Exercise Series 1

1. The following table contains some functions which can be linearized by a suitable transformation. Complete the table by inserting the needed transformations of x and y, and the resulting linear forms.

Function	Transformation	Linear form
$y = \alpha x^{\beta}$	$y' = \log(y),\ x' = \log(x)$	$y' = \log(\alpha) + \beta \cdot x'$
$y = \alpha e^{\beta \cdot x}$	....	....
$y = \alpha + \beta \cdot \log(x)$	....	....
$y = x/(\alpha \cdot x - \beta)$	....	....
$y = \frac{e^{\alpha+\beta x}}{1+e^{\alpha+\beta \cdot x}}$	....	....
$y = \alpha e^{\beta/x}$	....	....
$y = 1/(\alpha + \beta e^{-x})$	....	....

2. The behaviour of the least squares estimator can be investigated by a small simulation study. Here are the R-commands for linear regression:

```
## simple linear regression
x   <- seq(1,40,1)                # generates equidistant x-values
y   <- 2*x+1+5*rnorm(length(x))   # y-values=linear function(x) + error
reg <- lm(y~x)                    # fit of the linear regression
summary(reg)                      # output of selected results
plot(x,y)                         # scatter plot
abline(reg)                       # draws regression line

## modern smoothing
fit <- loess(y~x)                 # ``smoothed fit'' (to be introduced later)
lines(predict(fit),lty=2)         # draws fitted curve
txt <- c("Regression","Smooth")   # vector of strings for comment
legend(2,80,txt,lty=1:2)          # adds comment to plot
```

a) Write a sequence of R-commands which randomly generates 100 times a vector of y-values according to the above model with the given x-values and generates a vector of slopes of the regression lines.
R-hint: `help(for)`.

b) Draw a histogram of the 100 estimated slopes and add the normal density of the theoretically true distribution of the slopes to the histogram.
R-hints: Because of part d), you should use `par(mfrow=c(1,2))`. The histogram must be drawn with parameter `freq=FALSE`, so that the y-axis is suitably scaled for drawing the density. The density can be added by something like
`lines(seq(1.8,2.3,by=0.01),dnorm(seq(1.8,2.3,by=0.01),mean=?,sd=?))`,
where you have to find the correct values for `mean` and `sd`. To compute the inverse of a matrix use `solve()`.

c) Compute the mean and empirical standard deviation of the estimated slopes.

d) Repeat the simulation with a skew, non-normal error distribution. That is, replace the second line by

```
 y   <- 2*x+1+5*(1-rchisq(length(x), df=1))/sqrt(2)
# You may try hist(5*(1-rchisq(40, df=1))/sqrt(2))
# to explore the error distribution
```

Repeat part b) using the new slopes. Add the same normal density, which is only an asymptotic approximation to the true distribution in this setup.

3. The dataset `airline` contains the monthly number of flight passengers in the USA in the years 1949-1960.

a) Plot the data against time and verbally describe what you observe.

b) Compute the logarithms of the data and plot them against time. Comment on the differences.

c) Define a linear model of the form

$$\log(y_t) = \sum_{j=1}^{p} \theta_j f_j(t) + \epsilon_t$$

by choosing $f_1(t) = t$ (linear trend in time) and by defining $f_2, \ldots, f_{13}$ as indicator functions of the months, e.g.

$$f_2(t) = \begin{cases} 1 & \text{if } t \text{ corresponds to a January month} \\ 0 & \text{otherwise.} \end{cases}$$

Remark: It is not necessary to specify an intercept parameter. Why ? (optional)

d) Fit the model specified in c) and plot the fitted values and residuals against time. Do you think that the model assumptions hold?

R-hints: read data:

```
airline <- scan("http://stat.ethz.ch/Teaching/Datasets/airline.dat")
```

If this is not possible at home, save the data locally and read it by:

```
airline <- scan("filename")
```

Use `plot(airline,...)` for a) and consider the parameter `type="l"` of `plot` for the plots. Regression fit by

```
reg <- lm(log(airline) ~ f1+...+fp-1)
```

`fj` must be a vector of length 144 containing the values of $f_j(t)$. The inclusion of `-1` in the `lm`-command prevents the fit of an intercept parameter (which would be done by default otherwise). The `fj` can be generated by use of `rep()`. For example, take a look at the result of `c(1,rep(0,11))`. The commands `fitted(reg)` and `resid(reg)` extract fitted values and residuals from an `lm`-object.

Preliminary discussion: Friday, March 5.

Deadline: Friday, March 12.