# Madrid's air pollution analysis

*Alejandro Basco Plaza, Miriam Zaragoza Pastor, Jonatan Ruedas Mora, Raúl Cruz Benita and Laura Sánchez de Rojas Huerta.*

*20/12/2019*

## Abstract

Throughout the last years people has become more and more concern about the advantages and drawbacks of living in a polluted city. This report analyzes the evolution of the air quality of Madrid city from two different perspectives. The first one is a time series that shows the evolution of the ozone pollutant from 2012 to 2017 and tries to forecast the evolution of this pollutant. The second one is used to predicts how these values are going to evolve in the following year using machine learning techniques such as the K nearest neighbors. Thanks to theese predictions we will be able to discover which district of Madrid will have the less polluted air and as a consequence we will have the perfect candidate to answer the question and solve the problem of moving there there to live.

## Introduction and related work

Nowadays, the pollution levels in Madrid is a popular topic that generates a heated debate in the news (click here for more information about the high pollution levels that Madrid's air is facing off against). This has worried some of the group members who are thinking about moving to Madrid from the outskirts. That situation lead us to think about which area of Madrid is the most suitable for us to buy an apartment.

We have found some related works that works with the same data:

Madrid Air Quality Exhaustive Tutorial on Stationarity, Smoothing, and Seasonality. This projects is an introduction to time series while using Madrid's pollution datasets.

Explore air quality in Madrid. It wants to explore these datasets because of the measures taken by the authorities against the use of cars in the center of Madrid.

Madrid's pollution weekly mean values. You can see an interactive chart about the levels of pollution depending on the year that you want to see.

Airquality analysis. It is a project where it is plotted information about the most important pollutants of the datasets.

Madrid's Air Quality with ARIMA Forecasting. Worried about the short term and long term effects on health, the creator of this project wants to analyze the air quality and predict it in for the next two years.

## Document structure and R scripts organization

To begin with, an **exploratory data analysis** is made, where our data is described before performing any further analysis. Then we explain which features are the most relevant ones for the issues that we concern about, and at last we explain the wrangling carried out to deal with our data. After this previous step, we go through several visuals that show how our data behaves in terms of center, dispersion and distribution measures. The exploratory analysis is ended with some previous insights about the temporal evolution of our data, before starting with further analysis.

Secondly, a **predictive analysis first step** is performed, in which we have used **time series** to predict the behavior of one of the features in our data in the year 2018, using the ETS (Error Trend Seasonal) predictive model.

Apart from that, we have conducted a prediction method that uses two **machine learning approaches**, KNN (K nearest neighbors) and linear regression, to generate and compare two prediction models for one of

our data features, and chooses the best one according with the RMSE obtained. The process to generate the two models is described, and a prediction for year 2019 is ellaborated (year for which we don't have any available data), drawing some conclussions to assess our question of interest.

Finally, we present some **conclussions and future work** that could be carried out to follow this research.

We provide the following **scripts** that have been used to perform the different analysis and to generate the plots:

- **DataAdaptation.R**: script in which the missing value treatment is performed.
- **Exploratory_analysis.R**: script used to perform the exploratory data analysis.
- **Time_Series.R**: script in which the time series analysis is performed.
- **Machine_Learning.R**: script in which the machine learning approach is performed.

**Note: The project must be under the home directory because the path of the scripts is relative to that directory**

## Exploratory data analysis

### Dataset description

Before exposing the findings during the exploratory data analysis, important information and background about the dataset is introduced below:

The dataset used for this work was obtained from Kaggle's website. In this page, we can find a lot of information and datasets about the air quality in Madrid.

Our data is composed by **18 data sets** that contain observations of the level of presence of **various gases** in Madrid's air, **which can be noxious for human health**, during the last 18 years (from 2001 to 2018). This information is recorded by 24 different stations, each one located in a different area of the city, that measure the level of presence of each pollutant through several sensors. There is data about the presence of each gas during each hour of the day, every day of every month.

The data in this data set has been collected from the original files provided by Madrid Open Data. Decide soluciones organization processed them and uploaded to the Kaggle's website.

Regarding the **observations**, all csv files include data for every month, except the last one (2018) that only has records until May. According to this, we have the following number of rows for each data set (note that the stations.csv stores information about the 24 stations through which the information about the pollutants was measured):

| CSV | Observations |
|---|---|
| Stations.csv | 24 rows |
| Madrid2001.csv | 217872 rows |
| Madrid2002.csv | 217296 rows |
| Madrid2003.csv | 243984 rows |
| Madrid2004.csv | 245496 rows |
| Madrid2005.csv | 237000 rows |
| Madrid2006.csv | 230568 rows |
| Madrid2007.csv | 225120 rows |
| Madrid2008.csv | 226392 rows |
| Madrid2009.csv | 215688 rows |
| Madrid2010.csv | 209448 rows |
| Madrid2011.csv | 209928 rows |
| Madrid2012.csv | 210720 rows |
| Madrid2013.csv | 209880 rows |

| CSV | Observations |
|---|---|
| Madrid2014.csv | 210024 rows |
| Madrid2015.csv | 210096 rows |
| Madrid2016.csv | 209496 rows |
| Madrid2017.csv | 210120 rows |
| Madrid2018.csv | 69096 rows |

We can also differentiate two domains whose columns represent different data:

On the one hand, as it was mentioned above, we have the Stations.csv with six columns, which contain information about the stations:

- **id**: unique identifier for each station.
- **address**: location of the station in Madrid city.
- **Three additional columns** that stand for the exact location coordinates of the station.

On the other hand, we have the data sets named Madrid20xx.csv, which contain the observations about Madrid's air pollution itself. Some of these csv files have 14 columns, others 16 and others 17. These columns include information about the level of presence of each pollutant in Madrid's air, the station that measured the information, and the date (year, month, day, hour, minute and second) in which this information was obtained by the station.

We have the following number columns in the csv files:

| CSV | Features |
|---|---|
| Stations.csv | 6 columns |
| Madrid2001.csv | 16 columns |
| Madrid2002.csv | 16 columns |
| Madrid2003.csv | 16 columns |
| Madrid2004.csv | 17 columns |
| Madrid2005.csv | 17 columns |
| Madrid2006.csv | 17 columns |
| Madrid2007.csv | 17 columns |
| Madrid2008.csv | 17 columns |
| Madrid2009.csv | 17 columns |
| Madrid2010.csv | 17 columns |
| Madrid2011.csv | 14 columns |
| Madrid2012.csv | 14 columns |
| Madrid2013.csv | 14 columns |
| Madrid2014.csv | 14 columns |
| Madrid2015.csv | 14 columns |
| Madrid2016.csv | 14 columns |
| Madrid2017.csv | 16 columns |
| Madrid2018.csv | 16 columns |

### Data wrangling

Before getting to work with the previously described data, some steps were performed in order to make the data more manageable:

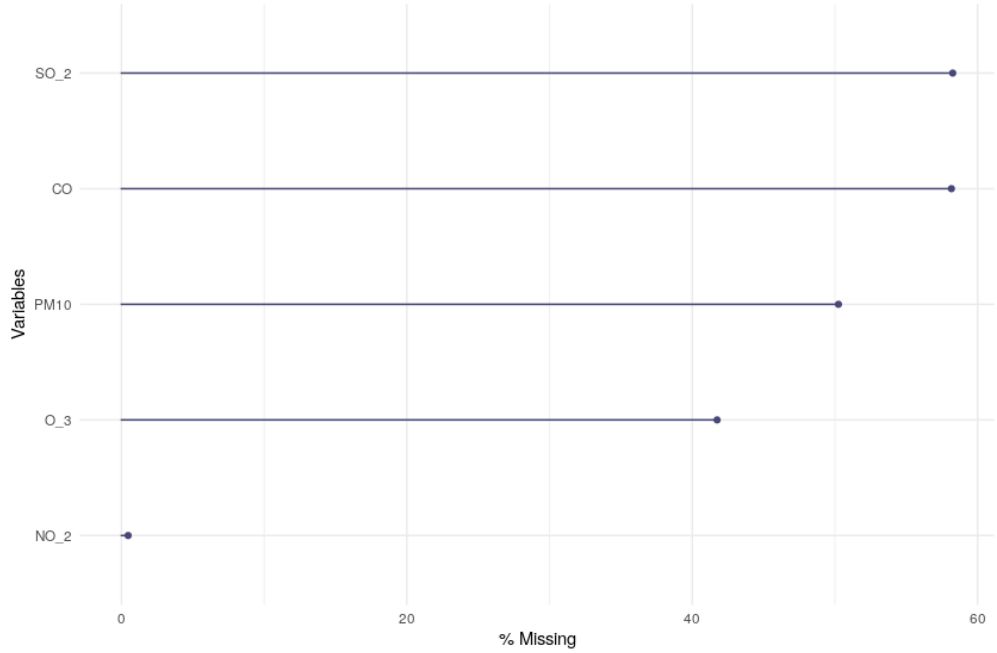
**Selected Features** As it was mentioned before, the data sets that record information about pollution in Madrid's air provide data about several gases that can be harmful for human health. To carry on the upcoming analysis, we are concerned about the most harmful ones: CO, O3, SO2, PM10 and NO2.

As all the previous links recall (the reader can click on the names of the pollutants mentioned above for more information on the environmental and health effects they cause), all of these 5 pollutants can cause different types of pain in humans, such as chest pain, and can hurt vital organs, like lungs. Some of them are responsible for environmental phenomenons like acid rain, that can harm sensitive ecosystems very easily, or haze (also known as reduced visibility). Some of them also contribute to contaminate lakes and other big water resources, changing their acidity.

After these insights, it is clear that when choosing a new place to live, it is very convenient to take a look at what environmental factors can harm our health, and these five pollutants are very important ones. Therefore in our upcoming analysis we will be focusing specifically on the behavior of these pollutants.

**Datasets unification** In order to work with the data, all the yearly datasets were unified in a single one, adding one additional feature to the existing date column that all of the datasets present: the year. Thus, after this transformation, a single dataset contained all the information about Madrid's air pollution in the last 17 years, instead of 17 different datasets (one per year). Two additional columns were added: the *only_month* and *only_year* columns, wich stand for the month and the year in which the observation was recorded, respectively.

**Missing values treatment** As we can observe in the plot below, there is a **high amount of missing values** in the variables that we concern about:




Thus, in order to deal with missing values, the **mean of each one of the variables for each year, month and station is computed**, so that all the missing values for each one of the pollutants in that same year, month and station are imputed with the corresponding mean. In case that one station has missing values all the month, they are replaced with the **mean of the column in that same year and month from all the stations**. We have developed a separate script to **regenerate all the csv files following this treatment to erase missing values**, and then employed these treated csvs to perform the rest of the analysis. The ultimate goal of this approach is **to reduce the time it takes to execute this script**: it would be very time consuming to execute it every time we need the data.

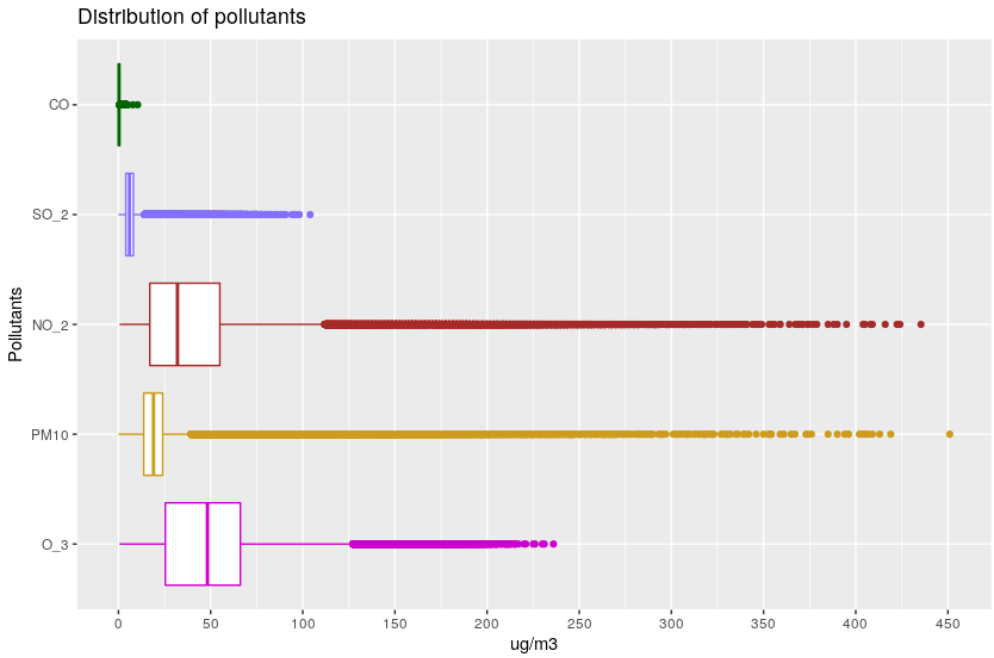
### Descriptive statistics analysis

After wrangling our data as it was necessary, we are ready to take a first look at our variables.

To start exploring our data, the distribution of the variables in our dataset in terms of **center, dispersion and distribution descriptive measures**, is analysed below.

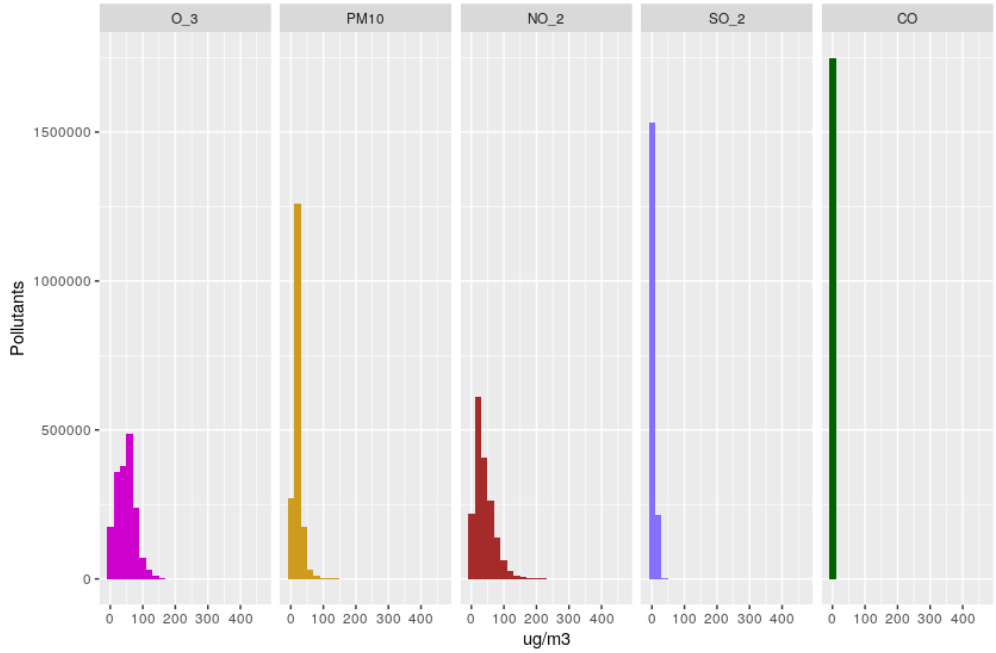
We will end this section providing the reader some **insights on the temporal evolution of our data**, what historical patterns can be extracted and how they may relate to other real world phenomena.

**Center, dispersion and distribution measures** The plot below is useful to visualize the center measures of the five selected variables:




As we can see, variables NO2, O3 and PMH10 spread along wide ranges, while variables SO2 and CO spread along narrower ranges. A characteristic that they all share is that all of them have many outliers or highly unexpected values.

The histograms below show a better insight of the shape of the distribution of the variables: as we can see, all of them are highly skewed to the right, so they don't follow a normal distribution.



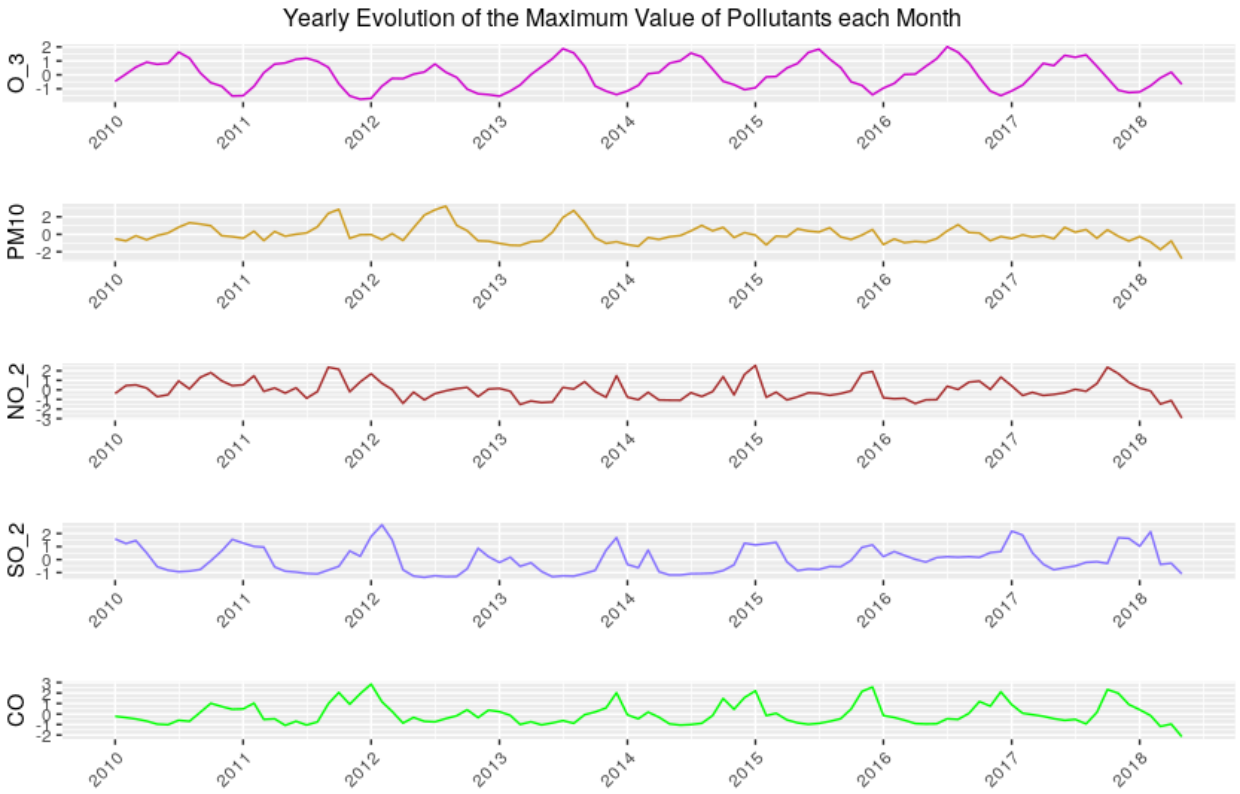

In the plot above, we can also observe a phenomenon that leads us to the conclussion that, depending on what we want to do with the data in upcoming sections, it may be a good approach to normalize it: variables SO2 and CO always take values that are very close to 0, while the other variables can reach much higher values.

### Data insights

It is interesting to answer some questions related to our data, in order to acquire some previous knowledge on how pollution behaves in Madrid's air, taking into account factors like year month and hour in the day, before procceeding to further steps in the project.

It is important to highlight that the questions that we intend to answer in this section are merely exploratory. We are still not trying to predict any phenomenon, but it may be interesting to **find out some patterns in the temporal evolution of the five pollutants**.

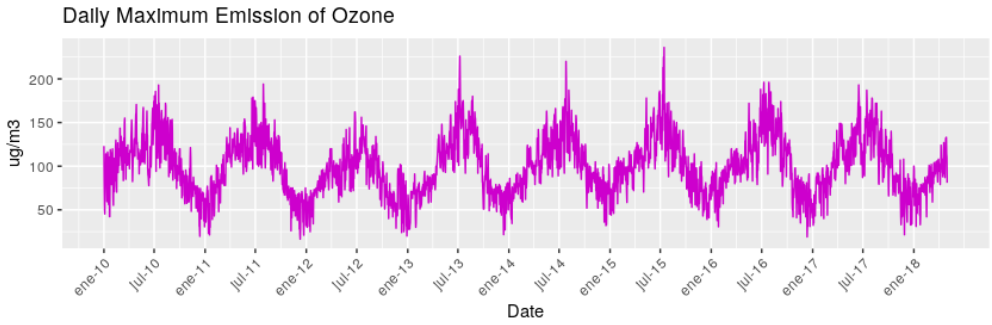
At first, it is interesting for us to observe the **yearly evolution of the maximum value that each pollutant reaches each month**. In order to do so, first the maximum value for each day on each month and year is computed. Then, the average value for each month per year is obtained. Note that the values have been normalized to be on the same scale. The resulting plot, looks as follows:

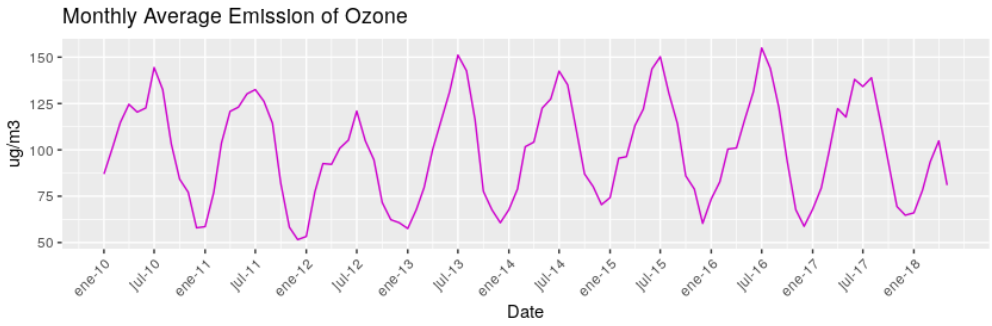


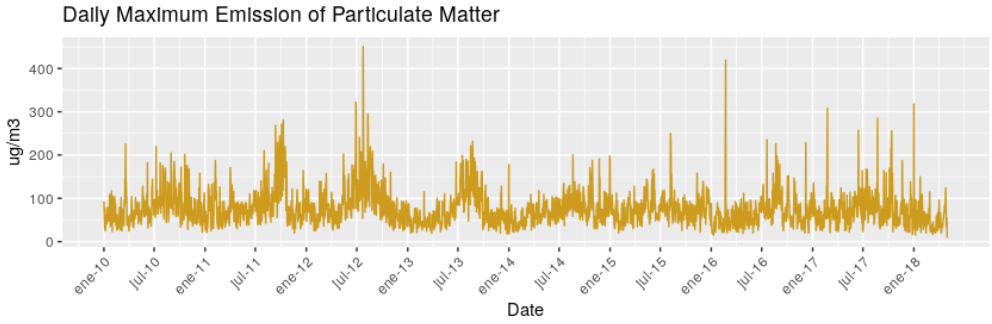

From the plot above we can see some curious information:

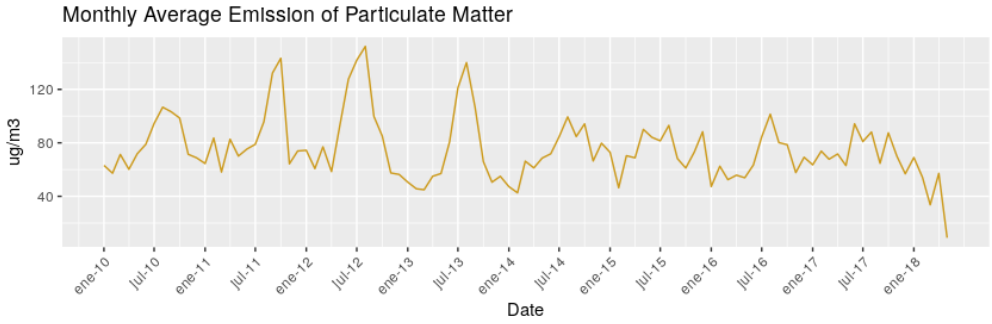
- the clear patterns followed by **O3** and **SO2** are the opposite: while the levels of O3 reach their yearly maximum during the months in the middle of the year (summer months) and decrease towards the end of the year, we observe the opposite trend for SO2, that reaches its yearly minimum values towards the middle of the year, and its maximum towards the end.
- **PM10** levels have reached a more regular trend since year 2014. The trend of this variable is very irregular. This also happens with variable **NO2**, which has an irregular trend aswell, but this variable seems to reach the yearly maximum value towards the end of the year, even though the maximum value can change a lot from one year to another.
- The temporal evolution of variable **CO** is more irregular than the one for SO2 and O3, but a pattern is also appreciated: the yearly maximum value is reached mostly towards the end of the current year and the beginning of the next one, in the coolest months.

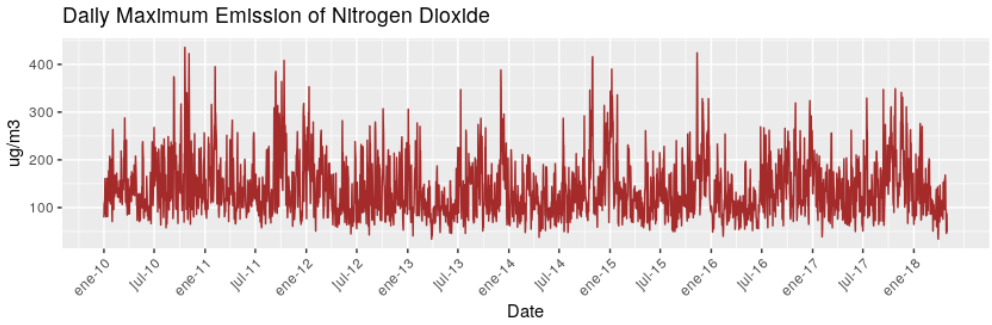
To support these findings, we provide the following set of plots:

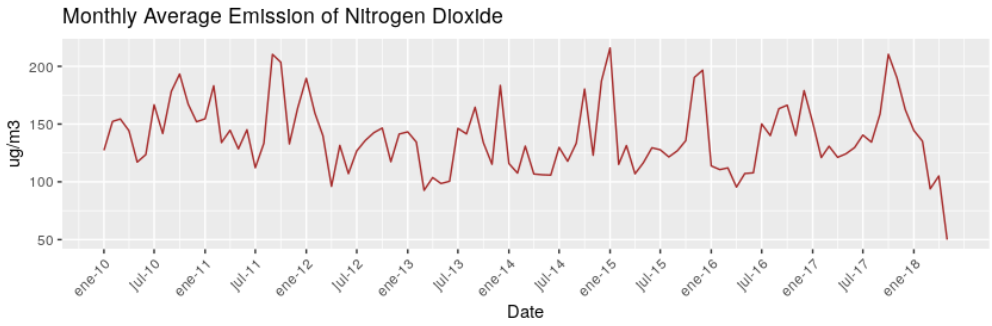

Dally Maximum Emission of Ozone
ug/m3
200
150
100
50
ene-10
jul-10
ene-11
jul-11
ene-12
jul-12
ene-13
jul-13
ene-14
jul-14
ene-15
jul-15
ene-16
jul-16
ene-17
jul-17
ene-18
Date

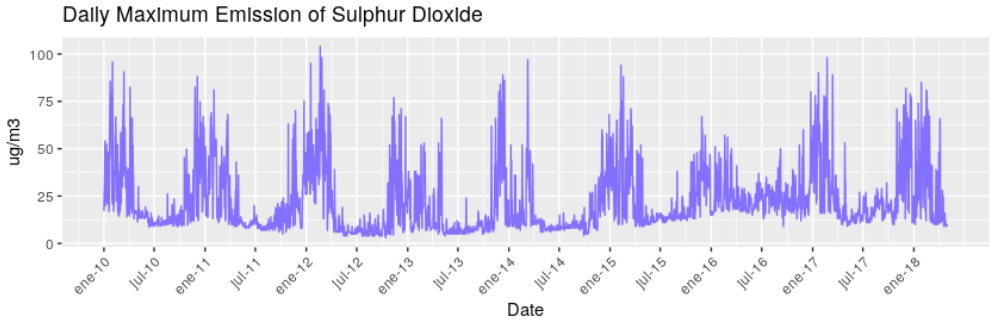

Monthly Average Emission of Ozone
ug/m3
150
125
100
75
50
ene-10
jul-10
ene-11
jul-11
ene-12
jul-12
ene-13
jul-13
ene-14
jul-14
ene-15
jul-15
ene-16
jul-16
ene-17
jul-17
ene-18
Date

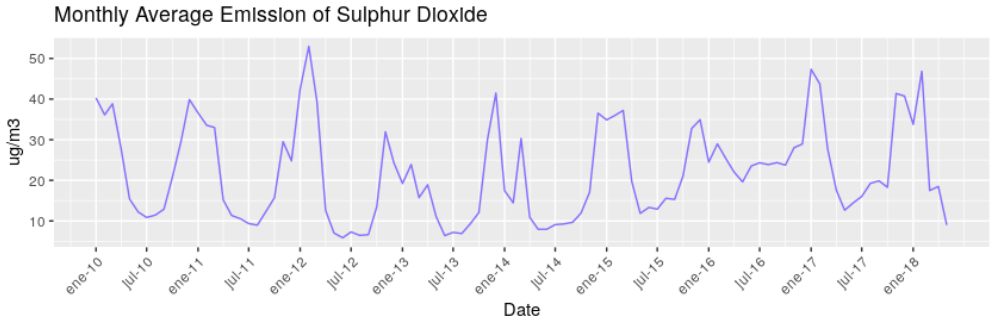

Dally Maximum Emission of Particulate Matter
ug/m3
400
300
200
100
0
ene-10
jul-10
ene-11
jul-11
ene-12
jul-12
ene-13
jul-13
ene-14
jul-14
ene-15
jul-15
ene-16
jul-16
ene-17
jul-17
ene-18
Date

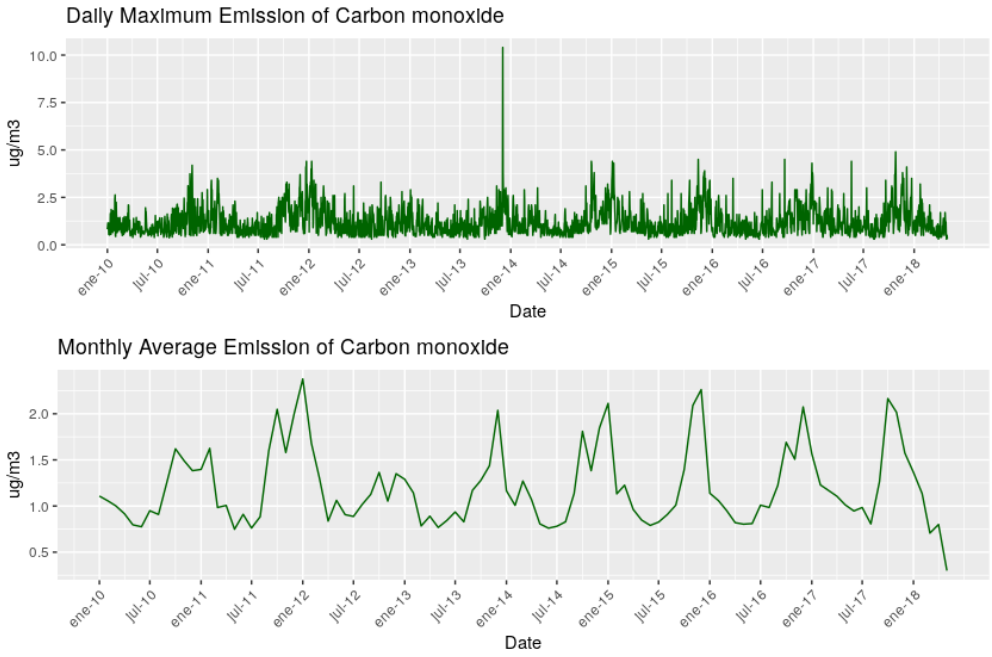

Monthly Average Emission of Particulate Matter
ug/m3
120
80
40
ene-10
jul-10
ene-11
jul-11
ene-12
jul-12
ene-13
jul-13
ene-14
jul-14
ene-15
jul-15
ene-16
jul-16
ene-17
jul-17
ene-18
Date


Daily Maximum Emission of Nitrogen Dioxide
ug/m3
400
300
200
100
ene-10
jul-10
ene-11
jul-11
ene-12
jul-12
ene-13
jul-13
ene-14
jul-14
ene-15
jul-15
ene-16
jul-16
ene-17
jul-17
ene-18
Date


Monthly Average Emission of Nitrogen Dioxide
ug/m3
200
150
100
50
ene-10
jul-10
ene-11
jul-11
ene-12
jul-12
ene-13
jul-13
ene-14
jul-14
ene-15
jul-15
ene-16
jul-16
ene-17
jul-17
ene-18
Date


Daily Maximum Emission of Sulphur Dioxide
ug/m3
100
75
50
25
0
ene-10
jul-10
ene-11
jul-11
ene-12
jul-12
ene-13
jul-13
ene-14
jul-14
ene-15
jul-15
ene-16
jul-16
ene-17
jul-17
ene-18
Date


Monthly Average Emission of Sulphur Dioxide
ug/m3
50
40
30
20
10
ene-10
jul-10
ene-11
jul-11
ene-12
jul-12
ene-13
jul-13
ene-14
jul-14
ene-15
jul-15
ene-16
jul-16
ene-17
jul-17
ene-18
Date




For **O3** and **SO2**, the previously observed patterns are maintained: the month of the year in which O3 reaches its maximum level is july, while the month of the year in which it reaches its lowest level is january. SO2 behaves the opposite way: winter months are more prone to have higher levels of this gas than summer months. However, we can find a common trend for both gases: the minimum level of SO2 and O3 that is present in Madrid's air has increased since 2015, which leads to the conclussion that the general level of presence of these two gases seems to be growing.

For **PMH10**, we can see that the daily minimum level of the gas seems to be slightly decreasing, which is good news. The overall trend of the temporal evolution of the level of presence of this pollutant is still irregular, reaching its maximum values in 2011, 2012 and 2013. Since 2014, the stiffness of irregularities has decreased, but the overall trend of PMH10 seems to be stabilizing in a higher level of presence of this gas, unitl january 2018, when it dramatically decreases.

**NO2** follows a similar trend to the one previously described for PMH10: in the last years, the overall level of presence of the gas seems to be increasing, to dramatically decrease in january 2018, and the overall pattern that it follows is more irregular than the one followed by PMH10.

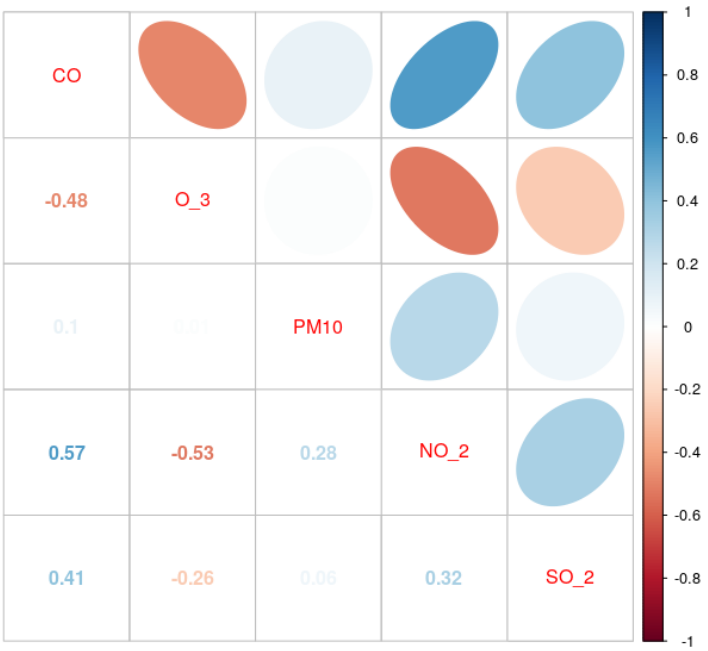
It also seems that the level of **CO** in Madrid's air is slightly increasing since 2015: the trend followed by minimum values is stabilizing towards a higher value than it was before this year. CO follows a more regular pattern than PMH10 and NO2, decreasing every july and increasing towards january, but there seems to be additional factors during the rest of the year that generate some fluctuations in the pattern.

The patterns followed by the most regular pollutants (O3, SO2 and CO) can be intuitively explained by some real world events:

**O3** is rarely emitted by anthropogenic sources: it is, instead, a direct consecuence of some quimical reactions that take place in the presence of sunlight, which explains why O3 levels are so high during the summer months. On the other hand, **SO2** emissions are a direct consecuence of human action: burning fossil fuels. One fact that could explain the high level of SO2 during winter may be that during these months, there is a higher industrial activity than in summer. **CO** emissions are caused by high motor-vehicle emissions, so the high level of CO during winter could be explained as a consecuence of higher traffic levels in an area like Madrid during winter than during summer. Probably, if we analysed the levels of CO in a turistic place, the level of this gas would be higher in summer than in winter.

## Strength of relationships

As our intention is to make predictions using our data, it is convenient to analyse the linear relationships among the five pollutants, if any. In order to do so, the correlation matrix is represented in the plot below:




In the plot, we can observe that the most correlated pair of variables is CO and NO_2, which are positively related with a value of 0.57. They are followed by O3 and NO2, which are negatively correlated with a value of -0.53.

In order to have an overall insight on whether there are high correlations among the five variables or not, we are going to calculate the determinant of the correlation matrix:

```
> det(r)
[1] 0.3320525
```

The value obtained is close to 0, but not small enough to conclude the existence of variables that are linearly dependent on others. There are correlations among the variables, but they are not high enough.

This result is coherent, as the variables recorded in our data seem to be the outcome of some other predictors which are not included in our dataset. For instance, a high number of manufacturing plants in the city, or high traffic levels, could be a direct cause of a high level of some of the gases that our five variables stand for, like SO2.

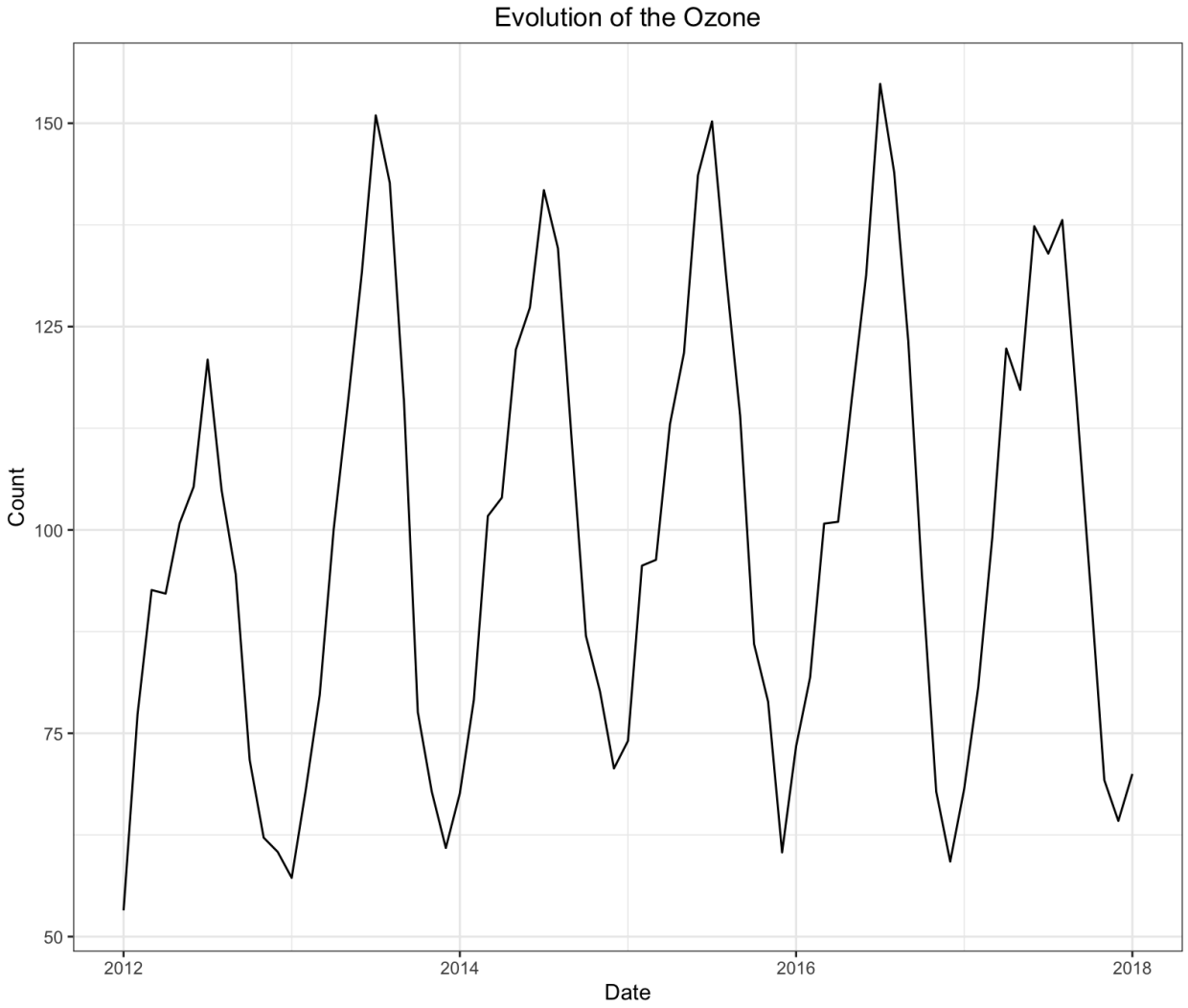
Due to these findings, in upcoming steps of this project we will not be trying to predict any of the five variables as an outcome of the others.

## Prediction

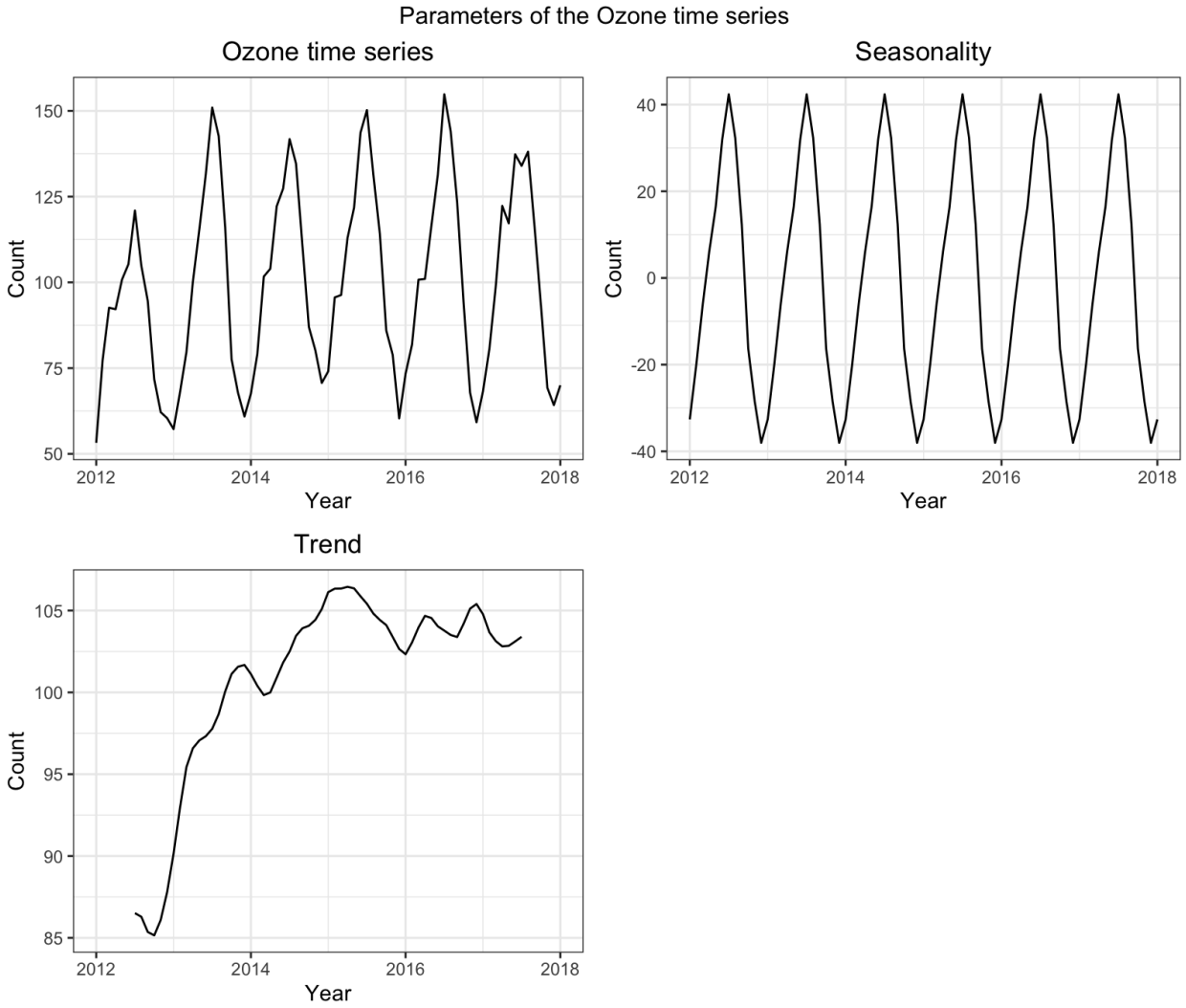
### Time series

After reading the data, the next thing that is done is the creation of the serial object and its sequential representation. In the figure shown below we can see that there is a presence of seasonality, that is, a periodic behavior of the series is observed caused by factors that are repeated over time. In our case it can be seen that at the beginning of the year the pollution is not very high but as the months go by it increases with its highest peak in summer and then goes back down. This may be due to the fact that temperatures affect pollution or that public transport is used more. Therefore, in the future it would be a possibility of field

of study to see what factors affect the different stations to have more or less pollution. On the other hand we could study the seasonality with the graph but in this case it is not clear, in this graph, no upward or downward trend since since we can see how from 2012 to 2014 it increases and yet from 2014 to 2016 it decreases although in smaller quantity than the previous increase, therefore it could not be said with this graph with certainty what its tendency is. Therefore the conclusion we draw with this graph is that we have seasonality.




After doing this, we decided to perform the ETS predictive model, used for time series. The following figure shows the time series we are studying with its decomposition of the components of the ETS model, trend and seasonality, which were what we tried to study in the previous case. In these graphs we can see more clearly that it has a very clear seasonality since as we see in the graph it grows and decreases depending on the passage of the seasons or the months, and has its highest peak in the middle of the year, which would correspond to the summer month. On the other hand we can also see the trend more clearly, just as we could deduce before, it does not have a linear trend, that is, we have a non-linear trend. This means that we have a changing trend since we do not only have pollution growth or decrease in it. However, if we look closely at the graph, pollution is usually mostly ascending since it has large growths from 2012 to 2014 and from 2014 to 2016, compared to small descending peaks that we find throughout these years, having the largest of all between 2015 and 2016 but being insignificant against rising peaks.

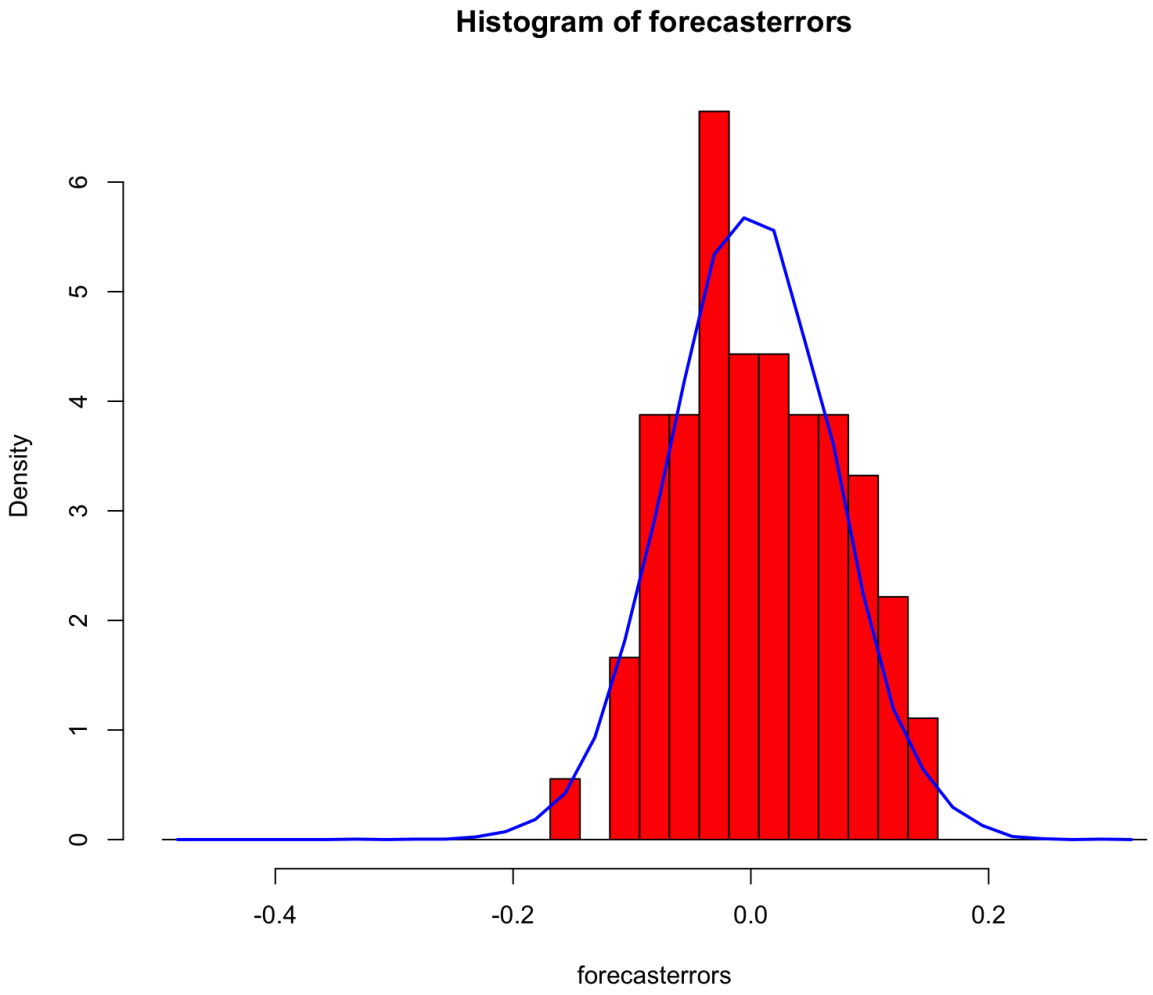



When making the ETS model we obtain the following:

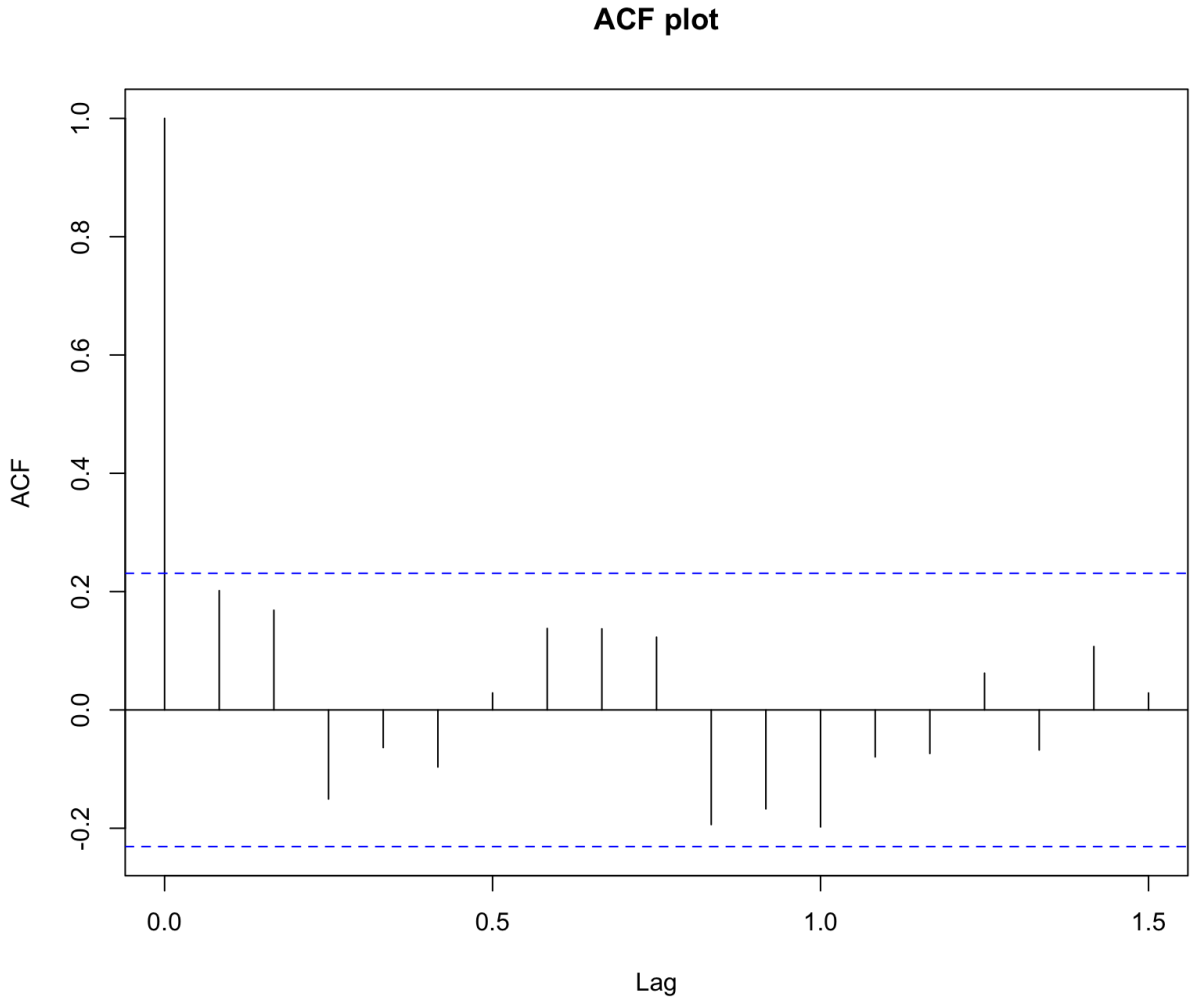
ETS(M,N,M)
Call: ets(y = train_ts)
Smoothing parameters:
alpha = 0.2495
gamma = 1e-04
Initial states:
l = 95.8235
s = 0.6168 0.7004 0.8541 1.1307 1.3166 1.433
1.3101 1.1716 1.0697 0.9337 0.8102 0.6532
sigma: 0.0766
AIC AICc BIC
609.0884 617.6598 643.2384

These data correspond to the ETS model that has been estimated for our data. Once this data is obtained, we make the plots to see everything graphically. The following graph shows the prediction you make based on the data we have provided. As you can see, it makes a pretty good prediction since, as you can see in the graph, the red line that represents our predictions has a shape that resembles that of the real data represented in green.

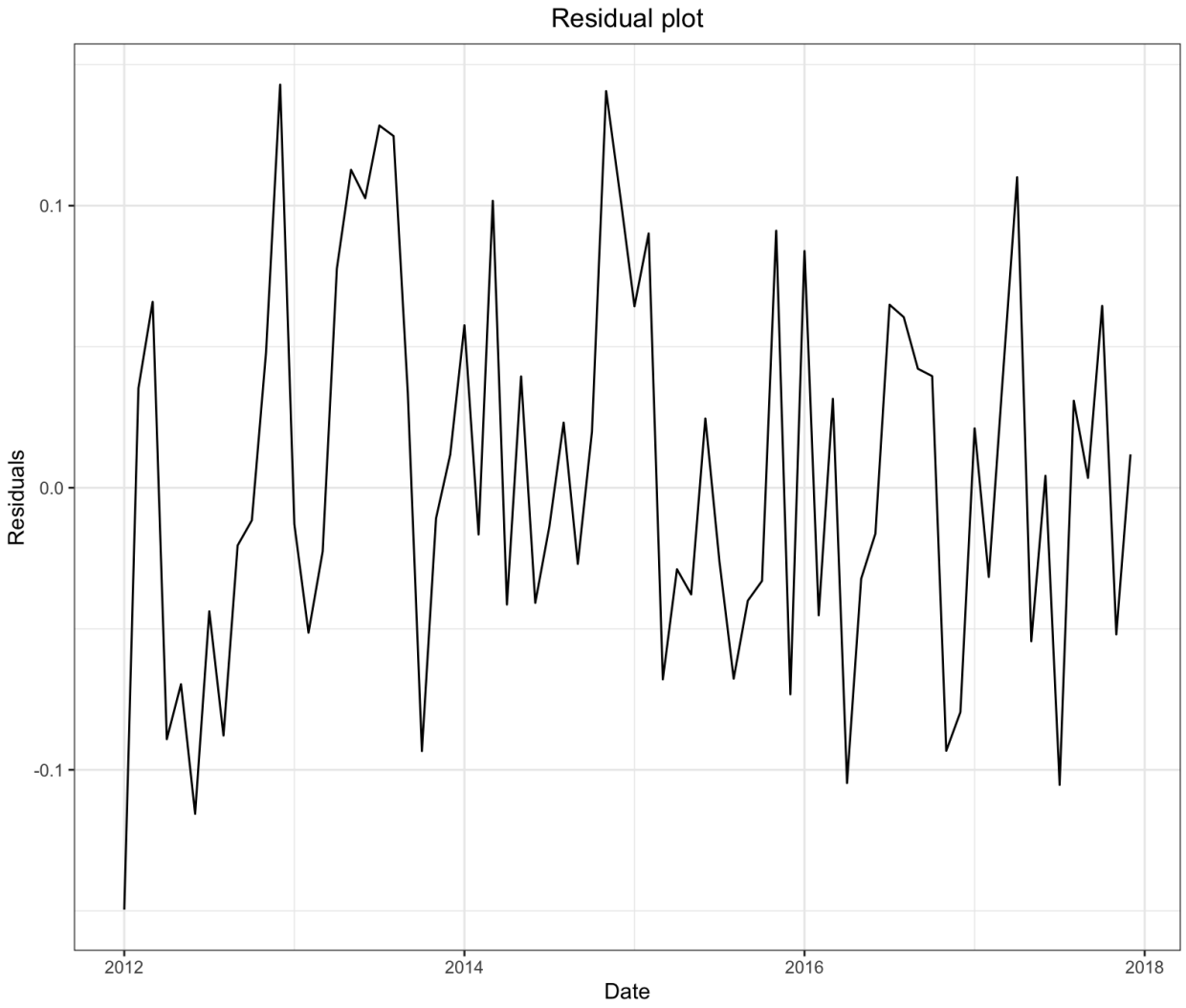

After making the prediction we have obtained the histogram of the errors produced during the prediction. The graph shows that the residues have a constant variation over time and their distribution tends to be normal, focusing on 0.




As we can see in the graph, a single significant lag is obtained, which would be lag 0, since, as observed in the graph, the rest of the values are within the range of the bands. That is why although the fas (auto correlation function) does not have 0 quickly since as we can see there are several lags before it takes that value it can be said that it falls within the bands which is considered sufficient. Therefore the graph shows a regular and seasonal lags correlation.




Finally, the plot of the waste was done to complete the validation of the model. To verify that the model is satisfactory, the residuals should look random. In our case, we could consider it valid since it does not follow any pattern but it has random values and grows and decreases randomly regardless of the season and the values obtained in previous years.




### Machine learning approach

In the preprocess work, we have obtained 9 files with the polution data of all the stationes in Madrid. Those files, are from 2010 to 2018, and their features are:

```
> colnames(madrid_2010)
 [1] "X"          "date"       "BEN"        "CO"         "EBE"        "MXY"        "NMHC"       "NO_2"       "NOx"        "OXY"
[11] "O_3"        "PM10"       "PM25"       "PXY"        "SO_2"       "TCH"        "TOL"        "station"    "only_month" "only_year"
```

We have to add another column "only_day" with the numeric valuer of the day of the month.

```
> #Insert column only_day
> madrid_2010$only_day <-unlist(lapply(madrid_2010$date, day))
> colnames(madrid_2010)
 [1] "X"          "date"       "BEN"        "CO"         "EBE"        "MXY"        "NMHC"       "NO_2"       "NOx"        "OXY"
[11] "O_3"        "PM10"       "PM25"       "PXY"        "SO_2"       "TCH"        "TOL"        "station"    "only_month" "only_year"
[21] "only_day"
```

To analyze polution data in Madrid, we will focus on C0 particles and we select those features that we'll need: CO, only_day, only_month, only_year, station

```
> #Select Co, day, month, year, station
> #Select optimal features for ourmodel
> madrid_2010 <- dplyr::select(madrid_2010, CO, only_day, only_month, only_year, station)
> colnames(madrid_2010)
[1] "CO"         "only_day"   "only_month" "only_year"  "station"
```

Our predictive model needs C0 levels from the year before for each day. We must join data for each year and add "CO_year_before" columns with C0 levels from the year before.

```
> #Inner_join between years to insert CO_year_before
> madrid_11_10 <- dplyr::inner_join(madrid_2011, madrid_2010, by = c("only_day", "only_month", "station"), suffix = c("", "_year_before"))
> colnames(madrid_11_10)
[1] "CO"                 "only_day"           "only_month"         "only_year"          "station"
[6] "CO_year_before"     "only_year_year_before"
```

Now, we can merge all the years and obtain just one dataset. We'll use this new one to generate our predictive model. We have no data from 2019, so as an exercise we'll predict them from the data of years before.

```
> #Merge all the information
> madrid_list <- list(madrid_11_10, madrid_12_11, madrid_13_12, madrid_14_13, madrid_15_14, madrid_16_15, madrid_17_16, madrid_18_17)
```

We don't need the column "on_year_year_before" so we delete it.

```
> madrid_list <- dplyr::select(madrid_list, -only_year_year_before)
```

This are the dimensions and features of Madrid_list dataset, that we'll use to generate our model.

```
> #Summary
> nrow(madrid_list)
[1] 36412969
> dim(madrid_list)
[1] 36412969        6
> summary(madrid_list)
       CO              only_day       only_month      only_year        station         CO_year_before
 Min.   : 0.1000   Min.   : 1.00   Min.   : 1.000   Min.   :2011   28079011: 1540800   Min.   : 0.0600
 1st Qu.: 0.2510   1st Qu.: 8.00   1st Qu.: 3.000   1st Qu.:2012   28079027: 1540800   1st Qu.: 0.2563
 Median : 0.3000   Median :16.00   Median : 6.000   Median :2014   28079039: 1540800   Median : 0.3066
 Mean   : 0.3570   Mean   :15.72   Mean   : 6.345   Mean   :2014   28079047: 1540800   Mean   : 0.3591
 3rd Qu.: 0.4034   3rd Qu.:23.00   3rd Qu.: 9.000   3rd Qu.:2016   28079049: 1540800   3rd Qu.: 0.4150
 Max.   :10.4000   Max.   :31.00   Max.   :12.000   Max.   :2018   28079054: 1540800   Max.   :10.4000
                                                                   (Other) :27168169
```

We have a set of 36412969 registers. The next step is to divide them in a training and a data set. We should use createDataPartition function, but this is a huge dataset and our personal computers can't deal with this data amount so we just use a set of 1.000.000 registers for training, and a set of 300.000 registers for testing. We get them sequencially, although we should get them random.

```
> #Split our data into:
> #Training data: to build our model
> #Testing data: to assess our model
> training_set <- madrid_list[1:1000000, ]
> test_set <- madrid_list[1000001:130000, ]
```

The predictive problem at hand is a supervised regression problem since we want to predict a continuous value and we have previous target data to be able to generate the model. In our case, the target will be C0, and the following features will be used to generate the model: only_day, only_month, only_year, station, C0_year_before. All are numerical values, except station which is a factor.

Trying to get a consistant model and to be honest with the geneartion of it, we apply k-fold cross validation with two iterations.

```
> # Train, specifying cross validation
> fitControl <- trainControl(
+    method = "cv",
+    number = 2
+    )
```

In this moment, we are ready to generate the model, we'll use two suited algorithms for the problems in which we are involved:

-Linear Regression

```
> #Linear Regression
> lm_mod = train(
+    CO ~ .,
+    data = training_set,
+    method = "lm",
+    trControl = fitControl,
+ )
```

-k-Nearest Neighbors

```
> #k-Nearest Neighbors
> knn_mod = train(
+    CO ~ .,
+    data = training_set,
+    method = "knn",
+    trControl = fitControl,
+ )
```

After more than three hours training the knn model with 1,000,000 records and applying k fold cross-validation, on a computer with Intel Core i7-6700 3.41GHz processor, we decided to stop the process and reduce the size of the training data and test to generate the models again. In order to generate a model with the 36412969 records, we should use clusters provided by AWS, Google Cloud or Microsoft Azure. Given the capacity of our personal equipment, we simply generate the models with 100,000 records.

```
> #Training data: to build our model
> #Testing data: to assess our model
> training_set <- madrid_list[1:100000, ]
> test_set <- madrid_list[100001:130000, ]
```

lm_mod and knn_mod are the generated models, to which before putting into production, we will evaluate them to assess the accuracy of our predictions.

```
> #Validation
> #Assess the accuracy of your predictions
> preds_lm_mod <- predict(lm_mod, test_set)
> preds_knn_mod <- predict(knn_mod, test_set)
```

We obtain the square root of the average difference between the observed known outcome values and the predicted values.

```
> #The RMSE corresponds to the square root of the average difference between the observed known outcome values and the predicted values
> #The lower the RMSE, the better the model.
> # Compute the prediction error RMSE
> RMSE(preds_lm_mod, test_set$CO)
[1] 0.1349137
> RMSE(preds_knn_mod, test_set$CO)
[1] 0.1269353
```

With this, we can observe that on the test data, the knn_mod model obtains a better RMSE and therefore with this data it is a better model and will be the selected one.

To apply the model and predict the 2019 data, we have to have the input data ready. The data to predict have CO_year_before those corresponding to 2018.

```
> #We have the model, so we can use it to predict 2019 CO levels
> # First of all, let's prepare the input data from the 2018 levels
> madrid_19_18 <- madrid_18_17
> madrid_19_18$CO_year_before <- madrid_18_17$CO
> madrid_19_18$only_year <- 2018
```

Madrid_19_18 has 1658304 records, again a figure too high for the resources we have.

```
> nrow(madrid_19_18)
[1] 1658304
```

For each season, we generate the average of CO_year_before of each day.

```
> madrid_19_18 <- madrid_19_18 %>%
+   group_by(only_day, only_month, only_year, station) %>%
+   summarise(CO_year_before = mean(CO_year_before))
> nrow(madrid_19_18)
[1] 2904
```

We apply the model and obtain the CO values in Madrid for the year 2019 in each season.

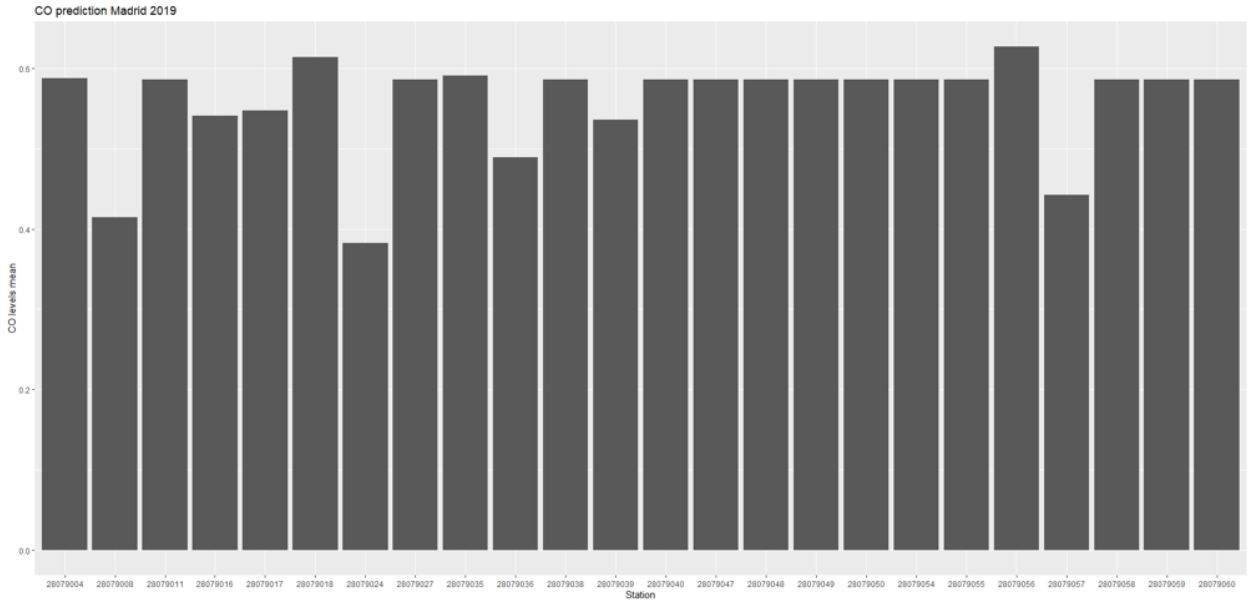
```
> #we apply knn model
> preds_knn_mod_2019 <- predict(knn_mod, madrid_19_18)

> #Insert the results give from knn model in 2019
> madrid_19_18$CO <- preds_lm_mod_2019
```

We calculate the average CO2 for each station.

```
> mean_madrid_19_18 <- madrid_19_18 %>%
+   group_by(station) %>%
+   summarise(CO_mean = mean(CO))
```

With this data, we can already make a bar chart with the average CO2 levels for each of the measuring stations in the city of Madrid and know which district is the most polluting and the least.

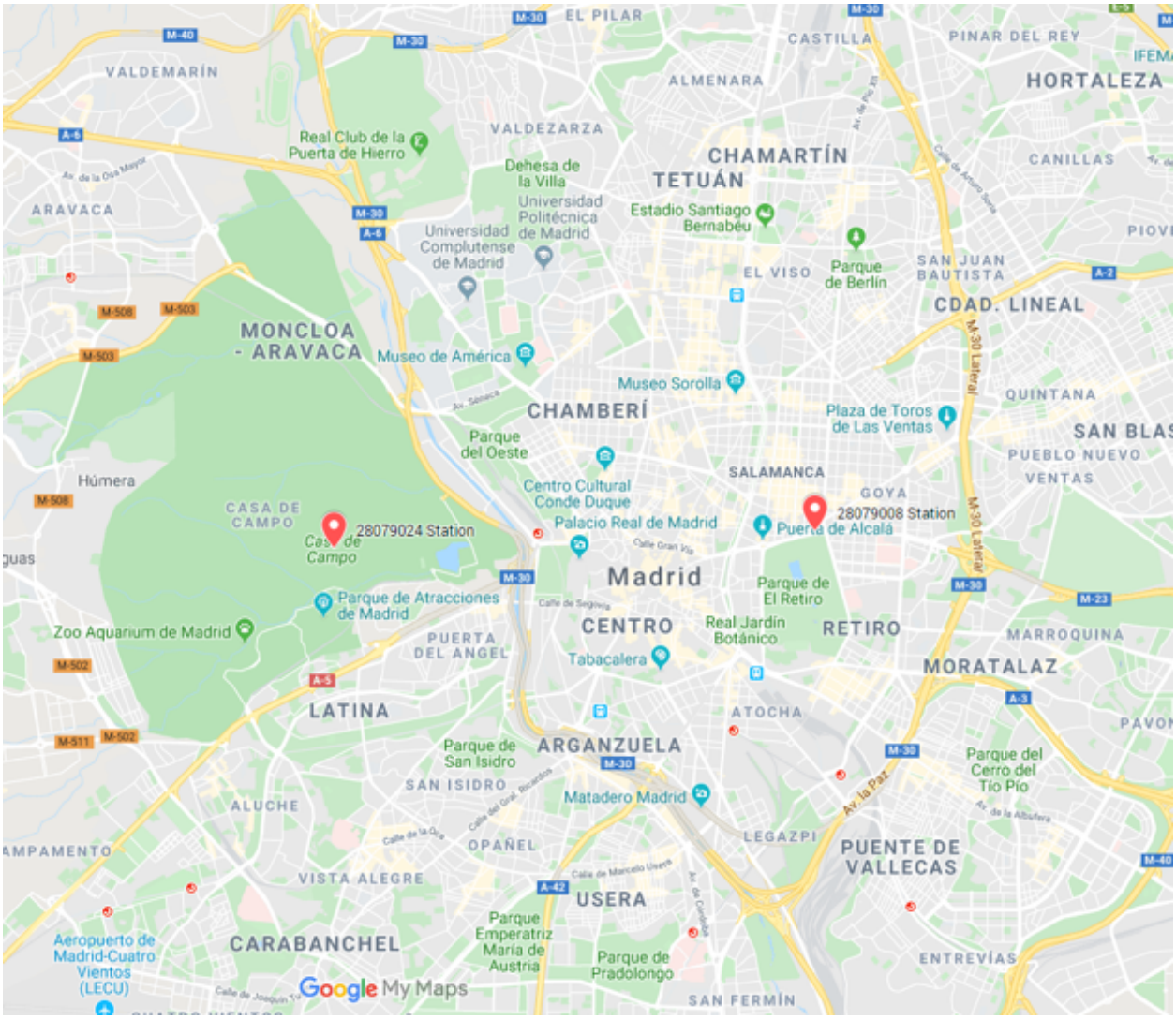
```
ggplot(data = mean_madrid_19_18, mapping = aes(mean_madrid_19_18$station, y = mean_madrid_19_18$CO_mean)) +
  geom_bar(stat = "identity") +
  ggtitle("CO prediction Madrid 2019") +
  labs(x = "Station", y = "CO levels mean")
```




With these data, and given our future home purchase, we discarded the station 28079056 in the Plaza de Fernández Ladreda in the Carabanchel district because it is the most polluted and we will have preference for the districts of Moncloa, Retiro where its stations 28079024 and 28079008 respectively where they have the lowest expected levels of C02.

We have checked the location of the stations, thanks to their description in the station.csv file and their location with the Google Maps tool. Delving into the goodness of the data obtained, it should be noted that the station 28079024 of Moncloa is located in the Casa de Campo, the great green lung of the city of Madrid. On the other hand, the 28079008 station of the Retiro district, is close to the Retiro park, another important green lung of the city.

In the following image, we can see the location of the stations with lower levels of C02 planned in 2019, around which we would look for apartments to move.




## Conclussions and future work

After the complete analysis of our data, we are able to draw some conclussions:

The exploratory data analysis gives us some accurate hints on how the five most harmful pollutants behave through time, and what patterns they follow. From the temporal evolution plots provided, we were able to see that O3, SO2 and CO are the pollutants that follow the most regular behavior: O3 increases in summer, probably due to sunlight and the quimical reactions that it promotes, while CO and SO2 increase notably during the rest of the year, most likely due to higher levels of traffic and industrial activity. The other two pollutants, PMH10 and NO2 are more irregular and its difficult to find a pattern. Regarding the most regular gases, we also saw that the overall levels of O3 and SO2 seem to be growing, as the minimum values for both gases seem to be higher than they used to be. If a person with a heart disease were to move to the city center, it would be wise to take this results into consideration

In the next step of the research, after analysing the linear relationships of the variables and seeing that its very possible that none of them can work as a predictor for any of the others, a time series prediction model has been developed to check if we are able to accurately predict the behavior of O3 in the year 2018: the residuals plot shows that the model fits well and can predict the behavior correctly. For the issue that we concern about, which is moving to Madrid's city center, this time series itself is not useful for the question that we concern about, as we are predicting events that have already happened. However, this model could

be used as a base to develop a more complex time series that could predict the behavior of the pollutants in upcoming years: that would be helpful to finally determine whether moving to Madrid's city center is a good idea or not.

In the machine learning method, we have also faced the challenge of the lack of predictors in our data. To solve it, we have decided to predict the values of pollutant CO in year 2019, using previous values of CO in the previous year. The features used as predictors are past values of CO, and the day, year, month and station in which they were recorded. Using these predictors, two models have been developed: one using linear regression, and one using KNN. Comparing the RMSE, KNN seems to be the one that fits the problem better. After choosing KNN to assess our question and predict the levels of CO for year 2019, we can finally draw some conclussions: districts of Moncloa and Retiro are the ones that the model predicts to have the lowest levels of CO. Extrapolating these results to other real world knowledge, they make sense: Moncloa is located in the middle of Casa de Campo, which is a very green zone, and Retiro district has the Retiro park, the biggest one in the city center. We can also discard Carabanchel as a future place to live, as it presents the highest levels of CO. These results seem coherent, but the model could be enriched with other predictors that are known CO sources in future research, for instance, the traffic level.

As an overall lesson, in future research on this domain using this data, it is convenient that additional variables that can work as predictors are used in the analysis, for both the machine learning and temporal series approaches. For example, the level of traffic in the city could be a good predictor to determine the growth of SO2 and CO, while the level of sunlight could be another predictor for O3. This would allow future contributors that are interested on this domain to perform a prediction based on factors other than temporal evolution or past values of a pollutant, to make the predictions more accurate.