Mini-research project

Maksym Protsyk, Diana Kmet

Chosen Data

For our research, we have taken a data set called “Air Quality Data in India (2015 - 2020)” and focused on the **station_day.csv**, which contained information about air quality and amount of different participles in it measured by stations across the India (measurements were taken every day).

Goals

We set ourselves three goals:

1. Examine the distribution of AQI (Air quality index)
2. Check how pandemic affected the AQI
3. Find some relations between the amounts of gases contained in the air (building linear regression)

Importing Libraries

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(diptest)
library(reshape2)
source("http://www.sthda.com/upload/rquery_cormat.r")
library(e1071)
```

Reading data

```r
data <- read.csv("station_day.csv")
data$Date <- as.Date(data$Date)
head(data)
```

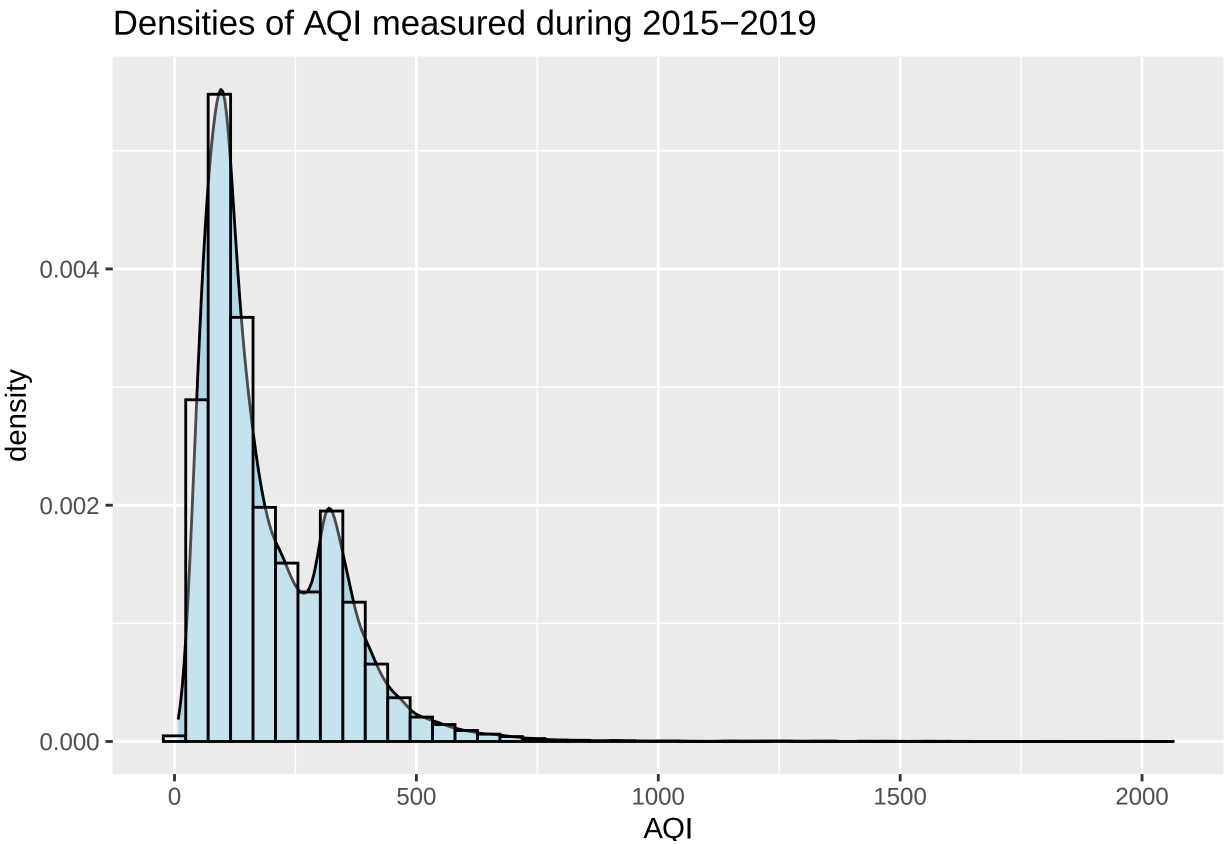
```
##   StationId       Date PM2.5   PM10   NO   NO2   NOx   NH3   CO   SO2     O3
## 1     AP001 2017-11-24 71.36 115.75 1.75 20.65 12.40 12.19 0.10 10.76 109.26
## 2     AP001 2017-11-25 81.40 124.50 1.44 20.50 12.08 10.72 0.12 15.24 127.09
## 3     AP001 2017-11-26 78.32 129.06 1.26 26.00 14.85 10.28 0.14 26.96 117.44
## 4     AP001 2017-11-27 88.76 135.32 6.60 30.85 21.77 12.91 0.11 33.59 111.81
## 5     AP001 2017-11-28 64.18 104.09 2.56 28.07 17.01 11.42 0.09 19.00 138.18
## 6     AP001 2017-11-29 72.47 114.84 5.23 23.20 16.59 12.25 0.16 10.55 109.74
##   Benzene Toluene Xylene AQI AQI_Bucket
## 1    0.17    5.92   0.10  NA           
## 2    0.20    6.50   0.06 184   Moderate
## 3    0.22    7.95   0.08 197   Moderate
## 4    0.29    7.63   0.12 198   Moderate
## 5    0.17    5.02   0.07 188   Moderate
## 6    0.21    4.71   0.08 173   Moderate
```

```r
# we added columns for year and month of measurements, which we will use later
data$Year = as.numeric(format(data$Date, "%Y"))
data$Month = as.numeric(format(data$Date, "%m"))
```

AQI distribution

First of all, we will examine the distribution of AQI before the pandemic

```r
air.quality.old <- na.omit(data[data$Year != 2020,][c("StationId", "Year", "AQI")])
ggplot(air.quality.old, aes(x=AQI)) +
  geom_density(fill="lightblue") +
  geom_histogram(alpha=0.3, aes(y=..density..), colour="black", fill="white", bins = 45) +
  ggtitle("Densities of AQI measured during 2015-2019")
```



This distribution, however, doesn't seem to follow any well-known distribution (it has 2 peaks, where a lot of values are concentrated). Actually, the distributions that has more than one peak are called multimodal and from the drawn plot we can assume that this distribution is also multimodal. We can use the Hartigans' Dip Test for Unimodality and check this (H0 - the distribution is unimodal, H1 - the distribution is multimodal) However, the proof that the this test works that we found in some book was too complex to write it here and we decided not to include this test (nevertheless, we tried perform this test and the p-value was very small, so we can reject the H0)

Other characteristics

```
aqi <- air.quality.old$AQI
skewness(aqi)
```

```
## [1] 1.845377
```

```
kurtosis(aqi)
```

```
## [1] 8.02885
```

```
mean(aqi)
```

```
## [1] 191.3153
```

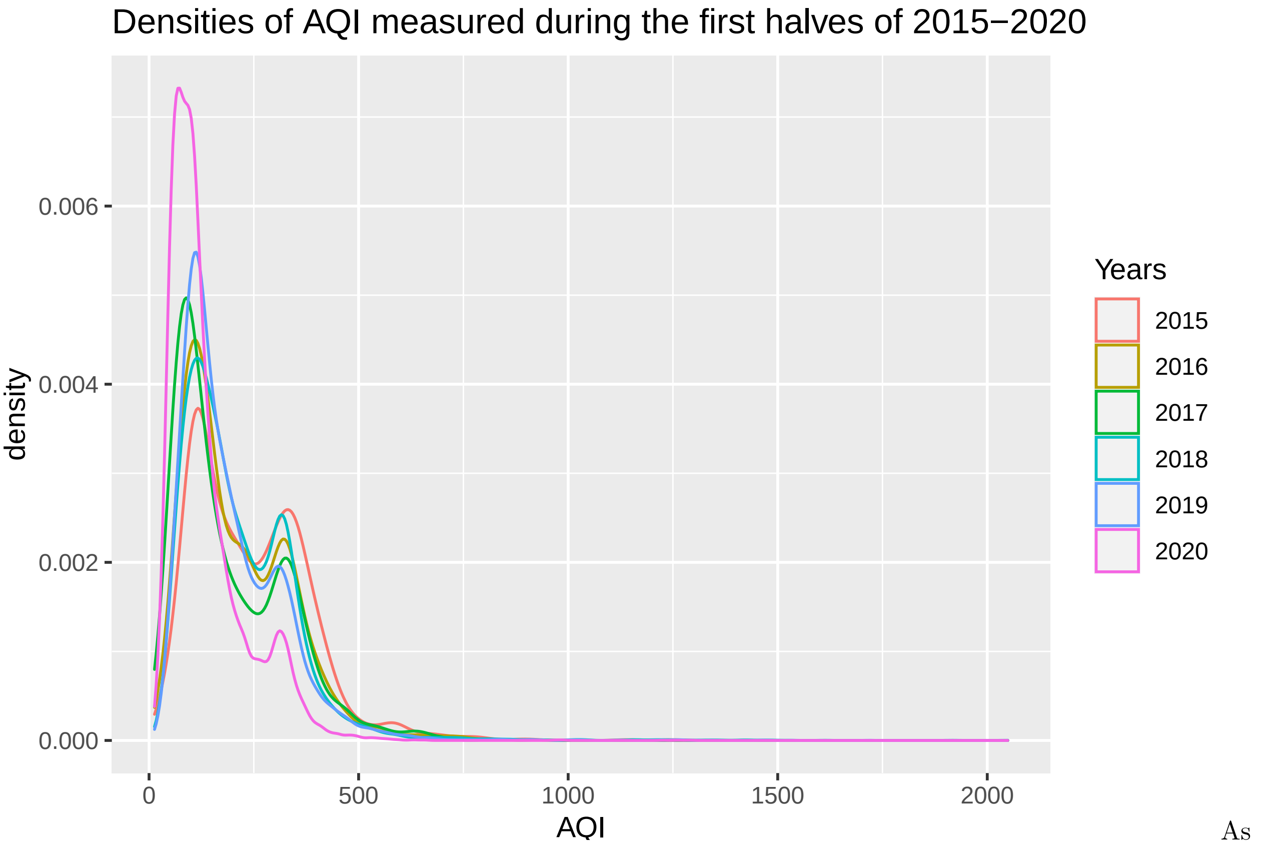
We see that our data is right-skewed and leptokurtic. The mean tells us that air in the India on average has almost poor quality (AQI > 200 means poor air quality)

Comparing 2020 to previous years

We build the plots only for first halves of years, because data contains information only about the first half of 2020

```
air.quality <- data[data$Month <= 6,][c("StationId", "Year", "AQI")]
air.quality <- na.omit(air.quality)

ggplot(air.quality, aes(x=AQI, color=factor(Year))) +
  geom_density() +
  ggtitle("Densities of AQI measured during the first halves of 2015-2020") +
  labs(color="Years")
```



As we can see from the plots, in 2020 the amount of low AQI measurements (better air quality) is much greater than in the previous years. Now let's compare some other characteristics:

```
air.quality %>%
  group_by(Year) %>%
  summarise(Mean=mean(AQI), Median=median(AQI), Sd=sd(AQI))
```

```
## ‘summarise()‘ ungrouping output (override with ‘.groups‘ argument)
```

```
## # A tibble: 6 x 4
##    Year  Mean Median    Sd
##   <dbl> <dbl>  <dbl> <dbl>
## 1  2015  242.    220  138.
## 2  2016  209.    175  126.
## 3  2017  193.    148 132.
## 4  2018  211.    180 134.
## 5  2019  194.    161 120.
## 6  2020  135.    107  91.6
```

Mean, median and standard deviation are all smaller in 2020 (smaller values and their spread), and it is reasonable to assume, that expected AIQ is less in 2020, than in other years. Let's now test this hypothesis for 2019 year. (H0 - expected value in 2020 = expected value in 2019, H1 - expected value in 2020 < expected value in 2019) We will use t-test, because we are comparing expected values and variance is unknown

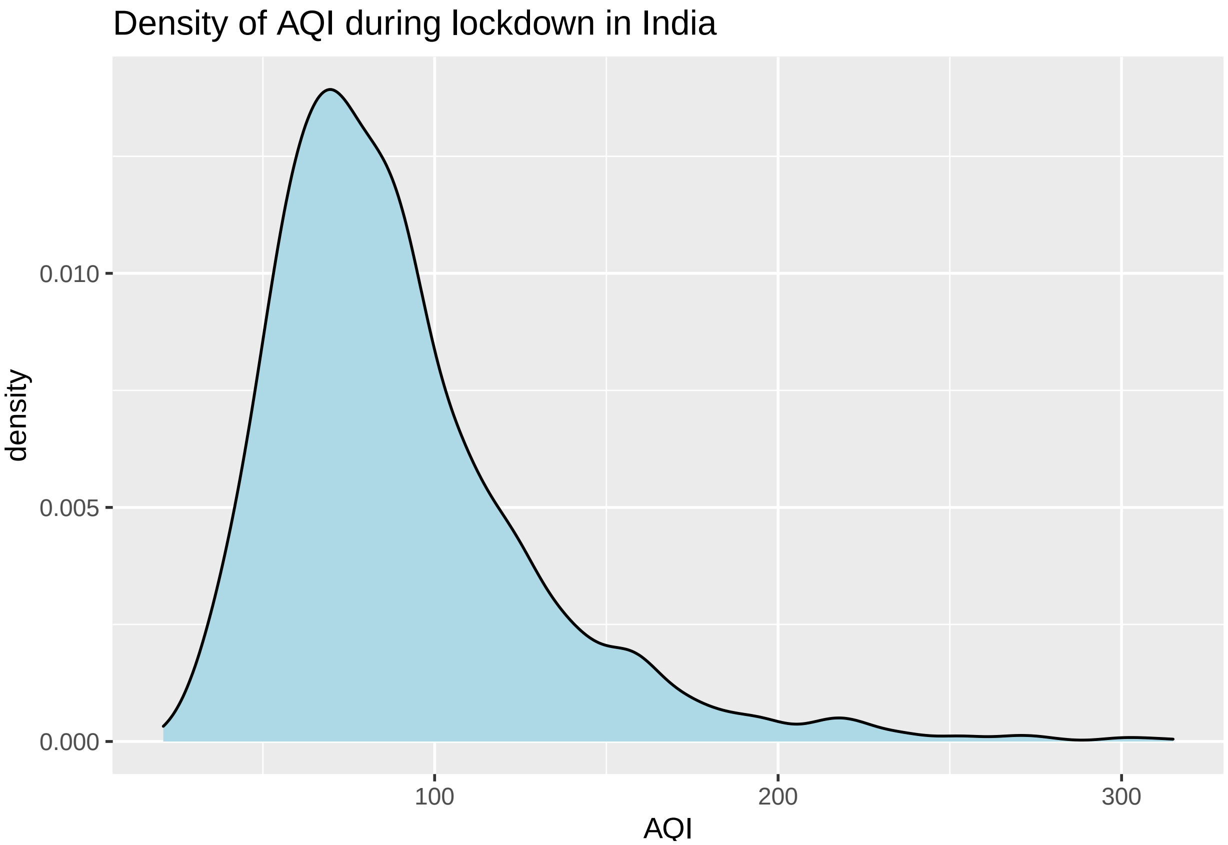
```
t.test(air.quality[air.quality$Year == 2020,]$AQI,
       air.quality[air.quality$Year == 2019,]$AQI,
       alternative = "l")
```

```
##
##  Welch Two Sample t-test
##
## data:  air.quality[air.quality$Year == 2020, ]$AQI and air.quality[air.quality$Year == 2019, ]$AQI
## t = -47.644, df = 24789, p-value < 2.2e-16
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##       -Inf -56.92107
## sample estimates:
## mean of x mean of y
##  135.4477  194.4042
```

The p-value is very small, which allows us to reject H0. So our test confirmed, that quarantine had positive effect on the air quality in India.

We also decided to check the values of AIQ during the lockdown, which took place from 25th of March to 14th of April

```
air.quality.lockdown <- na.omit(data[data$Date >= as.Date("2020-03-25") & data$Date <= as.Date("2020-04
ggplot(air.quality.lockdown, aes(x=AQI)) +
  geom_density(fill="lightblue") +
  ggtitle("Density of AQI during lockdown in India")
```



Here we see even better results. There is no second peak and it is logical, because all crowded regions which caused air quality to reduce before, were not crowded anymore. ## Relation between AQI and gases contained in the air First, we need to drop some unneccessary columns (Date, Year, Month, AQI_Bucket and StationId) and na values

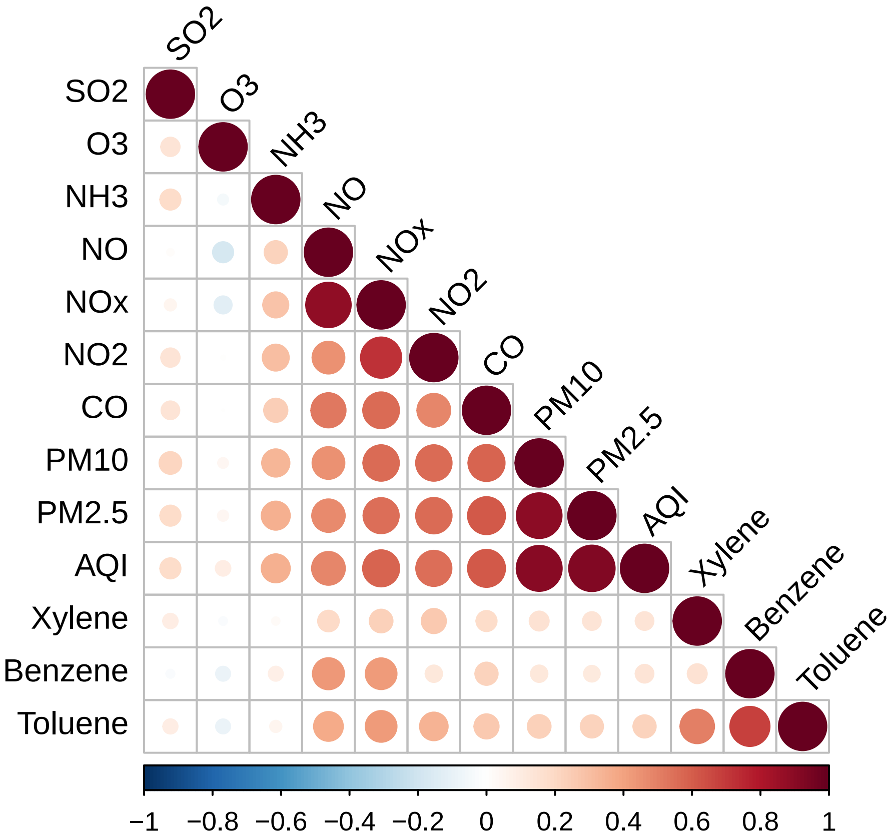
```
data.clear <- na.omit(data[c(-1, -2, -16, -18)])
head(data.clear)
```

```
##   PM2.5   PM10   NO   NO2   NOx   NH3   CO   SO2     O3 Benzene Toluene Xylene
## 2 81.40 124.50 1.44 20.50 12.08 10.72 0.12 15.24 127.09    0.20    6.50   0.06
## 3 78.32 129.06 1.26 26.00 14.85 10.28 0.14 26.96 117.44    0.22    7.95   0.08
## 4 88.76 135.32 6.60 30.85 21.77 12.91 0.11 33.59 111.81    0.29    7.63   0.12
## 5 64.18 104.09 2.56 28.07 17.01 11.42 0.09 19.00 138.18    0.17    5.02   0.07
## 6 72.47 114.84 5.23 23.20 16.59 12.25 0.16 10.55 109.74    0.21    4.71   0.08
## 7 69.80 114.86 4.69 20.17 14.54 10.95 0.12 14.07 118.09    0.16    3.52   0.06
##   AQI Year
## 2 184 2017
## 3 197 2017
## 4 198 2017
## 5 188 2017
## 6 173 2017
## 7 165 2017
```

Now let’s build the corellation matrix and see what parameters are related to each other

```r
rquery.cormat(data.clear[-14])
```

```
## corrplot 0.84 loaded
```



```
## $r
##            SO2     O3   NH3   NO  NOx  NO2   CO PM10 PM2.5  AQI Xylene Benzene
## SO2          1
## O3        0.15      1
## NH3       0.18 -0.045     1
## NO       0.019  -0.18  0.22    1
## NOx      0.056  -0.13  0.28 0.88    1
## NO2       0.15 0.0065   0.3 0.45 0.72    1
## CO        0.14 0.0011  0.24 0.52 0.57 0.48    1
## PM10      0.21  0.043  0.33 0.45 0.56 0.56 0.58    1
## PM2.5     0.18  0.047  0.35 0.47 0.55 0.56 0.61 0.89     1
## AQI       0.18  0.095  0.35 0.48 0.58 0.55 0.61  0.9  0.92    1
## Xylene   0.093 -0.027 0.025 0.19 0.23 0.26 0.18 0.16  0.14 0.14      1
## Benzene -0.029 -0.087 0.088 0.43 0.42 0.12 0.22 0.12  0.11 0.14   0.16       1
## Toluene   0.09 -0.087 0.056 0.37 0.42 0.34 0.26 0.23  0.22 0.22    0.5    0.68
##         Toluene
## SO2
## O3
## NH3
## NO
## NOx
## NO2
## CO
## PM10
## PM2.5
## AQI
## Xylene
## Benzene
## Toluene       1
##
## $p
##               SO2      O3      NH3      NO      NOx      NO2       CO     PM10
## SO2              0
## O3         1.7e-51       0
## NH3        1.2e-75 4.6e-06        0
## NO            0.06 1.3e-77 7.2e-109       0
## NOx        1.1e-08 5.3e-40 5.7e-190       0        0
## NO2        7.7e-51    0.51 8.6e-220       0        0        0
## CO         1.7e-45    0.91 9.2e-138       0        0        0        0
## PM10      4.9e-103   1e-05 1.4e-266       0        0        0        0        0
## PM2.5      8.4e-73 2.1e-06 2.4e-286       0        0        0        0        0
## AQI        1.1e-78 4.7e-22 1.5e-286       0        0        0        0        0
## Xylene     3.6e-21  0.0061    0.011 2.3e-85 4.9e-129   8e-156  8.8e-78    4e-60
## Benzene     0.0035 6.4e-19  2.7e-19       0        0  7.8e-32 1.7e-111  9.8e-35
## Toluene    3.5e-20 1.3e-18  1.1e-08       0        0 1.1e-273 8.9e-155 2.8e-120
##             PM2.5     AQI  Xylene Benzene Toluene
## SO2
## O3
## NH3
## NO
## NOx
## NO2
## CO
## PM10
## PM2.5            0
## AQI              0       0
## Xylene     6.5e-45 9.2e-48       0
## Benzene    3.4e-30 6.2e-48 6.7e-60       0
## Toluene   5.6e-114  1e-108       0       0       0
##
## $sym
##         SO2 O3 NH3 NO NOx NO2 CO PM10 PM2.5 AQI Xylene Benzene Toluene
## SO2     1
## O3          1
## NH3            1
## NO                 1
## NOx                +  1
## NO2                .  ,   1
## CO                 .  .   .   1
## PM10           .   .  .   .   .  1
## PM2.5          .   .  .   .   ,  +    1
## AQI            .   .  .   .   ,  +    *     1
## Xylene                                          1
## Benzene             .  .                              1
## Toluene             .  .   .                     .      ,       1
## attr(,"legend")
## [1] 0 ’ ’ 0.3 ’.’ 0.6 ’,’ 0.8 ’+’ 0.9 ’*’ 0.95 ’B’ 1
```

As we can see the strongest correlation is between PM2.5 and AQI, but if we build a model that calculates AQI depending on the parameters (the mass in micrograms of respective participles in the m^3 of air) given in the table, it will be useless, because, as we discovered, the formula of AQI actually contains all these parameters

However, we can try to predict the amount of one participle in the air, given another one. For example, we can see strong correlation between PM10 and PM2.5, which are representing the mass of participles, which are have diameter less than 10 and 2.5 respectively. Predicting PM10 based on PM2.5 can help scientists to calculate different coefficients without actually performing any measures of PM10.

We decided to build a simple linear regression model (our experiments showed, that other parameters lead to a very small increase of determinant coefficient)

Building the model

We want to test the Hypothesis that PM10 is linearly dependent of PM2.5 (H0 - the parameters are independent) For this purpose we are building the simple linear regression model.

```
predictor <- lm(PM10~PM2.5, data=data.clear )
summary(predictor)
```

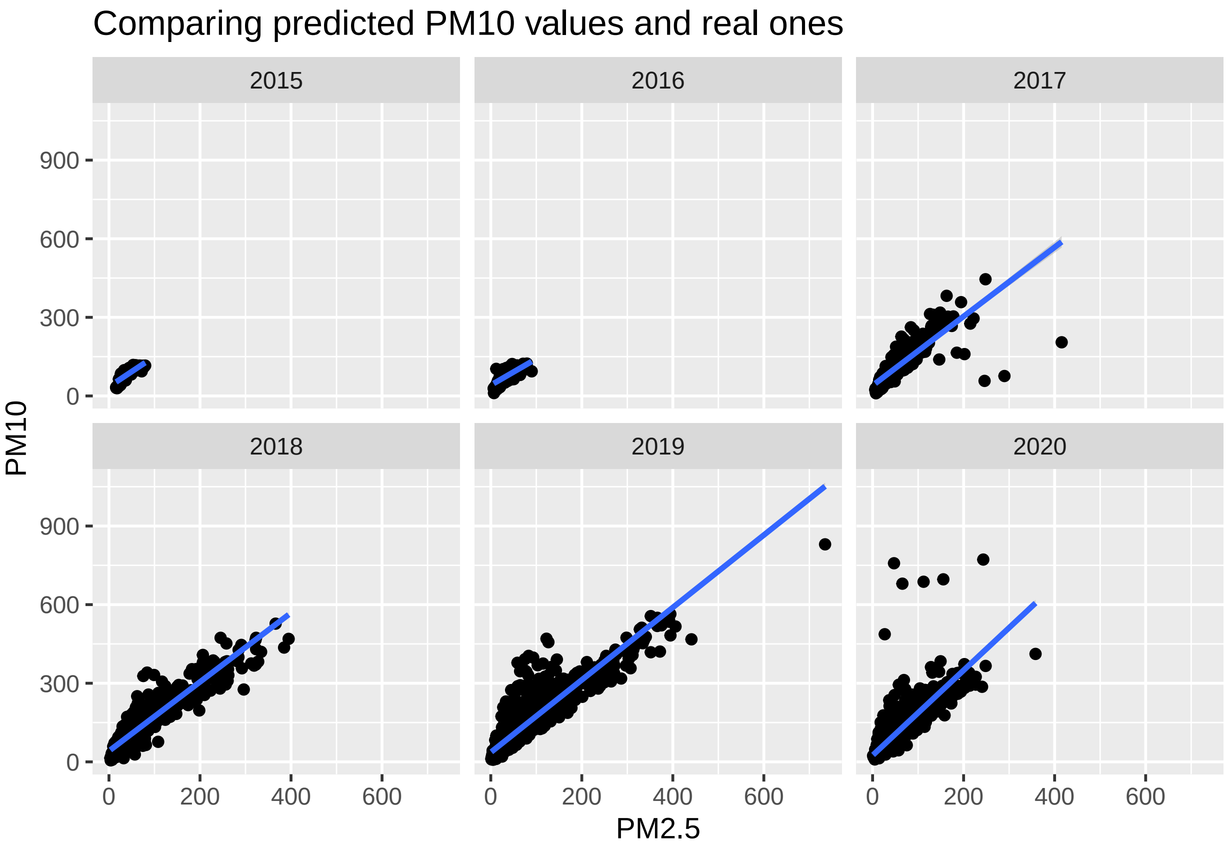
```
##
## Call:
## lm(formula = PM10 ~ PM2.5, data = data.clear)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -412.19  -18.48   -3.47   13.68  657.02
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 34.974913   0.481323   72.66   <2e-16 ***
## PM2.5        1.400857   0.007088  197.65   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 31.02 on 10312 degrees of freedom
## Multiple R-squared:  0.7912, Adjusted R-squared:  0.7911
## F-statistic: 3.907e+04 on 1 and 10312 DF,  p-value: < 2.2e-16
```

As we can see, the p-value for F-test is very small and also the determinant coefficient is equal to 0.79, which is good enough to reject the H0.

Let’s now build our predicted values and the real ones, and see how good our model fits the data

```
data.clear$Predicted = predict(predictor, newdata = data.clear)
ggplot(data.clear, aes(x=PM2.5, y=PM10)) +
  geom_point() +
  geom_smooth(method="lm") +
  facet_wrap(~factor(Year)) +
  ggtitle("Comparing predicted PM10 values and real ones")
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```


Comparing predicted PM10 values and real ones
PM10
PM2.5