ROSSMANN STORES

Sales Forecasting

NADIM SAAD

FEB, 11th | PARIS

DAFT NOV21

Business case

About the company

Rossmann is the 2nd largest drug store chain in Germany, founded in 1972 by Dirk Rossmann. It operates over 3,000 drug stores in 7 European countries, its HQ is situated in Burgwedel. The drugstore chain employs more than 28,000 people.

Rossmann Stats

Rossmann offers over 17,500 different items in its biggest retail outlets. Besides the pharmaceutical goods, they have pet food, healthy food and a large selection of wines. 800 of their drug stores offer toys and stationery as well.

Problem

Rossmann wishes to predict daily sales and the number of customers instead of relying only on their stores managers intuitions by creating effective sales forecasts based on a machine learning model aiming to minimize prediction errors.

By helping Rossmann create a robust prediction model to deal with objective data, I will be helping store managers stay focused on what's most important to them: their customers and their teams!

Plan

- **Data Mining, Data Preparation, Data Analysis (EDA)**
  - Data Exploration, Data Wrangling, Data Cleaning, …
  - EDA (*Exploring the Data and Analysing it*)
  - Data Correlation
  - Data Encoding
- Determine which Machine Learning (ML) models to use
- Compare my models
- Conclusion
- Prepare the visualisation on Tableau

Models

- **Since we are dealing here with a Sales Forecast case, I decided to implement the following ML models:**
  - *Linear Regression*
  - *Lasso Regression*
  - *Decision Tree Regressor*
  - *Prophet Regression*

About Data

- **I have been provided with 2 datasets:**

data_impr.csv

Rows: 1,017,209

Columns: 9

Column	VarType	Description
Store	Int64	1115
DayOfWeek	int64	**From** 1 to 7 (each day of the week)
Date	object (YYYY-MM-DD)	**From** 2013-01-01 **to** 2015-07-31
Sales	Int64	Daily sales (5,873,180,623)
Customers	Int64	Stores customers (644,041,755)
Open	Int64	**Indicator:** 0 = closed, 1 = open
Promo	int64	**Indicator**: 0 = no promo 1 = promo
StateHoliday	object	**Holiday Indicator**: a = public, b = Easter, c = Christmas, 0 = None
SchoolHoliday	Int64	**Indicator:** 0 = no, 1 = yes

store.csv

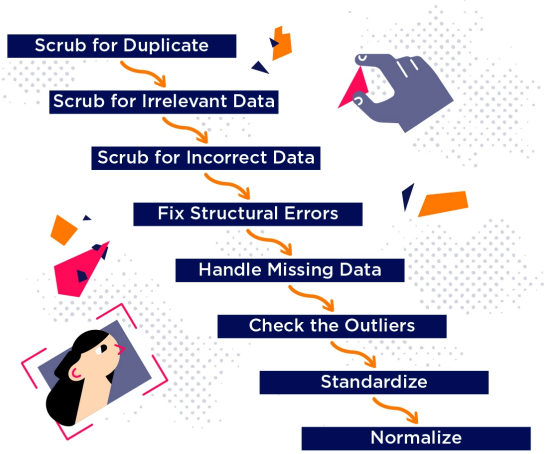
Rows: 1,115

Columns: 10

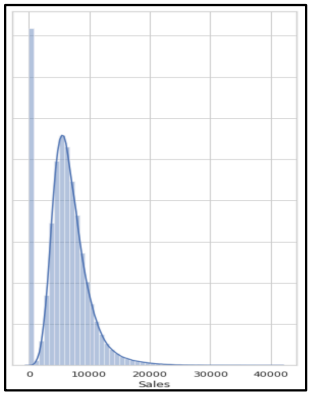
Column	VarType	Description
Store	Int64	1115
StoreType	object	Differentiates between 4 store models: a, b, c, d
Assortment	object (YYYY-MM-DD)	**From** 2013-01-01 **to** 2015-07-31
CompetitionDistance	float64	**From** 20 **to** 75,860 meters
CompetitionOpenSinceMonth	float64	12 values ranging from 1 to 12 (months)
CompetitionOpenSinceYear	float64	**From year** 1900 **to** 2015
Promo2	int64	**Indicator:** 0 = store is not participating, 1 = store is participating
Promo2SinceWeek	float64	**Promo2 indicator:** from 1 to 50 weeks
Promo2SinceYear	float64	**Promo2 indicator:** from year 2009 to 2015
PromoInterval	object	**Indicator:** Jan,Apr,Jul,Oct Feb,May,Aug,Nov or Mar,Jun,Sept,Dec 106

Data cleaning, data preparation

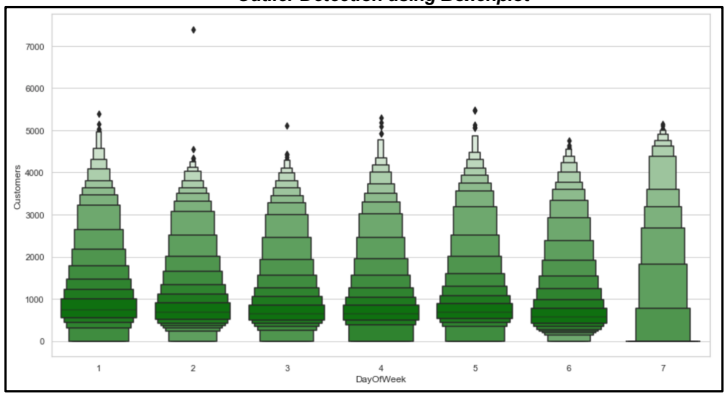
- **First, I merged the 2 datasets *(on store 'Id') and then:***
  - *Scrubbed for Duplicates looking for Irrelevant Data (i.e.datetime)*
  - *Handle Missing Values (i.e. null values)*
  - *Check for Outliers (i.e. Sales and Customers)*
  - *Convert data types (i.e. date to Year, Month, Week, Day, Season columns)*
  - *Created new columns (i.e. Avg CustomersPerWeek, PromoCountPerWeek)*



Outlier Detection using distplot



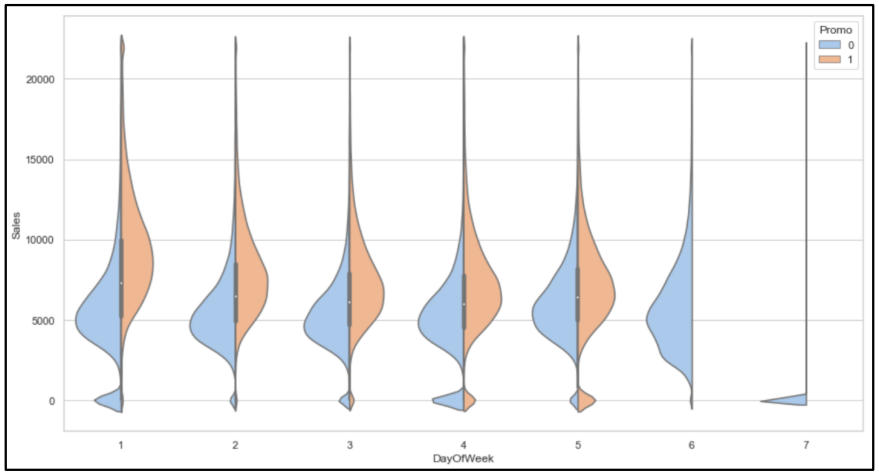
Outlier Detection using Boxenplot



EDA

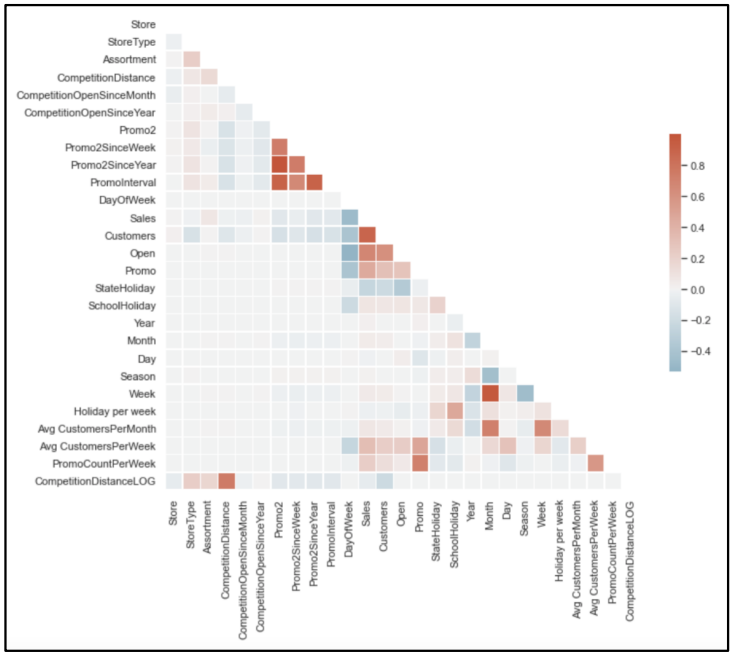
- **I analysed as per the Business Case prerequisites:**
  - *Finding the locations of the Rossmann Stores (timeanddate.com)*
  - *Analysed the effect of promotions on sales and visiting customers*
  - *Checked the impact of nearby competitors on sales*
  - *Verified the affect on school holidays on sales*

Checking the impact of promotions using Violinplot



Correlation

- **This analysis helped me reveal meaningful relationships among my variables**



Models

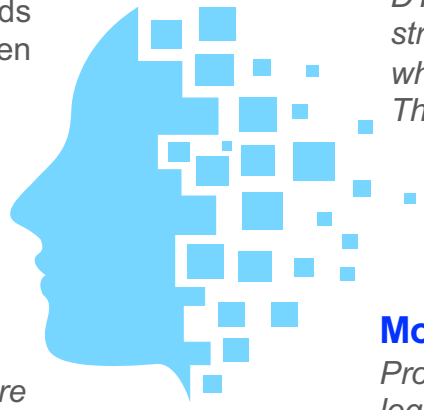
- **For my modelling task, I decide to use 4 different types of models to predict my forecasts:**

Model # 1: Linear Regression

LR is Supervised Machine Learning model which finds the best fit linear regression (relationship line) between the independent (X) and dependent (y) variables

Model # 3: Decision Tree Regressor

DTR builds regression models in the form of a tree structure. It breaks down a dataset into small subsets while developing in parallel an associated decision tree. The final result is a tree with decision and leaf nodes.



Model # 2: LASSO* Regression Model

LR is a type of LR model that uses shrinkage (where data values are shrunk towards a central point, like the mean). The lasso is good for simple/ sparse models with fewer parameters

Model # 4: Prophet Regression Model

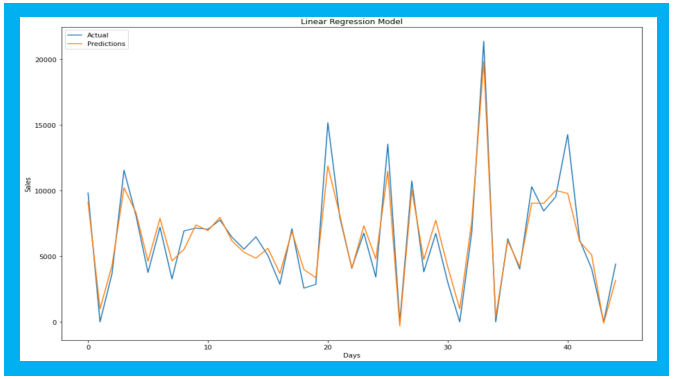
Prophet is an additive RM with a piecewise linear or logistic growth curve trend. It includes a yearly seasonal component model using series & a weekly seasonal component model using dummy variables

- *LASSO stands for Least Absolute Shrinkage and Selection Operator*
- *ARIMA stands for Auto Regressive Integrated Moving Average*

Models evaluation

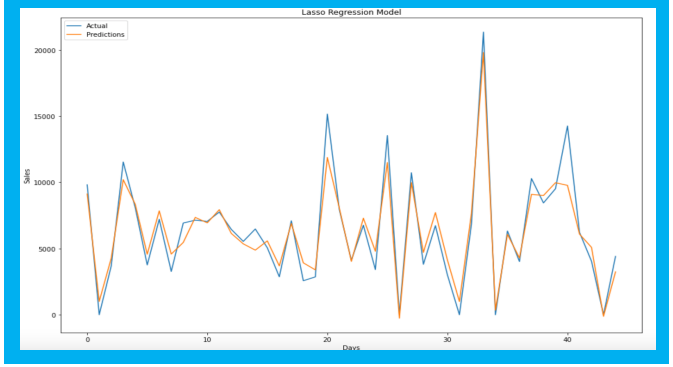
X_train.shape, y_train.shape, X_test.shape, y_test.shape
(813767, 27) (813767,) (203442, 27) (203442,)

Model # 1: Linear Regression



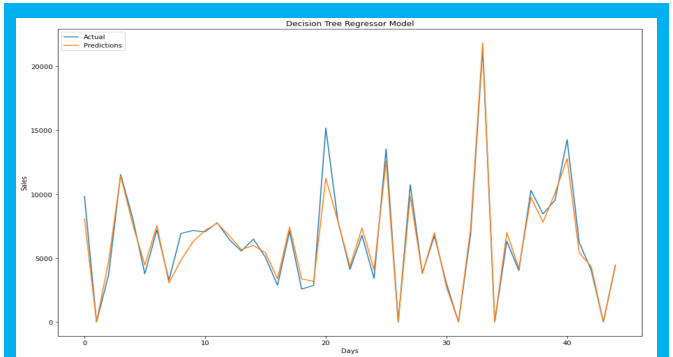
Model Score:
0.874293

Model # 2: LASSO Regression



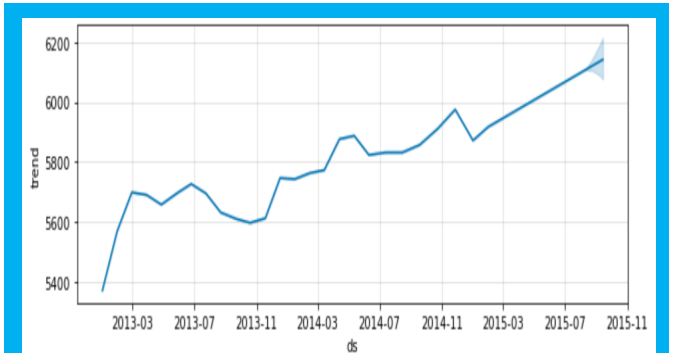
Model Score:
0.874151

Model # 3: Decision Tree Regression



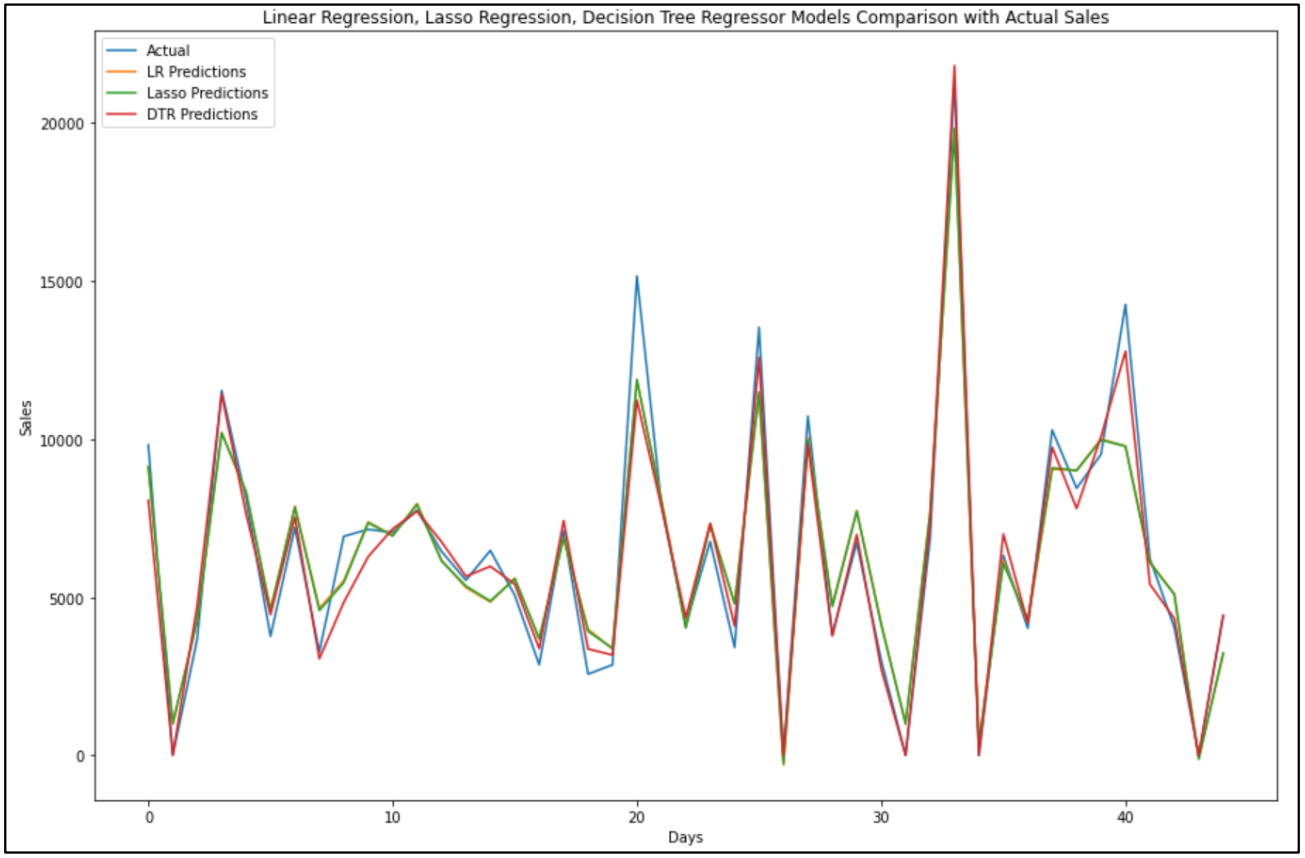
Model Score:
0.951652

Model # 4: Prophet Regression Model



Model Score:
0.868187

Results



Model	Actual	LR Predictions	Lasso Predictions	DTR Predictions
0	9,815	9,132	9,124	8,063
1	0	992	1,006	0
2	3,680	4,295	4,265	4,738
3	11,539	10,209	10,200	11,425
4	8,092	8,300	8,346	7,637

Prophet Model Results

	horizon	mse	rmse	mae	mdape	smape	coverage
0	3 days	9.746230e+06	3121.895307	2382.156933	0.319108	0.587139	0.868187
1	4 days	1.349443e+07	3673.476637	2679.528251	0.533699	0.879356	0.812321
2	5 days	1.562634e+07	3953.015989	2897.896543	0.753578	1.070386	0.763402
3	6 days	1.758346e+07	4193.263524	3245.220741	0.661657	0.920089	0.682275
4	7 days	1.734772e+07	4165.059941	3384.556228	0.649282	0.738014	0.653769

Furthermore

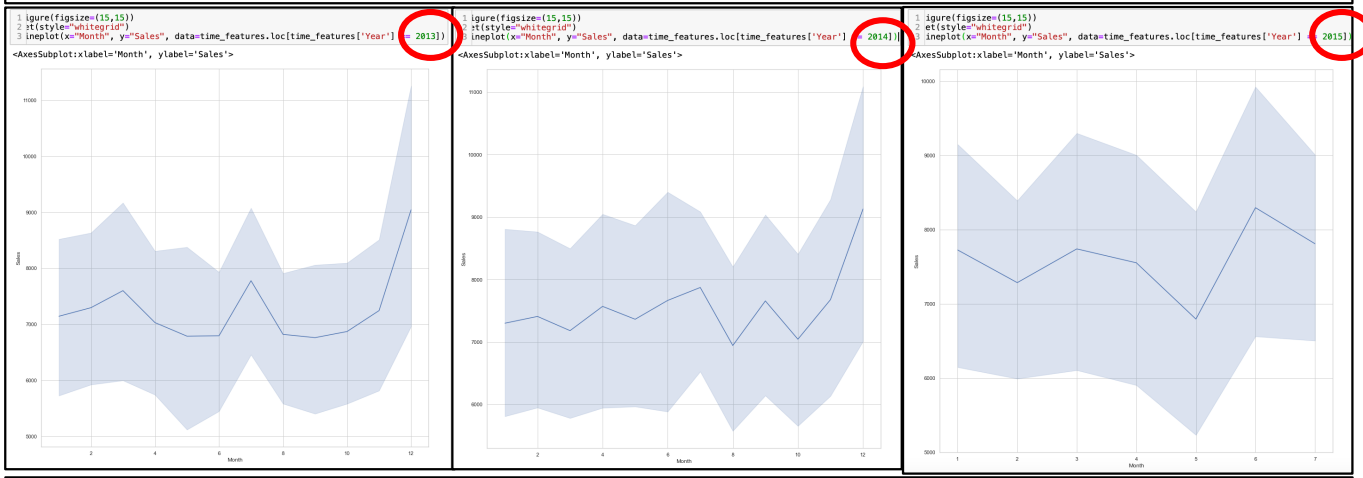
~~Model # 4: Time Series ARIMA Model~~**

Most popular/ widely used statistical method for Time Series forecasting. This model makes forecasts based on linear combinations of past variable values and forecasting errors

- **Time Series Analysis**

```
import random
time_features=df[["Year","Month","Day","Week","Season", "Promo", "Open","Store","Sales","Date"]]

#Choose a random store
x=random.randint(1,115)
print(x)
time_features=time_features.loc[time_features['Store'] == x]
```

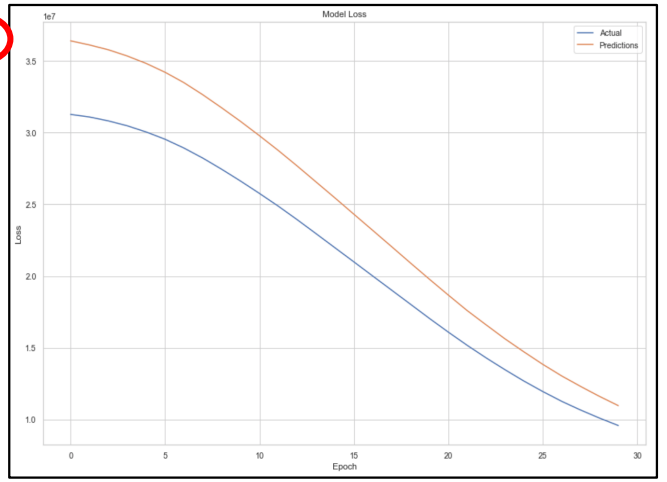


```
import tensorflow as tf
import keras
import keras.backend as kb
```

```
time_model=keras.models.Sequential()
```

```
6/6 [==============================] - 0s 9ms/step - loss: 9779709.0000 - mean_absolute_error: 2936.0776
loss : 9779709.0
mean_absolute_error : 2936.07763671875
```

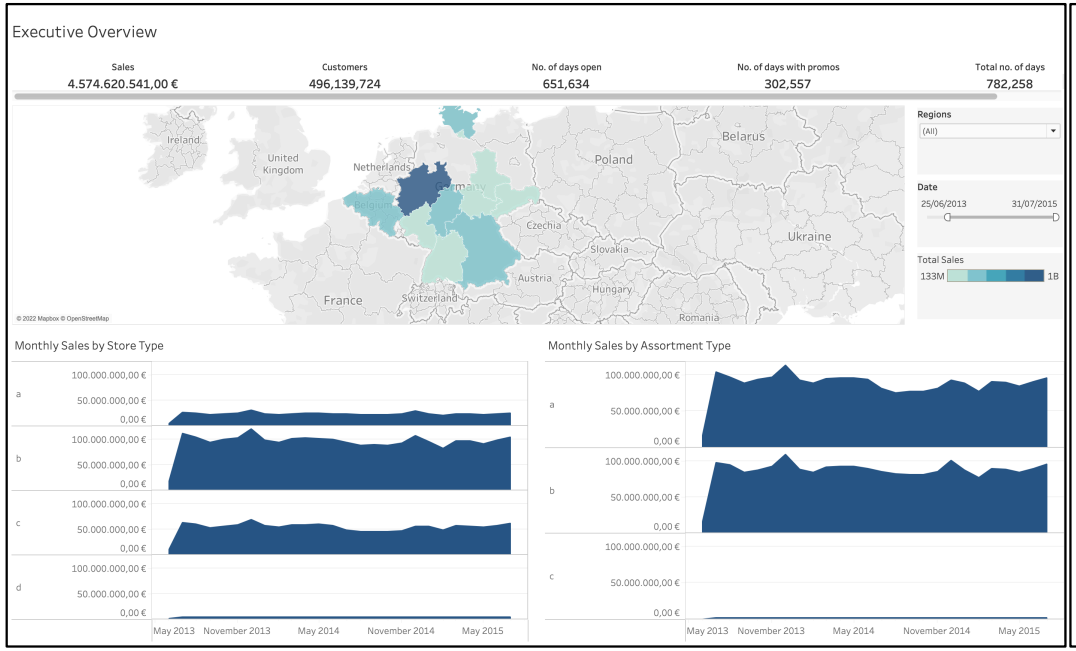
Loss: A scalar value that we try to minimize during our training of the model. Lower the loss, closer our predictions are to true labels. In Keras, a decrease in loss over n epochs



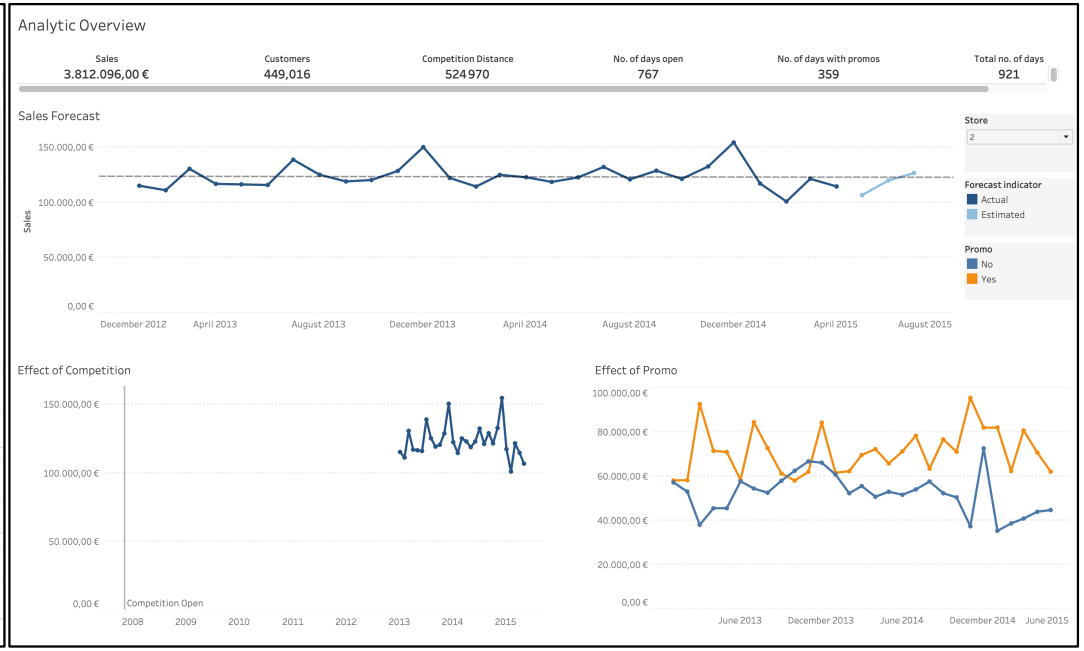
Conclusion

*The **Decision Tree Regressor (DTR) model** performed best with a **score of 95%** when comparing it with the other algorithms that I chose. Hence, I used the DTR model since it is my best suitable algorithm for predicting Rossmann Stores sales forecasts.*

Visualization on Tableau



https://public.tableau.com/shared/K7N79TQ6B?:display_count=n&:origin=viz_share_link



https://public.tableau.com/shared/98BF34FZB?:display_count=n&:origin=viz_share_link

Rossmann Sales

Nadim, Saad

Questions?

THANK YOU.

IRON HACK
PARIS