Classical and Bayesian Statistics

Problems 7

Problem 7.1

We look at a study conducted in the United States in 1979 (National Longitudinal Study of Youth, NLSY79): of 2584 Americans in 1981, the intelligence quotient (as per the AFQT - armed forces qualifying test score) was measured; in 2006, the same people were asked about their annual income in 2005 and the number of years of schooling.

We are naturally interested here in whether a high IQ or a long school education leads to a higher income.

In the file **income.dat** on ILIAS you will find the data set with the income, the number of years of completed school education and the intelligence quotients of 2584 Americans.

(∗) a) Read the data set **income.dat**.

```
... <- read.table(..., header = TRUE)
```

(∗∗) b) Generate scatter plots with the corresponding regression lines showing income versus number of years of schooling and income versus intelligence quotient. What do you find?

(∗∗) c) Determine the parameters a and b of the linear model $y = a + bx$, where y is the income and x is the number of years of schooling.

How do you interpret the parameters a and b?

(∗∗) d) Calculate the correlation between income and number of years of schooling. How appropriate is a regression model for this data set?

Problem 7.2

In this assignment we look at 4 data sets constructed by the statistician Francis John Anscombe (13 May 1918 – 17 October 2001). In each of the records there is a response variable y and an predictor x.

a) The file is already included in **R**.

```
head(anscombe)

##   x1 x2 x3 x4   y1   y2    y3   y4
## 1 10 10 10  8 8.04 9.14  7.46 6.58
## 2  8  8  8  8 6.95 8.14  6.77 5.76
## 3 13 13 13  8 7.58 8.74 12.74 7.71
## 4  9  9  9  8 8.81 8.77  7.11 8.84
## 5 11 11 11  8 8.33 9.26  7.81 8.47
## 6 14 14 14  8 9.96 8.10  8.84 7.04
```

(∗) b) Produce a scatter plot each of the 4 data sets, draw the regression line and comment on the results.

```
plot(anscombe$x1, anscombe$y1)
reg <- lm(anscombe$y1 ~ anscombe$x1)
abline(reg)
```

With `par(mfrow=c(2,2))` the graphics window is divided so that all 4 panes fit next to each other.

(∗) c) Compare a and b each, where $y = a + bx$.

```
lm(y1 ~ x1, data = anscombe) # or

lm(anscombe$y1 ~ anscombe$x1)
```

d) Determine the correlation coefficients. What stands out?

Problem 7.3

In this task we use the data set **Auto**, which is contained in the library **ISLR**.

```
library(ISLR)
```

If an error message appears, the library must be installed first (this only needs to be done once):

```
install.packages("ISLR")
```

(∗∗) a) Investigate the data record with `head(Auto)` and `?Auto` or `help(Auto)`.

(∗) b) Adjust the model to a simple linear regression with `mpg` as the target variable and `horsepower` as the predictor.

c) Use the `lm()` command to perform this regression.

Use the `summary()` command to output the results. Comment on it:

(**) i) Is there a connection between the target variable and the predictor?

(**) ii) How do you interpret the coefficients for `(intercept)` and `horsepower`? Is the correlation positive or negative?

(**) iii) How do you determine the confidence intervals (with `confint()`) and interpret them?

(**) iv) Interpret the R^2 value.

(**) d) Plot response variable and the predictor with the regression line (`abline`). How do you interpret this plot compared to the `summary()` output.

Problem 7.4

The `MASS` library contains the `Boston` data set, which records `medv` (median house value) for 506 neighborhoods around Boston. We will seek to predict `medv` using 13 predictors such as `rm` (average number of rooms per house), `age` (average age of houses), and `lstat` (percent of households with low socioeconomic status).

(**) a) To find out more about the data set, we can type `?Boston`.

(**) b) Which column names are available?

(*) c) Use the `attach(...)`-command to let `R` recognize the column names of the data set `Boston`.

d) We will start by using the `lm()` function to fit a simple linear regression model, with `medv` as the response and `lstat` as the predictor.

(*) i) Define the simple regression model using the two variables above.

(**) ii) The basic syntax is `lm(y~x,data)` , where `y` is the response, `x` is the predictor, and data is the data set in which these two variables are kept.

```
lm.fit <- lm(...)
summary(lm.fit)
```

(*) e) We can use the `names(...)` function in order to find out what other pieces of information are stored in `lm.fit`.

(**) f) Although we can extract these quantities by name — e.g. `lm.fit$coefficients` — it is safer to use the extractor functions like `coef(...)` to access them.

Interpret these values and the corresponding p–values in the summary above.

(∗∗) g) In order to obtain a confidence interval for the coefficient estimates, we can use the `confint(...)` command.

Give an interpretation of these values.

(∗∗) h) We will now plot `medv` and `lstat` along with the least squares regression line using the `plot(...)` and `abline()` functions (see exercise sheet 2).

Use `lty = ...`, `pch = ...` and `col = ...` to make the plot look nicer.

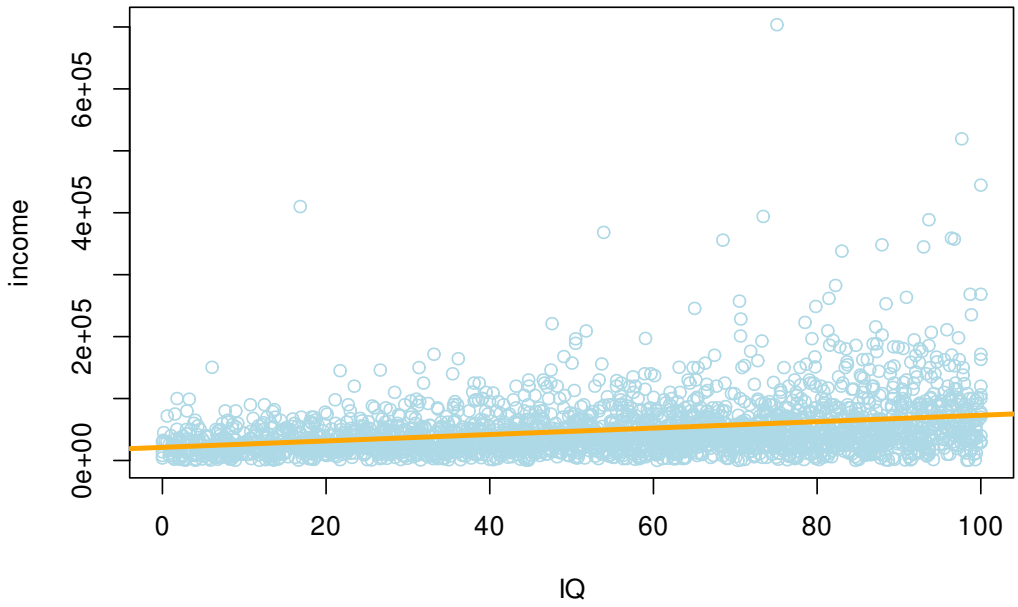
(∗∗) i) Interpret the R^2 value in the `summary`–output above.

Classical and Bayesian Statistic

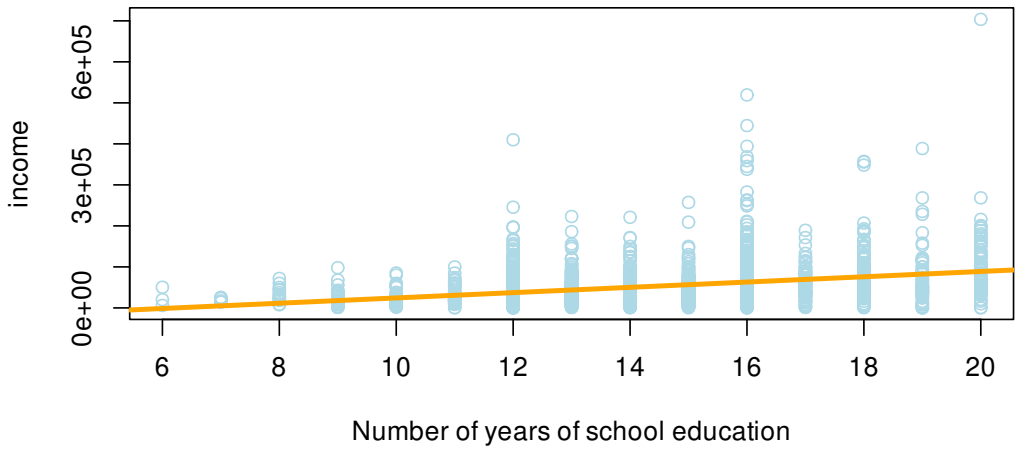
Sample solution for Problems 7

Solution 7.1

a)
```
income <- read.table(file="./Daten/income.dat", header=TRUE)
head(income)

##     AFQT Educ Income2005
## 1  6.841   12       5500
## 2 99.393   16      65000
## 3 47.412   12      19000
## 4 44.022   14      36000
## 5 59.683   14      65000
## 6 72.313   16       8000

iq <- income[,1]

number_years_of_school <- income[, 2]

income <- income[,3]

plot(iq,
     income,
     type = "p",
     xlab = "IQ",
     ylab = "income",
     col = "light blue"

)

abline(lm(income ~ iq),
       col = "orange",
       lwd = 3)
```




```
plot(number_years_of_school,
     income,
     type="p",
     xlab = "Number of years of school education",
     ylab="income",
     col = "light blue"

)

abline(lm(income ~ number_years_of_school),
       col = "orange",
       lwd = 3)
```



In both cases, the regression line is very flat and the points scatter quite a bit around the regression line.

b) With R we calculate for a and b

```
lm(income ~ number_years_of_school)

##
## Call:
## lm(formula = income ~ number_years_of_school)
##
## Coefficients:
##            (Intercept)  number_years_of_school
##                 -40200                    6451
```

So we find the values $a = -40'200$ and $b = 6451$ für the case of income versus number of years of schooling (and $a = 21'182$ and $b = 518.68$ für the case of income versus intelligence quotient). Thus, every additional year of schooling is accompanied by an annual increase in income of 6451 USD. But be careful: someone without education would have an income of $-40'200$ USD. Of course this makes no sense. Whenever extrapolating into areas where no data points were available, caution should be exercised in interpretation.

c) For the *empirical correlation* we get

```
cor(number_years_of_school, income)

## [1] 0.3456474
```

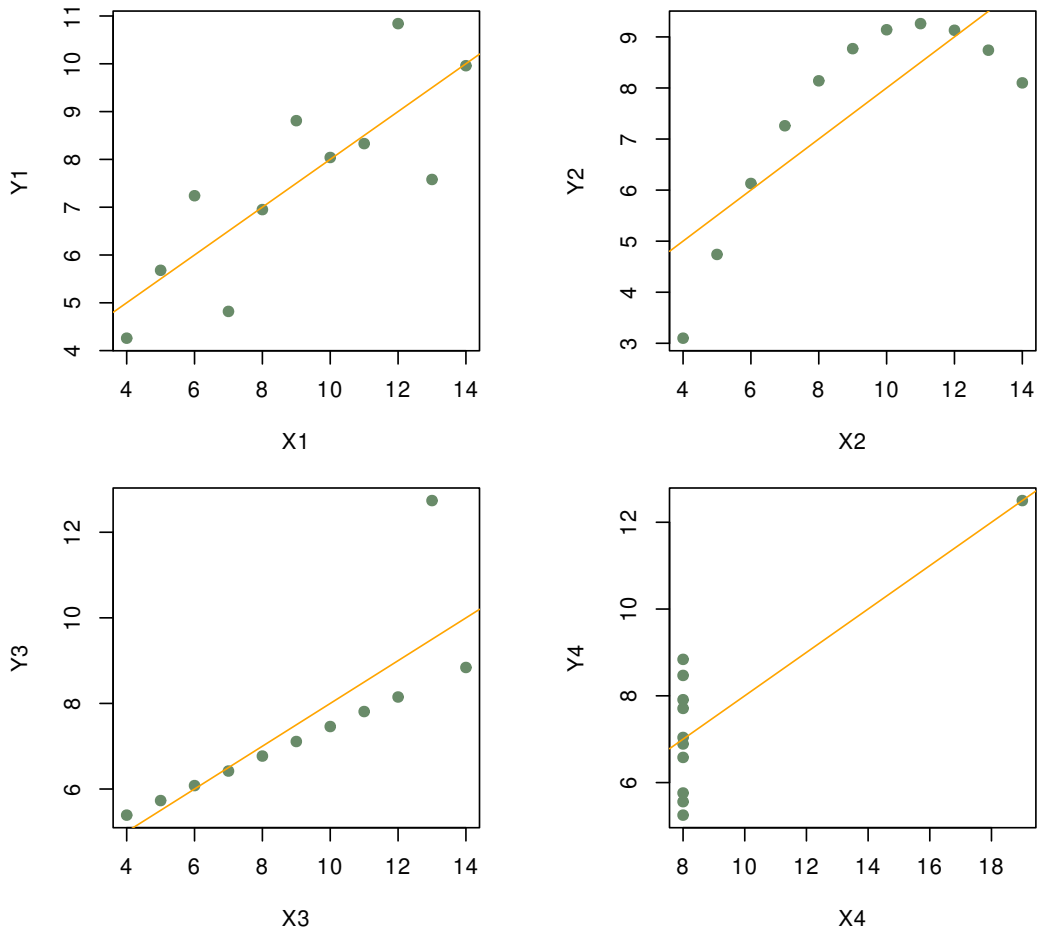
As the correlation coefficient is relatively small, a model based on a linear relationship between income and number of years of schooling does not seems to be appropriate.

Solution 7.2

a) If we look at the four scatter plots, we see that only in the first case a linear regression is correct. In the second case, the relationship between X and Y is not linear but quadratic. In the third case there is an outlier which strongly influences the estimated parameters. In the fourth case, the regression line is determined by a single point.

```
data(anscombe)
reg <- lm(anscombe$y1 ~ anscombe$x1)
reg2 <- lm(anscombe$y2 ~ anscombe$x2)
reg3 <- lm(anscombe$y3 ~ anscombe$x3)
reg4 <- lm(anscombe$y4 ~ anscombe$x4)
par(mfrow=c(2,2))
plot(anscombe$x1, anscombe$y1, ylab = "Y1", xlab = "X1", col="darkseagreen4", pch=
abline(reg, col = "orange")
plot(anscombe$x2, anscombe$y2, ylab = "Y2", xlab = "X2", col="darkseagreen4", pch=
abline(reg2, col = "orange")
plot(anscombe$x3, anscombe$y3, ylab = "Y3", xlab = "X3", col="darkseagreen4", pch=
abline(reg3, col = "orange")
plot(anscombe$x4, anscombe$y4, ylab= "Y4", xlab= "X4", col="darkseagreen4", pch=19
```

```
abline(reg4, col = "orange")
```



b) For all four models the estimates of the axis intercept β_0 and the slope β_1 are almost identical:

	model 1	model 2	model 3	model 4
Axis intercept (a)	3.000	3.001	3.002	3.002
Slope (β_1)	0.500	0.500	0.500	0.500

Conclusion: It is *not* enough to look at a and b. In all models these estimates are almost the same, but the records look very different. A (graphical) check of the model assumptions is therefore inevitable.

c)
```
cor(anscombe$x1, anscombe$y1)

## [1] 0.8164205

cor(anscombe$x2, anscombe$y2)
```

```
## [1] 0.8162365

cor(anscombe$x3, anscombe$y3)

## [1] 0.8162867

cor(anscombe$x4, anscombe$y4)

## [1] 0.8165214
```

Although the scatter plots look very different, the correlation coefficients are the same except for the 3rd digit after the decimal point.

Again: Don't just look exclusively at the correlation coefficient, but at the corresponding scatter plot.

Solution 7.3

a) Table:

```
library(ISLR)
head(Auto)

##   mpg cylinders displacement horsepower weight acceleration year
## 1  18         8          307        130   3504         12.0   70
## 2  15         8          350        165   3693         11.5   70
## 3  18         8          318        150   3436         11.0   70
## 4  16         8          304        150   3433         12.0   70
## 5  17         8          302        140   3449         10.5   70
## 6  15         8          429        198   4341         10.0   70
##   origin                       name
## 1      1 chevrolet chevelle malibu
## 2      1         buick skylark 320
## 3      1        plymouth satellite
## 4      1             amc rebel sst
## 5      1               ford torino
## 6      1          ford galaxie 500
```

```
help(Auto)
```

Auto {ISLR}

Auto Data Set

Description

Gas mileage, horsepower, and other information for 392 vehicles.

Usage

```
Auto
```

Format

A data frame with 392 observations on the following 9 variables.

`mpg`
: miles per gallon

`cylinders`
: Number of cylinders between 4 and 8

`displacement`
: Engine displacement (cu. inches)

`horsepower`
: Engine horsepower

`weight`
: Vehicle weight (lbs.)

`acceleration`
: Time to accelerate from 0 to 60 mph (sec.)

`year`
: Model year (modulo 100)

`origin`
: Origin of car (1. American, 2. European, 3. Japanese)

`name`
: Vehicle name

The orginal data contained 408 observations but 16 observations with missing values were removed.

Source

This dataset was taken from the StatLib library which is maintained at Carnegie Mellon University. The dataset was used in the 1983 American Statistical Association Exposition.

References

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013) *An Introduction to Statistical Learning with applications in R*, www.StatLearning.com, Springer-Verlag, New York

b) Linear regression:

$$\texttt{mpg} = \beta_0 + \beta_1 \cdot \texttt{horsepower}$$

c) Output:

```
fit <- lm(mpg ~ horsepower, data = Auto)
# Or: fit <- lm(Auto$mpg ~ Auto$horsepower)

summary(fit)

##
## Call:
## lm(formula = mpg ~ horsepower, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -13.5710  -3.2592  -0.3435   2.7630  16.9240
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 39.935861   0.717499   55.66   <2e-16 ***
## horsepower  -0.157845   0.006446  -24.49   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.906 on 390 degrees of freedom
## Multiple R-squared:  0.6059,Adjusted R-squared:  0.6049
## F-statistic: 599.7 on 1 and 390 DF,  p-value: < 2.2e-16
```

i) The p value for **horsepower** is almost 0 and so the null hypothesis ($\beta_1 = 0$) is rejected. The fuel consumption depends on the horsepower.

ii) The value 39.93 for the **intercept** indicates the fuel consumption (miles per gallon) at 0 hp. Of course this value has no practical meaning here.

More interesting is the value -0.15 for **horsepower**. This means that per hp the car gets 0.15 miles less for one gallon ($\approx 3.8\,\text{l}$) of petrol.

So the correlation is negative: the more horsepower the less distance is travelled per gallon.

iii) confidence interval:

```
confint(fit)

##                 2.5 %     97.5 %
## (Intercept) 38.525212 41.3465103
## horsepower  -0.170517 -0.1451725
```

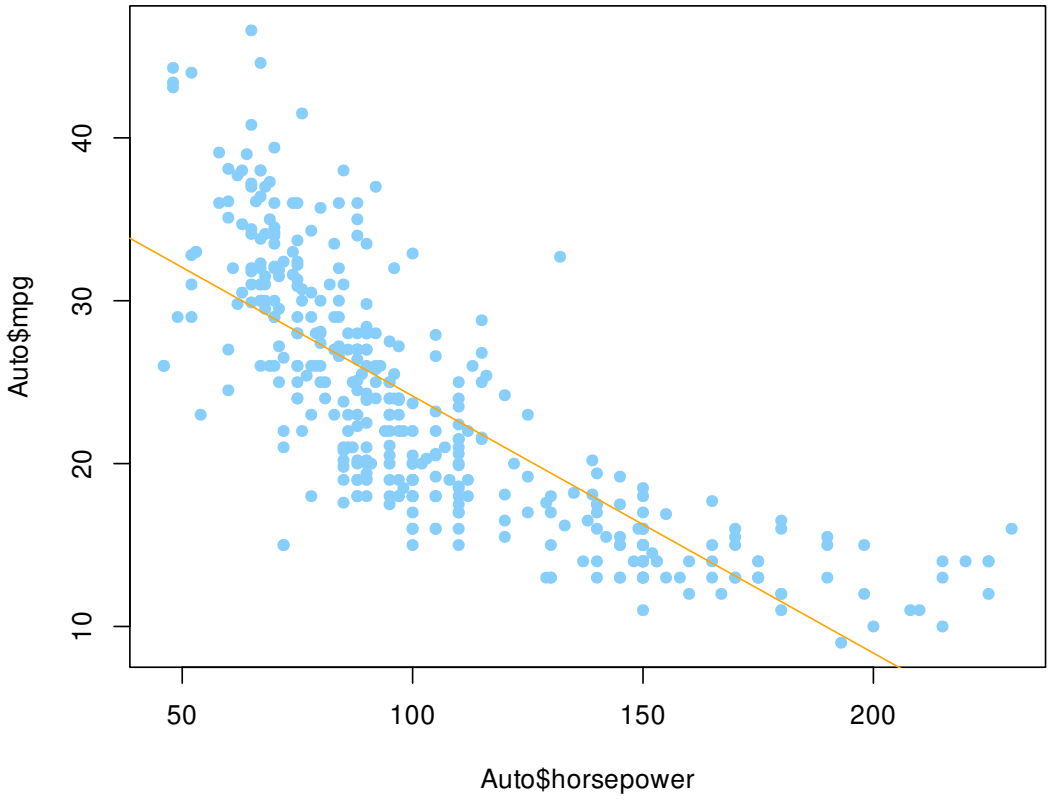
The true values for **intercept** and **horsepower** lie at 95 % in the corresponding intervals. The intervals are quite narrow, so that the significance of these intervals is quite large.

iv) The R^2 value is 0.606. This indicates that the variability to 60 % is through the model.

This is ok, but not very good, because other predictors also have an influence on the fuel consumption.

d) Plot:

```
plot(Auto$horsepower, Auto$mpg, pch=16, col="lightskyblue")
abline(lm(Auto$mpg~Auto$horsepower), col="orange")
```



The downward trend is clearly visible, hence the low p value. However, the point cloud does not fall linearly (weak R^2 value).

Solution 7.4

a)

b) Column names

```
library(MASS)
colnames(Boston)

##  [1] "crim"    "zn"      "indus"   "chas"    "nox"     "rm"
##  [7] "age"     "dis"     "rad"     "tax"     "ptratio" "black"
## [13] "lstat"   "medv"
```

c) Attach

```
attach(Boston)
```

d) i) The model is defined as follows:

$$\texttt{medv} = \beta_0 + \beta_1 \cdot \texttt{lstat}$$

ii) Output

```
lm.fit <- lm(medv ~ lstat)
summary(lm.fit)

##
## Call:
## lm(formula = medv ~ lstat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.168  -3.990  -1.318   2.034  24.500
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 34.55384    0.56263   61.41   <2e-16 ***
## lstat       -0.95005    0.03873  -24.53   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.216 on 504 degrees of freedom
## Multiple R-squared:  0.5441,Adjusted R-squared:  0.5432
## F-statistic: 601.6 on 1 and 504 DF,  p-value: < 2.2e-16
```

e)

```
names(lm.fit)

##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
```

f)

```
coef(lm.fit)

## (Intercept)       lstat
##  34.5538409  -0.9500494
```

Substitute these values in the simple linear regression model above

$$\texttt{medv} = 34.554 - 0.95 \cdot \texttt{lstat}$$

The values 34.55 is the intercept, which is the value for `lstat = 0` (zero percent of lower status of the population). The median house value is \$34 554 in neighborhoods with 0 percent population of lower status.

The value -0.95 is the slope of the regression line. We can interpret this value as follows: for each additional percent in population of lower status, the median house value drops by \$950.

The p-value for `lstat` is close to 0 and therefore highly significant because less than significance level of 0.05. The socioeconomic status does have an influence on the median house value.

g)
```
confint(lm.fit)

##                 2.5 %     97.5 %
## (Intercept) 33.448457 35.6592247
## lstat       -1.026148 -0.8739505
```

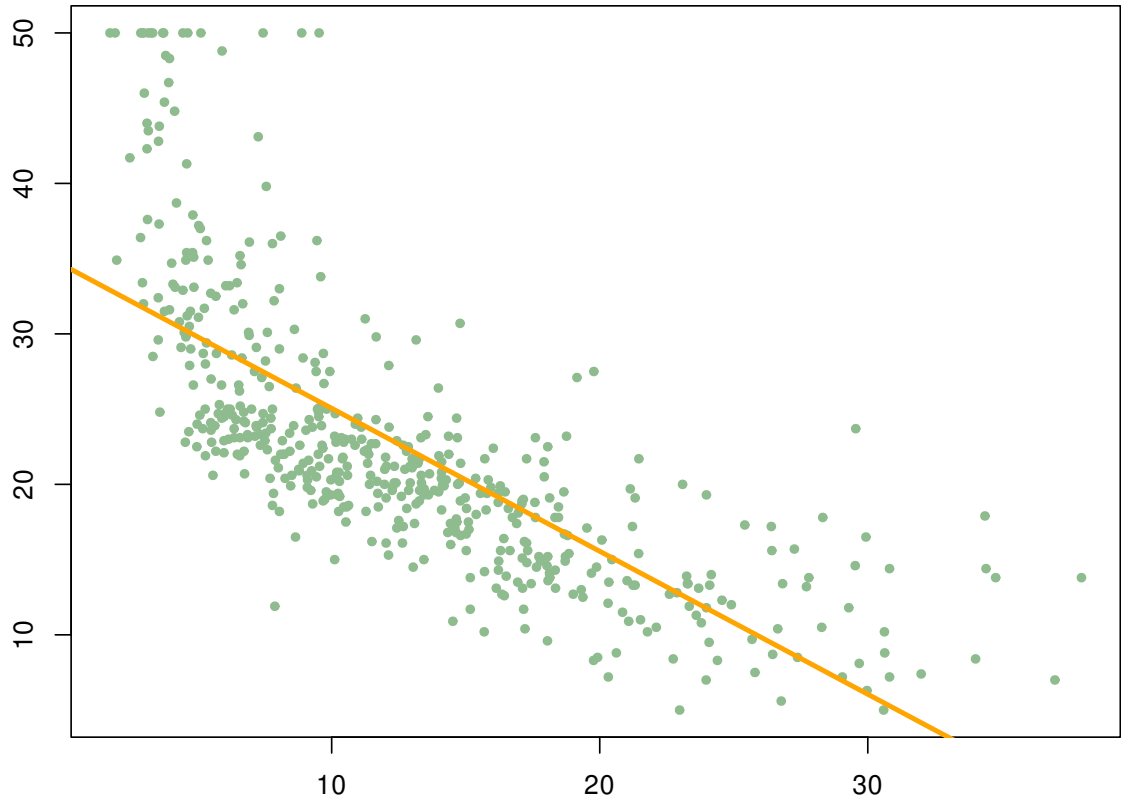
The true value of the intercept is with 95 % probability in the interval

$$[33.45,\ 35,66]$$

The true value of the slope is with 95 % probability in the interval

$$[-1.02,\ -0.87]$$

h) Plot:
```
plot(lstat, medv, col = "darkseagreen", pch = 20)
abline(lm.fit, col= "orange", lwd = 3)
```



i) The R^2-value is 0.5441, so about 54 % of the variability is explained by the model.