– Statistical Modeling – Prof Fulvia Pennoni

DAVIDE RONCHI - 903320

2024-04-03

Exercise 1

1.1

Illustrate the data using appropriate summary statistics. Draw the bivariate scatter plot and comment on it. Are the variables related to each other?

Data loading and preliminary analysis

Loading the dataset from "*heating.RData*" file using the *load()* function.

```r
load("heating.RData")
```

The file is loaded inside a variable called x, which is a matrix, as shown below by the function *class()*:

```r
class(x)
```

```
## [1] "matrix" "array"
```

We will transform the variable x into a dataframe for a more convenient handling using the function *data.frame()* and we will store it into a new variable called "*heating*". After that we delete the "*x*" variable from the environment using the *rm()* function:

```r
heating <- data.frame(x)
rm(x)
```

In this way inside our environment we just have the variable *heating* and we can explore it.

We use the functions:

- *class()* that returns what R considers as the *class* of the object;
- *typeof()* that returns what R considers as the *type* of the object.

```r
class(heating)
```

```
## [1] "data.frame"
```

```r
typeof(heating)
```

```
## [1] "list"
```

We can see that “*heating*” has a *data.frame* class and a *list* type.

Since *heating* is a data.frame we can use the function *head()* to look at it’s first 6 rows:

```r
head(heating)
```

```
##   temperature consumption
## 1       -0.99       17.69
## 2        2.49       17.21
## 3       -2.06       17.26
## 4       -0.78       16.88
## 5        0.07       17.40
## 6       -0.24       18.27
```

We can see that it contains 2 columns called **temperature** and **consumption** Now let’s explore the classes and types of them:

```r
class(heating$temperature)
```

```
## [1] "numeric"
```

```r
typeof(heating$temperature)
```

```
## [1] "double"
```

```r
class(heating$consumption)
```

```
## [1] "numeric"
```

```r
typeof(heating$consumption)
```

```
## [1] "double"
```

Both the *temperature* and the *consumption* columns are class *numeric* and type *double*, so can be considered as *continuous*. Alternatively we can use the function *skim* from the *skimr* library to evaluate variable types and also have a brief summary of data statistics. In particular we use the function *skim_without_charts* to avoid printing histograms of the data:

```r
library(skimr)
skim_without_charts(heating)
```

Table 1: Data summary

Name	heating
Number of rows	67
Number of columns	2
Column type frequency:	
numeric	2
Group variables	None

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
temperature	0	1	-0.17	1.60	-3.55	-1.12	-0.03	1.01	4.44
consumption	0	1	16.91	1.58	13.44	15.79	17.02	17.94	21.54

The variable "*heating*" contains 67 rows and 2 columns ("temperature" and "consumption"), both of them of type *numeric*. We can see that both of them has no missing values.

The variable **temperature** represents the external temperature, measured in Celsius degrees, for 67 houses. The variable **consumption** represents the Energy consumption for home heating, measured in kWh. With these information we could be interested in searching for associations between them and we could interpret the variable **consumption** as the **response** variable and the variable **temperature** as the covariate, so we could be interested in evaluating how the temperature influences consumption for home heating.

Comments on the observed values of the response:

- The **mean** value of **temperature** is around -0.17, suggesting that the distribution of the values is quite centered around zero;
- It has a **standard deviation** of 1.60. It has quite a contained variability, considering also that the **minimum** value (p_0) is $p_0 = -3.55$ and the **maximum** (p_{100}) is $p_{100} = 4.44$;
- The **range** is given by $p_{100} - p_0 = 4.44 - (-3.55) = 7.99$;
- The 1^{st} **quartile** (p_{25}) is $p_{25} = -1.12$, which means that a quarter of the temperature values are below -1.12 °C;
- The 3^{rd} **quartile** (p_{75}) is $p_{75} = 1.01$, which means that a quarter of the temperature values are above 1.010 °C;
- The **median** value is represented by p_{50} and has a value of $p_{50} = -0.03$, which is quite similar to the **mean**. This suggest that data are more or less symmetrical and centered around the mean with a low skewness.

Comments on the observed values of the covariate:

- The **mean** value of **consumption** is around 16.91;
- It has a **standard deviation** of 1.58, that means the consumption has a low variability;
- The **minimum** value (p_0) is $p_0 = 13.44$ and the **maximum** (p_{100}) is $p_{100} = 21.54$;
- The **range** is given by $p_{100} - p_0 = 21.54 - (13.44) = 8.10$;
- The 1^{st} **quartile** (p_{25}) is $p_{25} = 15.79$, which means that a quarter of the consumption values are below 15.785 [kWh].
- The 3^{rd} **quartile** (p_{75}) is $p_{75} = 17.94$, which means that a quarter of the consumption values are above 17.935 [kWh].
- The **median** value is represented by p_{50} and has a value of $p_{50} = 17.02$, which is quite similar to the **mean**. This suggest that data are more or less symmetrical and centered around the mean with a low skewness.

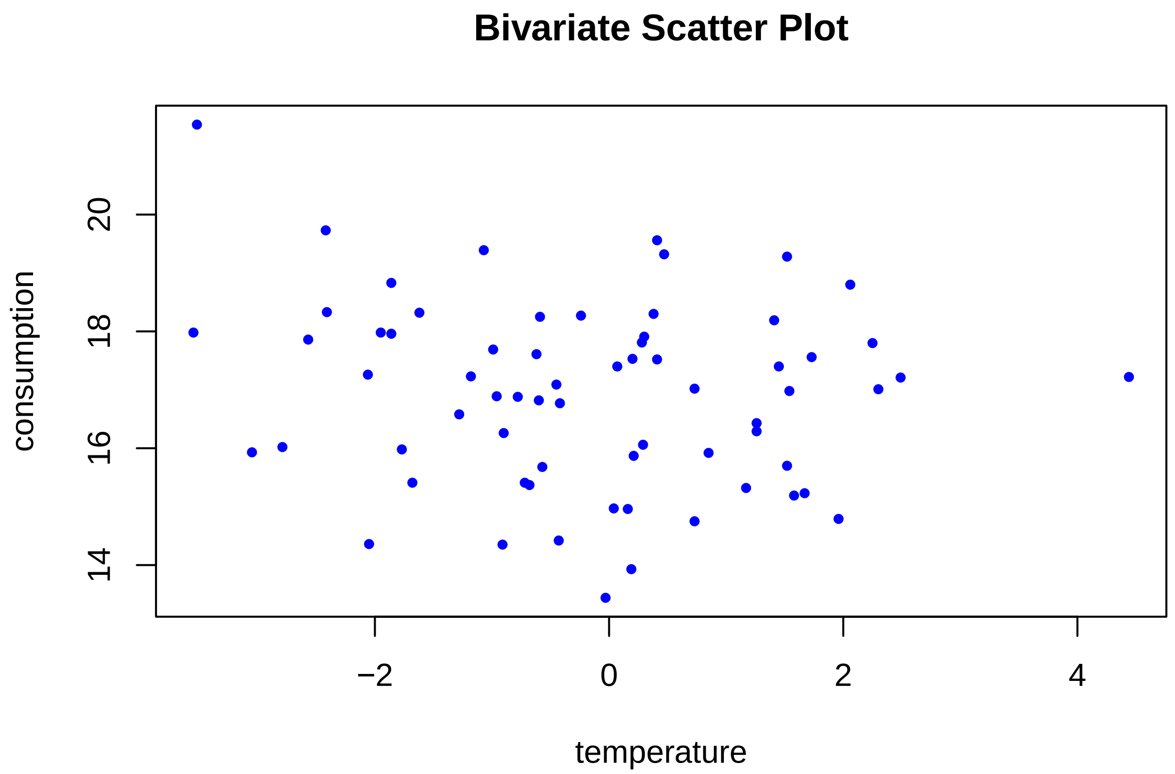
Bivariate Scatter Plot

To explore the association between the two variables we can use a **scatterplot** to have a first visual qualitative idea of their association. This plot visually depicts the relationship between the values of one variable "x" and the corresponding "y" values. To achieve this we use the *plot()* function which requires two lists of values as input: temperature (heating$temperature) as x coordinates and consumption (heating$consumption) as y coordinates.

- *main* set the title of the plot;

- *xlab* and *ylab* set the title for the x and y labels respectively;
- *col=“blue”* set the points color to blue
- *pch=16* choose the type of symbol to use, which in this case is “filled circle”;
- *cex = 0.7* reduce the dimension of the points to 70% of the original size.

```r
plot(heating$temperature,
     heating$consumption,
     main = "Bivariate Scatter Plot",
     xlab = "temperature",
     ylab = "consumption",
     col = "blue",
     pch = 16,
     cex = 0.7)
```



Looking at the plot we can see that the points are spread more or less randomly in the center without a particular trend, except for a slightly decreasing one in the first half of the plot. This suggest that there is not a particularly strong association between the two variable (at least not linear) except maybe for a weak negative one. In order to evaluate their actual association we need further investigations.

1.2

Calculate and comment on the value of the sample linear correlation coefficient.

A way of evaluating correlation between two random variables is by calculating the **Correlation Coefficient**.

The correlation coefficient between two random variables X,Y is defined as follows:

$$Cor(Y,X) = \rho_{YX} = \frac{\sigma_{YX}}{\sigma_Y \sigma_X} = \frac{Cov(Y,X)}{\sqrt{Var(Y)Var(X)}}$$

. Since we are dealing with a sample and not with a random variable distribution, we can evaluate the **sample correlation coefficient** which is defined as:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

To do so we can use the *cor()* function and provide it the values of temperature and consumption

```
cc <- round(cor(heating$consumption, heating$temperature), 3)
cc
```

```
## [1] -0.154
```

As we can see the **correlation coefficient** is $r_{x,y} = -0.154$, negative and really close to 0. This suggests that:

- The association between the two variables is a negative linear association;
- Since the magnitude is really close to zero (0.154) there is just a weak linear association between them.

So this suggests that, overall, when temperature increases, consumption slightly decreases. However this negative association is really weak, so it is required to further investigate to understand this association more comprehensively.
Furthermore, the **sample correlation coefficient** is a measure of linear association, it means that can help us excluding **linear** associations between variables, but we can't conclude that there is not association at all: e.g. in case their association is not linear.

1.3

Considering the correlation coefficient as a parameter of interest, use the non-parametric bootstrap to provide an accuracy measure for it. Obtain the bootstrap distribution from B = 1000 resamples and comment on the bootstrap estimate of the corresponding standard error.

Since before we calculated the Sample correlation coefficient through a plug-in estimator, we are interested in evaluating the accuracy of this estimator.
A way to do it is by using the **Non parametric bootstrap** method. It is very similar to the Monte Carlo simulation method, but with a conceptual difference: in the Monte Carlo simulation data are generated randomly starting from a theoretical distribution of random variable, while in Bootstrap data are *resampled* with replacement directly from the original data.
The sample data are treated as the original population, this method ensures that the non parametric bootstrap samples will have the distribution of the original data.

In our case we want to perform a Non parametric bootstrap procedure with $B = 1000$ resamples. To do so we use the following code:

- Set B equal to 1000, which will be the number of Bootstrap samples used;
- Set n equal to the sample size, which is *length(heating$temperature)* = 67;
- Create a variable *Tboot* as a sequence of B(=1000) zeros;
- Set the seed equal to "12391" for reproducibility;

```
B <- 1000
n <- length(heating$temperature)
Tboot <- rep(0, B)
set.seed(12391)
```

After that we use a for loop, in order to perform the resampling B=1000 times. For each iteration we will do as follow:

- *sample_indices* is a variable that contains the indices of the values from *heating* of which the current resample will be constituted of. These indices are extracted from a sequence of numbers from 1 to n(=67) via the *sample()* function. The sample size will be equal to n(=67) and the sampling will be performed with replacement. It means that every time a value is extracted from the original sample it will be reintroduced inside of it, such that every time we perform a sampling of a value, it will be equally likely to be extracted;
- *Xstar* will contains the values of *heating* at the previously extracted indeces;
- We set the value of Tboot[i] equal to the correlation coefficient of the current resample.

After that we will print the first 6 values of *Tboot* using the function *head()* and its summary, to evaluate the basic statistics.

```
for (i in 1:B){
  sampled_indeces <- sample(1:n, size = n, replace = TRUE)
  Xstar <- heating[sampled_indeces, ]

  Tboot[i] <- cor(Xstar$temperature, Xstar$consumption)
}

head(Tboot)
```

```
## [1]  0.05918733 -0.19498140  0.01629248 -0.05030264 -0.09728681 -0.12680828
```

```
summary(Tboot)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
## -0.59878 -0.23605 -0.15192 -0.14981 -0.06519  0.21033
```

Wave generated 1000 Bootstrap Replications, calculated the sample correlation coefficient for each of them and put the result in a variable called *Tboot*. So *Tboot* contains the sample correlation coefficient of each Bootstrap Replication.
By looking at the first 6 values we can see that they are all near to our plug-in estimation, that was -0.154. In particular the **mean** (-0.150) and **median**(-0.152) values of the *Tboot* distribution are really similar to our estimation. The range of variation is between -0.599 and 0.210.

After that we can calculate the **Standard Deviation** of these values.

```
se_Tboot <- sd(Tboot)
se_Tboot
```

```
## [1] 0.1254859
```

The standard error of the point estimate of the Correlation Coefficient of the reference population is around 0.125 (the estimate is -0.154). We notice that the magnitude of the standard deviation is quite high compared to the plug-in estimation, it means that we have a considerable variability in the estimates obtained from bootstrapping.

1.4

Provide the bootstrap percentile confidence interval for the parameter computed at 98% confidence level using the percentile method. Comment on its value and length.

Confidence Interval

Now we would like to calculate the **Confidence interval** of this estimate. The aim of the confidence interval is to find an interval of plausible values for an unknown parameter θ on basis of sampling information. The confidence interval specifies a *range* of possible values within which the parameter is *estimated* to lie, instead of a single value.
The probability that the confidence interval method produces an interval that truly contains the parameter θ is called **confidence level**. The confidence level is chosen *"a priori"* to be an high value (around 1), such as 0.95 or 0.99.
The interval is given by $[L(y), U(y)]$. This is called a $100(1-\alpha)\%$ confidence interval, with lower and upper confidence limits $L(y)$ and $U(y)$.
The probability $1-\alpha$ is the confidence level of the interval.

The probability, by definition, applies to random variables (before we observe the data), not to parameter values. Since the bootstrap distribution of an estimator $\widehat{T}$ is a simulation, by resampling, of the real distribution of $\widehat{T}$, the bootstrap histogram can be considered a Monte Carlo approximation of the latter

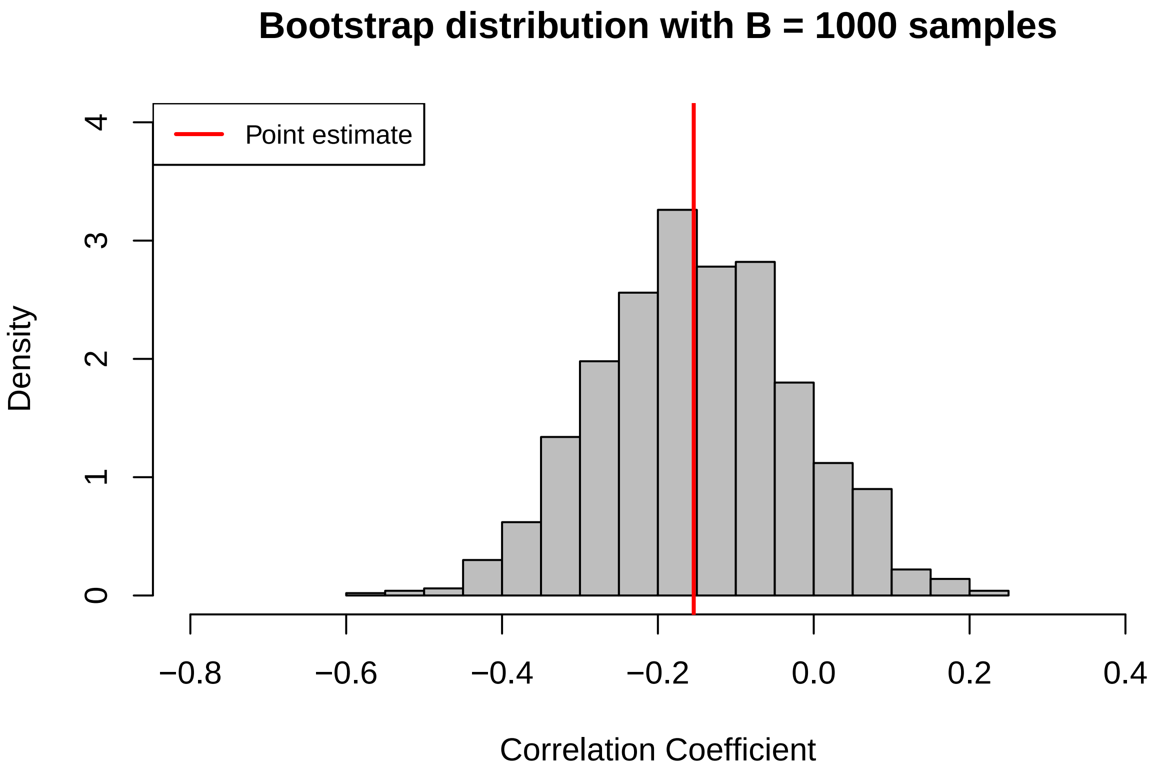
Percentile confidence interval

In our case we want to calculate the **98%** confidence interval, it means that the confidence level is $CL = 0.98$ and $\alpha = 1 - CL = 1 - 0.98 = 0.02$. First of all we can use an histogram to depict the distribution of the linear correlation coefficients obtained from each sample with the bootstrap method. We also plot the plug-in estimation from the original sample.

To do so we use the *hist* function:

- To determine the number of breaks we use breaks="FD" parameter. It corresponds to "Freedman-Diaconis" algorithm, which is based on the inter-quartile range;
- *freq* is set to FALSE in order to set the histogram a graphic representation of a probability density;
- *main* parameter is used to set the title of the plot;
- *xlab* and *ylab* to set the label of the x and y axis respectively;
- *xlim* and *ylim* to set the range of values to be displayed on the x and y axis respectively;
- *col* to set the color of the histogram bars to "gray"
  To depict the value of our estimation of the correlation coefficient from the original sample we use the function *abline()*, to draw a red vertical line at the value x = -0.154. The parameter *lwd* is used to set the line width to 2.
  We also plot a legend using the *legend()* function:
- *"topleft"* set the position of the legend in the *top-left* corner of the plot;
- The parameter "2" specifies the inset distance from the margins;
- c("Point estimate") contains the labels for the legend items;
- col = "red": This sets to *red* the color of the legend item;
- lty = 1 specifies the *line-type* of the item;
- cex = 0.8 adjust the size of the text (in this case set it to 80% of the default)

```r
hist(Tboot,
      breaks= nclass.FD(Tboot),
      freq=FALSE,
      main = paste("Bootstrap distribution with B =", B, "samples"),
      xlab = "Correlation Coefficient",
      ylab = "Density",
      xlim = c(-0.8,0.4),
      ylim = c(0,4),
      col= "gray"
      )
abline(v=cc, col="red", lwd=2)
legend("topleft", 2,
       c("Point estimate"),
       col = "red",
       lty= 1,
       cex = 0.8,
       lwd = 2)
```



By looking at the histogram we can see that the distribution of the data is **bell-shaped**, suggesting a normal distribution. Furthermore we can see that it is centered around our plug-in estimate.
We can also notice a that the distribution is not exactly symmetric and has a little longer tail on the right, indicating a small positive skewness.
To get a more informative insight we can calculate the confidence interval for that estimate, based on the distribution we obtained from the bootstrap method:

The variable Q will contains the quantiles of interest of the *Tboot* distribution. These will be used as boundaries for of the Confidence Interval; also we calculate the mean value. To do so we use the function *quantile()* providing it the variable *Tboot* that is the numeric vector whose sample quantiles are wanted, while with the parameter *probs* we provide the vector of the two values of the quantiles we are aiming to calculate: $1 - \frac{\alpha}{2} = 0.99$ and $\frac{\alpha}{2} = 0.01$.

```
CL <- 0.98
alpha <- 1 - CL
Q <- quantile(Tboot, probs = c(alpha/2, (1-alpha/2)))
B_cc_mean <- round(mean(Tboot), 3)
B_cc_mean
```

```
## [1] -0.15
```

```
round(Q, 3)
```

```
##     1%    99%
## -0.430  0.132
```

We have already seen that the mean is -0.150, which is similar to our estimation.
The results of the quantiles are the following:

- Q[1] (1%) ≈ -0.430. It means that 1% of the data in *Tboot* falls under this value. This will represent the **lower bound** of our confidence interval;
- Q[1] (99%) ≈ 0.132. It means that 99% of the data in *Tboot* falls above this value. This will represent the **upper bound** of our confidence interval.

$$CI_{0.98} \approx (-0.430, 0.132)$$

Its length is given by $0.132 - (-0.430) \approx 0.562$.
It means that the range over which we expect to find the true value of the Correlation Coefficient is small in absolute terms, but quite high in terms of relative magnitude with respect to the estimated value.

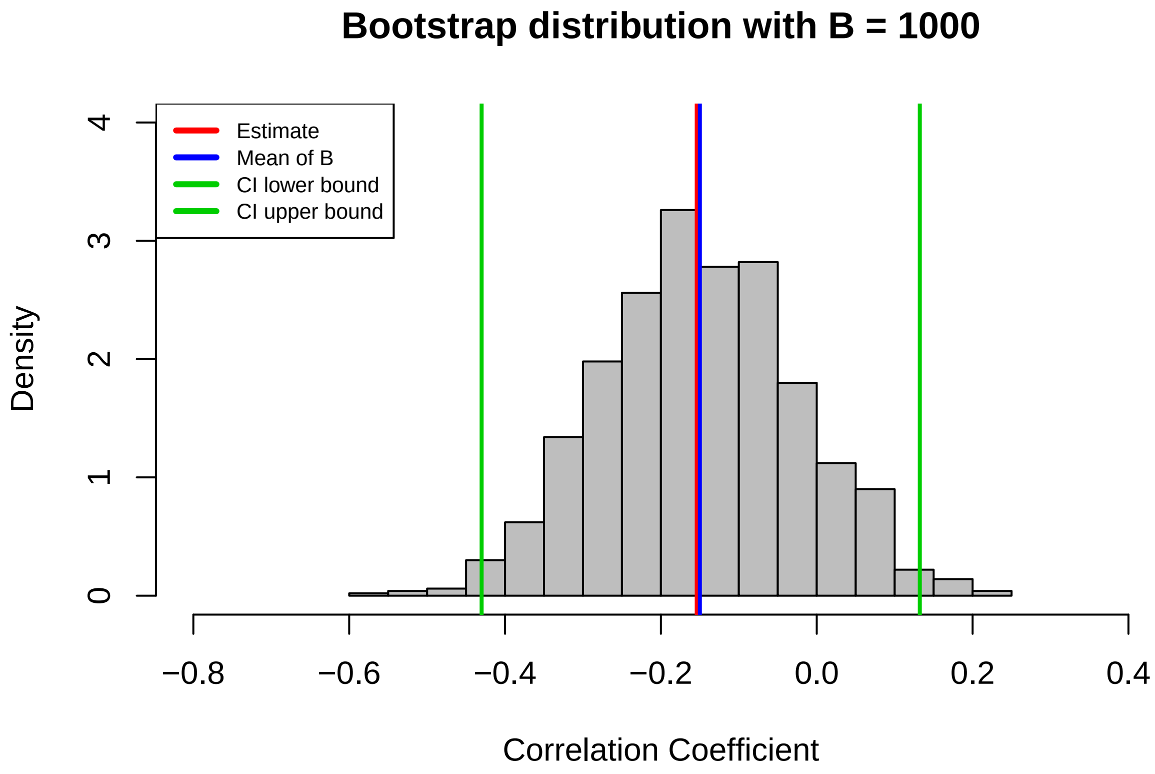
1.5

Depict the bootstrap distribution for the parameter of interest and add the lower and upper bound of the confidence interval and the legend. Comment on the shape of the distribution.

In order to better understand the meaning of these values we can plot again the histogram of the *Tboot* distribution, adding the values of the mean, the lower and upper bound of the Confidence Interval with the function *abline()* in a similar way as before:

```
hist(Tboot,
     breaks = nclass.FD(Tboot),
     freq=FALSE,
     main =paste("Bootstrap distribution with B =", B),
     xlab = "Correlation Coefficient",
     col = "gray",
     ylab = "Density",
     xlim = c(-0.8,0.4),
     ylim = c(0,4))
abline(v = c(cc,B_cc_mean, Q[1], Q[2]),
       col = c("red", "blue", "green3", "green3"),
       lwd = c(2,2,2,2))
```

```
legend("topleft",
       c("Estimate", "Mean of B",
         "CI lower bound", "CI upper bound"),
       col = c("red", "blue", "green3", "green3"),
       lty = c(1,1,1,1),
       lwd = c(3,3,3,3),
       cex = 0.7)
```



As we can see most of the data are contained inside the confidence interval boundaries and the mean value of *Tboot* nearly overlap with our plug-in estimator. We can make the same considerations about the shape as before.

Exercise 2

What may affect the chance of getting strong concrete? Consider data regarding the measurements of certain brands of concrete available in the file in concrete.Rdata1. Key variables to respond to the previous question are the following:

- Cement: density of different components (in kg/m3)
- Slag: blast furnace slag (in kg/m3)
- Ash: fly Ash (in kg/m3)
- Water: water (in kg/m3)
- Superplasticizer: superplasticizer (in kg/m3)

- Coarse: coarse aggregate (in kg/m3)
- Fine: fine aggregate (in kg/m3)

2.1

Explain if the data are collected with a randomized experiment or under an observation study. Define the sample size, and comment on the first six rows of the data.

Data loading

We load the data using the *load()* function:

```
load("concrete.Rdata")
class(concrete)
```

```
## [1] "data.frame"
```

Since the variable *concrete* is loaded as a *data.frame* we use the function *nrow()* to evaluate the number of rows contained in the dataframe, which can be considered as the sample size. The data are collected from an **observational study** and not from a randomized experiment.

Initial observations

Using the function *head()* we can also show its first 6 rows

```
sample_size <- nrow(concrete)
sample_size
```

```
## [1] 120
```

```
head(concrete)
```

```
##   Strength Cement Slag   Ash Water Coarse  Fine
## 1     7.75  16.80 4.21 16.38 12.18 10.587 7.801
## 2    18.00  21.37 9.81  2.45 18.17 10.658 7.854
## 3    13.18  21.38 9.81  2.45 18.17 10.660 7.855
## 4     7.32  18.20 4.52 12.20 17.02 10.594 7.807
## 5     7.40  16.89 4.22 12.43 15.83 10.808 7.962
## 6    30.45  27.72 9.78  2.45 16.07 10.617 7.825
```

We can see that the dataframe contains 120 rows, which is the sample size. By looking at the first 6 rows we can see that the dataframe contains 7 variables: *"Strength"*, *"Cement"*, *"Slag"*, *"Ash"*, *"Water"*, *"Coarse"*, *"Fine"*. We could be interested in analyzing the association between them, for example we can conduct analysis on the data considering *Strength* as the **response variable** and the others as **Explanatory variables**.
Among the first six rows we can observe the following:

- The variable **Strength** has quite an high variability, the lowest value of 7.32 in the 4^{th} row and the highest of 30.45 in the 6^{th}. We observe an high variability also for the variable **Ash**, that has a lowest value of 2.45 for the 6^{th} row and the highest of 16.38 to the 1^{st}. We notice that the lowest value of *Ash* corresponds to the highest of *Strength* and an high value (12.20), although not the highest, corresponds to the lowest value of *Strength*. In general we see high values of *Ash* corresponding to low values of *Strength*, suggesting a negative correlation;

- For the variable **Cement** we can see a more contained variability, with the lowest and highest values of 16.80 in the 1st row and 27.72 in the 6th respectively. The same is true for the variable **slag** with a lowest of 4.21 in the 1st and highest of 9.81 in the 2nd and 3rd. Again for the variable **Water**, that present the lowest value of 12.18 in the 1st row and the highest of 18.17 in the 2nd and 3rd. We notice that for three of them we have high values corresponding to high values of *Strength*, suggesting a positive association;
- Variables **Coarse** and **Fine** present a much lower variability, making it difficult to notice possible correlations at first glance. *Coarse* present a lowest value of *10.587* in the 1st row and the highest of 10.808 in the 5th; both of them corresponds to low values of *strength*, suggesting no particular associations. *Fine* present the lowest value of 7.801 in the 1st row and the highest of 7.962 in the 5th, suggesting the same conclusions.

2.2

Illustrate the data reporting and commenting on the descriptive statistics for each variable.

Using the *skim* function from the library *skimr* we show some basic summary statistics:

```r
library(skimr)
skim_without_charts(concrete)
```

Table 3: Data summary

Name	concrete
Number of rows	120
Number of columns	7
Column type frequency:	
numeric	7
Group variables	None

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
Strength	0	1	34.61	13.41	7.32	26.00	33.30	43.75	76.24
Cement	0	1	22.51	9.03	13.20	15.30	17.59	27.65	49.10
Slag	0	1	11.58	5.68	1.10	9.30	11.93	16.17	21.40
Ash	0	1	11.82	3.50	2.45	9.19	11.46	14.10	19.50
Water	0	1	18.47	2.09	12.18	17.20	18.09	19.62	24.70
Coarse	0	1	9.13	0.68	8.14	8.60	9.07	9.53	10.81
Fine	0	1	7.39	0.66	6.12	6.90	7.47	7.85	8.80

As stated before we have 7 variables. All of them have class *numeric* and type *double*. Along the 120 rows we have no missing values.

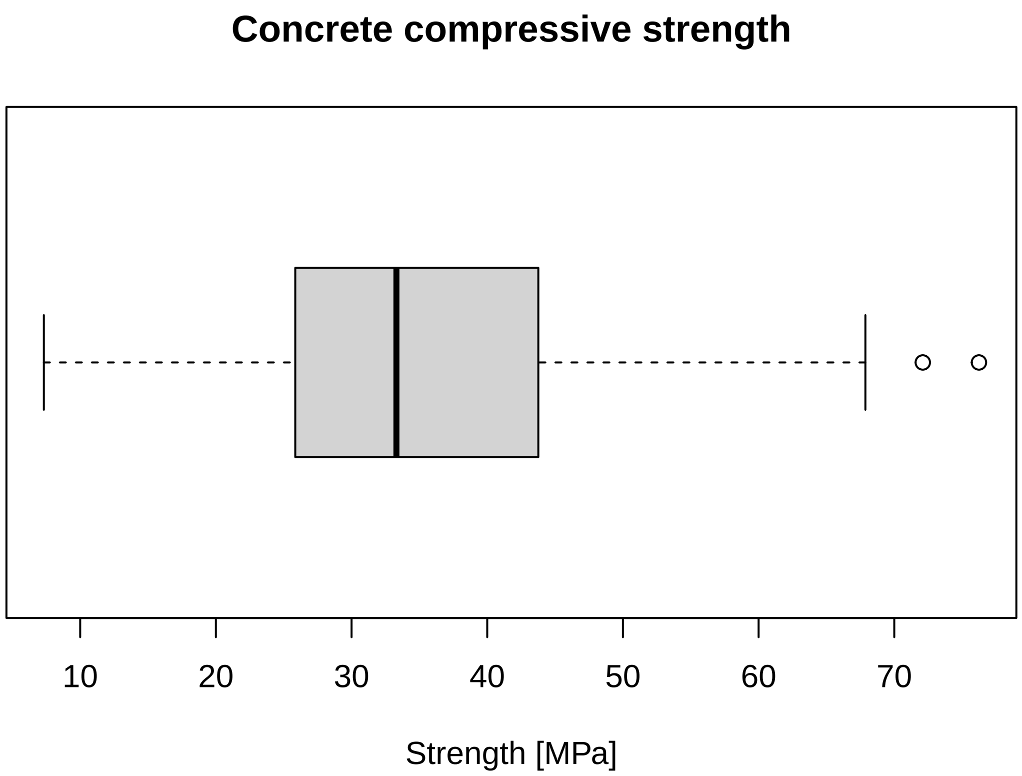
Comments on the observed values of the response (Strength):

- The **mean** value of **Strength** is around 34.61, with a **standard deviation** of 13.41. It has quite an high variability, considering that the **minimum** value (p_0) is **7.32** and the **maximum** (p_{100}) is **76.24**.
- The **range** is given by $p_{100} - p_0 \approx 76.24 - 7.32 \approx 68.92$.

- The 1^{st} **quartile** (p_{25}) is $p_{25} \approx 26.00$, which means that a quarter of the *Strength* values are below 26.00 MPa;
- The 3^{rd} **quartile** (p_{75}) is $p_{75} \approx 43.75$, which means that a quarter of the *Strength* values are above 43.75 MPa;
- The **median** value is represented by p_{50} and has a value of 33.31, which is really similar to the **mean** value. This means that the distribution of the data is quite symmetric.

We can Have a better picture of its distribution by looking at the boxplot. To do it we use the *boxplot()* function, providing the data of *Strength*, setting "Concrete compressive strength" as the main title, "Strength" as label for the x axis, and "horizontal" parameter equal to TRUE in order to draw an horizontal boxplot, instead of a vertical one.

```
boxplot(concrete$Strength,
        main="Concrete compressive strength",
        xlab="Strength [MPa]",
        horizontal = TRUE)
```



As expected the data are more or less symmetric, with a left whisker a little shorter than the right one.

Comments on the observed values of the covariates:

Cement

- The **mean** value of **Cement** is around 22.51, with a **standard deviation** of 9.03. As for the response variable it has quite an high variability, considering also a **minimum** value (p_0) is **13.20** and the **maximum** (p_{100}) is **49.10**.;
- The **range** is given by $p_{100} - p_0 \approx 49.10 - 13.20 \approx 35.90$.

- The 1^{st} **quartile** (p_{25}) is $p_{25} = 15.30$, which means that a quarter of the *Cement* values are below 15.30 [kg/m3].
- The 3^{rd} **quartile** (p_{75}) is $p_{75} = 27.66$, which means that a quarter of the *Cement* values are above 27.66 [kg/m3].
- The **median** value is represented by p_{50} and has a value of $p_{50} = 17.59$, which is quite lower than the **mean**. This suggest that data are a little positively (right) skewed.

Slag and **Ash**

- These two variables have both similar **mean** value of 11.58 and 11.82 respectively, but the **standard deviation** of the first is a little bigger than the second, indicating a little more variability.
- The higher variability of *Slag* is confirmed also by the higher range that is given by $p_{100} - p_0 \approx 21.40 - 1.10 = 20.30$ for *Slag* and $p_{100} - p_0 \approx 19.50 - 2.45 = 17.05$
- And also by the **inter-quartile range** $p_{75} - p_{25} \approx 16.17 - 9.31 \approx 6.86$ for *Slag* and $p_{75} - p_{25} \approx 14.10 - 9.19 \approx 4.91$, which is a little lower, for *Ash*.
- The median values are also similar and are similar to the mean, indicating for both of them a symmetric distribution. The values are given by $p_{50} \approx 11.92$ for *Slag* and $p_{50} \approx 11.46$ for *Ash*.

Water

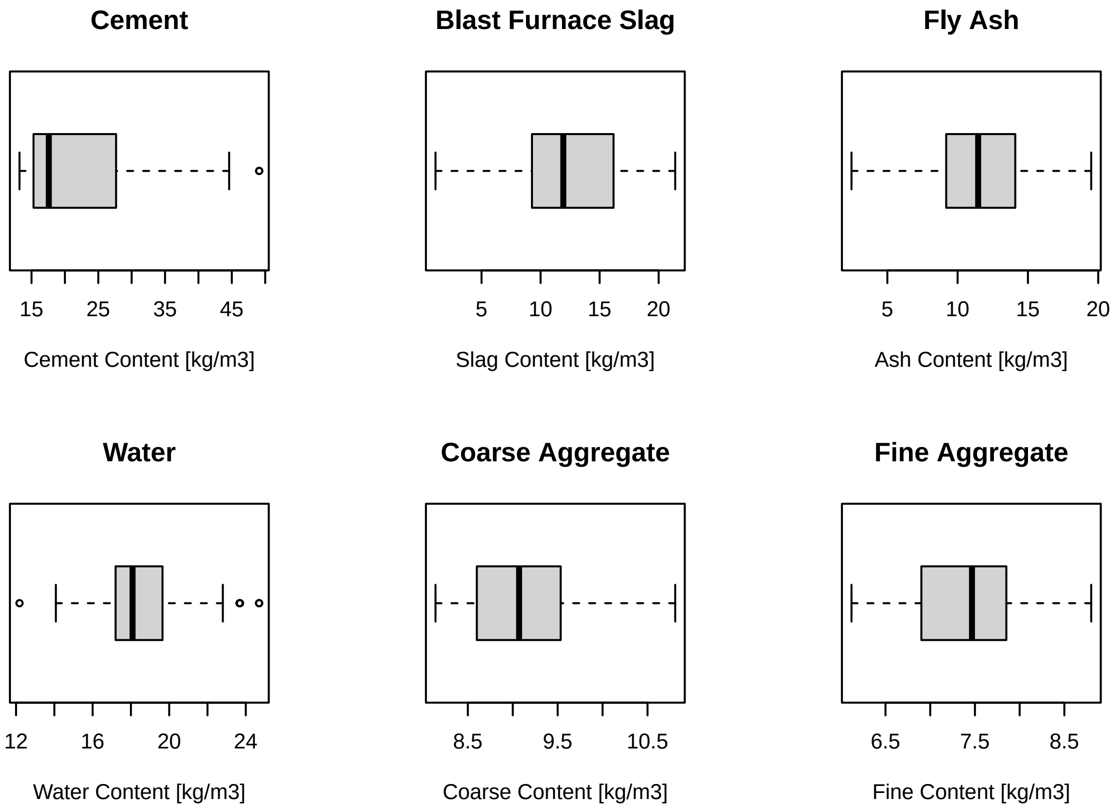
- The **mean** value of **Water** is around 18.47, with a **standard deviation** of 2.09, which is quite low compared to the absolute values. This is confirmed also by the **range**, which is given by $p_{100} - p_0 = 24.70 - 12.18 = 12.52$.
- The 1^{st} **quartile** (p_{25}) is $p_{25} \approx 17.20$, which means that a quarter of the *Water* values are below 17.20 [kg/m3].
- The 3^{rd} **quartile** (p_{75}) is $p_{75} \approx 19.63$, which means that a quarter of the *Water* values are above 19.63 [kg/m3].
- The **median** value is represented by p_{50} and has a value of $p_{50} \approx 18.09$, which is really similar to the **mean**. This suggest that data are more or less symmetric, thing confirmed by the boxplot below.

Coarse and **Fine**

- These two variables have a **mean** value of ≈ 9.13 for *Coarse* and ≈ 7.39 for *Fine*.
- Their **standard deviation** is really similar and much lower compared to the ones of the other variables: ≈ 0.68 and ≈ 0.66 respectively. This suggest a really low variability;
- This is confirmed also by their **range** of variation which is also very similar: $p_{100} - p_0 \approx 10.81 - 8.14 = 2.67$ for *Coarse* and $p_{100} - p_0 \approx 8.80 - 6.12 \approx 2.68$ for *Fine*.
- And also by the **inter-quartile range** $p_{75} - p_{25} \approx 9.53 - 8.60 \approx 0.93$ for *Coarse* and $p_{75} - p_{25} \approx 7.85 - 6.90 \approx 0.95$, for *Fine*, which is really similar as expected.
- The median values are similar to their respective mean values, indicating for both of them a symmetric distribution. The values are given by $p_{50} \approx 9.07$ for *Coarse* and $p_{50} \approx 7.47$ for *Fine*.
- We can conclude that their distribution is really similar in magnitude and variability, but *Coarse* distribution is "translated" of more or less 2 units to the right.

To better visualize the data and have a confirm of the above observations we can generate boxplots for all the covariates with the boxplot() function:

```
par(mfrow=c(2,3))

boxplot(concrete$Cement, main="Cement", xlab="Cement Content [kg/m3]", horizontal = TRUE)
boxplot(concrete$Slag, main="Blast Furnace Slag", xlab="Slag Content [kg/m3]", horizontal = TRUE)
boxplot(concrete$Ash, main="Fly Ash", xlab="Ash Content [kg/m3]",  horizontal = TRUE)
boxplot(concrete$Water, main="Water", xlab="Water Content [kg/m3]",  horizontal = TRUE)
boxplot(concrete$Coarse, main="Coarse Aggregate", xlab="Coarse Content [kg/m3]", horizontal = TRUE)
boxplot(concrete$Fine, main="Fine Aggregate", xlab="Fine Content [kg/m3]", horizontal = TRUE)
```

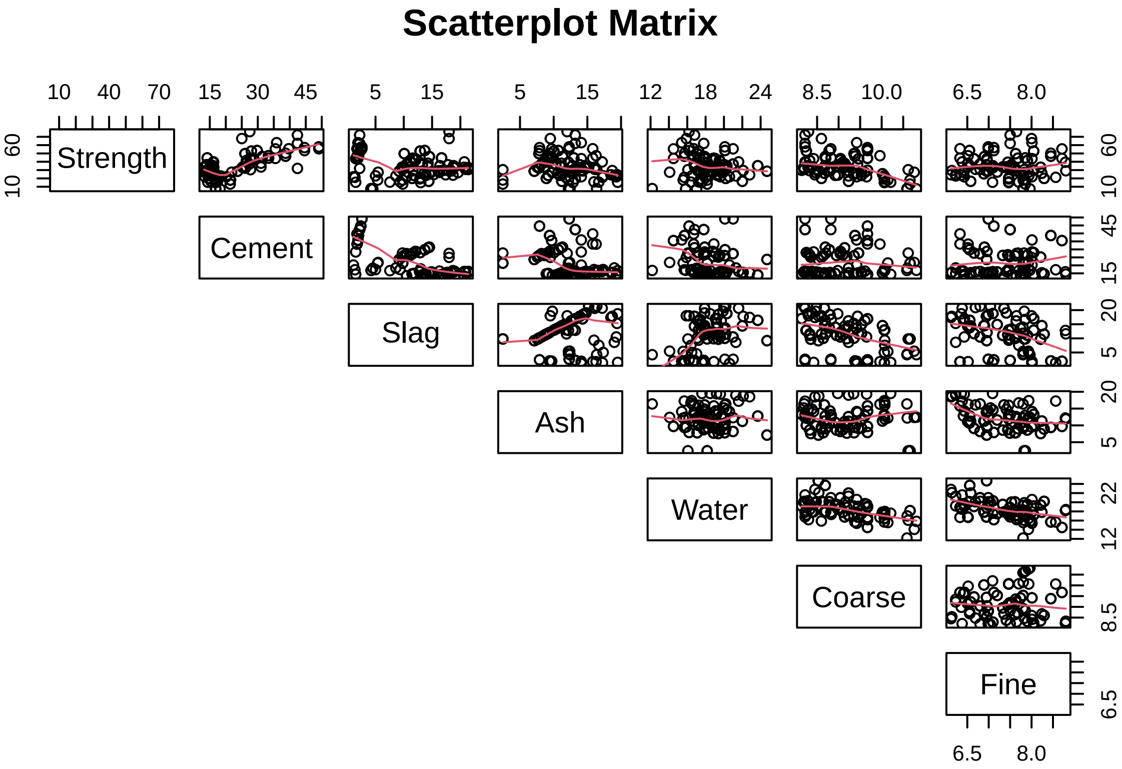


2.3

Depict the scatterplot matrix and comment on the plots depicted on the first row of the resulting matrix. The scatterplot matrix is a matrix where each row and column represent a variable and so each element is a scatterplot of two variable against each other. It allows us to visualize the pairwise relationships between the variables in the dataframe. In order to depict the scatterplot matrix we can use the function *pairs()*. We provide as input the entire dataframe *concrete*. We can also modify some parameters:

- "*upper.panel = panel.smooth*" enables us to draw trend lines for each plot in the upper triangular matrix;
- "*lower.panel = NULL*" avoid printing the lower triangular matrix. This is due to the fact that this matrix is symmetric: e.g. first row, third column depict the relationship between *Strength* and *Slag*, while third row, first column between *Slag* and *Strength*, which is the same mirrored plot. We use this parameter to avoid this redundancy;
- *main* set the title of the plot.

```r
pairs(concrete,
      upper.panel = panel.smooth,
      lower.panel = NULL,
      main = "Scatterplot Matrix")
```



We can comment on these plots in order to have a first idea on the associations between the variables. In particular we are interested in the association between the *response* and the *predictors*, so we focus our attention on the first line, where each covariate is plotted against the response variable.

- **Cement**: We can see that points form a dense group on the left, while are more spread out on the right, making it more difficult to evaluate the actual relationship between the two variables. Overall, except from the beginning where points are concentrated, we can see an increasing trend, suggesting a positive linear relationship (if we do not consider as relevant the first decreasing initial stretch).
- **Slag**: In this case points are divided into two groups making also in this case difficult to evaluate actual relationships. Up to 10 kg/m3 of *Slag* points seems to have a decreasing trend and a slightly increasing one above this value. This suggest that the relationship between the variables might not be linear.
- **Ash**: The plot presents a few points on the left that are well separated from the rest, that maybe represent outliers. Infact except for these points the overall trend seems to be slightly decreasing, suggesting that an increase of the content of *Ash* lead to a decrease of concrete compression *Strength*. Also points are more concentrated at the sides and more spread on the center of the plot.
- **Water**: Points in this plot are more spread in correspondence of low values and more concentrated on high values. Also they form a more dense group in the center of the plot. They overall follow a slightly decreasing trend, suggesting a negative linear relationship between the content of *Water* and *Strength* of the concrete.
- **Coarse**: The overall trend of the points is decreasing, suggesting a negative relationship between the content of **Coarse** and the **Strength** of the concrete, meaning that an increase in its content lead to a decrease in strength. Points are well distributed on the x axis, while on the y axis they seems to have an higher variability on low values and are more concentrated on higher ones. The trend is not perfectly linear, but it's not strong enough to highlight clearly a nonlinearity.
- **Fine**: The overall trend seems to be slightly increasing, but we can see that there are fewer observations on the sides than in the center, where the trend is slightly decreasing. This makes it difficult to evaluate

the actual relationship between the two variables. Also points are more spread out in the center while on the sides are more concentrated. It requires further investigations to better evaluate this, but there could be maybe a non linear association between *Strength* and *Fine*.

2.4

Fit a multiple linear regression model to explain Strength with all the other variables. Report the results using the summary function and comment on the estimated parameters.

In order to evaluate if the other variables are useful to determine the strength of the concrete we can try to build a **multiple linear regression model**. The idea is trying to evaluate association between the covariates and the response and building a model that enables us to make predictions. In our case we want to use all the variables in the dataframe, therefore our model will be as follows:

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 \cdot \text{Cement} + \hat{\beta}_2 \cdot \text{Slag} + \hat{\beta}_3 \cdot \text{Ash} + \hat{\beta}_4 \cdot \text{Water} + \hat{\beta}_5 \cdot \text{Coarse} + \hat{\beta}_6 \cdot \text{Fine}$$

To fit the regression model we use the *lm()* function providing all the data contained in the *concrete* dataframe and then we plot the summary:

```r
mod <- lm(data = concrete)
summary(mod)
```

```
##
## Call:
## lm(data = concrete)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -22.9645  -4.5167  -0.6057   3.3621  21.1820
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 176.4687    62.9286   2.804  0.00594 **
## Cement        0.9941     0.2193   4.533 1.46e-05 ***
## Slag          0.3690     0.3363   1.097  0.27477
## Ash          -0.2661     0.3423  -0.777  0.43860
## Water        -2.4945     0.7270  -3.431  0.00084 ***
## Coarse      -10.1744     2.4669  -4.124 7.13e-05 ***
## Fine         -3.5738     2.5325  -1.411  0.16094
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.091 on 113 degrees of freedom
## Multiple R-squared:  0.7347, Adjusted R-squared:  0.7206
## F-statistic: 52.14 on 6 and 113 DF,  p-value: < 2.2e-16
```

The summary shows a lot of statistics about the model which are very useful for evaluating its goodness.

- The first thing it shows is the **"Call"** section, where it print the formula used to build the regression model. In this case it is implicit in the syntax, in fact here we are using all the covariates to build the regression model using **Strength** as response variable. The model is as built as written above.

- **Residuals**:

- We can see that they have a contained range of variability, with a *minimum* value of -22.97 and a *maximum* value of 21.18; it follows that the *range* is *44.15*, indicating that, for most of the statistical units, the deviation between the observed value in the dataframe and the one predicted by the model is small. In the worst cases, the model underestimate the actual value of 22.97 units and overestimate the actual value of 21.18;

- The *median* is -0.61, which is near to zero, suggesting that the residuals overall distribution is symmetric;

- Also the 1^{st} and the 3^{rd} quartiles are equidistant from the median, which suggest symmetry. We can conclude that, although further investigations are required, the residuals could be distributed according to a gaussian normal distribution.

- **Model Coefficients Estimates**:
  - $\hat{\beta}_0$, which is the *estimate* of the intercept, has an high value of 176.47. The intercept represent the value of the response variable when all the others has a value of 0. In this case it means that, when we have zero content of all the other materials, the concrete has a Compression Strength resistance of 176.47 MPa. Practically the intercept value, as in many other cases, doesn't make sense, since we can't have concrete without anyone of the listed components. The *Standard Error* represents an estimate of the standard deviation of the coefficient β_0. It has a value of 62.93, which is quite high. with that value we could build a Confidence Interval for the estimated value, and we would get quite a wide interval.
    The *test statistic value* of its T test is 2.804. It simply is the ratio between the estimate and its standard error. Since we want the standard error to be small, we want this value to be as high as possible.
    Its corresponding *p-value* is 0.00594. This means that the hypothesis test have the following result: the null hypothesis $H_0 : \beta_0 = 0$ is rejected at a level of significance of 0.001 (as suggested by the two asterisks '**'). This leads us to conclude that there is enough evidence that the intercept in this model is not zero.
  - $\hat{\beta}_1$ is the *estimate* of the coefficient of *Cement*, has an high value of 0.99. It represents the expected increase of the response variable for a unit increase in the Cement variable, while fixing the others. In this case it means that, when we increase by 1 g/kg the content of Cement, we increase of 0.99 MPa the Strength of the Concrete. The *Standard Error* has a value of 0.22, which is quite small, and the *test statistic value* of its T test is 4.533. Its corresponding *p-value* is 1.46e-05. This means that the hypothesis test have the following result: the null hypothesis $H_0 : \beta_1 = 0$ is rejected at any level of significance (as suggested by the three asterisks '***'). This leads us to conclude that there is enough evidence that the intercept in this model is not zero.
  - The same interpretations can be made for $\hat{\beta}_4 = -2.50$ and $\hat{\beta}_5 = -10.17$, that are the estimated coefficient of *Water* and *Coarse*. They both are negative, so we expect a decreasing of 2.5 MPa in *Strength* for each unit increase (each kg/m^3) of *Water* and a decrease of 10.17 MPa for each unit increase (each kg/m^3) of *Coarse*. For both of them we have the rejection of the null hypothesis at any level of significance. In fact for *Water* we have a test statistics value of -3.43 and the corresponding p-value = 0.00084; for *Coarse* we have a test statistics value of -4.124 and the corresponding p-value = 7.13e-05.
  - For the remaining coefficients we have the following estimated values: $\hat{\beta}_2 = 0.37$, $\hat{\beta}_3 = -0.27$ and $\hat{\beta}_6 = -3.57$. They are the estimated coefficient of *Slag*, *Ash* and *Fine*. $\hat{\beta}_2$ is positive, it means and its value means that we expect an increase of 0.37 MPa in *Strength* for each unit increase (each kg/m^3) of *Slag*. $\hat{\beta}_3$ and $\hat{\beta}_6$ instead are negative. It means that we expect a decreasing of 0.27 MPa in *Strength* for each unit increase (each kg/m^3) of *Ash* and a decreasing of 3.57 MPa in *Strength* for each unit increase (each kg/m^3) of *Fine*. By looking at their test statistics values and their corresponding p-values, however, e can see that they cannot be considered significant. In

fact we don't have enough evidence to reject the null hypothesis that the true coefficients of these estimates are equal to 0. In other words the true coefficients $\beta_2, \beta_3, \beta_6$ may are not significantly different from zero, suggesting that the predictors *Slag*, *Ash* and *Fine* may not have a meaningful effect on the response variable in the model. This doesn't necessarily mean that the variables has no effect in reality. It might just indicate a weak relationship or insufficient power to detect an effect.

- **Residual Standard Error (RSE)**: This value, which is $RSE = 7.091$, is a measure of how well the model fits the data. Its degrees of freedom are: 113, which is given by $n - p - 1 = 120 - 6 - 1 = 113$, where $p = 6$ is the number of covariates. 1 is subtracted if the model has an intercept, which is our case. RSE is defined as the square root of the ratio of [the sum of squares of residuals] and [the number of degrees of freedom]:

$$\mathrm{RSE}(y, f_i) = \sqrt{\frac{\sum_{i=1}^{n}(y_i - f_i)^2}{n - p - 1}}$$

It represents the average amount that the observed values deviate from the fitted ones. In our case, on average, the observed values deviate from the fitted regression line by approximately 7.091 units.

- **Multiple R-squared** and **Adjusted R-squared**: Both of them are measures of how well the covariates "explain" the variability of the response.
  - *Multiple R-squared* indicates the proportion of variance in the dependent variable that is explained by the independent variables included in the model. In our case it has a value of 0.7347, which means that 73.47% of the variability is explained by the covariates we are considering. It is quite an high value, which means that the model fits the observed data well;
  - *Adjusted R-squared* has a value of 0.7206. This indicator adjusts the R-squared value with the number of predictors in the model penalizing the addition of unnecessary predictors. Both those values are quite high, meaning that the variability is well explained by the covariates, but we cannot detect whether the model has been correctly specified.
- **F-statistic**: It tests the overall significance of the regression model ad has a value of 52.14. It has two degrees of freedom, which are given by the number of predictors contained in the model (6) the $n - p - 1 = 113$. This value measures how much better the regression model fits the data compared to a model with no predictors. Its p-value is < 2.2e-16, suggesting strong evidence against the null hypothesis (all coefficients in the model are zero). This means that at least one of the predictors in the model has a non-zero coefficient.

2.5

Report and comment on the plots of residuals: scatter plot of residuals alone, against and the fitted values and against each covariate.

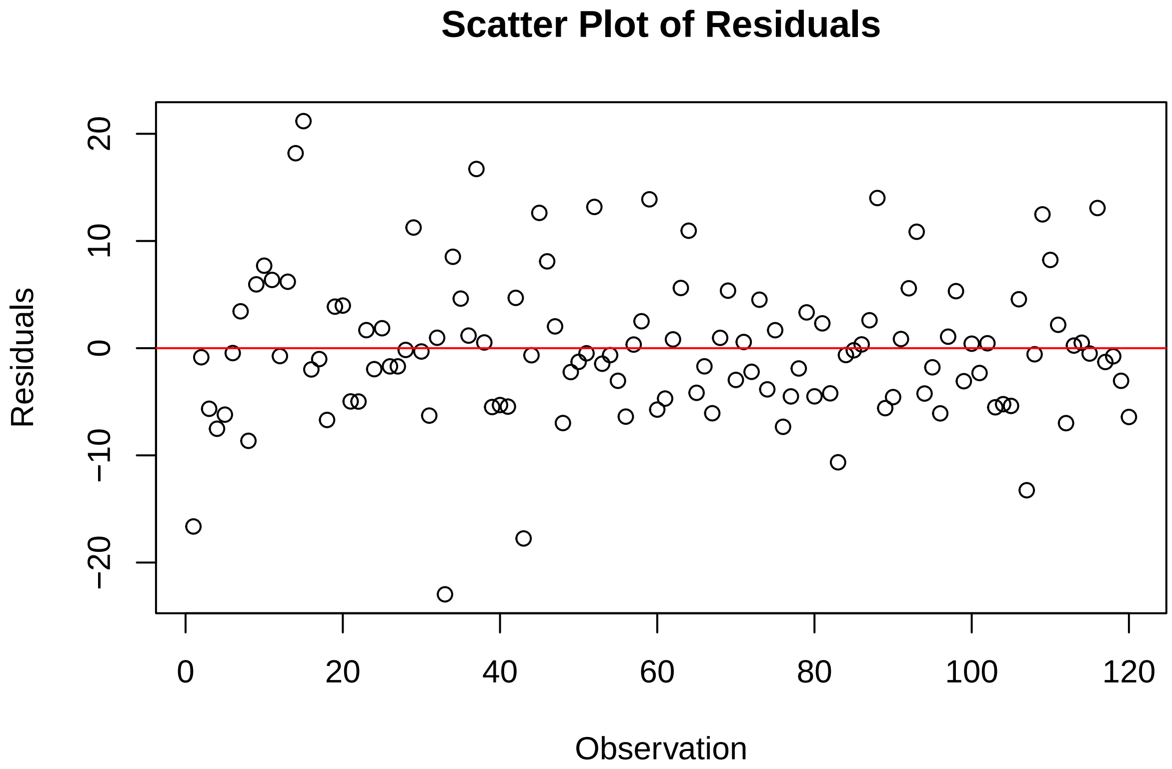
In order to analyze if the residuals are in accordance to the theoretical assumptions we can perform different analysis. First of all we can look at the summary of the model: As we stated before we can see, for example, that they have a median of -0.61, which is really close to zero, in accordance with the assumptions of symmetry. Other that the observation made in the previous section we can depict different plots about residuals in order to further investigate them.

Scatterplot of the residuals

In order to plot the residuals alone we first extract them from the *mod* variable and assign them to a new variable called *residuals*. It will be a numeric list containing the residuals of the model.
After that we can plot them in a scatter plot using the *plot()* function. After that we depict an horizontal

line at the value 0 to evaluate if the residuals are approximately distributed symmetrically around this value.

```
residuals <- residuals(mod)
plot(residuals, main = "Scatter Plot of Residuals", xlab = "Observation", ylab = "Residuals")
abline(h=0, col='red')
```

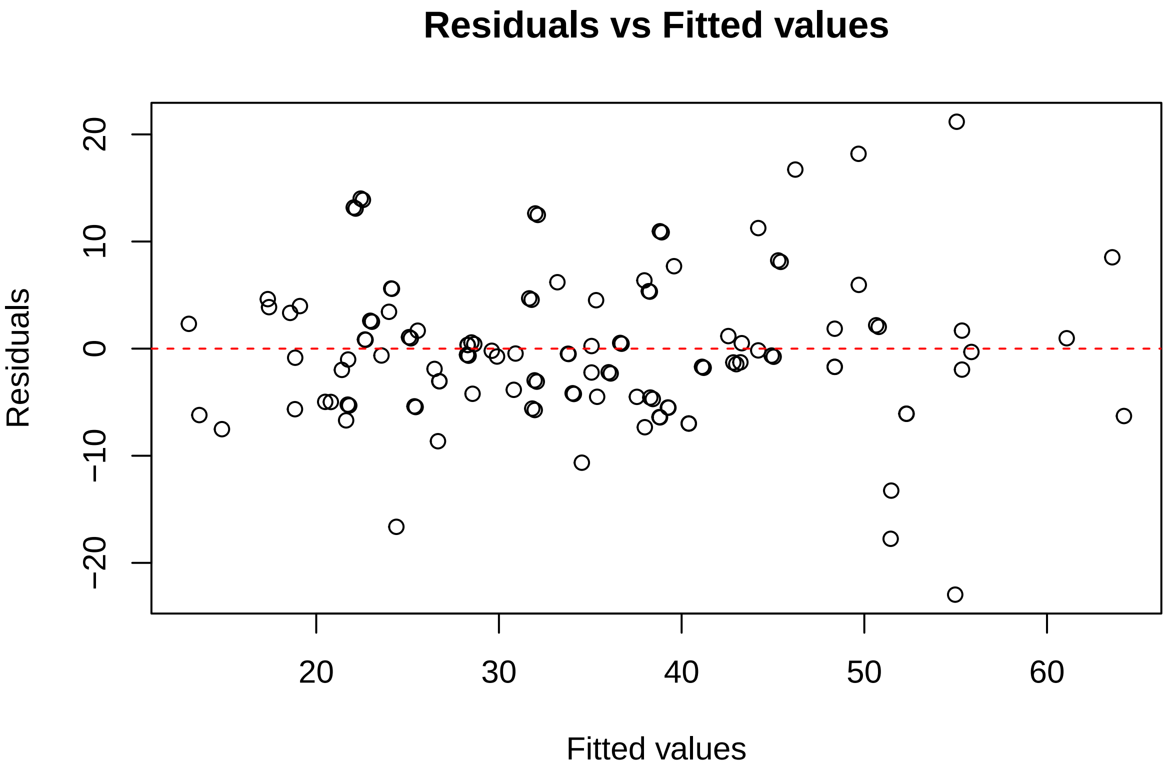


- As we can see all the points are approximately distributed around this line, indicating quite a symmetric distribution. Points, as expected, seem to be distributed randomly and there is no particular concentration of them above or below the : positive and negative residuals seems to be alternating without a particular pattern. - As we previously noticed in the summary of the model the range of variation is between ≈ -21 and $\approx +21$. These values seems to slightly deviate from the rest of the points, that seems be more concentrated around the central line. - Points seem to have the same overall variance along the horizontal axis: they don't appear more concentrated near to zero and more spread as the independent variable value increase. So we can affirm that the hypothesis of constant variance of the residuals (as the units vary) is confirmed.

Residuals vs Fitted

The second plot we can depict is the **Residuals vs Fitted** plot, that depict the residuals values on the y axis against the values fitted by the model on the x axis. To do so we first assign the fitted values of the model to a new variable called *fitted* and then we plot it using the *plot()* function as x values. The *residuals* variable is given as input for the y value of the function. As we did before we plot an horizontal red dashed line at the value 0.

```
fitted <- mod$fitted
plot(fitted,
     residuals,
     main = "Residuals vs Fitted values",
     xlab = "Fitted values",
     ylab = "Residuals")
abline(h=0, col='red', lty=2)
```

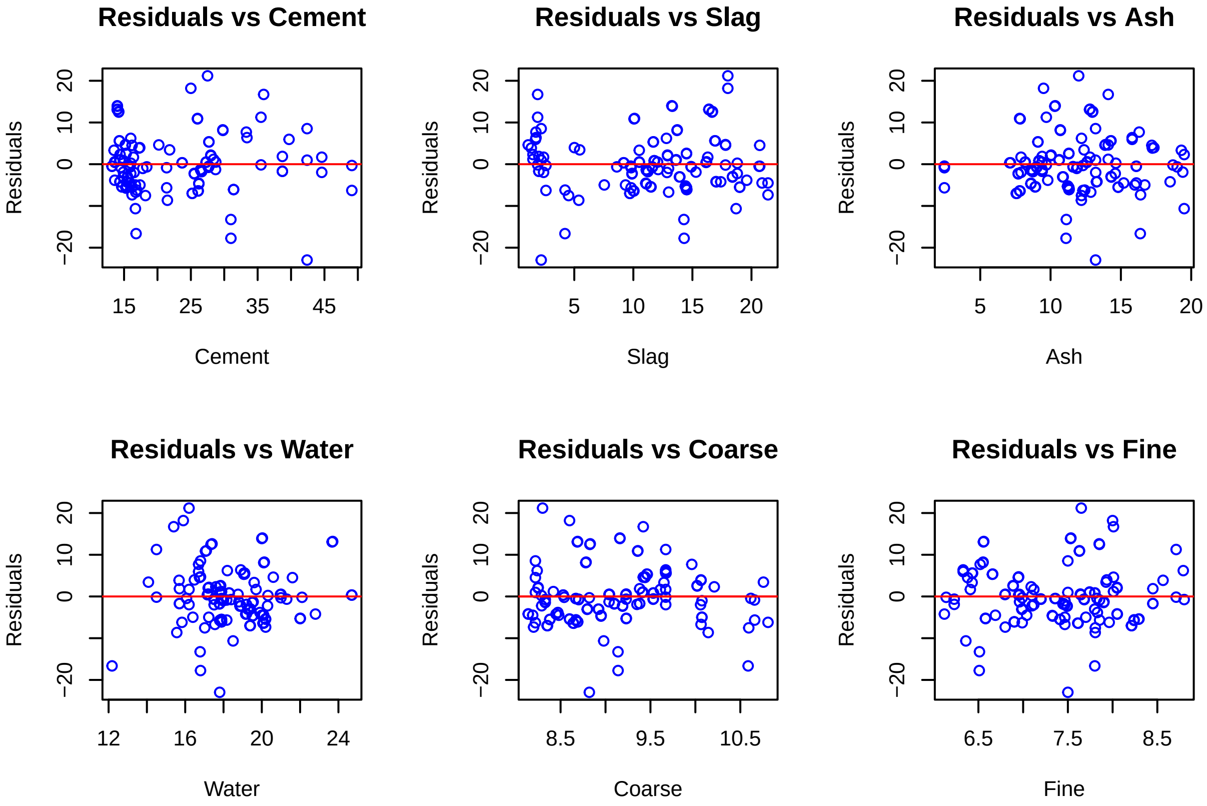


- The first thing we notice is that points seems to be randomly equally distributed above and below the red line, indicating the they have more ore less mean equal to zero, indicating symmetry of the distribution. - This observation about the mean seems to be true even when varying the value of *fitted values*, indicating a constant mean. - If we look at the variance, instead, the assumption of constancy doesn't seem to hold so strictly. In fact points are more concentrated around the origin and more spread out as the value of *fitted values* increase. It suggests that the assumption of homoscedasticity (constant variance) might not hold strictly, violating the theoretical assumption.

Residuals against covariates

We can also depict the residuals values against each covariate: the residuals are plotted on the y-axis, while the covariates are plotted on the x-axis. Each data point represents the residual corresponding to a specific value of the covariate. We do it in the same way for each variable using the function *plot()*. We provide as x-axis data each covariate and as y-axis the residuals. *par(mfrow = c(2,3))* is used to create a multi-panel plotting layout, in this case we will have 2 rows and 3 columns for subsequent plots.

```r
par(mfrow = c(2, 3))

plot(concrete[, 2],
     mod$residuals,
     main = "Residuals vs Cement",
     xlab = "Cement", ylab = "Residuals",
     col = "blue")
abline(h = 0, col = "red")

plot(concrete[, 3],
     mod$residuals,
     main = "Residuals vs Slag",
     xlab = "Slag", ylab = "Residuals",
     col = "blue")
abline(h = 0, col = "red")

plot(concrete[, 4],
     mod$residuals,
     main = "Residuals vs Ash",
     xlab = "Ash", ylab = "Residuals",
     col = "blue")
abline(h = 0, col = "red")

plot(concrete[, 5],
     mod$residuals,
     main = "Residuals vs Water",
     xlab = "Water", ylab = "Residuals",
     col = "blue")
abline(h = 0, col = "red")

plot(concrete[, 6],
     mod$residuals,
     main = "Residuals vs Coarse",
     xlab = "Coarse", ylab = "Residuals",
     col = "blue")
abline(h = 0, col = "red")
abline(lm(residuals ~ concrete$Cement), col = "red")

plot(concrete[, 7],
     mod$residuals,
     main = "Residuals vs Fine",
     xlab = "Fine", ylab = "Residuals",
     col = "blue")
abline(h = 0, col = "red")
```



- **Residuals vs Cement**: data points are more concentrated on the left side of the plot and are not symmetrical in the positive and negative area of the plot. Also points seems to follow a particular pattern, providing evidence against the hypothesis of linear association between the response variable *Strength* and *Cement*. The pattern suggest that maybe higher order term should be introduced in the model.

- **Residuals vs Slag**: data points are not symmetric, are not randomly distributed. They are concentrated in two groups, on lower and higher terms. Also points seems to follow a particular pattern, providing evidence against the hypothesis of linear association between the response variable *Strength* and *Slag*.

- **Residuals vs Ash**: data points seems to be distributed uniformly and symmetrically in the positive and negative area of the plot and don't assume any particular pattern. There are a few points that are detached from the majority, but it is due to the values assumed by the covariate,there are no particular high or low values. This supports the hypothesis of a linear association between the response variable *Strenght* and *Ash*.

- **Residuals vs Water**: data points seems to be distributed uniformly and symmetrically in the positive and negative area of the plot and don't assume any particular pattern. Also there are no particular high or low values. This supports the hypothesis of a linear association between the response variable *Strength* and *Water*.

- **Residuals vs Coarse**: data points are not symmetrically distributed on the positive and negative area of the plot. They are more spread out in correspondence of low values and more concentrated on the negative side. In the middle of the plot are conversely more concentrated on the positive side and back on the negative for high values of the covariate. This pattern suggest evidence against the hypothesis of linear correlation between the response *Strength* and the covariate *Coarse*.

- **Residuals vs Fine**: Also in this case points are not randomly distributed around positive and negative values, but follows a particular trend suggesting that an higher order term should be introduced in the model.