if exact methods are too computer-intensive.

3.2.3 Confidence intervals and hypothesis tests

Many software packages provide estimates of $\boldsymbol{\beta}$, but the hazard ratio $HR = \exp\{\boldsymbol{\beta}\}$ is usually the parameter of interest.

For each covariate of interest, the null hypothesis is

$$H_0 : HR_j = 1 \Leftrightarrow \beta_j = 0$$

Wald test is used. For *nested* models, a *likelihood ratio test* is constructed.

3.2.4 Predicted survival

For the i-th patient with covariates $\mathbf{Z}_i$, we have

$$S_i(t) = [S_0(t)]^{\exp\{\boldsymbol{\beta}\mathbf{Z}_i\}}.$$

Say we are interested in the survival pattern for single males in the nursing home study. Based on the previous formula, if we had an estimate for the survival function in the reference group, i.e., $\widehat{S}_0(t)$, we could get estimates of the survival function for any set of covariates $\mathbf{Z}_i$.

How can we estimate the survival function, $S_0(t)$? We could use the KM estimator, but there are a few disadvantages of that approach:

- It would only use the survival times for observations contained in the reference group, and not all the rest of the survival times.
- It would tend to be somewhat choppy, since it would reflect the smaller sample size of the reference group.
- It's possible that there are no subjects in the dataset who are in the 'reference' group (e.g., say covariates are age and sex; there is no one of age=0 in our dataset).

Instead, we will use a baseline hazard estimator which takes advantage of the proportional hazards assumption to get a smoother estimate

$$\widehat{S}_i(t) = [\widehat{S}_0(t)]^{\exp\{\widehat{\boldsymbol{\beta}}\mathbf{Z}_i\}}.$$

Using the above formula, we substitute $\widehat{\boldsymbol{\beta}}$ based on fitting the Cox PH model, and calculate $\widehat{S}_0(t)$ by one of the following approaches:

- Breslow estimator (R) ... extending the concept of the Nelson-Aalen estimator to the proportional hazards model.
- Kalbfleisch/Prentice estimator (R, SAS) ... analogous to the Kaplan-Meier estimator.

3.2.5 Model selection

Collett (2014, Section 3.6) has an excellent discussion of various approaches for model selection. In practice, model selection proceeds through a combination of:

- knowledge of the science,
- trial and error, common sense,
- automatic variable selection procedures:
  - forward selection,
  - backward selection,
  - stepwise selection.

Many advocate the approach of first doing a univariate analysis to 'screen' out potentially significant variables for consideration in the multivariate model (Collett, 2014). Moreover, typically univariate analysis is a part of a larger analysis, in order to identify the importance of each predictor taken in itself.

3.2.6 Model selection approach

1. Fit a univariate model for each covariate, and identify the predictors significant at some level p_1, say 0.20 (Hosmer and Lemeshow recommend $p_1 = 0.25$).
2. Fit a multivariate model with all significant univariate predictors, and use backward selection to eliminate nonsignificant variables at some level p_2, say 0.10.
3. Starting with final step 2. model, consider each of the non-significant variables from step 1. using forward selection, with significance level p_3, say 0.10.
4. Do final pruning of main-effects model (omit variables that are non-significant, add any that are significant), using stepwise regression with significance level p_4. At this stage, you may also consider adding interactions between any of the main effects currently in the model, under the hierarchical principle.

Collett recommends using a likelihood ratio test for all variable inclusion/exclusion decisions. Each step uses a 'greedy' approach:

- *Backward step*: among several candidates for elimi- nation from the model, the one with the smallest effect on the criterion (AIC, LR), is eliminated. None is eliminated if AIC increases (AIC) or LR significant (LR)
- *Forward step*: among several candidates for addition to the model, the one with the largest effect on the criterion (AIC, LR), is added. None is added if AIC increases (AIC) or LR not significant (LR).

Remarks:

- The forward, backward and stepwise options, the same final model was reached. However, this will not always (in fact, rarely) be the case.
- Variables can be forced into the model using the include option in SAS. Any variables that you want to force inclusion of must be listed first in your model statement. In `stepAIC`, these are included in the scope part.
- SAS uses the score test to decide what variables to add and the Wald test for what variables to remove.
- When might we want to force certain variables into the model?
  (i) to examine interactions,
  (ii) to keep main effects in the model,
  (iii) to include scientifically important variables, e.g., treatment, gender, race in a medical study, or other covariates which are known from previous studies or from theory to be important.
- Model selection is an imperfect art!
  (i) No method is universally agreed upon
  (ii) Choosing between two models using LR or AIC has an information theory grounding,
  (iii) Forward/backward/stepwise selection have no theoretical bases,
  (iv) Leads to potential biases, nominal p-values smaller than true p-values,
  (v) The model selection step is usually not taken into account, and inference is done as if the chosen model were 'true'.

3.2.7 Model diagnostics – Assessing overall model fit

How do we know if the model fits well?

- *Always look at univariate plots* (Kaplan-Meiers). Construct a Kaplan-Meier survival plot for each of the important predictors.

- Check proportional hazards assumption.
- Check residuals:
  (i) generalized (Cox-Snell),
  (ii) martingale,
  (iii) deviance,
  (iv) Schoenfeld,
  (v) weighted Schoenfeld.

3.2.8 Assessing the PH assumption

There are several options for checking the assumption of proportional hazards:

1. Graphical:
   - Plots of $\log \widehat{\Lambda}(t) = \log[-\log \widehat{S}(t)]$ vs $\log t$ for two subgroups.
   - Plots of weighted Schoenfeld residuals vs time.
2. Tests of time-varying coefficient $\beta = \beta(t)$:
   - Tests based on Schoenfeld residuals (Grambsch and Thereneau, 1994).
   - Including interaction terms between covariates Z_j and time, or $\log t$.
3. Overall goodness of fit tests available in R via `cox.zph`.

How do we interpret the above?

Kleinbaum (and other texts) suggest a strategy of assuming that PH holds unless there is very strong evidence to counter this assumption:

- estimated survival curves are fairly separated, then cross,
- estimated log cumulative hazard curves cross, or look very unparallel over time,
- weighted Schoenfeld residuals clearly increase or decrease over time (you could fit a OLS regression line and see if the slope is significant),
- test for log(time) x covariate interaction term is significant (this relates to time-dependent covariates).

If PH doesn't exactly hold for a particular covariate but we fit the PH model anyway, then what we are getting is sort of an average HR, averaged over the event times.

In most cases, this is not such a bad estimate. Allison claims that too much emphasis is put on testing the PH assumption, and not enough to other important aspects of the model.

3.3 Practical applications of SDA

All the analyses will be performed by *R* software using the following packages:

```
library(survival)
library(KMsurv)
library(ggplot2)
```

3.3.1 Simple SDA's exercises

Exercise 3.1. Consider a distribution for the following discrete survival time (denoted by T):

a_j	1	3	4	5	10	12
$\mathsf{P}[T = a_j]$	0.40	0.25	0.10	0.10	0.10	0.05

(a) Compute by hand the survival, hazard and cumulative hazard functions for T.

(b) Plot the survival function $S(t)$ versus time for the data above.

(c) What are the quantities in part (a) at times 1.2, 4.3, and 13.0?

(a) Survival function for discrete survival time T:

$$S(t) = \sum_{a_j \geqslant t} \mathsf{P}(T = a_j) = \begin{cases} 1 & \text{if } t \in (-\infty, 1]; \\ a_2 + a_3 + a_4 + a_5 + a_6 = 1 - a_1 = 0.6 & \text{if } t \in (1, 3]; \\ a_3 + a_4 + a_5 + a_6 = 0.35 & \text{if } t \in (3, 4]; \\ a_4 + a_5 + a_6 = 0.25 & \text{if } t \in (4, 5]; \\ a_5 + a_6 = 0.15 & \text{if } t \in (5, 10]; \\ a_6 = 0.05 & \text{if } t \in (10, 12]; \\ 0 & \text{if } t \in (12, +\infty). \end{cases}$$

This survival function is left-continuous, because a definition used is $S(t) := \mathsf{P}(T \geqslant t)$. Using the other version of the definition of survival function one could end up with almost the same result, only the boundaries in the steps would exchange—a right-continous function.

Hazard function for T:

$$\lambda(t) = \begin{cases} \frac{\mathsf{P}(T=a_1)}{S(a_1)} = 0.4 & \text{if } t = a_1 = 1; \\ \frac{\mathsf{P}(T=a_2)}{S(a_2)} = \frac{5}{12} \approx 0.4167 & \text{if } t = a_2 = 3; \\ \frac{\mathsf{P}(T=a_3)}{S(a_3)} = \frac{2}{7} \approx 0.2857 & \text{if } t = a_3 = 4; \\ \frac{\mathsf{P}(T=a_4)}{S(a_4)} = \frac{2}{5} = 0.4 & \text{if } t = a_4 = 5; \\ \frac{\mathsf{P}(T=a_5)}{S(a_5)} = \frac{2}{3} \approx 0.6667 & \text{if } t = a_5 = 10; \\ \frac{\mathsf{P}(T=a_6)}{S(a_6)} = 1 & \text{if } t = a_6 = 12; \\ 0 & \text{if otherwise .} \end{cases}$$

Density f for a discrete random variable is positive only in a_j's (the step points of distribution function). Also if $\mathsf{P}(T \geqslant a_j) = 0$ then we define $\mathsf{P}(T = a_j | T \geqslant a_j) = 0$. Conditioning by an event with probability zero could be seen meaningless, but we do it for completeness (it is absolutely consistent with the whole theory of probability).

Cumulative hazard function for T:

$$\Lambda(t) = \sum_{a_j < t} \lambda(a_j) = \begin{cases} 0 & \text{if } t \in (-\infty, 1]; \\ 0.4 & \text{if } t \in (1, 3]; \\ \frac{49}{60} \approx 0.8167 & \text{if } t \in (3, 4]; \\ \frac{463}{420} \approx 1.102 & \text{if } t \in (4, 5]; \\ \frac{631}{420} \approx 1.502 & \text{if } t \in (5, 10]; \\ \frac{911}{420} \approx 2.169 & \text{if } t \in (10, 12]; \\ \frac{1331}{420} \approx 3.169 & \text{if } t \in (12, +\infty). \end{cases}$$

(b) Survival function S for our data is shown in Figure 3.2.

(c) Values of three quantities mentioned in part (a) at times 1.2, 4.3 and 13.0 are shown in Table 3.1.

Time t	1.2	4.3	13.0
Survival function S	0.6	0.25	0
Hazard function λ	0	0	0
Cumulative hazard function Λ	0.4	1.102	3.169

Table 3.1: Values of survival function S, hazard function λ and cumulative hazard function Λ for survival time T at times 1.2, 4.3 and 13.0.

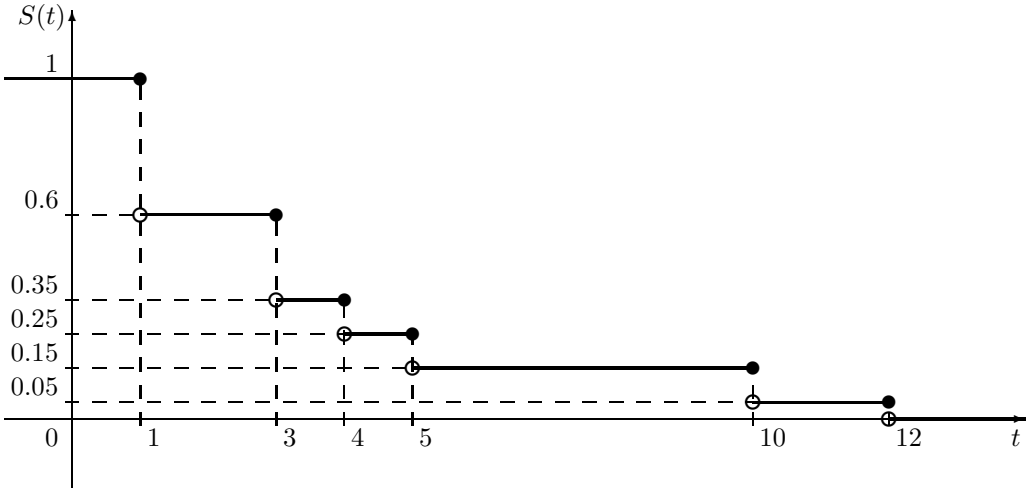

Figure 3.2: Survival function $S(t)$ versus time t.

Exercise 3.2. Given $\gamma > 0$ and the survival function for the random variable T

$$S_T(t) = \exp\{-t\gamma\}, \quad t > 0$$

derive

(a) the probability density function,

(b) the hazard function of T,

(c) find the distribution of $Y = T^\gamma$. Hint: compute S_Y.

(a) Probability density function for the continuous random variable T:

$$f_T(t) = -S_T'(t) = \left\{ \begin{array}{l} |\gamma = 1| = \exp\{-t\} \\ |\gamma \neq 1| = -\exp\{-t^\gamma\}\left[-\gamma t^{\gamma-1}\right] \end{array} \right\} = \gamma t^{\gamma-1} \exp\{-t^\gamma\}, \quad t > 0.$$

(b) Hazard function of T:

$$\lambda_T(t) = -\frac{\mathrm{d}}{\mathrm{d}t}\left[\log S_T(t)\right] = \gamma t^{\gamma-1}, \quad t > 0.$$

(c) One way how to obtain the distribution of $Y := T^\gamma$ (characterized by its density) is applying transformation theorem (change of variables formula) with corresponding Jacobian $\mathcal{J}(y) = \frac{1}{\gamma} y^{\frac{1-\gamma}{\gamma}}$ (for $\gamma \neq 1$, otherwise it is trivial) and get

$$f_Y(y) = f_T\left(y^{\frac{1}{\gamma}}\right)|\mathcal{J}(y)| = \exp\{-y\}, \quad y > 0$$

and hence Y is exponentially distributed, e.g. $Y \sim \mathsf{Exp}(1)$.

The other way to obtain the distribution of Y (characterized by its distribution function) is computing its survival function S_Y:

$$S_Y(y) = \mathsf{P}(Y \geqslant y) = \mathsf{P}\left(Y^{\frac{1}{\gamma}} \geqslant y^{\frac{1}{\gamma}}\right) = \mathsf{P}\left(T \geqslant y^{\frac{1}{\gamma}}\right) = S_T\left(y^{\frac{1}{\gamma}}\right) = \exp\{-y\}, \quad y > 0$$

and hence we can get distribution function for exponentially distributed random variable Y:

$$F_Y(y) = 1 - S_Y(y) = 1 - \exp\{-y\}, \quad y > 0.$$

Exercise 3.3. Given a hazard function for a survival time T

$$\log \lambda(t) = \lambda_0 + \lambda_1 (t - \tau)_+,$$

where τ, λ_0, and λ_1 are constant, $\tau > 0$, and $x_+ = x$ if $x > 0$, and 0 otherwise, derive

(a) the cumulative hazard,

(b) the survival function,

(c) the probability density function,

(d) the cumulative distribution function for T.

(a) Cumulative hazard for T:

$$\Lambda(t) = \int_0^t \exp\{\lambda_0 + \lambda_1 (u - \tau)_+\}\, \mathrm{d}u = \begin{cases} \int_0^t \exp\{\lambda_0\}\, \mathrm{d}u = te^{\lambda_0} & \text{if } 0 \leqslant t \leqslant \tau; \\ e^{\lambda_0} \int_0^\tau \mathrm{d}u + e^{\lambda_0} \int_\tau^t e^{\lambda_1 (u-\tau)} \mathrm{d}u & \\ = \tau e^{\lambda_0} + e^{\lambda_0 - \lambda_1 \tau} \left[\frac{e^{\lambda_1 u}}{\lambda_1}\right]_\tau^t & \\ = \tau e^{\lambda_0} - \frac{e^{\lambda_0}}{\lambda_1} + \frac{e^{\lambda_0 + \lambda_1 (t-\tau)}}{\lambda_1} & \text{if } t > \tau. \end{cases}$$

(b) Survival function for T:

$$S(t) = \exp\{-\Lambda(t)\} = \begin{cases} \exp\{-t \exp\{\lambda_0\}\} & \text{if } 0 \leqslant t \leqslant \tau; \\ \exp\left\{\frac{\exp\{\lambda_0\}}{\lambda_1} - \frac{\exp\{\lambda_0 + \lambda_1 (t-\tau)\}}{\lambda_1} - \tau \exp\{\lambda_0\}\right\} & \text{if } t > \tau. \end{cases}$$

(c) Probability density function for T:

$$f(t) = \lambda(t) S(t) = \begin{cases} \exp\left\{\lambda_0 - te^{\lambda_0}\right\} & \text{if } 0 \leqslant t \leqslant \tau; \\ \exp\left\{\lambda_0 + \lambda_1 (t-\tau) - \tau e^{\lambda_0} + \frac{e^{\lambda_0}}{\lambda_1} - \frac{e^{\lambda_0 + \lambda_1 (t-\tau)}}{\lambda_1}\right\} & \text{if } t > \tau. \end{cases}$$

The same result can be obtained using $f(t) = -S'(t)$.

(d) Cumulative distribution function for T:

$$F(t) = 1 - S(t) = \begin{cases} 1 - \exp\{-t\exp\{\lambda_0\}\} & \text{if } 0 \leqslant t \leqslant \tau; \\ 1 - \exp\left\{\frac{\exp\{\lambda_0\}}{\lambda_1} - \frac{\exp\{\lambda_0+\lambda_1(t-\tau)\}}{\lambda_1} - \tau\exp\{\lambda_0\}\right\} & \text{if } t > \tau. \end{cases}$$

Exercise 3.4. Consider a discrete survival time T taking nonnegative values $a_1 < a_2 < \ldots < a_n < \ldots$. Show that for $a_j < t \leqslant a_{j+1}$ holds

(a) $\mathsf{P}[T \geqslant t] = \mathsf{P}[T \geqslant a_1, \ldots, T \geqslant a_{j+1}]$,

(b) $\mathsf{P}[T \geqslant t] = \mathsf{P}[T \geqslant a_1]\mathsf{P}[T \geqslant a_2|T \geqslant a_1]\ldots\mathsf{P}[T \geqslant a_{j+1}|T \geqslant a_j]$. Hint: use multiplicative law of probability.

(a) Since $a_{j+1} > a_j > \ldots > a_1$ then

$$\mathsf{P}\left(T \geqslant a_1, \ldots, T \geqslant a_{j+1}\right) = \mathsf{P}\left(T \geqslant a_{j+1}\right) \stackrel{def}{=} \sum_{i:a_i \geqslant a_{j+1}} \mathsf{P}\left(T = a_i\right).$$

There is none a_i, $i = 1, \ldots, j$ between t and a_{j+1} and therefore for our index set holds

$$\{i : a_i \geqslant a_{j+1}\} = \{j+1, j+2, \ldots\} = \{i : a_i \geqslant t\},$$

because these two (actually only one) sets of indices contain the same elemets. Hence

$$\mathsf{P}\left(T \geqslant a_1, \ldots, T \geqslant a_{j+1}\right) = \sum_{i:a_i \geqslant t} \mathsf{P}\left(T = a_i\right) = \mathsf{P}\left(T \geqslant t\right).$$

(b) To prove this simple property, we use mathematical induction. For $j = 2$ our property simply follows from the definition of conditional probability

$$\mathsf{P}\left(T \geqslant a_1, T \geqslant a_2\right) = \mathsf{P}\left(T \geqslant a_1\right)\mathsf{P}\left(T \geqslant a_2|T \geqslant a_1\right).$$

Let us suppose that for all $k \in \{2, \ldots, j\}$ the following formula holds

$$\mathsf{P}\left(T \geqslant a_1, \ldots, T \geqslant a_j\right) = \mathsf{P}\left(T \geqslant a_1\right)\prod_{k=2}^{j}\mathsf{P}\left(T \geqslant a_k|T \geqslant a_{k-1}\right).$$

Now we just proceed the induction step by combining the definition of conditional probability, the equality for two random events $[T \geqslant a_j] \equiv [T \geqslant a_1, \ldots, T \geqslant a_j]$ (due to descending

ordering) and the induction hypothesis

$$\begin{aligned}\mathsf{P}(T \geqslant a_1, \ldots, T \geqslant a_{j+1}) &= \mathsf{P}(T \geqslant a_1, \ldots, T \geqslant a_j)\,\mathsf{P}(T \geqslant a_{j+1} | T \geqslant a_1, \ldots, T \geqslant a_j) \\ &= \mathsf{P}(T \geqslant a_1) \prod_{k=2}^{j} \mathsf{P}(T \geqslant a_k | T \geqslant a_{k-1}) \\ &\quad \times \mathsf{P}(T \geqslant a_{j+1} | T \geqslant a_j) \\ &= \mathsf{P}(T \geqslant a_1) \prod_{k=2}^{j+1} \mathsf{P}(T \geqslant a_k | T \geqslant a_{k-1})\,.\end{aligned}$$

Exercise 3.5. Consider the survival times (in months) of 30 melanoma patients given below.

- Group 1 (BCG, Bacillus Calmette-Guerin, treatment): 33.7+, 3.9, 10.5, 5.4, 19.5, 23.8+, 7.9, 16.9+, 16.6+, 33.7+, 17.1+.
- Group 2 (C. parvum treatment): 8.0, 26.9+, 21.4+, 18.1+, 16.0+, 6.9, 11.0+, 24.8+, 23.0+, 8.3, 10.8+, 12.2+, 12.5+, 24.4, 7.7, 14.8+, 8.2+, 8.2+, 7.8+.

(a) Ignore the censoring indicators for treatment group 1 (BCG) for the moment (that is, treat all times as failure times in group 1). Compute and plot by hand the KM estimator of the survivor function for this treatment group.

(b) Now use the censoring indicators as given in the table and compute and plot on the same graph the KM estimator for group 1. Interpret any differences between the curves.

(c) Compute by hand 95% confidence limits on the survivor function at times $t = 5, 10, 20$ years for treatment group 1 using Greenwood's formula. Comment on the adequacy of the limits.

(d) Using R (or any other software), make a plot of the KM estimators for treatment groups 1 and 2. Comment on which treatment appears better.

(e) Compute (by hand) the median survival for each group and the 1-year survival for each group. Which statistic (median or 1-year survival) do you think is more adequate for these data? Why?

(a) The calculation of the Kaplan-Meier estimator of the survivorship function $S(t)$ for BCG data set (ignoring the censoring) is shown in Table 3.2. By ignoring the censoring the Kaplan-Meier estimator become exactly the same as the empirical distribution function of T.

According to Table 3.2, the Kaplan-Meier estimator $\hat{S}$ of the survivorship function for

τ_j	d_j	c_j	r_j	$1-\frac{d_j}{r_j}$	$\hat{S}(\tau_j^+)$
3.9	1	0	11	$\frac{10}{11}$	$\frac{10}{11}$
5.4	1	0	10	$\frac{9}{10}$	$\frac{9}{11}$
7.9	1	0	9	$\frac{8}{9}$	$\frac{8}{11}$
10.5	1	0	8	$\frac{7}{8}$	$\frac{7}{11}$
16.6	1	0	7	$\frac{6}{7}$	$\frac{6}{11}$
16.9	1	0	6	$\frac{5}{6}$	$\frac{5}{11}$
17.1	1	0	5	$\frac{4}{5}$	$\frac{4}{11}$
19.5	1	0	4	$\frac{3}{4}$	$\frac{3}{11}$
23.8	1	0	3	$\frac{2}{3}$	$\frac{2}{11}$
33.7	2	0	2	0	0

Table 3.2: Calculation of the Kaplan-Meier estimator of the survivorship function for BCG data set.

BCG data set (ignoring the censoring) is ploted in Figure 3.3.

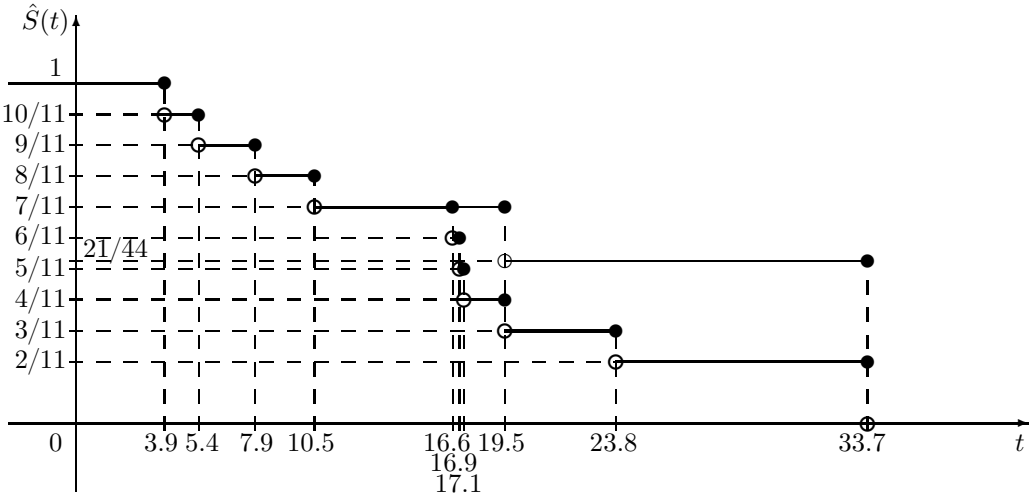

Figure 3.3: Kaplan-Meier estimator of the survivorship function for BCG data set ignoring the censoring (thick line) and for original BCG data set taking into account the censoring (thin line).

(b) The calculation of the Kaplan-Meier estimator of the survivorship function $S(t)$ for original BCG data set (taking into account the censoring) is shown in Table 3.3.

The difference between two curves in Figure 3.3—the Kaplan-Meier estimators of the survivorship function for BCG data set ignoring the censoring and original BCG data set—could be predicted. Due to ignoring the censoring (all times treaten as failure times) makes all the values $\hat{S}(t)$ of the KM estimator for the original data set bigger or at most equal

τ_j	d_j	c_j	r_j	$1-\frac{d_j}{r_j}$	$\hat{S}(\tau_j^+)$
3.9	1	0	11	$\frac{10}{11}$	$\frac{10}{11}$
5.4	1	0	10	$\frac{9}{10}$	$\frac{9}{11}$
7.9	1	0	9	$\frac{8}{9}$	$\frac{8}{11}$
10.5	1	0	8	$\frac{7}{8}$	$\frac{7}{11}$
19.5	1	3	4	$\frac{3}{4}$	$\frac{21}{44}$

Table 3.3: Calculation of the Kaplan-Meier estimator of the survivorship function for original BCG data set.

than the values of the KM estimator for modified BCG data set, because the proportion of patiens 'from original data' alive in a particular time t have to be bigger or at most equal than for patients 'from modified data'—here some patients that should be alive (and are censored) are strictly considered as dead ones. Simply speaking, the KM estimator for the original data set does not have steps at those time points (the time points of censoring) where the KM estimator for the modified data set has these steps. On the other hand, the KM estimator for the original data set 'ends' at the last time point, because the last time point is a censored time point and we cannot say that from this point on the estimator will be equal zero (like in the case of modified data set).
(c) With respect to already computed values in Table 3.3, one can calculate standard errors for survival estimate at 5, 10 and 20 months using Greenwood's formula:

$$\begin{aligned}
\mathsf{se}\left[\hat{S}(5)\right] &= \sqrt{\widehat{\mathsf{Var}}\left(\hat{S}(5)\right)} = \left[\hat{S}(5)\right]\sqrt{\sum_{j:\tau_j<5}\frac{d_j}{(r_j-d_j)r_j}} \\
&= \frac{10}{11}\sqrt{\frac{1}{(11-1)\times 11}} = \sqrt{\frac{10}{11^3}} \approx 0.08668; \\
se\left[\hat{S}(10)\right] &= \frac{8}{11}\sqrt{\left(\frac{1}{(11-1)\times 11}+\frac{1}{(10-1)\times 10}+\frac{1}{(9-1)\times 9}\right)} \approx 0.13428; \\
se\left[\hat{S}(20)\right] &= \frac{21}{44}\sqrt{\left(\frac{1}{110}+\frac{1}{90}+\frac{1}{72}+\frac{1}{56}+\frac{1}{12}\right)} \approx 0.17554.
\end{aligned}$$

For constructing 95% confidence limits, we can use direct method $\hat{S}(t) \pm z_{0.975} se\left[\hat{S}(t)\right]$ or a transformation of $S(t)$ approach (log or log-log)—stabilizing variance, allowing for non-symmetric confidence intervals and log-log transformation also yields values of the limits within $[0,1]$. The confidence intervals for all three methods are shown in Table 3.4.

Direct method is not very plausible and satisfactory because of above mentioned general disadvantages (lack of advanteges of log or log-log transformations).
(d) A plot of the KM estimators for treatment groups 1 and 2 is shown in Figure 3.4.

Approach	Time t [months]	95% lower CI	$\hat{S}(t)$	95% upper CI
direct method	5	0.739	0.909	1.000
	10	0.464	0.727	0.990
	20	0.133	0.477	0.821
log transformation	5	0.754	0.909	1.000
	10	0.506	0.727	1.000
	20	0.232	0.477	0.981
log-log transformation	5	0.508	0.909	0.987
	10	0.371	0.727	0.903
	20	0.141	0.477	0.756

Table 3.4: 95% confidence limits for survival in BCG data set.

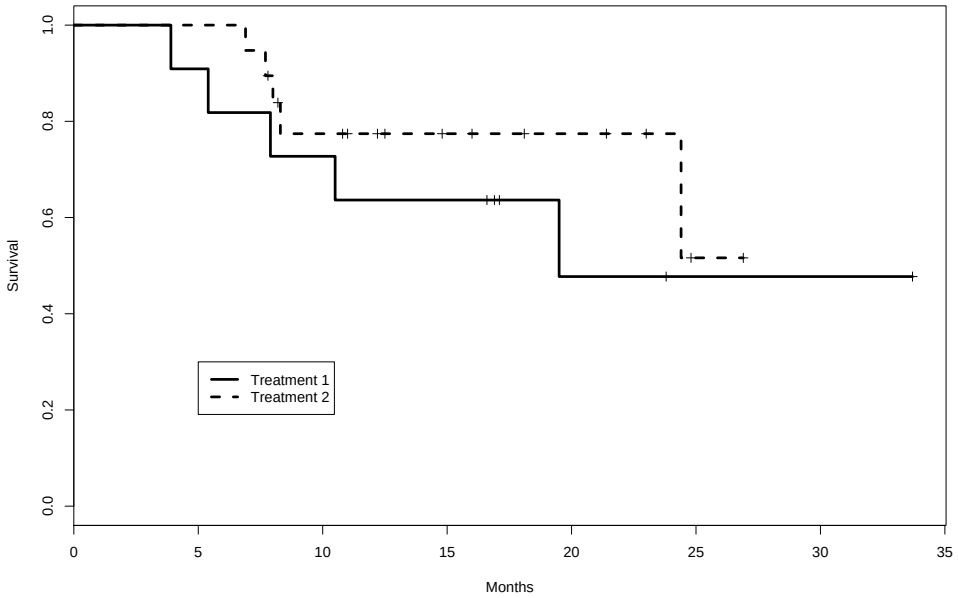

Figure 3.4: KM estimators for treatment groups 1 and 2.

Strictly according to the appearance of the two KM estimators ploted in Figure 3.4, it can be concluded that treatment 2 is better that treatment 1 in terms of survival time. The reason is simple—at each time point, there is bigger proportion of patients alive in treatment group 2 than in treatment group 1.

(e) The median survival for treatment group 1 is 19.5 months, but for treatment group 2 it cannot be computed—is not a relevant statistic for the second group. One-year survival for

treatment group 1 is equal 0.636 and for treatment group 2 is equal 0.774.

For these data, one-year survival is more appropriate and adequate than median. The main reason is that for the second treatment group median cannot be computed. And survival at one year is a reasonable characteristic, because 12 months is a 'meaningful' value according to provided survival times in the data.

Exercise 3.6. Construct KM estimator for the leukemia data set (see below) using software.

The file `leukAB.dat` contains the failure/censoring times and the event indicators for the Leukemia example (treatment arm).

```
# read dataset, skip 2 lines of file:
Link="http://miso.matfyz.cz/prednasky/NMFM404/Data/leukAB.dat"
leuk =
read.table(Link, sep=" ", head=T, skip=4)
# print a few observations; t = time, f = event indicator
leuk[1:3,]
```

```
  t f
1 6 0
2 6 1
3 6 1
```

```
fit.leuk = survfit(Surv(leuk$t, leuk$f) ~ 1)
summary(fit.leuk)
```

```
Call: survfit(formula = Surv(leuk$t, leuk$f) ~ 1)

 time n.risk n.event survival std.err lower 95% CI upper 95% CI
    6     21       3    0.857  0.0764        0.720        1.000
    7     17       1    0.807  0.0869        0.653        0.996
   10     15       1    0.753  0.0963        0.586        0.968
   13     12       1    0.690  0.1068        0.510        0.935
   16     11       1    0.627  0.1141        0.439        0.896
   22      7       1    0.538  0.1282        0.337        0.858
   23      6       1    0.448  0.1346        0.249        0.807
```

```
plot(fit.leuk, xlab="Weeks", ylab="Proportion Survivors", col=3)
```

A plot of the KM estimator for leukemia patients is shown in Figure 3.5.

Exercise 3.7. Calculate three types of confidence intervals for the KM estimator from the leukemia data set in Exercise 3.6 using software.

Confidence intervals for the KM using the log stabilizing transformation, which is default in *R*. Hence, `conf.type = "log"` can be omitted from the input:

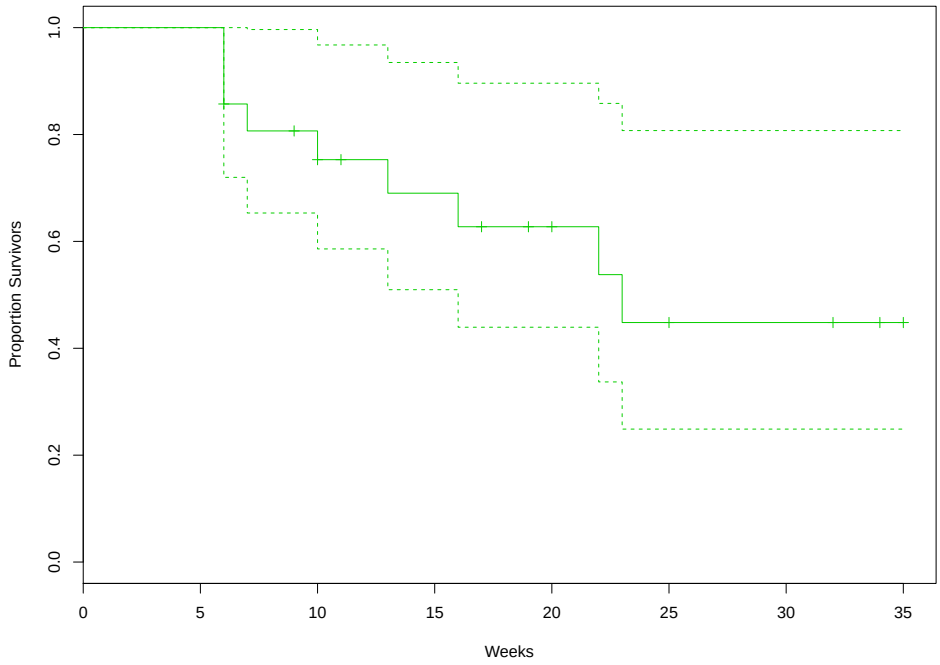

Figure 3.5: KM estimators for leukemia data.

```
fit.leuk = survfit(Surv(leuk$t, leuk$f) ~ 1, conf.type = "log")
summary(fit.leuk)
```

```
Call: survfit(formula=Surv(leuk$t,leuk$f)~1,conf.type="log")

 time n.risk n.event survival std.err lower 95% CI upper 95% CI
    6     21       3    0.857  0.0764        0.720        1.000
    7     17       1    0.807  0.0869        0.653        0.996
   10     15       1    0.753  0.0963        0.586        0.968
   13     12       1    0.690  0.1068        0.510        0.935
   16     11       1    0.627  0.1141        0.439        0.896
   22      7       1    0.538  0.1282        0.337        0.858
   23      6       1    0.448  0.1346        0.249        0.807
```

CIs using the *Greenwood's formula* (`conf.type = "plain"`):

```
fit.leuk2 = survfit(Surv(leuk$t, leuk$f) ~ 1, conf.type = "plain")
summary(fit.leuk2)
```

```
Call: survfit(formula=Surv(leuk$t,leuk$f)~1,conf.type="plain")
```

```
 time n.risk n.event survival std.err lower 95% CI upper 95% CI
    6     21       3    0.857  0.0764        0.707        1.000
    7     17       1    0.807  0.0869        0.636        0.977
   10     15       1    0.753  0.0963        0.564        0.942
   13     12       1    0.690  0.1068        0.481        0.900
   16     11       1    0.627  0.1141        0.404        0.851
   22      7       1    0.538  0.1282        0.286        0.789
   23      6       1    0.448  0.1346        0.184        0.712
```

CIs using the *log-log* transformation (`conf.type = "log-log"`):

```
fit.leuk3 = survfit(Surv(leuk$t,leuk$f)~1,conf.type="log-log")
summary(fit.leuk3)
```

```
Call: survfit(formula=Surv(leuk$t,leuk$f)~1,conf.type="log-log")

 time n.risk n.event survival std.err lower 95% CI upper 95% CI
    6     21       3    0.857  0.0764        0.620        0.952
    7     17       1    0.807  0.0869        0.563        0.923
   10     15       1    0.753  0.0963        0.503        0.889
   13     12       1    0.690  0.1068        0.432        0.849
   16     11       1    0.627  0.1141        0.368        0.805
   22      7       1    0.538  0.1282        0.268        0.747
   23      6       1    0.448  0.1346        0.188        0.680
```

Exercise 3.8. Constructing the lifetable for the car crash data below.

Constructing the lifetable in *R* with `KMsurv` package requires three elements:

- a vector of $k + 1$ interval endpoints,
- a k vector of death counts in each interval,
- a k vector of censored counts in each interval.

Incidentally, KM in `KMsurv` stands for Klein and Moeschberger, not Kaplan-Meier!

Car crash data contain numbers of the first car crash during the first, second, etc. year of the car's service (up to 16 years). Censorings can correspond to a fact that the car was sold or stolen.

```
### car crash data ###
intEndpts = 0:16    # interval endpoints
crash = c(456, 226, 152, 171, 135, 125, 83, 74, 51, 42, 43, 34, 18, 9, 6,
    0)
cens = c(0, 39, 22, 23, 24, 107, 133, 102, 68, 64, 45, 53, 33, 27, 23,
   30)
# n censored
```

```
fitlt = lifetab(tis = intEndpts, ninit=2418, nlost=cens, nevent=crash)
round(fitlt, 4) # restrict output to 4 decimals
```

```
       nsubs nlost  nrisk nevent   surv    pdf hazard se.surv se.pdf
0-1     2418     0 2418.0    456 1.0000 0.1886 0.2082  0.0000 0.0080
1-2     1962    39 1942.5    226 0.8114 0.0944 0.1235  0.0080 0.0060
2-3     1697    22 1686.0    152 0.7170 0.0646 0.0944  0.0092 0.0051
3-4     1523    23 1511.5    171 0.6524 0.0738 0.1199  0.0097 0.0054
4-5     1329    24 1317.0    135 0.5786 0.0593 0.1080  0.0101 0.0049
5-6     1170   107 1116.5    125 0.5193 0.0581 0.1186  0.0103 0.0050
6-7      938   133  871.5     83 0.4611 0.0439 0.1000  0.0104 0.0047
7-8      722   102  671.0     74 0.4172 0.0460 0.1167  0.0105 0.0052
8-9      546    68  512.0     51 0.3712 0.0370 0.1048  0.0106 0.0050
9-10     427    64  395.0     42 0.3342 0.0355 0.1123  0.0107 0.0053
10-11    321    45  298.5     43 0.2987 0.0430 0.1552  0.0109 0.0063
11-12    233    53  206.5     34 0.2557 0.0421 0.1794  0.0111 0.0068
12-13    146    33  129.5     18 0.2136 0.0297 0.1494  0.0114 0.0067
13-14     95    27   81.5      9 0.1839 0.0203 0.1169  0.0118 0.0065
14-15     59    23   47.5      6 0.1636 0.0207 0.1348  0.0123 0.0080
15-16     30    30   15.0      0 0.1429     NA     NA  0.0133     NA
      se.hazard
0-1      0.0097
1-2      0.0082
2-3      0.0076
3-4      0.0092
4-5      0.0093
5-6      0.0106
6-7      0.0110
7-8      0.0135
8-9      0.0147
9-10     0.0173
10-11    0.0236
11-12    0.0306
12-13    0.0351
13-14    0.0389
14-15    0.0549
15-16        NA
```

Exercise 3.9. Test whether there is a significant difference in survival time with respect to the treatment for the leukemia data from Exercise 3.6.

```
Link3 = "http://miso.matfyz.cz/prednasky/NMFM404/Data/leuk.dat"
leuk = read.table(Link3, sep=" ", head=T, skip=7)
```

```
summary(leuk)
```

```
     weeks           remiss           trtmt
 Min.   : 1.00   Min.   :0.0000   Min.   :0.0
 1st Qu.: 6.00   1st Qu.:0.0000   1st Qu.:0.0
 Median :10.50   Median :1.0000   Median :0.5
 Mean   :12.88   Mean   :0.7143   Mean   :0.5
 3rd Qu.:18.50   3rd Qu.:1.0000   3rd Qu.:1.0
 Max.   :35.00   Max.   :1.0000   Max.   :1.0
```

Using Cochran-Mantel-Haenszel logrank test:

```
(leuk.lrt = survdiff(Surv(weeks,remiss) ~ trtmt, data = leuk))
```

```
Call:
survdiff(formula = Surv(weeks, remiss) ~ trtmt, data = leuk)

         N Observed Expected (O-E)^2/E (O-E)^2/V
trtmt=0 21       21     10.7      9.77      16.8
trtmt=1 21        9     19.3      5.46      16.8

 Chisq= 16.8  on 1 degrees of freedom, p= 4.17e-05
```

We reject the null. So, there is a significant effect of treatment on the survival time.

Using Peto-Peto-Prentice-Gehan-Wilcoxon rank test:

```
## notice: rho=1
(leuk.pppgw = survdiff(Surv(weeks,remiss) ~ trtmt, data = leuk, rho=1))
```

```
Call:
survdiff(formula = Surv(weeks, remiss) ~ trtmt, data = leuk,
    rho = 1)

         N Observed Expected (O-E)^2/E (O-E)^2/V
trtmt=0 21    14.55     7.68      6.16      14.5
trtmt=1 21     5.12    12.00      3.94      14.5

 Chisq= 14.5  on 1 degrees of freedom, p= 0.000143
```

We reject the null as well. Additionally, one can be interested in plotting the KM for the treated and untreated arm.

```
leuk.km = survfit(Surv(weeks, remiss) ~ trtmt, data=leuk)
plot(leuk.km, xlab = "Weeks", ylab = "Survival", col=c(2,3))
legend(18, .95, legend=c("No treatment", "Treatment"),
  col=c(2:3), lty=1)
title(main="Treatment Comparison, Leukemia Data", cex=.7)
```

Estimated survival curves in Figure 3.6 assures ourselves that there is a significant difference in survival for the treated and untreated patients with leukemia.

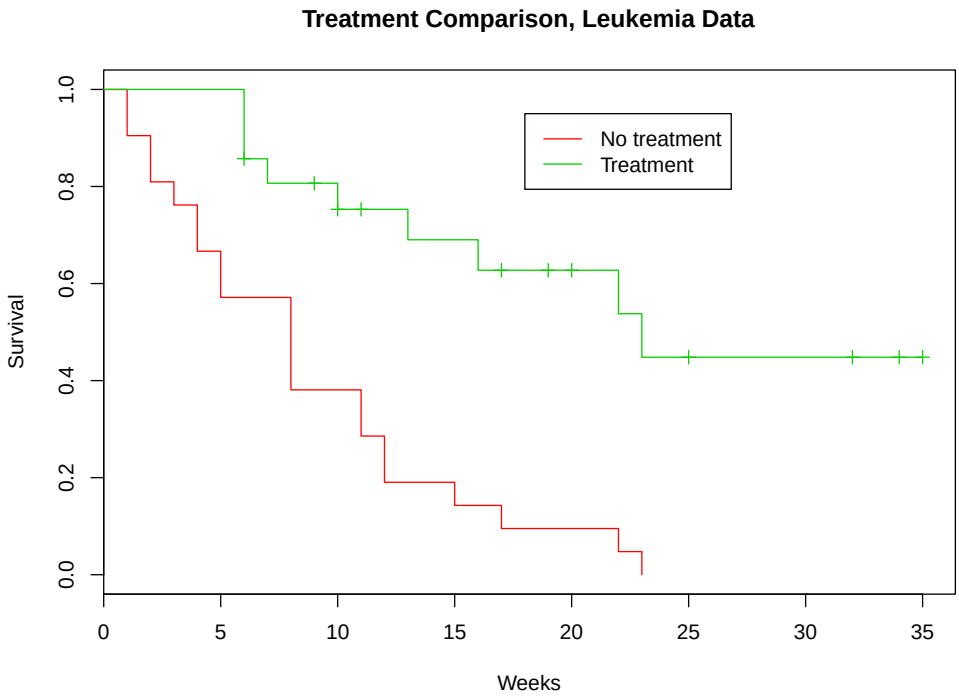

Figure 3.6: Comparing survival curves for leukemia data.

Exercise 3.10. Time taken to finish a test with 3 different noise distractions is recorded (see below). All tests were stopped after 12 minutes. Perform the three sample logrank test.

```
time = c(9, 9.5, 9, 8.5, 10, 10.5, 10, 12, 12, 11, 12,
  10.5, 12, 12, 12, 12, 12, 12)
event = c(1,1,1,1,1,1,1,1,0,1,1,1,1,0,0,0,0,0)
group = rep(1:3, times=c(6,6,6))
(cbind(group,time,event))
```

```
      group time event
 [1,]     1  9.0     1
 [2,]     1  9.5     1
 [3,]     1  9.0     1
 [4,]     1  8.5     1
 [5,]     1 10.0     1
```

```
 [6,]     1 10.5     1
 [7,]     2 10.0     1
 [8,]     2 12.0     1
 [9,]     2 12.0     0
[10,]     2 11.0     1
[11,]     2 12.0     1
[12,]     2 10.5     1
[13,]     3 12.0     1
[14,]     3 12.0     0
[15,]     3 12.0     0
[16,]     3 12.0     0
[17,]     3 12.0     0
[18,]     3 12.0     0
```

```
(noise.lrt = survdiff(Surv(time, event) ~ group, rho=0))
```

```
Call:
survdiff(formula = Surv(time, event) ~ group, rho = 0)

        N Observed Expected (O-E)^2/E (O-E)^2/V
group=1 6        6     1.57   12.4463   17.2379
group=2 6        5     4.53    0.0488    0.0876
group=3 6        1     5.90    4.0660    9.4495

 Chisq= 20.4  on 2 degrees of freedom, p= 3.75e-05
```

We reject the null. Hence, there is a significant impact of the noise distraction.

```
noise.km = survfit(Surv(time, event) ~ group)
plot(noise.km, col = 2:4, xlab = "Minutes",
  ylab = "Time to finish test",
  main = "Time to finish test for 3 noise levels")
```

Corresponding estimated survival curves are shown in Figure 3.7.

Exercise 3.11. Fit a Cox PH model to the leukemia data from Exercise 3.6 using Efron's, Breslow's and exact method of handling the ties.

```
LinkL="http://miso.matfyz.cz/prednasky/NMFM404/Data/leuk.dat"
leuk = read.table(LinkL, sep=" ", head=T, skip=7) # read data
head(leuk)
```

```
  weeks remiss trtmt
1     6      0     1
2     6      1     1
```

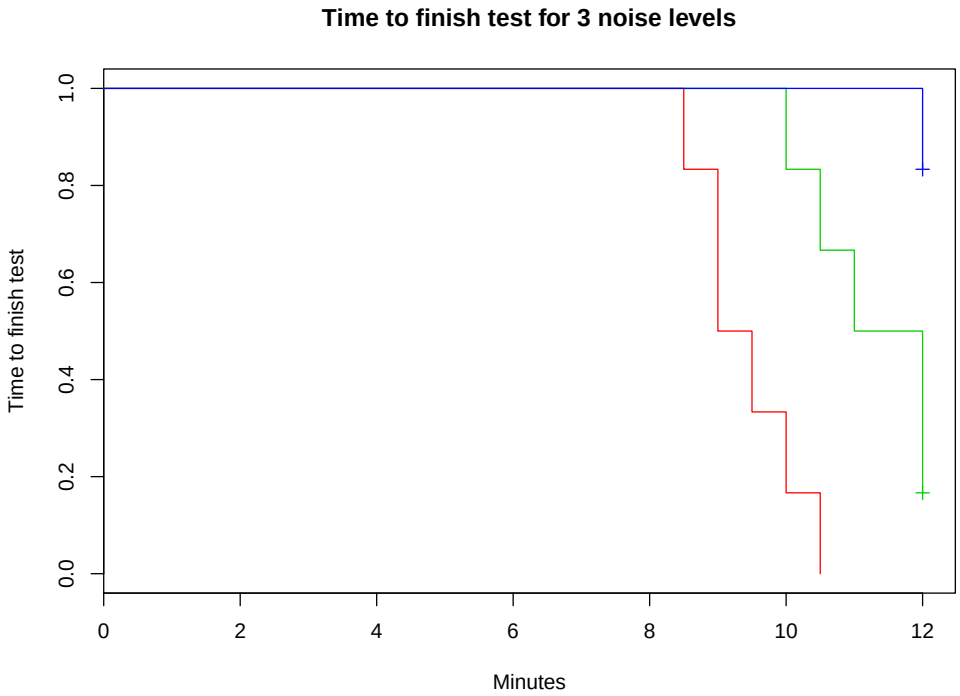

Figure 3.7: Comparing survival curves for noise distraction data.

```
3      6       1      1
4      6       1      1
5      7       1      1
6      9       0      1
```

```
# Cox PH Model
leuk.ph = coxph(Surv(weeks,remiss) ~ trtmt, data=leuk)
# Note: default = Efron method for handling ties
summary(leuk.ph)
```

```
Call:
coxph(formula = Surv(weeks, remiss) ~ trtmt, data = leuk)

  n= 42, number of events= 30

          coef exp(coef) se(coef)      z Pr(>|z|)
trtmt -1.5721    0.2076   0.4124 -3.812 0.000138 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
      exp(coef) exp(-coef) lower .95 upper .95
trtmt    0.2076      4.817   0.09251    0.4659

Concordance= 0.69  (se = 0.053 )
Rsquare= 0.322   (max possible= 0.988 )
Likelihood ratio test= 16.35  on 1 df,   p=5.261e-05
Wald test            = 14.53  on 1 df,   p=0.0001378
Score (logrank) test = 17.25  on 1 df,   p=3.283e-05
```

```
# Breslow handling of ties
leuk.phb=coxph(Surv(weeks,remiss)~trtmt,data=leuk,method="breslow")
summary(leuk.phb)
```

```
Call:
coxph(formula = Surv(weeks, remiss) ~ trtmt, data = leuk, method = "
    breslow")

  n= 42, number of events= 30

         coef exp(coef) se(coef)      z Pr(>|z|)
trtmt -1.5092    0.2211   0.4096 -3.685 0.000229 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

      exp(coef) exp(-coef) lower .95 upper .95
trtmt    0.2211      4.523   0.09907    0.4934

Concordance= 0.69  (se = 0.053 )
Rsquare= 0.304   (max possible= 0.989 )
Likelihood ratio test= 15.21  on 1 df,   p=9.615e-05
Wald test            = 13.58  on 1 df,   p=0.0002288
Score (logrank) test = 15.93  on 1 df,   p=6.571e-05
```

```
# Exact handling of ties
leuk.phe = coxph(Surv(weeks, remiss) ~ trtmt, data=leuk, method="exact")
summary(leuk.phe)
```

```
Call:
coxph(formula = Surv(weeks, remiss) ~ trtmt, data = leuk, method = "exact
    ")

  n= 42, number of events= 30

         coef exp(coef) se(coef)      z Pr(>|z|)
```

```
trtmt -1.6282    0.1963   0.4331 -3.759  0.00017 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

      exp(coef) exp(-coef) lower .95 upper .95
trtmt    0.1963      5.095   0.08398    0.4587

Rsquare= 0.321   (max possible= 0.98 )
Likelihood ratio test= 16.25  on 1 df,   p=5.544e-05
Wald test            = 14.13  on 1 df,   p=0.0001704
Score (logrank) test = 16.79  on 1 df,   p=4.169e-05
```

Based on the Cox PH model, it can be concluded that there is a significant effect of treatment ($\widehat{\beta}_1 \approx -1.5$ or -1.6 for all three methods of handling ties with the corresponding p-value < 0.001).

3.3.2 SDA case studies

Duration of nursing home stay data

The National Center for Health Services Research studied 36 for-profit nursing homes to assess the effects of different financial incentives on length of stay. 'Treated' nursing homes received higher per diems for Medicaid patients, and bonuses for improving a patient's health and sending them home. Study included 1601 patients admitted between May 1, 1981 and April 30, 1982. Variables include:

- *los* ... Length of stay of a resident (in days),
- *age* ... Age of a resident,
- *rx* ... Nursing home assignment (1:bonuses, 0:no bonuses),
- *gender* ... Gender (1:male, 0:female),
- *married* ... (1:married, 0:not married),
- *health* ... Health status (2:second best, 5:worst),
- *censor* ... Censoring indicator (1:censored, 0:discharged).

Case study 3.1. Perform the following steps within the survival data analysis for the duration of nursing home stay:

- construct the lifetable (the actuarial method), when the data are not grouped,
- based on the previous lifetable, calculate the estimated survival and estimated hazard,

- obtain the Fleming-Harrington estimate of the survival function for married females, in the healthiest initial subgroup, who are randomized to the untreated group of the nursing home study,
- compare the above mentioned FH estimate with the corresponding Kaplan-Meier estimate,
- perform a stratified logrank test with respect to gender in order to test whether there is an effect of under 85 years of age or not,
- fit a Cox proportional hazards model when married and health status as regressors are taken into account,
- from the previously fitted Cox PH model, predict the survival for single individuals with the worst and second worst health status,
- get the baseline survival and plot the cumulative hazard.

We wish to construct the lifetable (the actuarial method), but the data *do not come grouped*. Consider the treated nursing home patients, with length of stay (*los*) grouped into 100 day intervals:

```
Link2 = "http://miso.matfyz.cz/prednasky/NMFM404/Data/NursingHome.dat"
nurshome = read.table(Link2, head=F, skip=14, col.names = c("los", "age",
    "rx", "gender", "married", "health", "censor"))
nurshome$int = floor(nurshome$los/100)*100 # group in 100 day intervals
nurshome = nurshome[nurshome$rx==1,] # keep only treated homes
nurshome[1:3,] # check out a few observations
```

```
   los age rx gender married health censor  int
1   37  86  1      0       0      2      0    0
2   61  77  1      0       0      4      0    0
3 1084  75  1      0       0      3      1 1000
```

```
tab = table(nurshome$int, nurshome$censor)
intEndpts = (0:11)*100
ntotal = dim(nurshome)[1]  # nr patients
cens = tab[,2]   # nr censored in each interval
released = tab[,1]   # nr released in each interval
fitlt = lifetab(tis = intEndpts, ninit=ntotal, nlost=cens, nevent=
   released)
round(fitlt, 4)        # restrict output to 4 decimals
```

```
          nsubs nlost nrisk nevent   surv    pdf hazard se.surv
0-100       712     0 712.0    330 1.0000 0.0046 0.0060  0.0000
100-200     382     0 382.0     86 0.5365 0.0012 0.0025  0.0187
```

```
200-300     296     0 296.0     65 0.4157 0.0009 0.0025  0.0185
300-400     231     0 231.0     38 0.3244 0.0005 0.0018  0.0175
400-500     193     1 192.5     32 0.2711 0.0005 0.0018  0.0167
500-600     160     0 160.0     13 0.2260 0.0002 0.0008  0.0157
600-700     147     0 147.0     13 0.2076 0.0002 0.0009  0.0152
700-800     134    30 119.0     10 0.1893 0.0002 0.0009  0.0147
800-900      94    29  79.5      4 0.1734 0.0001 0.0005  0.0143
900-1000     61    30  46.0      4 0.1647 0.0001 0.0009  0.0142
1000-1100    27    27  13.5      0 0.1503     NA     NA  0.0147
          se.pdf se.hazard
0-100      2e-04     3e-04
100-200    1e-04     3e-04
200-300    1e-04     3e-04
300-400    1e-04     3e-04
400-500    1e-04     3e-04
500-600    1e-04     2e-04
600-700    1e-04     3e-04
700-800    0e+00     3e-04
800-900    0e+00     3e-04
900-1000   1e-04     5e-04
1000-1100     NA        NA
```

To calculate the estimated survival and display it (Figure 3.8):

```
names(fitlt)  # check out components of fitlt object
x = rep(intEndpts, rep(2,12))[2:23]
y = rep(fitlt$surv, rep(2,11))
plot(x, y, type="l", col=4, xlab="Time Interval (Days)", ylab="Survival (
    Life Table)")
title(main = "Duration of stay in nursing homes", cex=.6)
```

To calculate the estimated hazard and display it (Figure 3.9):

```
y = rep(fitlt$hazard, rep(2,11))
plot(x, y, type="l", col=6, xlab="Time Interval (Days)", ylab="Hazard (
    Life Table)")
title(main = "Duration of stay in nursing homes", cex=.6)
```

Say we want to obtain the Fleming-Harrington estimate of the survival function for married females, in the healthiest initial subgroup, who are randomized to the untreated group of the nursing home study.

```
nurshome = read.table(Link2,
  head=F, skip=14, col.names = c("los", "age", "rx", "gender", "married",
     "health", "censor"))
(nurs2 = subset(nurshome, gender==0 & rx==0 & health==2 & married==1))
```

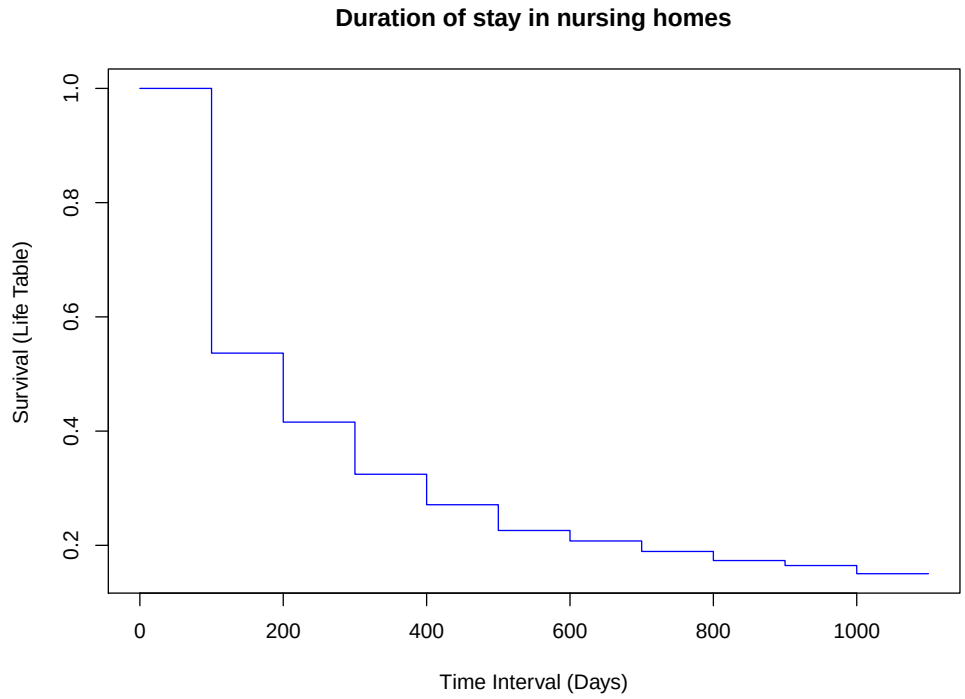

Figure 3.8: Estimated survival for nursing home data.

```
     los age rx gender married health censor
144   89  78  0      0       1      2      0
362  123  75  0      0       1      2      0
427   38  78  0      0       1      2      0
601   24  82  0      0       1      2      0
719  185  86  0      0       1      2      0
736  113  73  0      0       1      2      0
1120  25  71  0      0       1      2      0
1343  14  73  0      0       1      2      0
1362 149  81  0      0       1      2      0
1461 168  72  0      0       1      2      0
1472  64  81  0      0       1      2      0
1489 234  72  0      0       1      2      0
```

```
fit.fh = survfit(Surv(nurs2$los, 1-nurs2$censor) ~ 1,
    type="fleming-harrington", conf.type="log-log")
summary(fit.fh)
```

```
Call: survfit(formula = Surv(nurs2$los, 1 - nurs2$censor) ~ 1, type = "
    fleming-harrington",
```

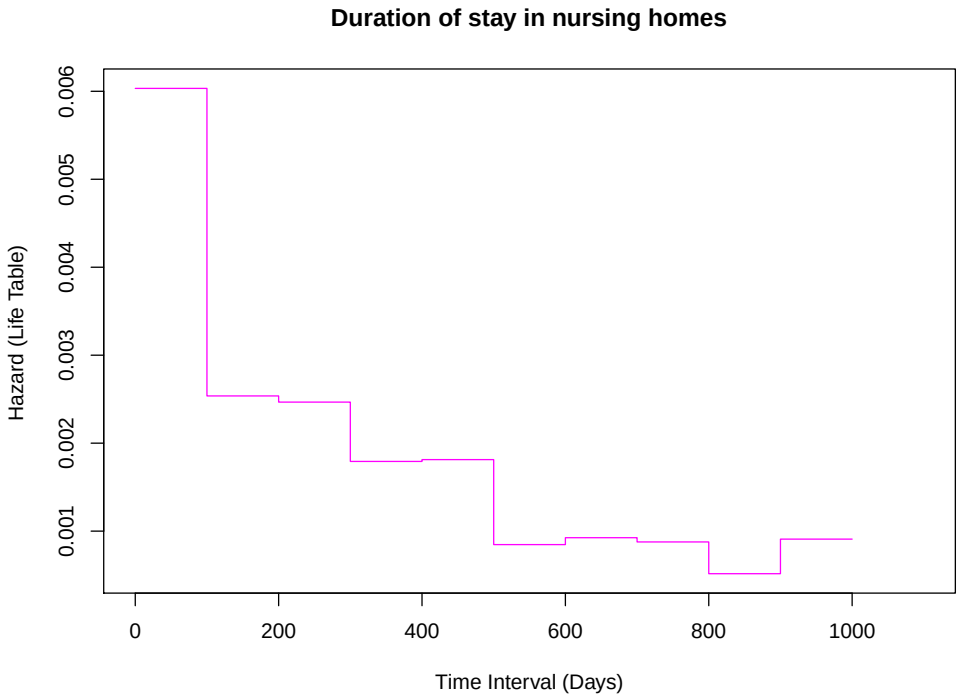

Figure 3.9: Estimated hazard for nursing home data.

```
    conf.type = "log-log")

 time n.risk n.event survival std.err lower 95% CI upper 95% CI
   14     12       1   0.9200  0.0801       0.5244        0.989
   24     11       1   0.8401  0.1085       0.4750        0.960
   25     10       1   0.7601  0.1267       0.4056        0.920
   38      9       1   0.6802  0.1388       0.3368        0.872
   64      8       1   0.6003  0.1465       0.2718        0.819
   89      7       1   0.5204  0.1502       0.2116        0.760
  113      6       1   0.4405  0.1505       0.1564        0.696
  123      5       1   0.3606  0.1472       0.1070        0.628
  149      4       1   0.2809  0.1404       0.0641        0.556
  168      3       1   0.2012  0.1299       0.0293        0.483
  185      2       1   0.1221  0.1169       0.0059        0.422
  234      1       1   0.0449     Inf       0.0000        1.000
```

And to compare it with KM:

```
fit.km = survfit(Surv(nurs2$los, 1-nurs2$censor) ~ 1,
    type="kaplan-meier", conf.type="log-log")
```

```
summary(fit.km)
```

```
Call: survfit(formula = Surv(nurs2$los, 1 - nurs2$censor) ~ 1, type = "
    kaplan-meier",
    conf.type = "log-log")

 time n.risk n.event survival std.err lower 95% CI upper 95% CI
   14     12       1   0.9167  0.0798      0.53898        0.988
   24     11       1   0.8333  0.1076      0.48171        0.956
   25     10       1   0.7500  0.1250      0.40842        0.912
   38      9       1   0.6667  0.1361      0.33702        0.860
   64      8       1   0.5833  0.1423      0.27014        0.801
   89      7       1   0.5000  0.1443      0.20848        0.736
  113      6       1   0.4167  0.1423      0.15247        0.665
  123      5       1   0.3333  0.1361      0.10270        0.588
  149      4       1   0.2500  0.1250      0.06014        0.505
  168      3       1   0.1667  0.1076      0.02651        0.413
  185      2       1   0.0833  0.0798      0.00505        0.311
  234      1       1   0.0000     NaN           NA           NA
```

A logrank test comparing length of stay for those under and over 85 years of age suggests a significant difference ($p = 0.03$). However, we know that gender has a strong association with length of stay, and also age. Hence, it would be a good idea to *stratify* the analysis by gender when trying to assess the age effect.

```
Link3 = "http://miso.matfyz.cz/prednasky/NMFM404/Data/NursingHome.dat"
nurshome = read.table(Link3, head=F, skip=14,
    col.names=c("los","age","rx","gender","married","health","censor"))
nurshome$discharged = 1 - nurshome$censor # event indicator
nurshome$under85 = (nurshome$age < 85) + 0
(nurs.slrt = survdiff( Surv(los, discharged) ~ under85 + strata(gender),
    data = nurshome))
```

```
Call:
survdiff(formula = Surv(los, discharged) ~ under85 + strata(gender),
    data = nurshome)

             N Observed Expected (O-E)^2/E (O-E)^2/V
under85=0 689      537      560     0.926      1.73
under85=1 912      742      719     0.721      1.73

 Chisq= 1.7  on 1 degrees of freedom, p= 0.189
```

We do not reject the null. This means that there is no significance effect of age on the length of stay, when stratifying for gender.

We fit the Cox PH model

$$\lambda(t) = \lambda_0(t) \exp\{\beta_1[\text{married}] + \beta_2[\text{health status}]\}.$$

```
LinkN="http://miso.matfyz.cz/prednasky/NMFM404/Data/NursingHome.dat"
nurshome = read.table(LinkN, head=F, skip=14, col.names = c("los", "age",
    "rx", "gender", "married", "health", "censor"))
nurshome$discharged = 1 - nurshome$censor # event indicator
head(nurshome)
```

```
   los age rx gender married health censor discharged
1   37  86  1      0       0      2      0          1
2   61  77  1      0       0      4      0          1
3 1084  75  1      0       0      3      1          0
4 1092  77  1      0       1      2      1          0
5   23  86  1      0       0      4      0          1
6 1091  71  1      1       0      3      1          0
```

```
table(nurshome$health)
```

```
  2   3   4   5
343 576 513 169
```

```
# Fit Cox PH model
fit.nurs = coxph(Surv(los,discharged) ~ married + health, data=nurshome)
```

Let's predict the survival for single individuals with `health status = 5` (worst), and for those with `health status = 4` (second worst), see Figure 3.10.

```
# Predict survival for single patients with health = 5
newdat = data.frame(married=0, health=5)
newdat    # data frame with same predictors as fit.nurs
```

```
  married health
1       0      5
```

```
ps5 = survfit(fit.nurs, newdata=newdat)    # predicted survival
# Compare with Kaplan-Meier
nursh5 = nurshome[nurshome$health==5 & nurshome$married==0,]
fit.km5 = survfit(Surv(los, discharged) ~ 1, data = nursh5)
# Predict survival for single patients, health =4
newdat[1,] = c(0,4)
ps4 =  survfit(fit.nurs, newdata=newdat)
# Compare with Kaplan-Meier
nursh4 = nurshome[nurshome$health==4 & nurshome$married==0,]
```

```
fit.km4 = survfit(Surv(los, discharged) ~ 1, data = nursh4)
plot(ps5, xlab= "Length of stay in nursing home (days)",
    ylab = "Proportion not discharged", col=2, conf.int=F)
lines(fit.km5, mark.time=F, col=2)     # add line to existing plot
lines(ps4, xlab= "Length of stay in nursing home (days)",
    ylab = "Proportion not discharged", col=4, conf.int=F)
lines(fit.km4, mark.time=F, col=4)     # add line to existing plot
```

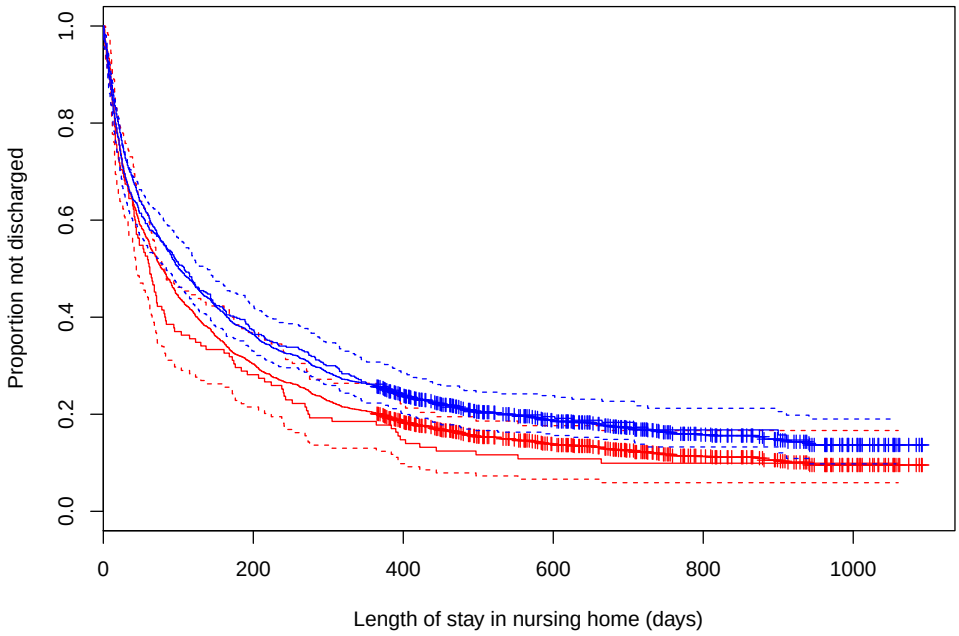

Figure 3.10: Cox PH model against KM.

Let's get the baseline survival $S_0(t)$ and plot cumulative hazard $\Lambda_0(t)$. They correspond to `married=0`, `health=0`, cf. Figure 3.11.

```
newdat[1,] = c(0,0)
ps0 = survfit(fit.nurs, newdata=newdat)    # S_0 = ps0$surv
bh = basehaz(fit.nurs, centered=F)
Lambda0 = bh$hazard   # it's cummulative hazard!
# same as Lambda0 = -log(ps0$surv)
plot(bh$time, Lambda0, type="l",
    xlab="Length of stay in nursing home (days)",
    ylab="Cummulative hazard", col=4)
```

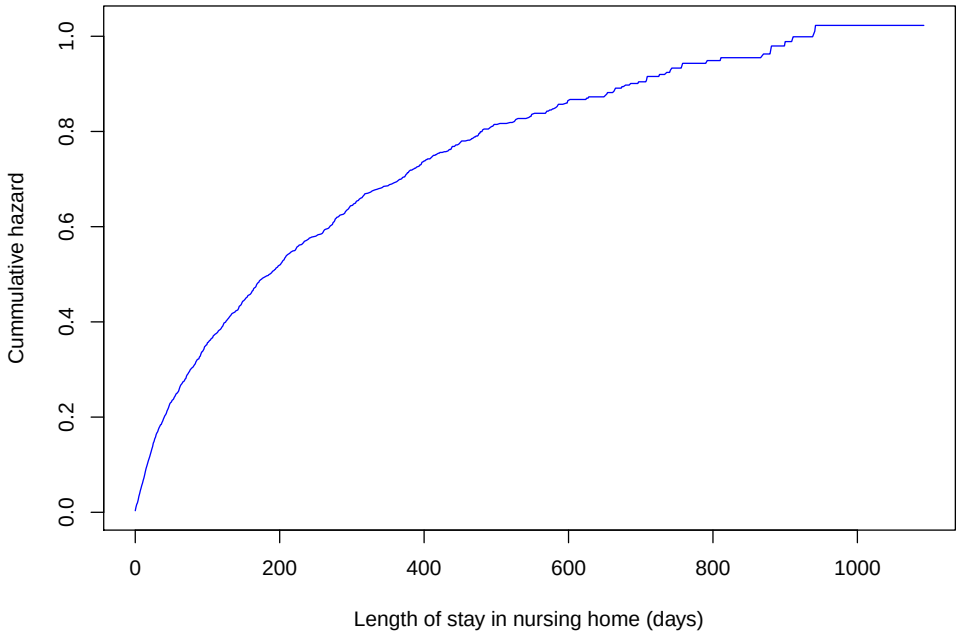

Figure 3.11: Baseline hazard function.

Lapses of insurance policy

Lapses for a special type of insurance policy with automatic renewal are of interest. Our data contain

- duration of the policy in weeks (lapse can be done online; that is why non-integer values are feasible),
- lapse indicator whether the policy has lapsed or is still in force,
- part-time/full-time indicator,
- premium paid by the insured monthly,
- age of the insured,
- monthly income of the insured,
- external rating of the insured (the higher the better).

Case study 3.2. Perform the following steps within the survival data analysis for the lapses of insurance policy:

- categorize continuous explanatory variables in a reasonable way based on descriptive statistics,
- construct KM estimators univariately (and separately) for categorized explanatory variables,
- find a suitable Cox PH model for lapses using automatic model selection criteria,
- obtain deviance residuals for the Cox PH model chosen before.

First of all, let us read the data and go through some descriptive statistics.

```
LapsesL="http://miso.matfyz.cz/prednasky/NMFM404/Data/lapse.csv"
lapse = read.csv(LapsesL, head=T)
dim(lapse)
```

```
[1] 294   8
```

```
# Time = Length of the policy (survival time) in weeks
# Lapse = Event indicator
# Job = Part-time (=50), Full-time (=100)
# Premium = Premium paid monthly
# Age = Age of the insured
# Income = Monthly income
# Rating = External rating of client
head(lapse)
```

```
  No Time Lapse Job Premium Age Income Rating
1  1  209     1  50      13  41      8  6.992
2  2  209     1  50      13  44      8  6.992
3  3  209     1  50      13  47     10  6.992
4  4  209     1  50      13  34     10  6.992
5  5   38     1  50      13  40     11  6.992
6  6  209     1  50      13  42     11  6.992
```

```
### Univariate analysis
# Part-time/Full-time:
lapse$Job <- as.factor(lapse$Job)
table(lapse$Job)
```

```
 50 100
107 187
```

```
kmjob = survfit(Surv(Time, Lapse) ~ Job, data=lapse)
# Age of insured:
summary(lapse$Age)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  29.00   38.00   43.00   42.51   46.75   57.00
```

```
lapse$Agecat = (lapse$Age > 42) + 0
table(lapse$Agecat)
```

```
  0   1
134 160
```

```
kmage = survfit(Surv(Time, Lapse) ~ Agecat, data=lapse)
# Premium:
table(lapse$Premium)
```

```
 1  2  3  4  5  6  8 10 13 15 23 49 58
 6 17  8  6 45 46 40 68  8 12 32  3  3
```

```
summary(lapse$Premium)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   5.000   8.000   9.966  10.000  58.000
```

```
lapse$Premiumcat = cut(lapse$Premium, breaks=c(0,5,9,11,60))
kmpre = survfit(Surv(Time, Lapse) ~ Premiumcat, data=lapse)
# Income:
summary(lapse$Income)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1.00    6.00   10.00   12.51   17.00   38.00
```

```
lapse$Incomecat = cut(lapse$Income, breaks = c(0,6,10,17,40))
kminc = survfit(Surv(Time, Lapse) ~ Premiumcat, data=lapse)
# Rating:
summary(lapse$Rating)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  2.904   4.059   4.203   4.886   5.323   8.690
```

```
lapse$Ratingcat = cut(lapse$Rating, breaks=c(0,4.06,4.89,5.33,9))
kmrat = survfit(Surv(Time, Lapse) ~ Ratingcat, data = lapse)
```

KM estimators for univariate categorized explanatory variables are shown in Figure 3.12.

```
par(mfrow=c(3,2))
plot(kmjob, lty=1:2, col=2:3, xlab="Weeks", ylab="Survival", main="Job")
plot(kmage, lty=1:2, col=2:3, xlab="Weeks", ylab="Survival", main="Age")
plot(kmpre, lty=1:4, col=1:4, xlab="Weeks", ylab="Survival",
    main="Premium")
plot(kminc, lty=1:4, col=1:4, xlab="Weeks", ylab="Survival",
    main="Income")
plot(kmrat, lty=1:4, col=1:4, xlab="Weeks", ylab="Survival",
    main="Rating")
par(mfrow=c(1,1))
```

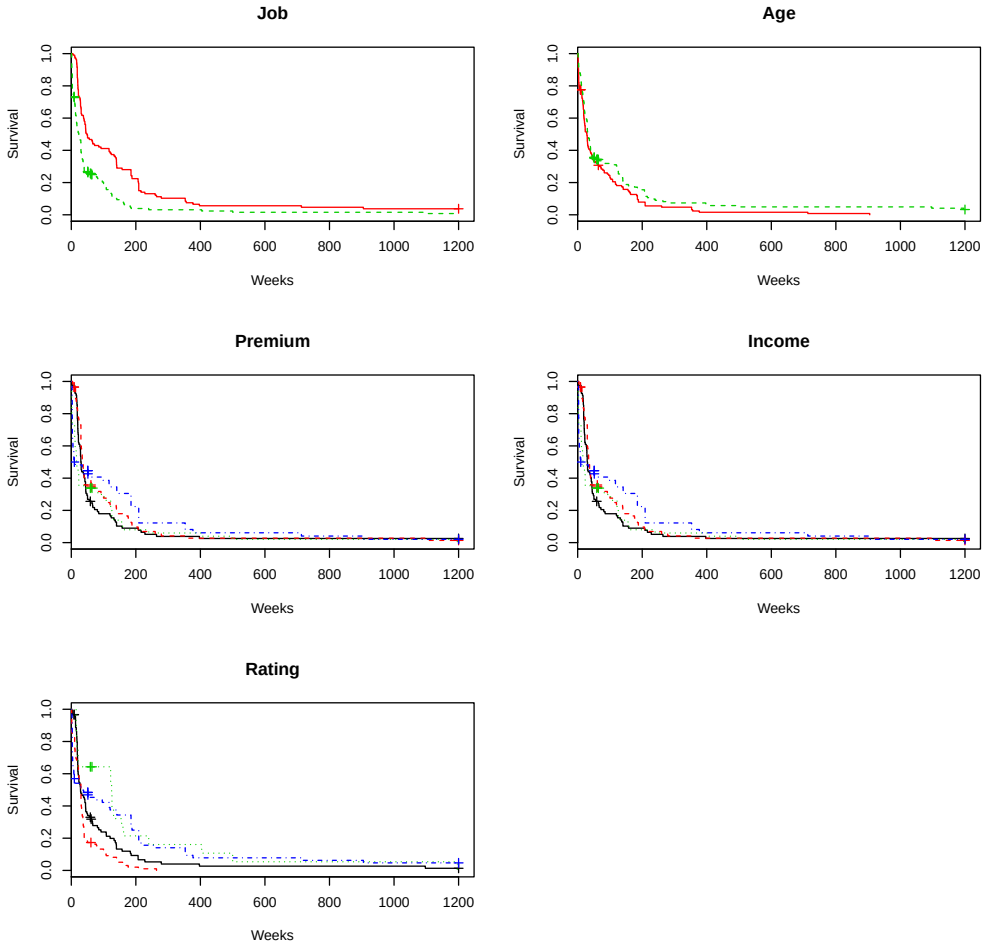

Figure 3.12: Univariate KM estimators for lapses.

Let us use automatic model selection approaches for finding a suitable Cox PH model for lapses.

```
library(MASS)
args(stepAIC)
```

```
function (object, scope, scale = 0, direction = c("both", "backward",
    "forward"), trace = 1, keep = NULL, steps = 1000, use.start = FALSE,
    k = 2, ...)
NULL
```

```
# Model selection for Lapse data
fit = coxph(Surv(Time, Lapse) ~ Job + Age + Premium + Income + Rating,
   data=lapse)
# Backward selection using AIC
fitb = stepAIC(fit, direction="backward", k=2)
```

```
Start:  AIC=2520.15
Surv(Time, Lapse) ~ Job + Age + Premium + Income + Rating

          Df    AIC
- Premium  1 2519.7
<none>       2520.2
- Job      1 2531.7
- Rating   1 2532.8
- Age      1 2533.0
- Income   1 2544.9

Step:  AIC=2519.7
Surv(Time, Lapse) ~ Job + Age + Income + Rating

         Df    AIC
<none>      2519.7
- Rating  1 2530.8
- Age     1 2531.2
- Job     1 2532.7
- Income  1 2547.7
```

```
summary(fitb)
```

```
Call:
coxph(formula = Surv(Time, Lapse) ~ Job + Age + Income + Rating,
    data = lapse)

  n= 294, number of events= 273

           coef exp(coef) se(coef)      z Pr(>|z|)
```

```
Job100  0.54205   1.71954  0.14119  3.839 0.000123 ***
Age    -0.03696   0.96371  0.01003 -3.686 0.000228 ***
Income  0.05495   1.05648  0.00987  5.567 2.59e-08 ***
Rating -0.18337   0.83246  0.05098 -3.597 0.000322 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

       exp(coef) exp(-coef) lower .95 upper .95
Job100    1.7195     0.5816    1.3039    2.2677
Age       0.9637     1.0377    0.9450    0.9828
Income    1.0565     0.9465    1.0362    1.0771
Rating    0.8325     1.2013    0.7533    0.9199

Concordance= 0.683  (se = 0.02 )
Rsquare= 0.25   (max possible= 1 )
Likelihood ratio test= 84.5  on 4 df,   p=0
Wald test            = 90.71  on 4 df,   p=0
Score (logrank) test = 94.51  on 4 df,   p=0
```

```
# Forward selection using AIC
fit0 = coxph(Surv(Time, Lapse) ~ 1, data=lapse) # (starting model)
fitf = stepAIC(fit0, scope=formula(fit), direction="forward", k=2)
```

```
Start:  AIC=2596.2
Surv(Time, Lapse) ~ 1

          Df    AIC
+ Income   1 2556.8
+ Job      1 2568.0
+ Age      1 2586.1
+ Premium  1 2594.5
<none>       2596.2
+ Rating   1 2596.8

Step:  AIC=2556.85
Surv(Time, Lapse) ~ Income

          Df    AIC
+ Rating   1 2540.3
+ Job      1 2543.6
+ Age      1 2549.9
<none>       2556.8
+ Premium  1 2558.8
```

```
Step:  AIC=2540.3
Surv(Time, Lapse) ~ Income + Rating

          Df    AIC
+ Job      1 2531.2
+ Age      1 2532.7
<none>       2540.3
+ Premium  1 2541.4

Step:  AIC=2531.23
Surv(Time, Lapse) ~ Income + Rating + Job

          Df    AIC
+ Age      1 2519.7
<none>       2531.2
+ Premium  1 2533.0

Step:  AIC=2519.7
Surv(Time, Lapse) ~ Income + Rating + Job + Age

          Df    AIC
<none>       2519.7
+ Premium  1 2520.2
```

```
summary(fitf)
```

```
Call:
coxph(formula = Surv(Time, Lapse) ~ Income + Rating + Job + Age,
    data = lapse)

  n= 294, number of events= 273

           coef exp(coef) se(coef)      z Pr(>|z|)
Income  0.05495   1.05648  0.00987  5.567 2.59e-08 ***
Rating -0.18337   0.83246  0.05098 -3.597 0.000322 ***
Job100  0.54205   1.71954  0.14119  3.839 0.000123 ***
Age    -0.03696   0.96371  0.01003 -3.686 0.000228 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

       exp(coef) exp(-coef) lower .95 upper .95
Income    1.0565     0.9465    1.0362    1.0771
Rating    0.8325     1.2013    0.7533    0.9199
Job100    1.7195     0.5816    1.3039    2.2677
```

```
Age       0.9637     1.0377    0.9450    0.9828

Concordance= 0.683  (se = 0.02 )
Rsquare= 0.25   (max possible= 1 )
Likelihood ratio test= 84.5  on 4 df,   p=0
Wald test            = 90.71  on 4 df,   p=0
Score (logrank) test = 94.51  on 4 df,   p=0
```

```
# same final model as backward selection!
# Stepwise model selection (backward and forward)
fits = stepAIC(fit, direction="both", k=2)
```

```
Start:  AIC=2520.15
Surv(Time, Lapse) ~ Job + Age + Premium + Income + Rating

          Df    AIC
- Premium  1 2519.7
<none>       2520.2
- Job      1 2531.7
- Rating   1 2532.8
- Age      1 2533.0
- Income   1 2544.9

Step:  AIC=2519.7
Surv(Time, Lapse) ~ Job + Age + Income + Rating

          Df    AIC
<none>       2519.7
+ Premium  1 2520.2
- Rating   1 2530.8
- Age      1 2531.2
- Job      1 2532.7
- Income   1 2547.7
```

```
summary(fits)
```

```
Call:
coxph(formula = Surv(Time, Lapse) ~ Job + Age + Income + Rating,
    data = lapse)

  n= 294, number of events= 273

           coef exp(coef) se(coef)      z Pr(>|z|)
Job100  0.54205   1.71954  0.14119  3.839 0.000123 ***
```

```
Age    -0.03696   0.96371  0.01003 -3.686 0.000228 ***
Income  0.05495   1.05648  0.00987  5.567 2.59e-08 ***
Rating -0.18337   0.83246  0.05098 -3.597 0.000322 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

       exp(coef) exp(-coef) lower .95 upper .95
Job100    1.7195     0.5816    1.3039    2.2677
Age       0.9637     1.0377    0.9450    0.9828
Income    1.0565     0.9465    1.0362    1.0771
Rating    0.8325     1.2013    0.7533    0.9199

Concordance= 0.683  (se = 0.02 )
Rsquare= 0.25   (max possible= 1 )
Likelihood ratio test= 84.5  on 4 df,   p=0
Wald test            = 90.71  on 4 df,   p=0
Score (logrank) test = 94.51  on 4 df,   p=0
```

```
# Stepwise selection with alpha=k=3
fitk3 = stepAIC(fit, direction="both", k=3)
```

```
Start:  AIC=2525.15
Surv(Time, Lapse) ~ Job + Age + Premium + Income + Rating

          Df    AIC
- Premium  1 2523.7
<none>       2525.2
- Job      1 2535.7
- Rating   1 2536.8
- Age      1 2537.0
- Income   1 2548.9

Step:  AIC=2523.7
Surv(Time, Lapse) ~ Job + Age + Income + Rating

          Df    AIC
<none>       2523.7
+ Premium  1 2525.2
- Rating   1 2533.8
- Age      1 2534.2
- Job      1 2535.7
- Income   1 2550.7
```

```
summary(fitk3)
```

```
Call:
coxph(formula = Surv(Time, Lapse) ~ Job + Age + Income + Rating,
    data = lapse)

  n= 294, number of events= 273

           coef exp(coef) se(coef)      z Pr(>|z|)
Job100  0.54205   1.71954  0.14119  3.839 0.000123 ***
Age    -0.03696   0.96371  0.01003 -3.686 0.000228 ***
Income  0.05495   1.05648  0.00987  5.567 2.59e-08 ***
Rating -0.18337   0.83246  0.05098 -3.597 0.000322 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

       exp(coef) exp(-coef) lower .95 upper .95
Job100    1.7195     0.5816    1.3039    2.2677
Age       0.9637     1.0377    0.9450    0.9828
Income    1.0565     0.9465    1.0362    1.0771
Rating    0.8325     1.2013    0.7533    0.9199

Concordance= 0.683  (se = 0.02 )
Rsquare= 0.25   (max possible= 1 )
Likelihood ratio test= 84.5  on 4 df,   p=0
Wald test            = 90.71  on 4 df,   p=0
Score (logrank) test = 94.51  on 4 df,   p=0
```

Our final Cox PH model based on forward, backward and stepwise automatic model selection criteria is

$$\lambda(t) = \lambda_0(t) \exp\{\beta_1 \mathcal{I}[Job = 100] + \beta_2 Age + \beta_3 Income + \beta_4 Rating\}.$$

Here, an insured person working full time has a higher hazard to lapse the policy. The older insured person is the smaller hazard rate for her/his policy to lapse. People with higher income tend to have higher lapse hazard rate. Higher rated clients have lower hazard of lapse. Finally, there is no significant effect of premium on the hazard of lapse.

Note that when the lapse data were analyzed with the forward, backward and stepwise options, the same final model was reached. However, this will not always (in fact, rarely) be the case.

Deviance residuals (Figure 3.14) for the above fitted model can be used to judge the suitability of the chosen Cox PH model.

```
devres = residuals(fitb, type="deviance")  # Deviance residuals
devres[1:3]
```

```
         1          2          3
-0.2217595 -0.1047756 -0.1037553
```

```
plot(lapse$Age, devres, xlab="Age", ylab="Deviance Residuals", col=2)
abline(h=0, lty=2)
```

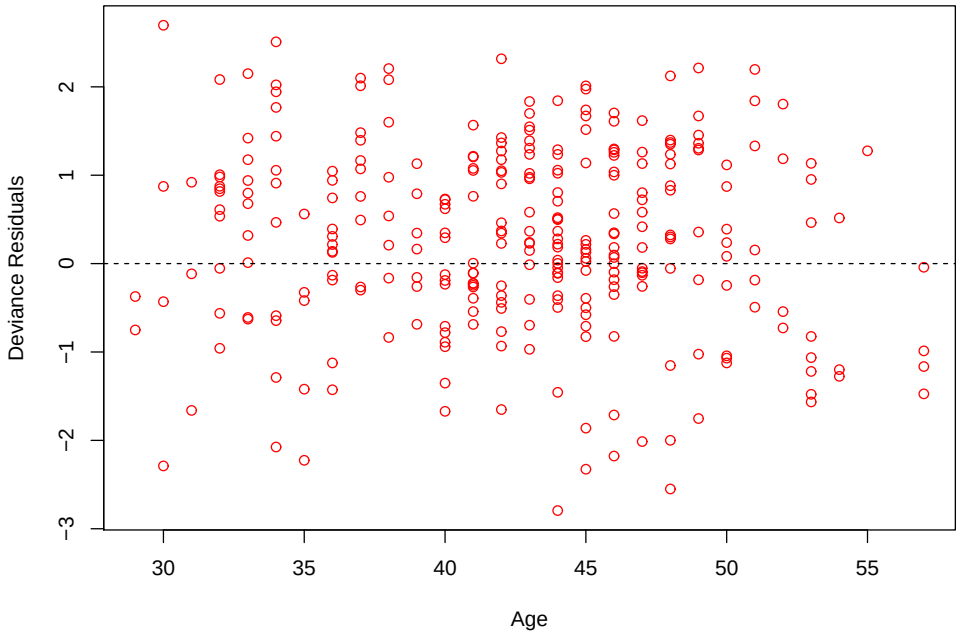

Figure 3.13: Deviance residuals from the Cox PH model for lapse data.

Claim development duration

We are interested in time between the claim origin and the claim closing. Realize that some of the observed claim developments are still not closed. Our data—provided by the Czech Insurer's Bureau—contain

- id number of the claim [*id*],
- time between claim origin and actual date in days [*time*],
- closing indicator whether the claim has already been closed [*closed*],
- type of the claim [*type*],
- final reserve set on the claim [*reserve_final*].

Case study 3.3. Fit the Cox PH model for the claim development duration taking into account the type of claim.

First of all, let us read the data and go through some descriptive statistics.

```
ckp="http://miso.matfyz.cz/prednasky/NMFM404/Data/ckp.csv"
CKP = read.csv(ckp, head=T, sep=";")
dim(CKP)
```

```
[1] 54469     5
```

```
head(CKP)
```

```
    id time type reserve_final closed
1 2625  325    7             0      1
2 2636  494    7             0      1
3 2652 1507    1             0      1
4 2682  431    5             0      1
5 2684  471    7             0      1
6 2688  309    7            -1      1
```

```
summary(CKP)
```

```
       id              time         type
 Min.   :  2625   Min.   :   1.0   1: 6743
 1st Qu.: 58685   1st Qu.: 145.0   2:   63
 Median :101712   Median : 290.0   3:  198
 Mean   :106401   Mean   : 543.3   4: 5246
 3rd Qu.:152639   3rd Qu.: 746.0   5: 6244
 Max.   :216069   Max.   :5426.0   6:   22
                                   7:35953
 reserve_final          closed
 Min.   :  -36007   Min.   :0.0000
 1st Qu.:       0   1st Qu.:1.0000
 Median :       0   Median :1.0000
 Mean   :   17610   Mean   :0.9344
 3rd Qu.:       0   3rd Qu.:1.0000
 Max.   :33389383   Max.   :1.0000
```

Firstly, we test whether there is a significant difference in claim development durations with respect to the type of claim.

```
library(survival)
CKP <- as.data.frame(CKP)
CKP$type <- as.factor(CKP$type)
(ckp.lrt = survdiff(Surv(time, closed) ~ type, rho=0, data=CKP))
```

```
Call:
survdiff(formula = Surv(time, closed) ~ type, data = CKP, rho = 0)

           N Observed Expected (O-E)^2/E (O-E)^2/V
type=1  6743     6735   9184.1   653.111   819.321
type=2    63       61    118.7    28.039    28.187
type=3   198       46    707.5   618.526   661.907
type=4  5246     4225   9028.1  2555.332  3198.515
type=5  6244     6040   6780.1    80.792    93.750
type=6    22       22     18.6     0.634     0.635
type=7 35953    33768  25059.8  3026.033  6318.684

 Chisq= 7460  on 6 degrees of freedom, p= 0
```

We can also obtain Kaplan-Meier estimates for all the claim types.

```
(ckp.km = survfit(Surv(time, closed) ~ type, data=CKP))
```

```
Call: survfit(formula = Surv(time, closed) ~ type, data = CKP)

           n events median 0.95LCL 0.95UCL
type=1  6743   6735    893     870     915
type=2    63     61    802     616    1401
type=3   198     46   4127    3437      NA
type=4  5246   4225   1149    1102    1198
type=5  6244   6040    458     441     472
type=6    22     22    310     268     376
type=7 35953  33768    237     234     239
```

Consequently, we can fit the Cox PH taking into account the type of claim.

```
ckp.ph = coxph(Surv(time, closed) ~ type, data=CKP)
summary(ckp.ph)
```

```
Call:
coxph(formula = Surv(time, closed) ~ type, data = CKP)

  n= 54469, number of events= 50897

          coef exp(coef) se(coef)       z Pr(>|z|)
type2 -0.37466   0.68752  0.12866  -2.912  0.00359 **
type3 -2.52853   0.07978  0.15008 -16.848  < 2e-16 ***
type4 -0.45945   0.63163  0.01970 -23.324  < 2e-16 ***
type5  0.21863   1.24438  0.01777  12.302  < 2e-16 ***
type6  0.53947   1.71511  0.21357   2.526  0.01154 *
type7  0.65444   1.92406  0.01359  48.160  < 2e-16 ***
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

      exp(coef) exp(-coef) lower .95 upper .95
type2   0.68752     1.4545   0.53428    0.8847
type3   0.07978    12.5351   0.05945    0.1071
type4   0.63163     1.5832   0.60771    0.6565
type5   1.24438     0.8036   1.20178    1.2885
type6   1.71511     0.5831   1.12849    2.6067
type7   1.92406     0.5197   1.87349    1.9760

Concordance= 0.616  (se = 0.001 )
Rsquare= 0.142   (max possible= 1 )
Likelihood ratio test= 8328  on 6 df,   p=0
Wald test            = 6684  on 6 df,   p=0
Score (logrank) test = 7460  on 6 df,   p=0
```

Finally, the estimated proportions of closed claims with respect to the claim type can be plotted.

```
plot(survfit(ckp.ph, newdata=data.frame(type=as.factor(1))), col=1, xlab=
     "Time (days)", ylab = "Proportion of closed claims", conf.int=F)
lines(survfit(ckp.ph, newdata=data.frame(type=as.factor(2)), conf.int=F),
     col=2)
lines(survfit(ckp.ph, newdata=data.frame(type=as.factor(3)), conf.int=F),
     col=3)
lines(survfit(ckp.ph, newdata=data.frame(type=as.factor(4)), conf.int=F),
     col=4)
lines(survfit(ckp.ph, newdata=data.frame(type=as.factor(5)), conf.int=F),
     col=5)
lines(survfit(ckp.ph, newdata=data.frame(type=as.factor(6)), conf.int=F),
     col=6)
lines(survfit(ckp.ph, newdata=data.frame(type=as.factor(7)), conf.int=F),
     col=7)
legend("topright",inset=0.05,title="Type",legend=1:7,fill=1:7)
```

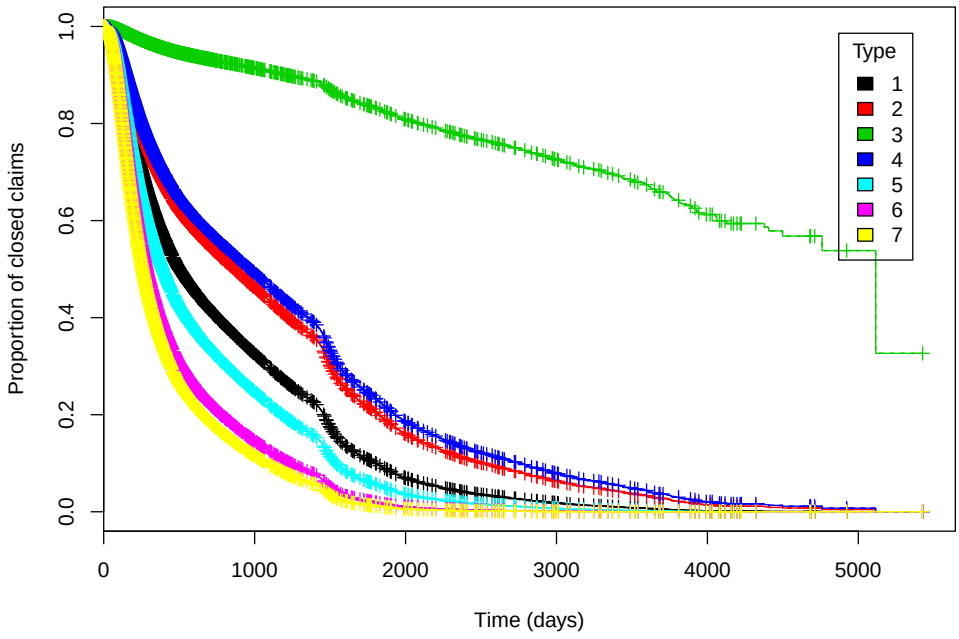

Figure 3.14: Cox PH model for the claim development duration taking into account the type of claim.

Appendix A

Useful Things

xyz

List of Procedures

List of Figures

List of Tables

Bibliography

Cebrián, A. C., Deniut, M., and Lambert, P. (2003). Generalized pareto fit to the society of actuaries' large claims database. *North American Actuarial Journal*, 7:18–36.

Coles, S. G. (2001). *An introduction to statistical modelling of extreme values*. Springer Series in Statistics, London.

Coles, S. G. and Dixon, M. J. (1999). Likelihood-based inference for extreme value models. *Extremes*, 2(1):5–23.

Collett, D. (2014). *Modelling Survival Data in Medical Research*. CRC Press, Boca Raton, FL, 3rd edition.

Cox, D. R. and Hinkley, D. V. (1974). *Theoretical Statistics*. Chapman and Hall, London.

Cunnane, C. (1979). Note on the poisson assumption in partial duration series model. *Water Resources Research*, 15(2):489–494.

Davison, A. C. and Smith, R. L. (2012). Models for exceedances over high thresholds (with discussion). *Journal of the Royal Statistical Society*, 52:237–254.

Fleming, T. R. and Harrington, D. P. (2005). *Counting Processes and Survival Analysis*. Wiley, New York, 2nd edition.

Grazier, K. L. (2004). Group medical insurance claims database collection and analysis. Technical report, Society of Actuaries.

Hosking, J. R. M. (1990). L-moments: analysis and estimation of distributions using linear combination of order statistics. *Journal of the Royal Statistical Society*, 52(1):105–125.

Hosking, J. R. M. and Wallis, J. R. (1987). Parameter and quantile estimation for the generalized pareto distribution. *Technometrics*, 29:339–349.

Hosking, J. R. M., Wallis, J. R., and Wood, E. F. (1985). Estimation of the generalized extreme value distribution by the method of probability weighted moments. *Technometrics*, 27:251–261.

Juaréz, S. F. and Schucany, W. R. (2004). Robust and efficient estimation for the generalized pareto distribution. *Extremes*, 7(3):237–251.

Kalbfleisch, J. D. and Prentice, R. L. (2002). *The Statistical Analysis of Failure Time Data*. Wiley, New York, 2nd edition.

Kleinbaum, D. G. and Klein, M. (2012). *Survival Analysis, A Self-Learning Text*. Springer, New York, 3rd edition.

Klugman, S. A., Panjer, H. H., and Willmot, G. E. (1998). *Loss models: from data to decisions*. Wiley, New York.

Landwehr, J. M., Matalas, N. C., and Wallis, J. R. (1979). Probability weighted moments compared with some traditional techniques in estimating gumbel parameters and quantiles. *Water Resources Research*, 15:1066–1064.

Luceno, A. (2006). Fitting the generalized pareto distribution to data using maximum goodness-of-fit estimators. *Computational Statistics & Data Analysis*, 51(2):904–917.

Ohlsson, E. and Johansson, B. (2010). *Non-Life Insurance Pricing with Generalized Linear Models*. Springer, New York.

Peng, L. and Welsh, A. H. (2001). Robust estimation of the generalized pareto distribution. *Extremes*, 4(1):53–65.

Pickands, J. (1975). Statistical inference using extreme order statistics. *Annals of Statistics*, 3:119–131.

Ribatet, M. (2011). *A user's guide to the POT package (version 1.4)*. University of Montpellier II.

Scarrott, C. and MacDonald, A. (2012). A review of extreme value threshold estimation and uncertainty quantification. *Statistical Journal*, 10(1):33–60.

Smith, R. L. (1985). Maximum likelihood estimation in a class of non-regular cases. *Biometrika*, 72:67–92.

Society of actuaries (2004). Group medical claims database 1997, 1998 and 1999. http://www.soa.org. Accessed 9th October 2013.

Stephens, M. A. (1974). EDF statistics for goodness of fit and some comparisons. *Journal of the American Statistical Association*, 69(347):730–737.

Zvang, J. (2007). Likelihood moment estimation for the generalized pareto distribution. *Australian and New Zealand Journal of Statistics*, 49(1):69–77.