Data Mining and Text Mining Project.

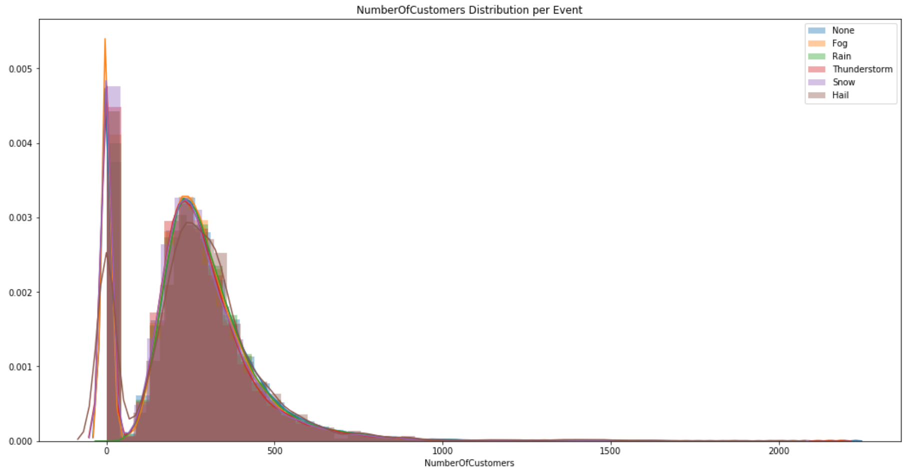
Bellini Alberto - 10455206
Biasielli Matteo - 10482071
Capo Emilio - 10493029
Di Fatta Mattia - 10480719
Di Palo Flavio - 10482911

1. Data Visualization and Exploration.

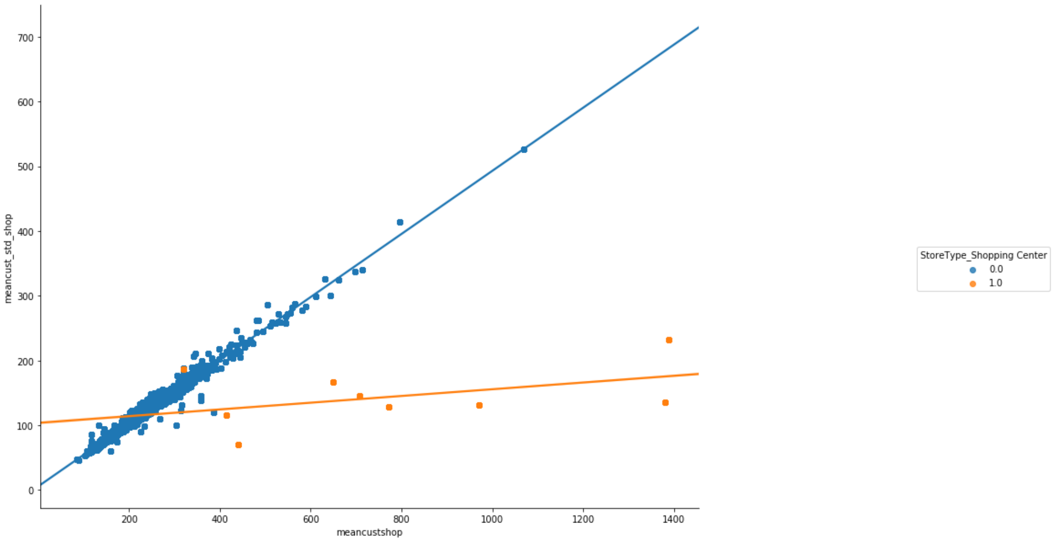
1.1 Visualization

We started by visualizing data in order to understand if some of our main guess about the data trends could be visible, we briefly report some of the main results.

- We plotted the graph that shows the distributions of **NumberOfCustomers** with reference to atmospheric events. This graph led us to conclude that there is no significative difference between the various atmospheric events and **NumberOfCustomers** distribution since no difference stands out in the plot.

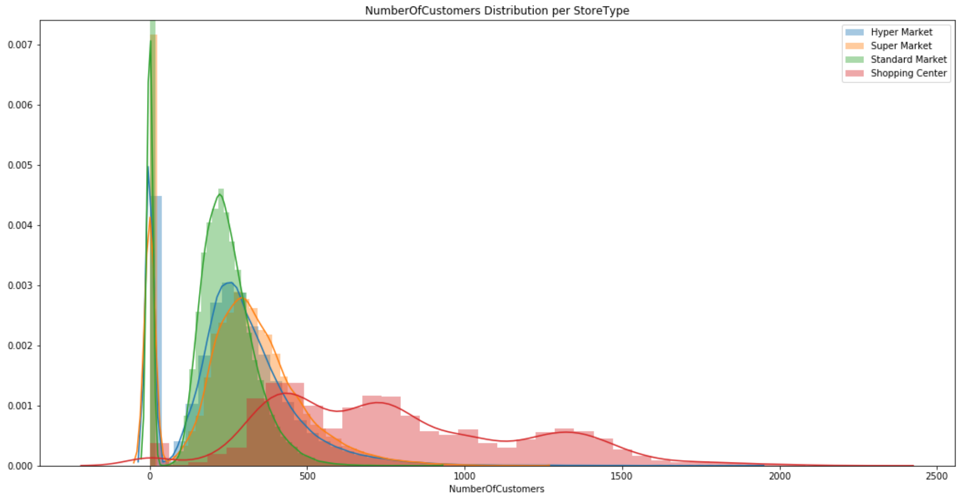


- We plotted the relationship between **NumberOfCustomers** mean and **NumberOfCustomers** standard deviation, with a distinction for each StoreType. We noticed that shopping centers (in orange) are less sensible to the variance of the customers. On the other hand, all other types of store (in blue) result to be more sensitive to the variance of customers. The same trend applied even to the **NumberOfSales**.
  Therefore we decided to enrich the dataset with this additional information (mean and std. deviation) and noticed significant results.



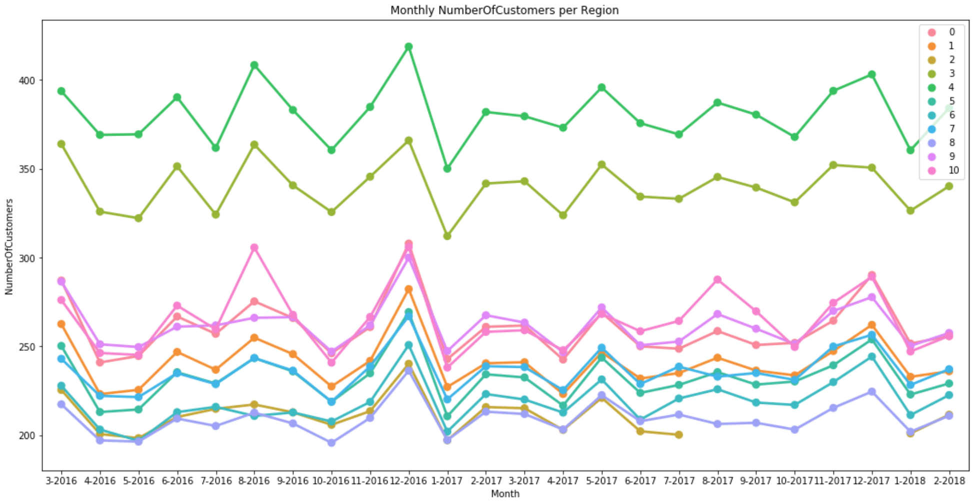
- Throughout more data exploration we didn’t notice any significant relationship neither between **NearestCompetitor** and **NumberOfCustomers** nor **NearestCompetitor** and **NumberOfSales**.

- We plotted the **NumberOfCustomers** distribution per **StoreType**. We saw that all types of stores exhibit a near-gaussian distribution, with the exception of Shopping Centers, which anyway approximately constitute only the 1.2% of the stores in the dataset, so we did not take this difference in consideration while building our model.



- We wondered whether or not promotions affected the **NumberOfCustomers** distribution for each store type. From the exploration we performed it appeared that promotions benefit all types of stores but the **Shopping Centers**.

- We noticed that **NumberOfCustomers, NumberOfSales** and all other data are missing for the data points regarding Region 2 from August 2017 on.



1.2 Feature Extraction

We performed many techniques for feature extraction in order to identify which of the features present in the dataset could effectively be more significant.
We applied **Reduced Variance Feature Selection** in order to obtain a reduced set of features. From that technique we obtained 33 Features that we mainly used in the next parts of the process.
We applied **Random Forest** and used its scoring to select the attributes that have more importance, we also plotted the ranking for the most important attributes.
Initially we performed **PCA** in order to obtain 4 principal components used to train simple models like linear and polynomial regression, the PCA improved results on these simple models but then we noticed that the

results obtained by that model could be outperformed by other model that are further explained in detail in the next pages.

We also noticed that models described in the following sections were able to perform implicit feature selection so we didn't notice a significant improvement in using the above mentioned feature selection techniques.

2. Imputation and dataset enrichment

The original training set needed to be imputed since some attributes had a significant number of missing values.

For this reason attributes like "**Max_VisibilityKm**", "**Mean_VisibilityKm**" and "**Min_VisibilitykM**" were imputed with their mean value, whereas others, like "**Events**" and "**CloudCover**", were assigned respectively "none" and "0" where values were missing.

However imputation did not solve all the problematics related to our dataset since was not suited for a particular attribute, namely "**Max_Gust_Speed_h**", because more than 78% of its values were missing. Therefore we decided to discard the latter since it couldn't be imputed and the available 22% wouldn't have provided enough information to work with.

Nevertheless, since regression models requires numerical inputs, we turned into binary attributes some relevant categorical attributes,r such as "**Events**" and "**StoreType**", throughout one hot encoding.

After this whole imputation process we were able to get our model defining a first prediction baseline for the customers and the sales, but we soon realized there was still room for improvements.

Thus, we started reasoning on what kind of dataset enrichment would have lead us to a performance increase with respect to the accuracy of predictions.

Eventually we understood that enriching the dataset with **means** and **standard deviations**, plus other additions, such as regional or monthly information, would have boosted the accuracy of the predictions and therefore, once for the sales and once for the customers, we added the following fields:

- Mean and standard deviation (respectively for sales or customers) for each shop.
- Max and min values (respectively for sales or customers) for each shop.
- Mean (respectively for sales or customers) for each shop for each month.
- Mean (respectively for sales or customers) for each region for each month

After these additions we noticed significant improvements in accuracy and discovered that models such as Decision Regressor Trees made great use of this new attributes and chose them as first splitting points in several different cases.

3. The model

At this point, the samples in the enriched dataset can be considered independent, since they already contain information about the "history" of the shops in different conditions (statistics per shop, per day of the week and per region).

We thought that the prediction of the **NumberOfSales** day-by-day would have been more accurate if we had the **NumberOfCustomers** available.

For this reason we decided to follow two different approaches, building two separate models before deciding which one to use:

- The first one makes a direct prediction of the **NumberOfSales**, starting from the enriched training dataset.
- The second one is a "pipeline" that first predicts the **NumberOfCustomers** and then proceeds with the prediction of the **NumberOfSales**, using as input the enriched training dataset plus the customers predicted in the previous stage.

No one of the models significantly outperformed the other one and the paired t-test we applied on their performances confirmed that the slight difference in the average performances of the two models is not significant. However, the second model has a slightly lower variance on the predictions, so we decided to use that one in the end.
The prediction has to be done day-by-day but the overall performances are evaluated on an aggregation of the predicted data, so some of them will most likely compensate each other.
Our workflow to train the models is then structured as follows:

- the training dataset enriched with statistics about the **NumberOfCustomers** (and obviously without the **NumberOfSales** variable) is repeatedly subsampled (Bootstrapping) to train several DecisionTreeRegressor models, that are then used to predict the **NumberOfCustomers**.
 The final prediction is the arithmetic mean of the outcomes of all the trees;
- the training dataset enriched with statistics about the **NumberOfSales** and taking in consideration the real **NumberOfCustomers** is repeatedly subsampled (Bootstrapping) to train several DecisionTreeRegressor models.The final prediction is the arithmetic mean of the outcomes of all the trees. In this step we aimed to obtain a model that could accurately predict the **NumberOfSales** knowing the real **NumberOfCustomers** for the period

4. Results

It’s important to notice that, when evaluating the model, in order to guarantee the most accurate simulation possible, the enrichment on the test set is done using statistics (as mean and variance for **NumberOfCustomers)** calculated on the train set and the target variables of test set are used only to evaluate the performances. This is necessary in order to obtain an unbiased performance estimation.
To estimate the overall performances, after dropping the **NumberOfCustomers** and **NumberOfSales** variables from the test set, the previously trained model is used to make the predictions and the original target is used only for the evaluation.
The results are then aggregated and the final error (mean per-region error) can be evaluated using the provided error measure.
The final predictions that we submit are obtained by re-training the model on the entire train dataset.

Before diving into numerical results, the following facts have to be taken into account:

- though this is a regression problem, R2 metric is not forcibly a good indicator. The “Total Error”(mean of per-region errors) is evaluated on an aggregation of the predictions, so it is theoretically possible to have a really low R2 measure but also a really low final error and this would happen if mistakes made in the predictions perfectly balance each other. This is however really unlikely to happen so from now on we will talk about both the measures (R2 and the final per-region error);
- the train dataset lacks 6 months of data for Region 2 (precisely, from 07/2017 to 12/2017). This, considering the enrichment we did on the dataset, prevented us from doing full cross validation. We were however able to evaluate the model on five different train/test splits with each test split containing two consequent months and and we are quite confident in giving the following results.

The average R squared of the model is 0.87 and this is calculated using the day-by-day predictions.

The average Total Error is 5.4% and its standard deviation is 0.86% computed on the five splits.
In particular, the latter (which is the final evaluation metric that was requested by the company), ranges from 3.6% to 6.6% considering the 5 splits.