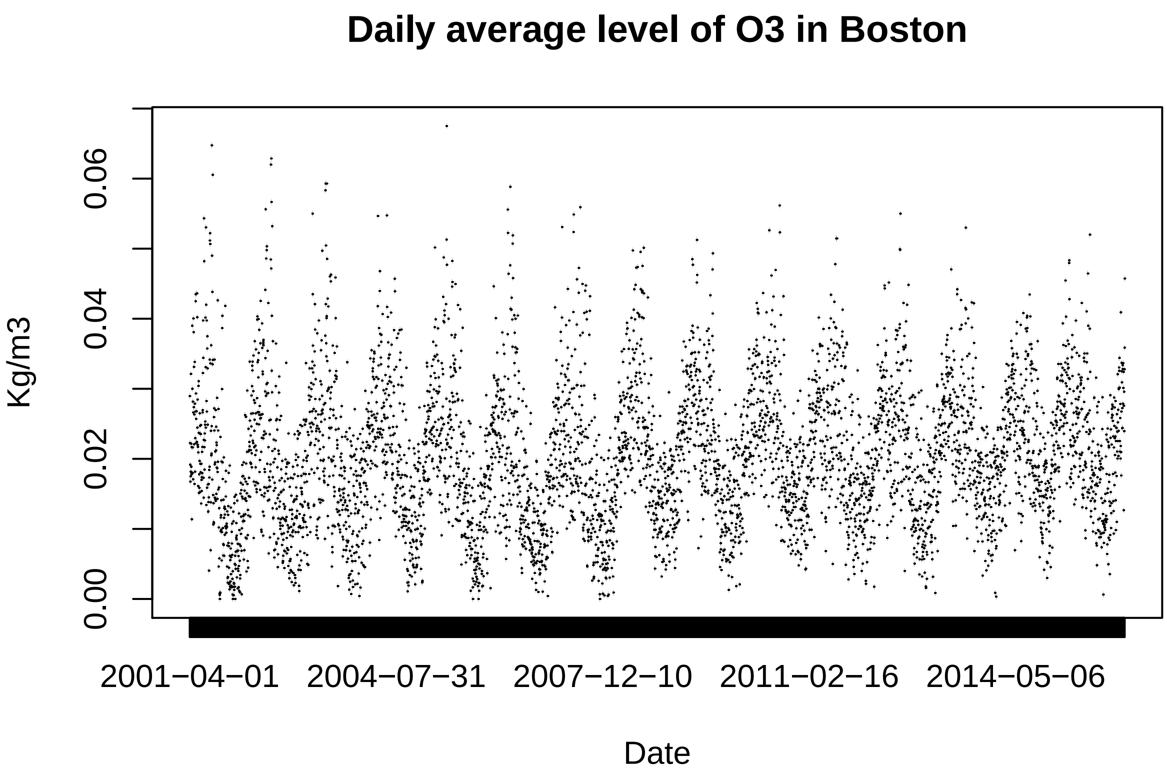
Prediction of ozone level in Boston

Lucas Emanuel Resck Domingues   Lucas Machado Moscheb*

Predicting O3 in Boston

Load and visualize



Data treatment

We noticed that some days do not exist in the dataset, for example, the day August 31, 2001 does not have information in the dataset.

```
##       X   City         State Site.Num Date.Local  O3.Mean
## 148 148 Boston Massachusetts       42 2001-08-28 0.024583
## 149 149 Boston Massachusetts       42 2001-08-29 0.015000
## 150 150 Boston Massachusetts       42 2001-08-30 0.022333
## 151 151 Boston Massachusetts       42 2001-09-01 0.021958
## 152 152 Boston Massachusetts       42 2001-09-02 0.018750
## 153 153 Boston Massachusetts       42 2001-09-03 0.028708
```

Also, there is duplicated days, as June 9, 2002:

*Escola de Matemática Aplicada

```
##        X   City         State Site.Num Date.Local  O3.Mean
## 412 412 Boston Massachusetts       42 2002-06-08 0.022917
## 413 413 Boston Massachusetts       42 2002-06-09 0.036190
## 414 414 Boston Massachusetts       42 2002-06-09 0.037000
## 415 415 Boston Massachusetts       42 2002-06-10 0.023389
```

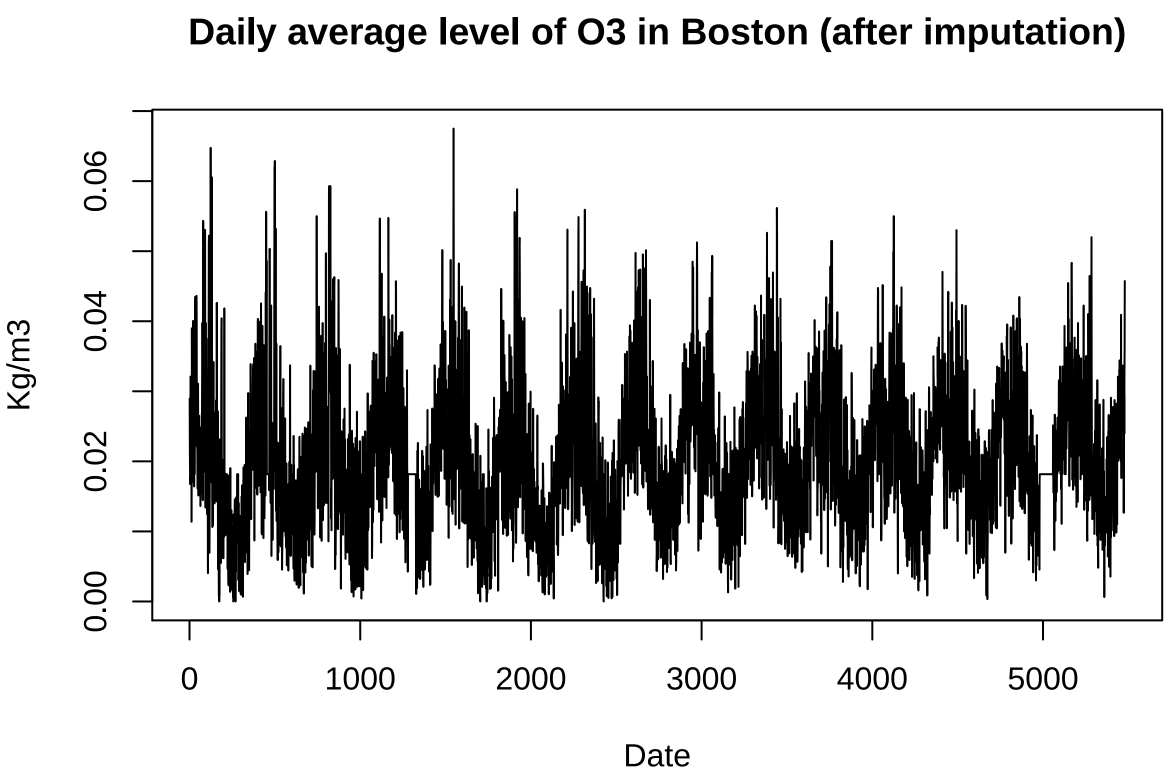
The duplicated one is easier to deal, but the nan values are harder. First we calculate the mean value between the duplicated.

The rate of NA values is almost 5% of the dataset.

```
## [1] 0.04453367
```

So as to solve that problem, we make a knn imputation using the month ($k = 30$)

```
o3.clean <- knn.impute(as.matrix(o3.ts), k = 30)
o3.clean <- as.ts(o3.clean)
plot(o3.clean, main = 'Daily average level of O3 in Boston (after imputation)',
                xlab = 'Date', ylab = 'Kg/m3')
```



Models: case 1

Now we develop some models using the train data.

The metric to compare is the Mean Absolute Error (MAE) in the predictions:

```
mae <- function(ytrue, ypred)
{
    return(mean(abs(ytrue - ypred)))
}
```

We will use `rollyapply` in order to calculate the error, considering the last two years to predict one week forward.

Decompose

First of all we make a seasonality test using Kruskal-Wallis. Actually it tests whether samples originate from the same distribution. We can organize it to be samples for each corresponding day. We compare two different frequencies: monthly and yearly. The second one showed the smallest p-value, in particular less than 0.05. FOr that reason, we will use 365 in the seasonality.

```
##
##  Kruskal-Wallis rank sum test
##
## data:  o3_train and g
## Kruskal-Wallis chi-squared = 32.983, df = 30, p-value = 0.3233
```

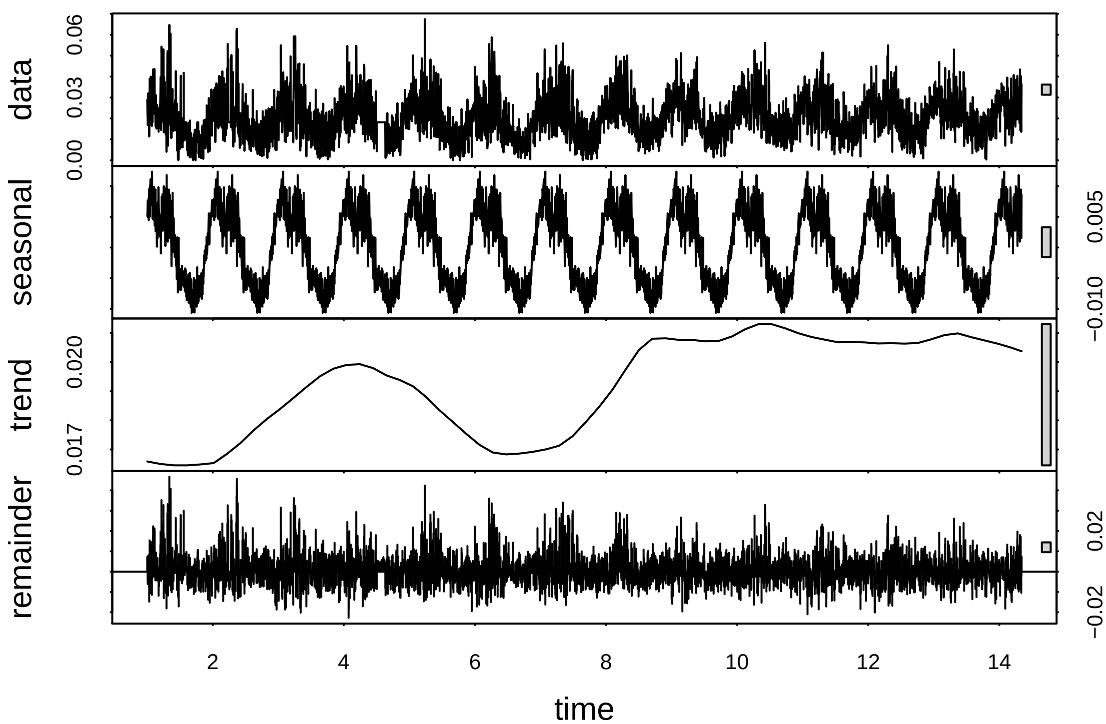
```
##
##  Kruskal-Wallis rank sum test
##
## data:  o3_train and g
## Kruskal-Wallis chi-squared = 2122.9, df = 364, p-value < 2.2e-16
```

Additive model

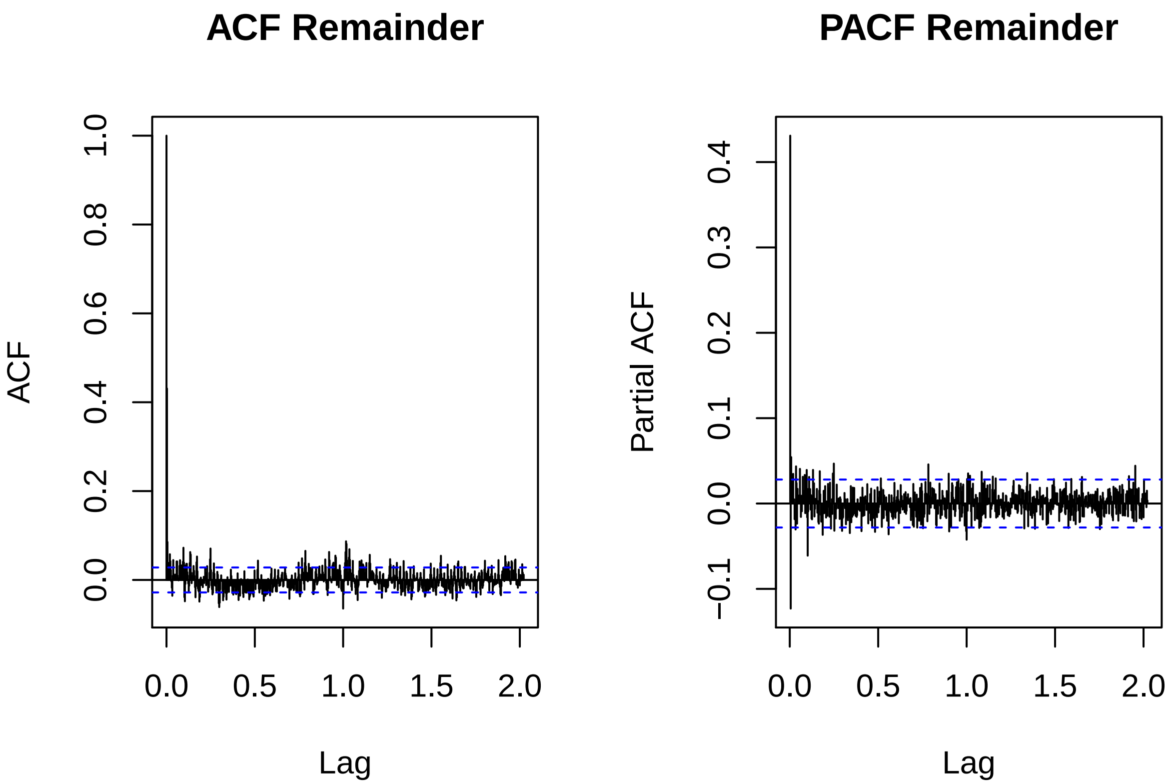
First we analyse the MAE.

```
## [1] 0.007963981
```

We also can fit the model using `t.window` and analyse the reminder of the method.



The ACF and the PACF of the reminder:



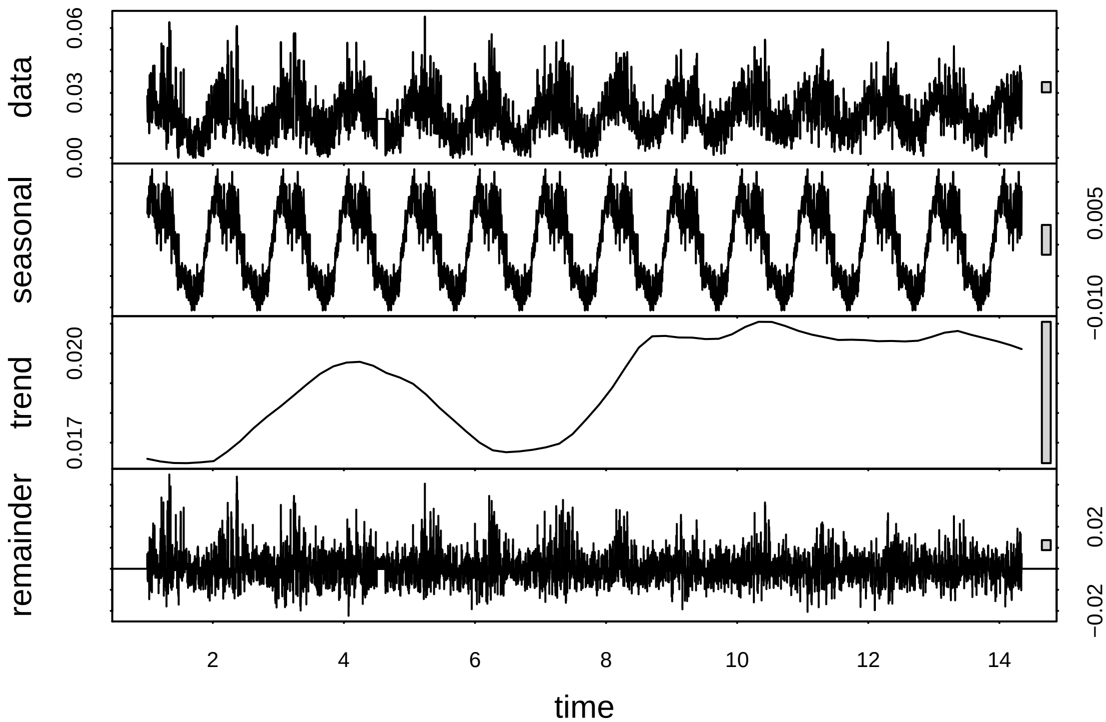
We see that there are a big spike when lag = 365. It seems not so good for a reminder. We could fit an ARMA model in this reminder yet.

Multiplicative model

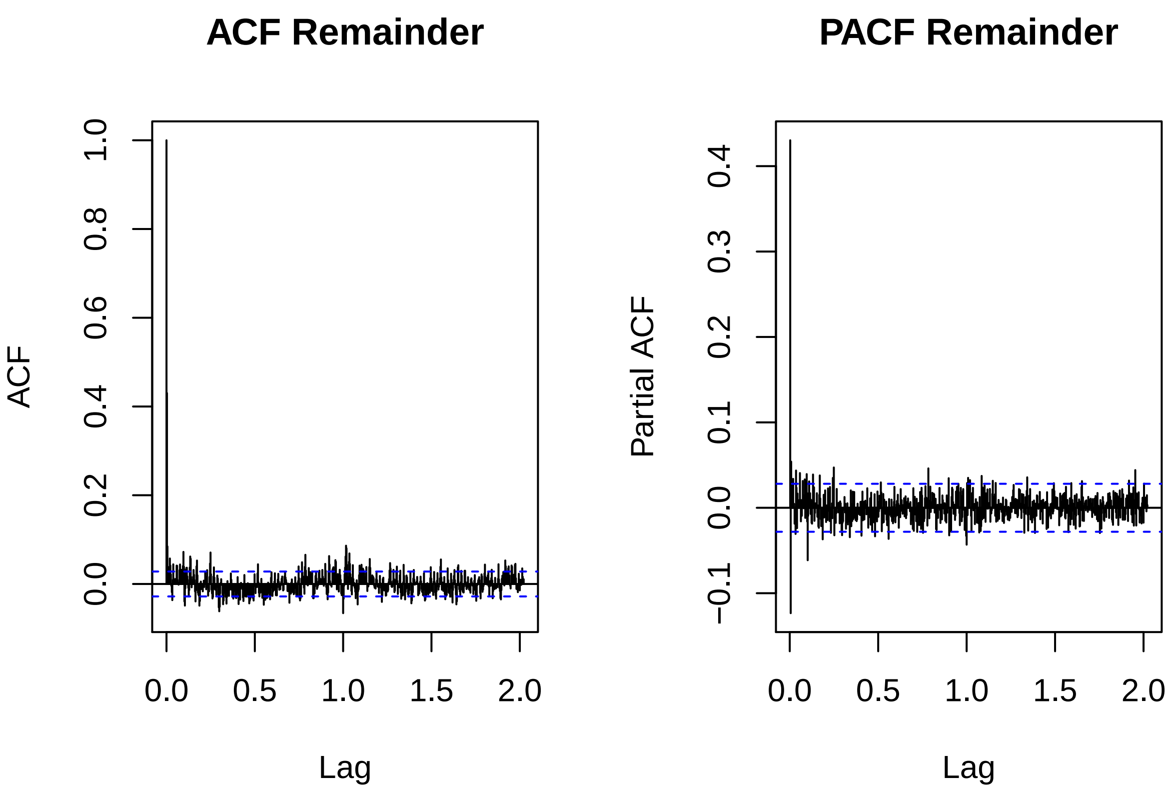
First we analyse the MAE.

```
## [1] 0.00795386
```

We also can fit the model using `t.window` and analyse the reminder of the method.



The ACF and the PACF of the reminder:



We see that there are a big spike when lag = 365. It seems not so good for a reminder. We could fit an ARMA model in this reminder yet. The same problem as before.

Regression

Holt-Winters

ARMA

Models: case 2