

# Forest Fires in Portugal

*João Teixeira - up200705307, Nuno Peixoto - up200804621*

## Contents

<b>Introduction</b>	<b>2</b>
<b>Exploratory analysis of the data</b>	<b>2</b>
Global Summary . . . . .	2
Main Variables Summary . . . . .	2
Target Variable Summary . . . . .	6
<b>Data Pre-Processing</b>	<b>10</b>
Remove None importance Variables . . . . .	10
Normalizing Value . . . . .	10
<b>Predictive Models</b>	<b>12</b>
Multiple Linear Regression . . . . .	12
Regression Trees . . . . .	12
k-Nearest Neighbors(KNNs) . . . . .	13
Support Vector Machines(SVMs) . . . . .	14
Artificial neural networks(ANNs) . . . . .	16
Assembles . . . . .	16
Conclusions . . . . .	16
<b>Clustering</b>	<b>17</b>
<b>References</b>	<b>18</b>

# Introduction

The data set of the fires in the Portugal forests allow monitoring and forecasting the fire outbreaks taking into account a number of variables that may influence the amount of burned area. (Torgo 2010)

In 2003, Portugal faced the worst forest fire season ever, during which the burned area greatly exceeded the average for the last few decades. The year 2003 was marked by the loss of 8.6% (423,949 hectares) of the total area of Portuguese forests, representing a value four times the annual average of the 90's.

The catastrophic forest fires involving human and infrastructures losses are becoming increasingly common in different parts of the world, particularly in bioclimatic regions of the Mediterranean due to dry and hot seasons. Besides leaving a long trail of burnt areas, the catastrophic fires have caused a significant number of human victims.

Each data set entry contains a set of specifications (variables) into an forest area, such as elevation, slope, the percentage area of each existing type, density and **the total burnt area**.

The purpose of this work consists in the data analysis and exploration of predictive models in order to predict the burned forest area and thereby create prevention actions to fighting forest fires.

## Exploratory analysis of the data

### Global Summary

The train dataframe has the following main attributes:

- Number of Columns: 81.
- Number of Rows: 990.
- Number of Data: 80190.
- Target Value: 1 (TotalBurntArea) - The variable is a numeric value so the predict model will be regression.
- Number of Unknown Values: 0.

The data are organized into four categories that affect somehow the area of burned forest:

**Climate Variables** - The climatic conditions may affect fuel accumulation and moisture having an effect on the probability of a fire to occur;

**Landscape Variables** - The landscape features of the Earth's surface has been extensively associated with fire occurrence;

**Socio-economic Variables** - Human factors have been used in predictive modeling of historical fire patterns;

**Topographic Variables** - The topographic features and compositions may influence the fire ignitions and the accessibility limitations to reach the fire occurrences;

### Main Variables Summary

In order to know and summary only the variables that have more impact in the target value we can calculate the gain ratio between each variable with the TotalBurntArea.

In the following table we have the **TOP5** main variables:

attr_importance	attribute
0.2037	ELEV_MAX
0.1962	bio1
0.1926	ELEV_MEAN
0.1898	bio7
0.1844	DensPop01

Table 1: Variables and the importance for the target value.

There's a short description of the main variables:

- **ELEV\_MAX** - Maximum altitude;
- **bio1** - Annual Mean Temperature;
- **ELEV\_MEAN** - Mean altitude;
- **bio7** - Temperature Annual Range (between max temperature of warmest month and min of coldest month);
- **DensPop01** - Population density in 2001;

## 1. Summary

The summarization of the data allows to get an overview of the fundamental properties of the data submitted for analysis, with the aim to describe the properties of the data between the observations measures.

With *summary* command is possible to view the following statistics:

ELEV_MAX	bio1	ELEV_MEAN	bio7	DensPop01
Min. : 14.0	Min. : 90.04	Min. : 3.081	Min. :124.3	Min. : 1.40
1st Qu.: 211.2	1st Qu.:136.00	1st Qu.: 100.619	1st Qu.:200.6	1st Qu.: 30.45
Median : 414.0	Median :143.25	Median : 244.958	Median :225.9	Median : 102.85
Mean : 480.6	Mean :143.38	Mean : 312.015	Mean :224.6	Mean : 400.66
3rd Qu.: 714.5	3rd Qu.:153.52	3rd Qu.: 476.824	3rd Qu.:250.3	3rd Qu.: 265.30
Max. :1990.0	Max. :175.20	Max. :1355.150	Max. :286.7	Max. :15597.00

Table 2: Summarization of top5 variables

## 2. Number of Outliers

An outlier is an observation that is outside the overall pattern of a distribution. Usually, the presence of an outlier indicates some sort of problem.

It is important to check the number and the rate of existing outliers for the most important variables of the data that may influence in the predictive models.

- ELEV\_MAX: 8 (0.81%)
- Bio1: 21 (2.12%)
- ELEV\_MEAN: 9 (0.91%)
- Bio7: 1 (0.1%)
- DensPop01: 132 (13.33%)

It can be concluded that the main variables (except **DensPop01**) have a low outlier rate which does not significantly affect the forecast values.

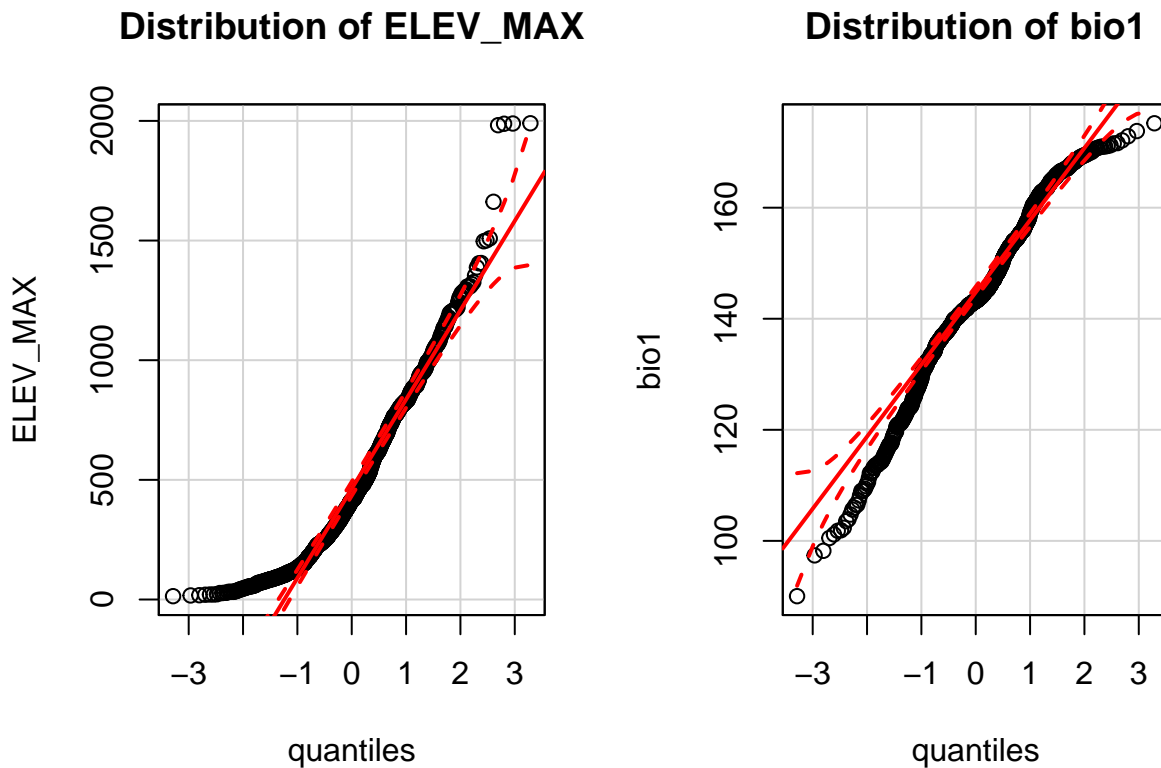
With the variable DensPop01 it may have to do some sort of analysis or pre-processing once has a higher value of outliers.

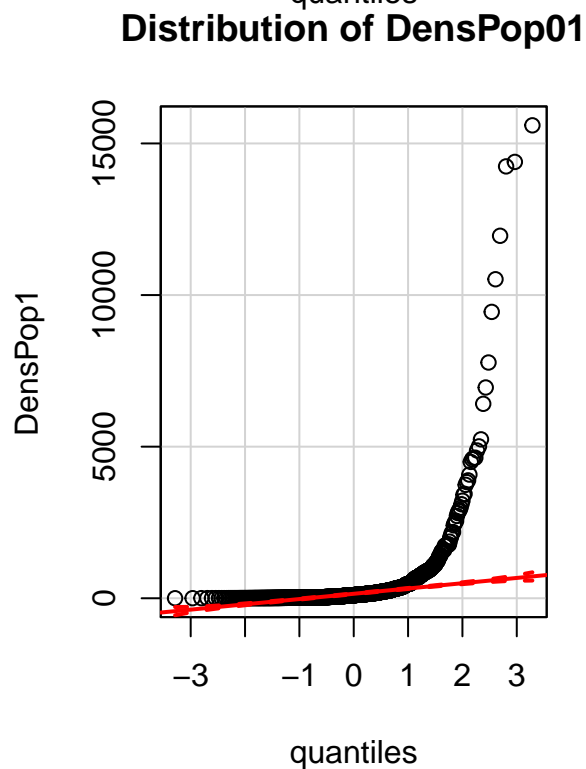
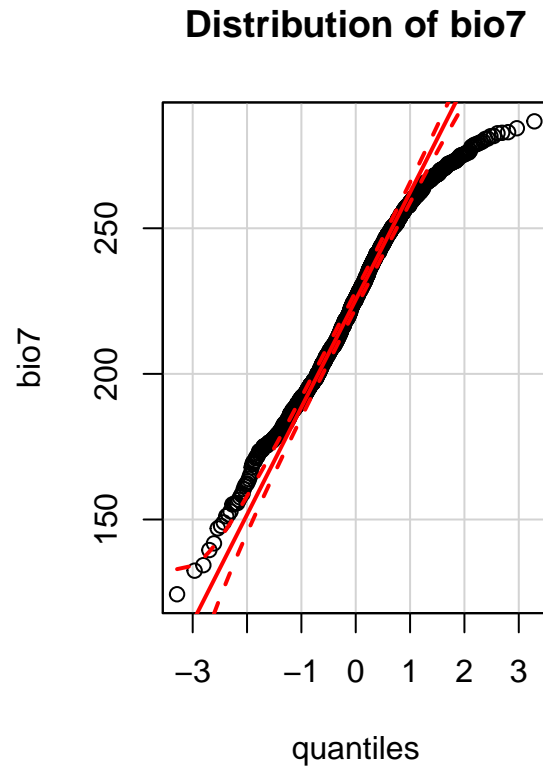
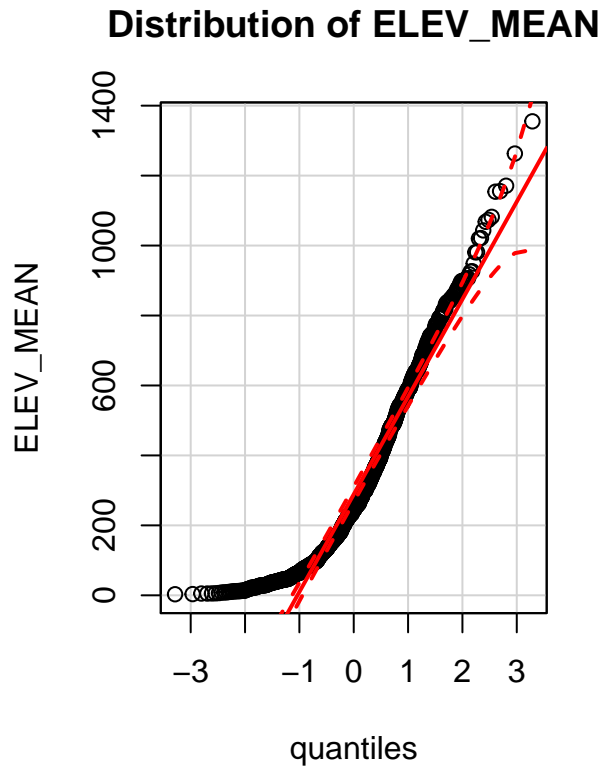
### 3. Distribution of the values

The most common graphical tool to assess the normality of the data is a Quantile-Quantile (Q-Q) plot.

In a Q-Q plot quantile the values of a theoretical distribution are plotted against quantile values of the observed sample distribution. Normally the normal distribution is used to make a judgment if the 2 quantiles are in the same distribution(Razali and Wah 2011).

In the following plots we analyse the variables with the e qqplot to check if the values follow the normal distribution.





We can see that most of the variables follow in most of the values of a normal distribution. However, and due to the high presence of outlier the **DensPop01** those not follow this distribution.

#### 4. Standard Deviation

The Standard Deviation is a measure of how spread out numbers are. If the standard deviation is low than the observation values are close to the median, otherwise the data are spread out across a wide range of values.

- ELEV\_MAX: 339.100654
- Bio1: 14.710837
- ELEV\_MEAN: 251.9971412
- Bio7: 30.9059137
- DensPop01: 1222.3683295

In 3 of 5 cases show previously showed the standard deviation is to higher. This values (ELEV\_MAX, ELEV\_MEAN and DensPop01) must influence the results of the target variable.

### Target Variable Summary

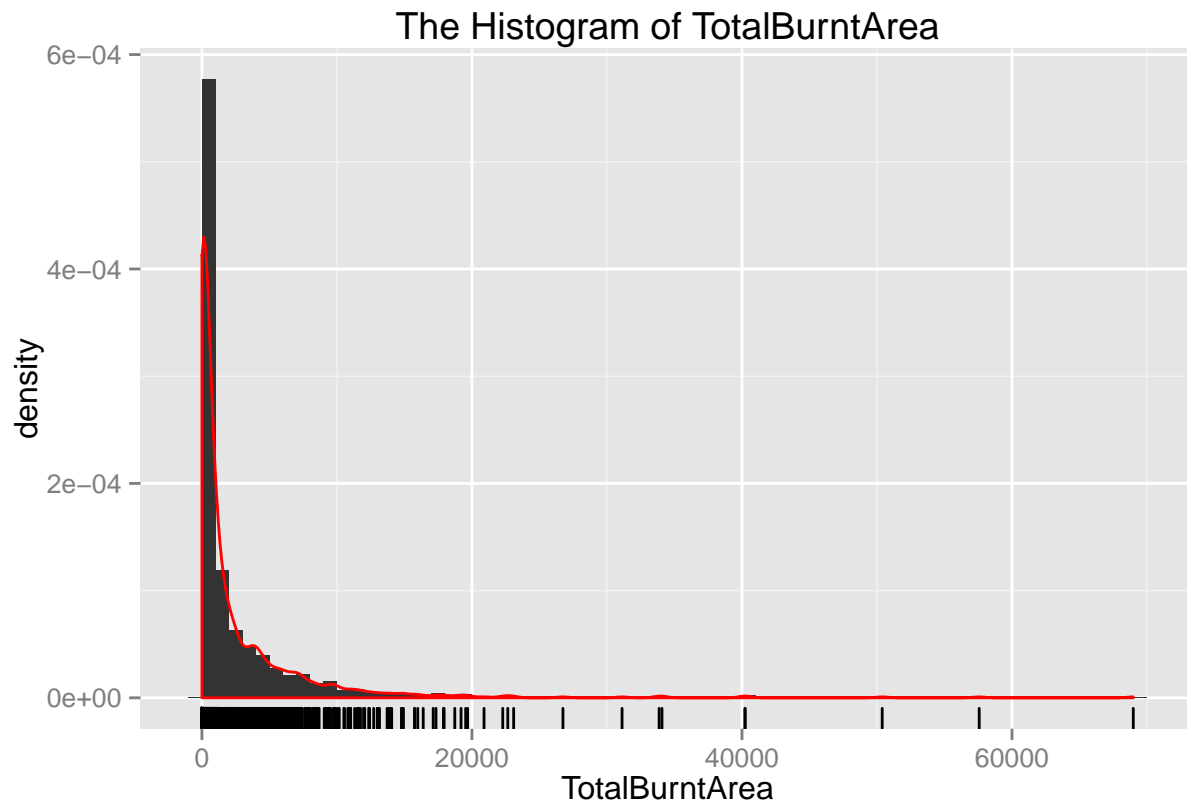
In this chapter we will make an exploratory analisys of the target variable, **TotalBurntArea**.

#### 1. Summary

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0	0	609	2550	2752	68981

Table 3: Summarization of target variable

## 2. Histogram



Through the histogram above we can see that there is a large concentration of values below 20,000 square meters burned.

## 3. Outliers

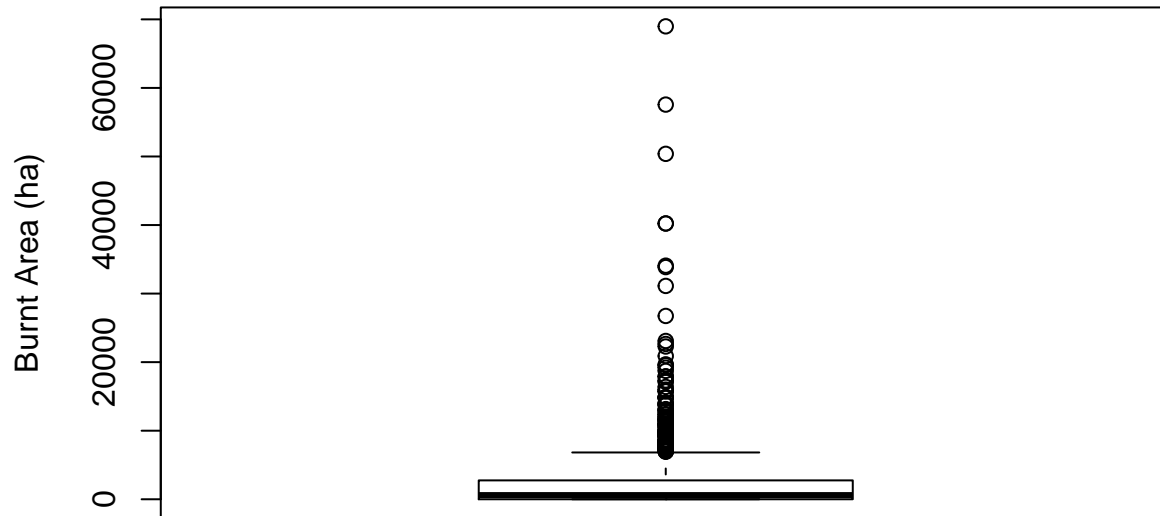
- TotalBurntArea: 106 (10.71%)

We can see that more than 10% of the total burnt area values are considered outliers.

On the one hand despite containing a large number it means that the overall burned area values are low.

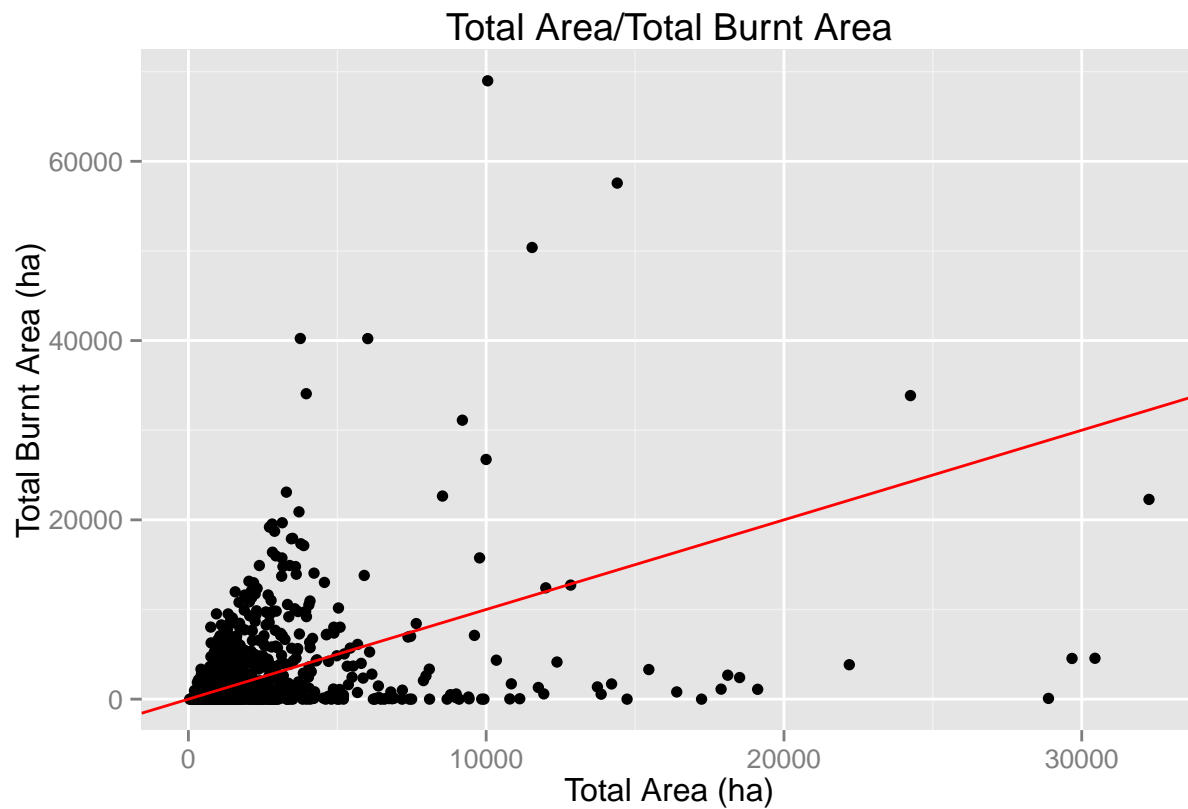
On the other hand some outlier values are pretty high which means that there are situations where the area of burned forest is very high.

### Burnt Area Boxplot



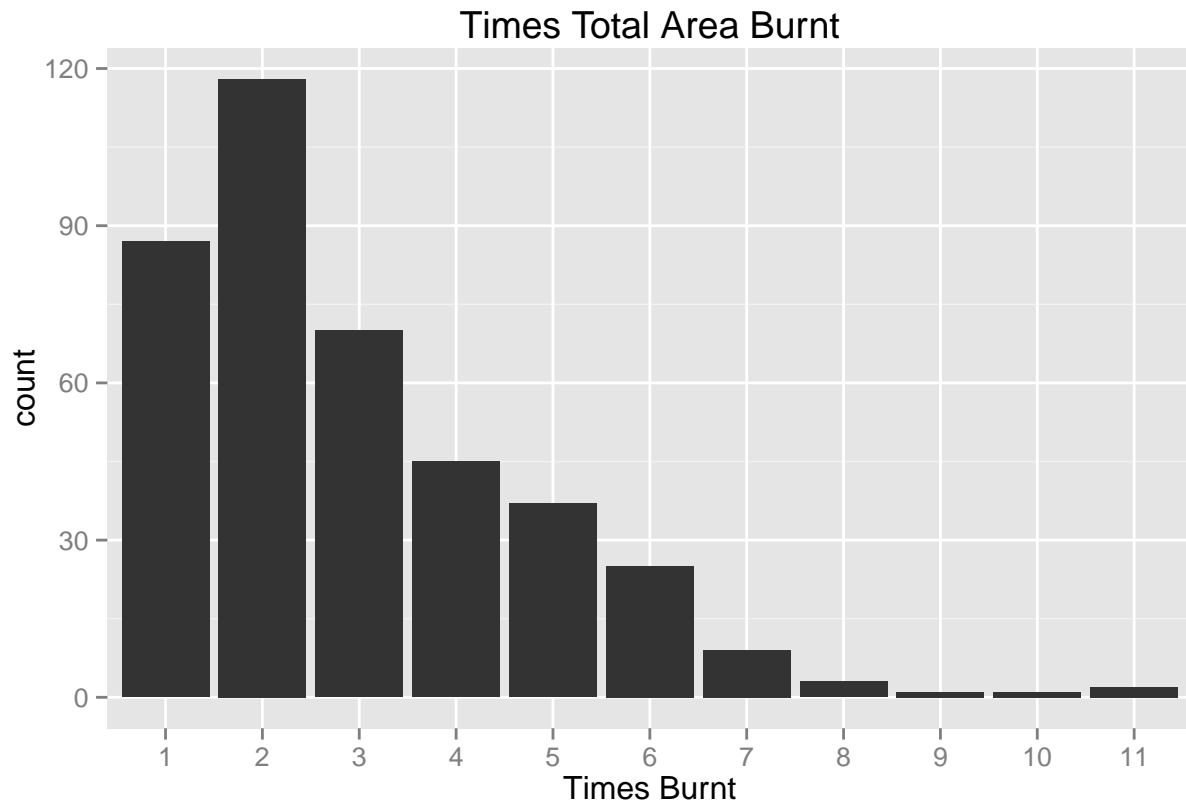
#### 4. Total Area vs. Total Burnt Area

Compare the total burnt area for each of the locations of the data set with the total area may be able to draw some conclusions.



We checked out from the graph that there are some cases (40.2 %) where the burnt area is greater than the total area. This means that in these cases the land area was completely burned repeatedly over time.





By the graphic the majority of areas analyzed burns between 1-3 times its total area. However, there are regions where their area has been completely decimated by fire repeatedly over time which allows us to conclude that these regions are more conducive to fires.

# Data Pre-Processing

## Remove None importance Variables

In order to reduce the size of data to analyze/evaluate, we can determine as was done in chapter 2.2, the importance of each variable in the training data in the target variable.

In the following table we have the variables with no relevance to the calculation of the target variable:

attr_importance	attribute
NaN	TCI_STD
NaN	LPI
NaN	ED
NaN	FRAC_SD
NaN	IJI
NaN	ENN_AM
NaN	eucalipto_AREA_perc
NaN	outfolhosas_AREA_perc

Table 4: Variables with no importance

This way we will remove these 8 variables that will help to improve the processing efficiency in building models since we eliminate 8 columns of 990 entries.

With this pre-processing that does not prejudice the information analysis could be reduced 9.88% of the total data.

From now on we will no longer use the *train\_data* provided. We will use the same data frame but without the data previously removed.

## Normalizing Value

Data normalization is one of the pre-processing techniques that we will use for the analysis of the data.

This technique allows the values and variables of the dataframe are all on the same scale (typically with mean 0 and standard deviation 1).

This way, the prediction of some regression models can be more efficient since the range of values can influence the performance of the data is provided.

Not all models that we will present ahead perform this procedure internally, so we will normalize the training dataframe.

```
norm.new_train_data <- scale(new_train_data[,])
colnames(norm.new_train_data) <- colnames(new_train_data)
norm.new_train_data <- data.frame(norm.new_train_data)
```

Comparing the values before and after the normalized data we can see that before the pre-processing the scale of values was very dispersed (e.g. comparing a 88 with a 706), while after normalization the values are in a much more “harmonic” scale.

```
head(new_train_data[,c(1:6)], 3)
```

```
##      ELEV_MAX ELEV_MEAN ELEV_STD SLOPE_MAX SLOPE_MEAN SLOPE_STD
## 1         396  168.5080  76.7385   44.9590   18.87750   11.90550
## 2         706  604.4890  42.7725   39.1152    8.99396    6.03132
## 3          88   34.2032  23.7021   14.3287    2.32026    1.76678
```

```
head(norm.new_train_data[,c(1:6)], 3)
```

```
##      ELEV_MAX ELEV_MEAN  ELEV_STD  SLOPE_MAX SLOPE_MEAN  SLOPE_STD
## 1 -0.2493703 -0.569479  0.2047242  0.5829093  1.0775585  1.73288356
## 2  0.6648126  1.160624 -0.4090371  0.2295771 -0.3401967 -0.09093957
## 3 -1.1576551 -1.102441 -0.7536369 -1.2690826 -1.2975129 -1.41499956
```

## Predictive Models

In this chapter we will look several forecasting models in order to figure out which the best regression model that can predict the target variable of the test data set with less error.

The metric evaluation to be considered for these forecasts is the **MAE** - Mean Absolute Error.

The Mean Absolute Error (MAE) measures the average absolute deviation between the predictions and the true values. The MAE is measured in the same unit as the original variable scale. That means if the values are scaled the mae is also scaled.

For evaluate the modles we will use the **performanceEstimation** package that provides a set of functions and arguments that allow us to change the values of parameters in order to check the best fit for an specific model.

## Multiple Linear Regression

### Description

Multiple linear regression attempts to model the relationship between two or more explanatory variables and a response variable by fitting a linear equation to observed data. Every value of the independent variable x is associated with a value of the dependent variable y.

### Evaluation

**MAE:** 2201.692.

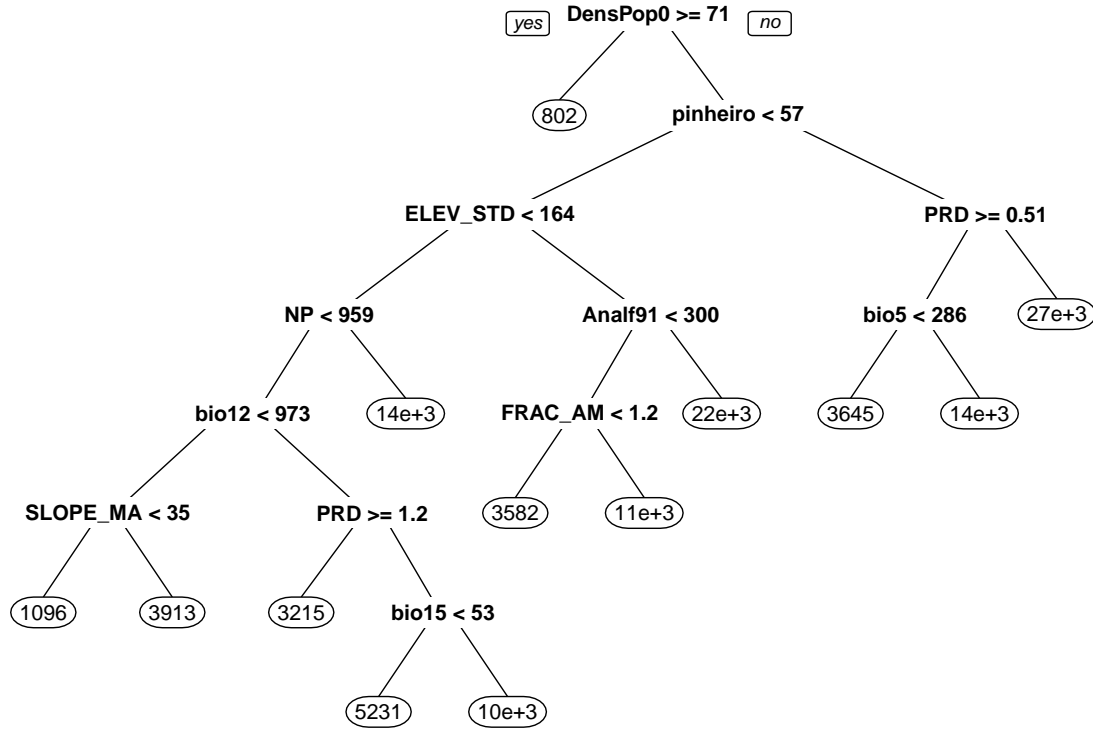
## Regression Trees

### Description

Tree-based models are models that provide as result a model based on logical tests on the input variables.

This partitioning is defined based on carefully chosen logical tests on these variables. Within each partition all cases are assigned the same prediction (either a class label or a numeric value).

Here is the tree regression generated for this case study:



## Evaluation

se	minspl
1	20

Table 5: Regression Tree Parameters

MAE: 2788.954.

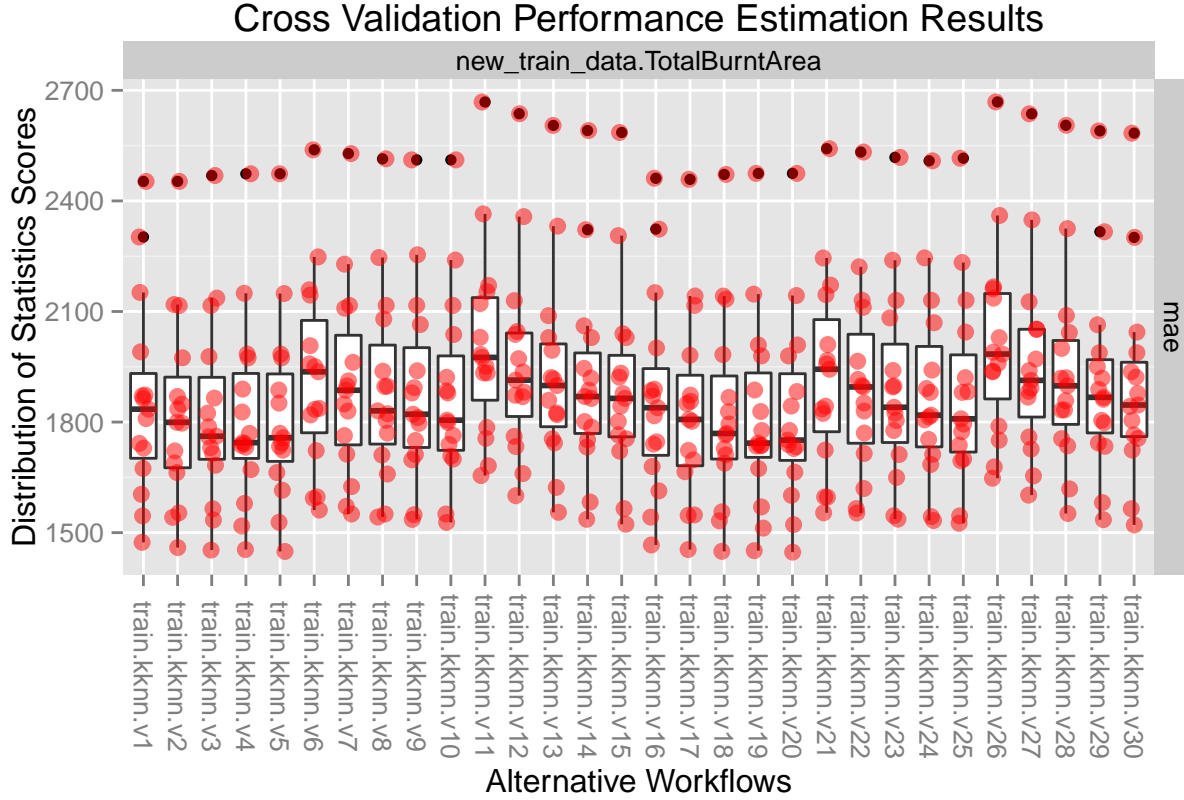
## k-Nearest Neighbors(KNNs)

### Description

The purpose of the k Nearest Neighbours (kNN) algorithm is to use a data set where the data points are separated into several separate classes to predict the classification or regression of a new sample point. This sort of situation is best motivated through examples.

### Evaluation

We made several tests to the KNN model, changing the value of the number of the neighbours considered ( $k$  parameter). We tested with variations from 5 to 50 neighbors and obtained the following result:



It is possible to verify that the result of estimation with the different values of parameter  $k$  is very similar.

scale	k	distance	kernel
TRUE	11	1	epanechnikov

Table 6: KKN module Parameters

The best MAE value obtained with this modules is: 1816.351.

## Support Vector Machines(SVMs)

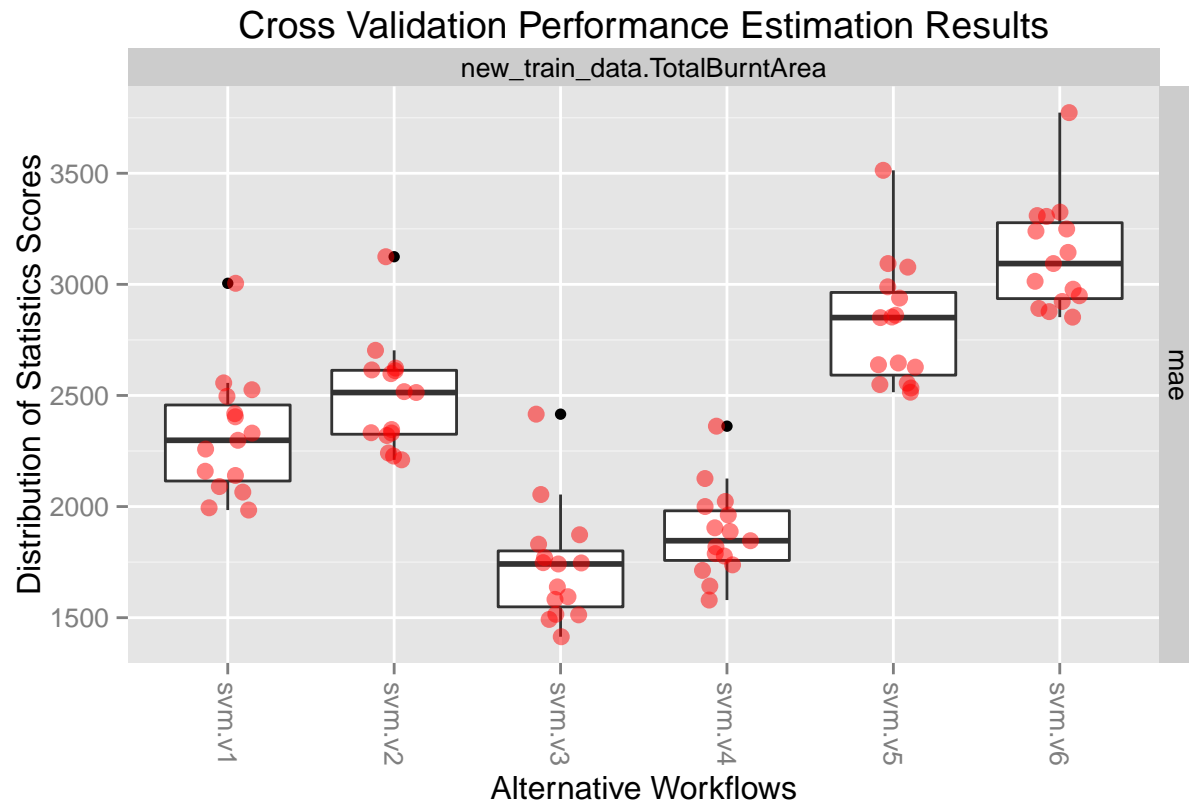
### Description

The main goal of SVMs model is mapping the original data into a new, high-dimensional space, where it is possible to apply linear models to obtain a separating hyper plane. The mapping of the original data into this new space is carried out with the help of kernel functions. (Torgo 2010)

### Evaluation

For the study of svm models and its performance we had into account two main parameters:

- cost: Cost of constraints violation (1 and 10);
- gamma: Parameter needed for kernels (1, 0.1 and 0.01);

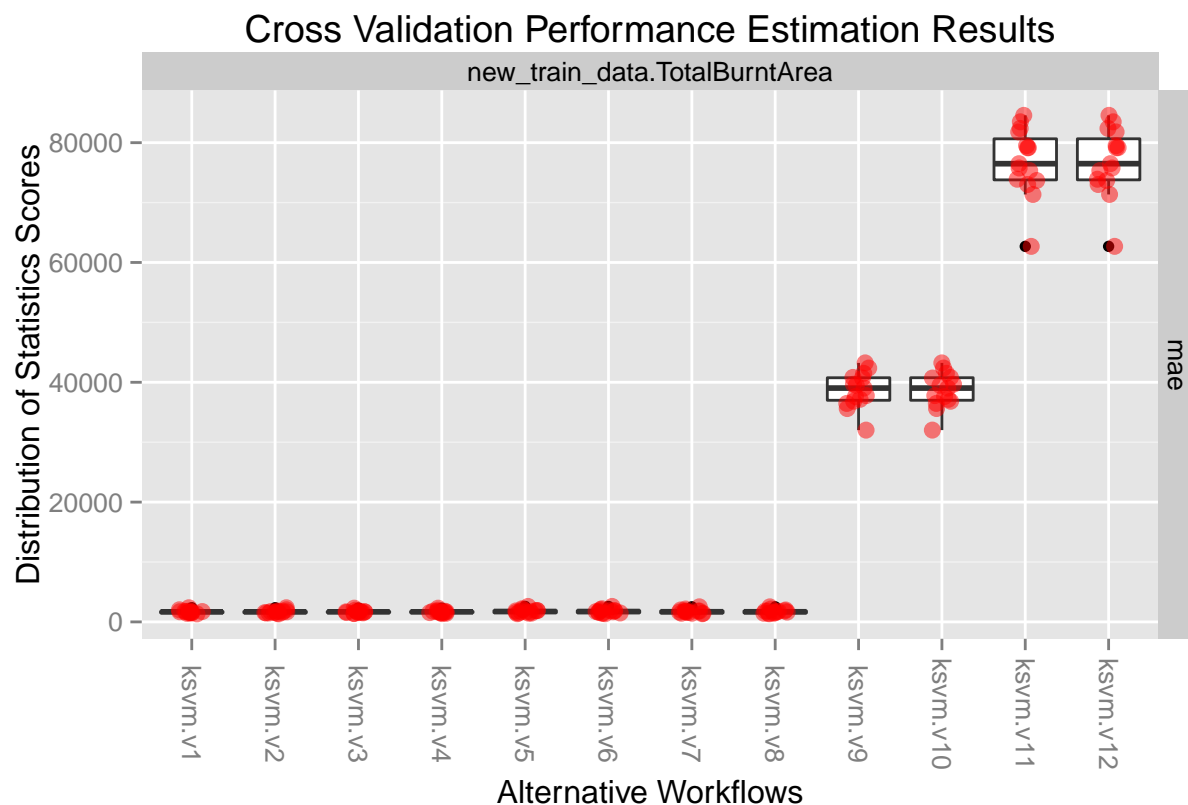


cost	gamma
1	0.01

Table 7: SMV module Parameters

The best MAE value obtained with this modules is: 1728.543.

It was made a different approach using SVMs package **ksvm** (Karatzoglou et al. 2004) which allows us to take advantage of more parameters and a greater variety of kernels.



epsilon	C	kernel
1e-09	1	rbfdot

Table 8: KSVM module Parameters

The best MAE value obtained with this modules is: 1686.583.

## Artificial neural networks(ANNs)

### Description

Artificial neural networks are models with a strong biological inspiration composed by a set of units (neurons) that are connected between them by associated weights.

These basically consist of inputs which are multiplied by weights and then computed by a mathematical function which determines the activation of the neuron ANNs combine artificial neurons in order to process information(Rojas 1996).

### Evaluation

### Assembles

### Conclusions



# Clustering

In this chapter we will explore techniques of clustering.

The goal of clustering techniques finding structure on what we have observed. Clustering is a division of data into groups of similar objects.

Each group, called cluster, consists of objects that are similar between themselves and dissimilar to objects of other groups. Representing data by fewer clusters necessarily loses certain fine details but achieves simplification. Data modeling puts clustering in a historical perspective rooted in mathematics, statistics, and numerical analysis(Berkhin 2002).

## References

- Berkhin, Pavel. 2002. “Survey of Clustering Data Mining Techniques.”
- Karatzoglou, Alexandros, Alexandros Smola, Kurt Hornik, and Achim Zeileis. 2004. “kernlab - an S4 Package for Kernel Methods in R.” *Journal of Statistical Software* 11 (9) (November 2): 1–20. <http://www.jstatsoft.org/v11/i09>.
- Razali, Nornadiah, and Yap B. Wah. 2011. “Power Comparisons of Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors and Anderson-Darling Tests.” *Journal of Statistical Modeling and Analytics* 2 (1) (June).
- Rojas, R. 1996. *Neural Networks: A Systematic Introduction*. Springer Berlin Heidelberg. <http://books.google.pt/books?id=txsjjYzFJS4C>.
- Torgo, Luis. 2010. *Data Mining with R: Learning with Case Studies*. 1st ed. Chapman & Hall/CRC.