

Bivariate Descriptive Statistics

Bachelor in Computer Science and Engineering

2020/21

1. Introduction

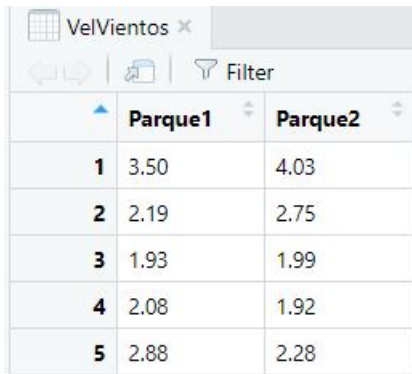
In this guide we are going to analyze two variables observed simultaneously by using R. We are going to study their linear dependence and to find the regression line that will help us to predict one variable as a function of the other one.

The data consists of measures of the wind speeds obtained by two anemometers arranged in two wind parks close to each other. The `VelVientos730.xlsx` file have the records of 730 hours, and for every hour it contains the wind speeds measured in both parks. The speeds, measured in meters per seconds (m/s), of each park are contained in the variables `Parque1` and `Parque2`, respectively.

We want to have a computer system that records the wind speeds in these parks in time real. This information is very important to manage the energy production in the park and also to detect malfunctions of the turbines. The computer system that is going to be installed is very expensive, requiring a network where some links use microwave radio to communicate, periodic calibrations and staff and procedures that monitor the transmissions of data. For this reason we decide to install this facility only for `Parque1`. The ultimate goal we want to reach by this data analysis is to use wind measurements at `Parque1` to make predictions of the wind speed at `Parque2` by means of a regression line and in such a way we save doubling the cost of the system.

First we read and view the data file. The figure shows the first five observations of this datafile. Note that the line `View(VelVientos)` appears as a comment, to execute it, simply delete the symbol `#`.

```
library(readxl)
VelVientos <- read_excel("VelVientos730.xlsx")
#View(VelVientos)
```



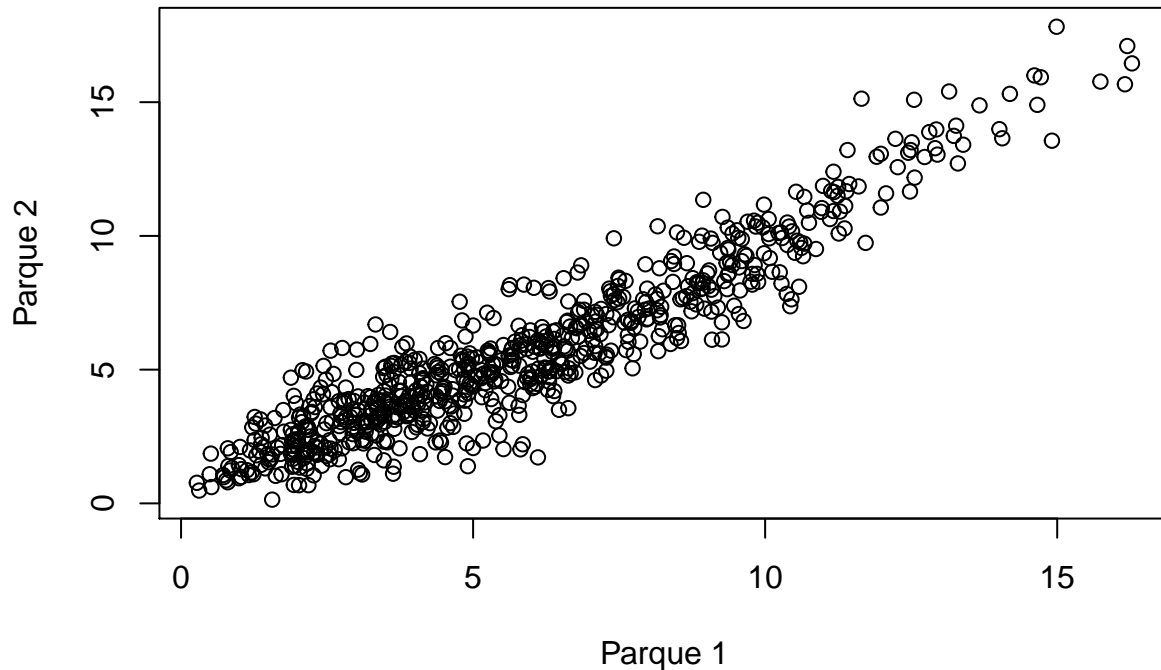
	Parque1	Parque2
1	3.50	4.03
2	2.19	2.75
3	1.93	1.99
4	2.08	1.92
5	2.88	2.28

2. Graphical Analysis

We will obtain the scatter plot of these two variables using the function `plot`. Since our goal is to use the `Parque1` as input variable and `Parque2` as output variable, we will use `Parque1` as x and `Parque2` as y . In

general, if the purpose is generating the graph, this distinction is arbitrary. The resulting graph is

```
plot(VelVientos$Parque1, VelVientos$Parque2, xlab = "Parque 1", ylab = "Parque 2")
```



where we can see that the relationship between the two variables is linear and strong. It thus seems reasonable to use the regression line to predict y as function of x . The fact that the distributions of both variables look similar helps to make the prediction more accurate.

3. Bivariate Characteristic Measures

To compute the characteristic measures that summarize this linear relation, we can use the following instructions:

```
cov(VelVientos$Parque1, VelVientos$Parque2)
```

```
## [1] 9.841527
```

```
cov(VelVientos)
```

```
##           Parque1  Parque2
## Parque1 10.505743  9.841527
## Parque2  9.841527 10.594773
```

The first line provides the covariance between the two variables and the second the covariance matrix. Using the information contained in this matrix we could get the correlation coefficient and the terms determining the regression line. For example, the correlation coefficient of the two variables is given by

$$\text{corr}(x, y) = \frac{\text{cov}(x, y)}{s_x s_y} = \frac{9.841527}{\sqrt{10.505743} \sqrt{10.594773}} = 0.9328317.$$

This correlation coincides with the result computed by R:

```
cor(VelVientos$Parque1, VelVientos$Parque2)
```

```
## [1] 0.9328317
```

```
cor(VelVientos)
```

```
##           Parque1    Parque2
## Parque1 1.0000000 0.9328317
## Parque2 0.9328317 1.0000000
```

A scatter plot so linear and a correlation coefficient so high imply that the regression line will be accurate in making predictions.

4. Regression Line

To compute the regression line, called simple regression as well (this is due to the fact that we have only one independent variable), we can use

```
RegressionModel <- lm(Parque2 ~ Parque1, data=VelVientos)
print(RegressionModel)
```

```
##
## Call:
## lm(formula = Parque2 ~ Parque1, data = VelVientos)
##
## Coefficients:
## (Intercept)      Parque1
##      0.1979         0.9368
```

The relation that we want to compute is the Least-Square line

$$\hat{y}_i = a + bx_i,$$

where $b = \frac{\text{cov}(x, y)}{s_x^2}$ and $a = \bar{y} - b\bar{x}$.

The values of a and b that the previous instructions computes are $a = 0.1979$ and $b = 0.9368$.

The following instruction gives a detailed summary of the obtained regression model

```
summary(RegressionModel)
```

```
##
## Call:
## lm(formula = Parque2 ~ Parque1, data = VelVientos)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.2016 -0.6728 -0.0155  0.6701  4.0187
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.19787    0.08911   2.221  0.0267 *
```

```
## Parque1      0.93678    0.01341  69.854   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.174 on 728 degrees of freedom
## Multiple R-squared:  0.8702, Adjusted R-squared:  0.87
## F-statistic: 4880 on 1 and 728 DF, p-value: < 2.2e-16
```

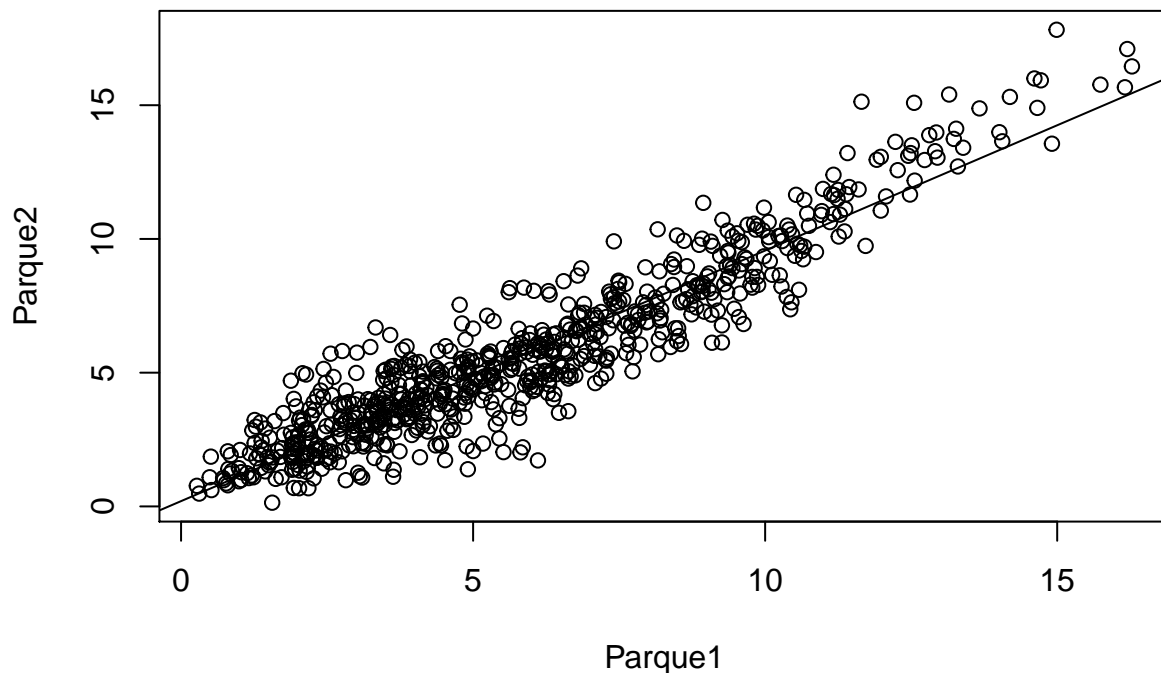
However, we are interested only to the values corresponding to the column **Estimate**. The parameter b correspond to the coefficient associated to variable **Parque1**, that is the slope of the regression line. The parameter a is the **Intercept** coefficient, that is the point of intersection of the line with the y -axis ($x = 0$).

Our regression line is finally given by

$$\text{Wind speed at Parque2} = 0.19787 + 0.93678 \times \text{Wind speed at Parque1}.$$

The regression line can be plotted by using

```
plot(VelVientos)
abline(RegressionModel)
```



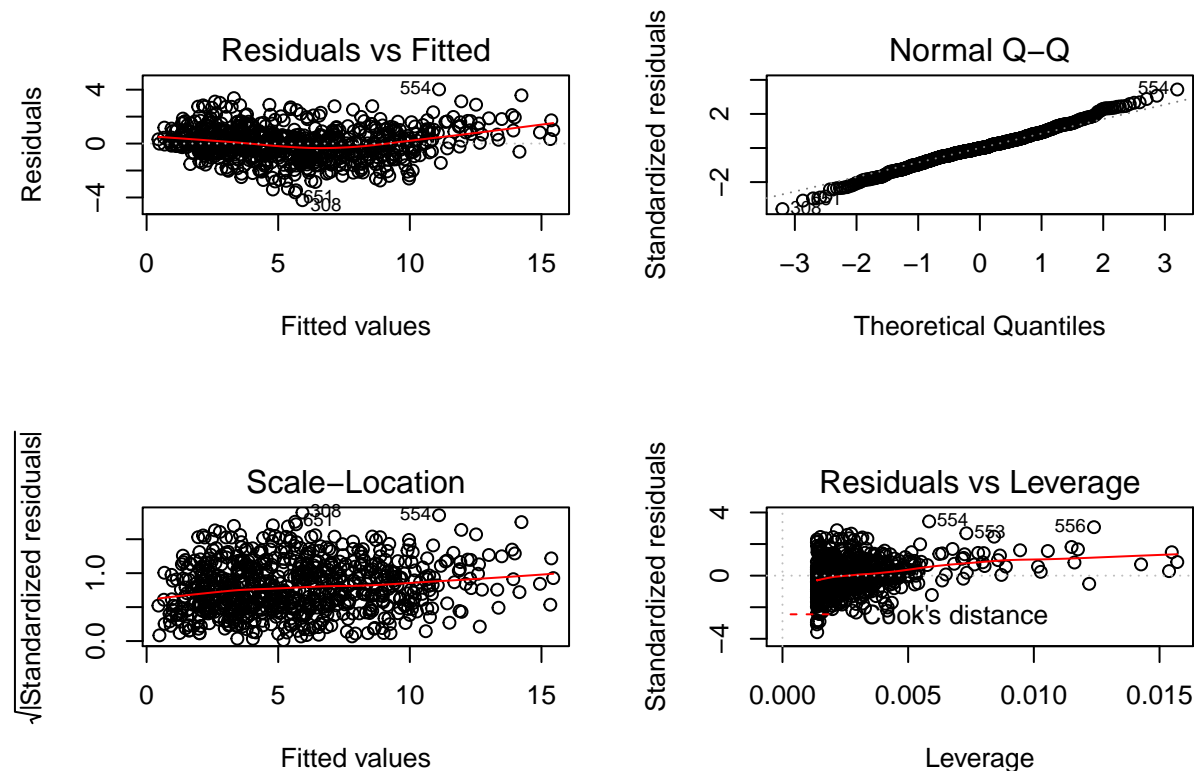
The regression line together with the cloud of points shows that the accuracy of the prediction is good for almost all values of the velocities. For higher speeds in **Parque1** however the regression line underestimates the values of the wind speeds in **Parque2**. By using more advanced concepts of the regression technique we could improve the current accuracy of the predictions in this area of the values, but at the moment this concepts are beyond the scope of this document.

The determination coefficient, R^2 , displayed at the **summary** output is $R^2 = 0.8702$. Therefore if we assume

that the linear relation between the two variables is acceptable we can say that the wind speed in **Parque2** can be expressed for an 87% of its value as a function of the wind speed in **Parque1**. That is we have a good predictor. However, the linearity diagnosis of the model shows that the linear model is not entirely adequate.

To do a diagnosis of linearity for this model we show the graph of residuals vs. predictions, which is available in

```
par(mfrow=c(2,2))
plot(RegressionModel)
```



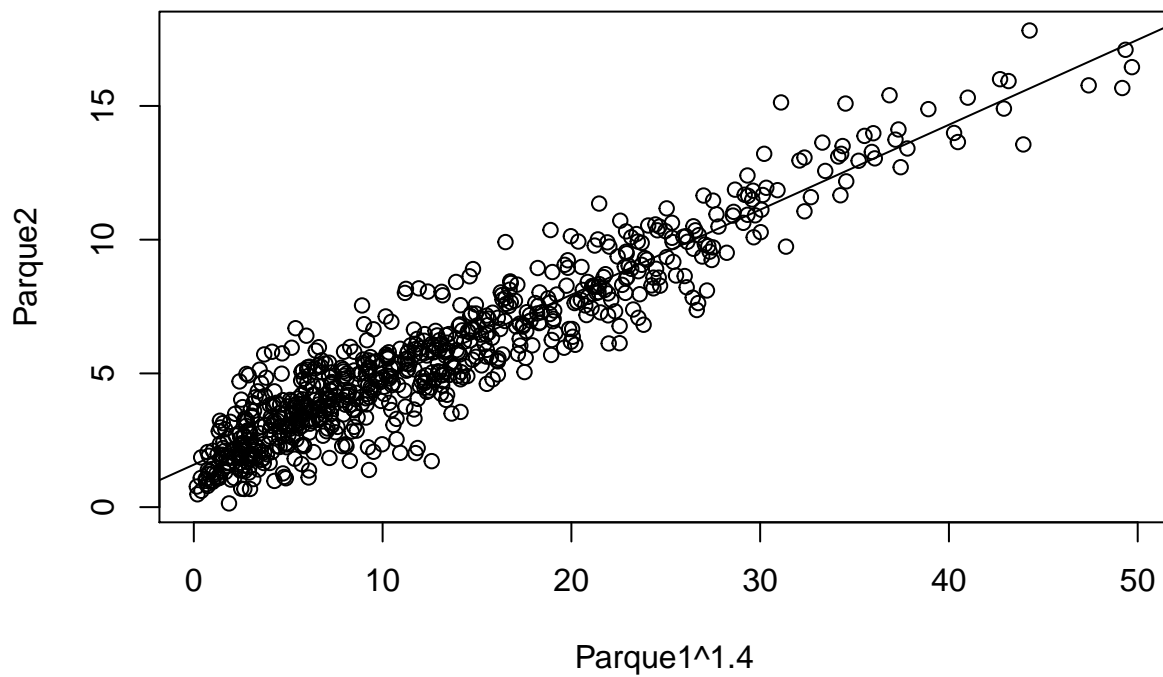
The above residuals versus fitted values graph that we get is not very convincing, since as was noticed before, it shows that for high values the predictions loose accuracy. The graph shows indeed a curvature that The regression line underestimates the wind speeds in **Parque2** in this region evidences a non-linearity in the data. This lack of linearity can be appreciated also in the scatterplot with the regression line drawn on top of it.

The non linearity contained in the data can be corrected by using transformations of the power type, x^c , with $c > 1$. In that way we could expand the data on the x-axis in such a way to shift more the data with higher value and correcting the individuated curvature. Next picture shows the results of the transformation $x^{1.4}$ that seems to solve the problem.

```
VelVientos$Parque1power <- VelVientos$Parque1^1.4
RegressionPowerModel <- lm(Parque2 ~ Parque1power, data=VelVientos)
print(RegressionPowerModel)
```

```
##
## Call:
## lm(formula = Parque2 ~ Parque1power, data = VelVientos)
##
## Coefficients:
```

```
## (Intercept) Parque1power
##      1.5875      0.3177
plot(VelVientos$Parque1power, VelVientos$Parque2, xlab = "Parque1^1.4", ylab = "Parque2")
abline(RegressionPowerModel)
```



The final model is

$$\text{Wind speed at Parque2} = 1.5875 + 0.3177 \times \text{Wind speed at Parque1}^{1.4}.$$

Its diagnosis reveals a better adjustment than the previous linear model.

```
par(mfrow=c(2,2))
plot(RegressionPowerModel)
```

