**ROSSMANN STORE SALES**

-Kruthika Potlapally

**1. Introduction**

Sales forecasting is one of the most important factors that contribute towards the success of a company, be it a small, medium or a large scale organization. With the constantly evolving market trends, refining one’s sales and marketing strategy becomes extremely vital in order for survival. Every company, regardless of the type of product or service it provides has planning departments, econometric models, customer polls, sales pipelines, quotas, and commitments that need resolute refinement. The publicly traded companies too have to forecast for stock analysts, and when they miss their projections, their stock prices suffer.

The entire sales forecasting process is intended to not only generate better business results but also help the management make better business decisions that are in alignment with the company’s business goals and objectives.

Timely identification of newly emerging trends, sales patterns of customer segments, upward and downward trends in sales and predictive modeling are some of the few study areas that can be used to identify new market trends. They help develop a better understanding of overall sales trends of the organization and ultimately devise more efficient marketing strategies for the growth of the company, customer retention and expansion of existing client base.

**2. Problem Statement**

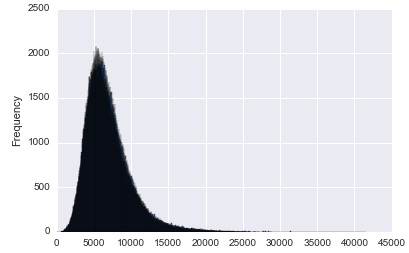
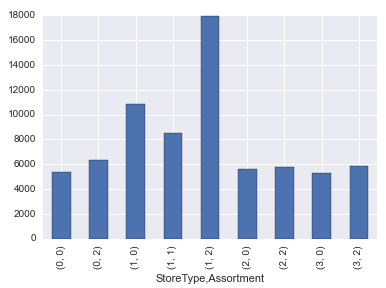
I worked on the dataset of a drugstore, Rossmann. Rossmann is Germany's second-largest drug store chain, with over 3,000 stores in Europe. The problem statement involves the prediction of their daily sales for 1115 stores for a period of 6 weeks. Store sales are influenced by various factors, including promotions, competition, holidays, seasonality, and locality.

I took the data set from Kaggle. I chose to work on this project as I have never been involved with a business problem before. Hence, I believed that working on this project would be a good learning experience and a good exposure to the field of sales and marketing. The data set contains training, testing and store data. Training data represents historical data of 1115 stores including sales; testing contains historical data without sales and store data contains information about the stores. The training set contains over 1 million rows representing the sales of each store on a particular day (starting from January 2013). The store data contains 1115 stores representing data of each store.

|  |  |  |
| --- | --- | --- |
| Train set | Store set | Test set |
| Shape – (1017209, 9) | Shape - (1115, 10) | Shape - (41088, 8) |
| 1. Store 2. DayOfWeek 3. Date 4. Customer - the number of customers on a given day 5. Open - an indicator for whether the store was open: 0 = closed, 1 = open 6. Promo- indicates whether a store is running a promo on that day 7. StateHoliday- indicates a state holiday. Normally all stores, with few exceptions, are closed on state holidays. Note that all schools are closed on public holidays and weekends. a = public holiday, b = Easter holiday, c = Christmas, 0 = None) 8. SchoolHoliday- indicates if the (Store, Date) was affected by the closure of public schools | 1. Store 2. Store Type - differentiates between 4 different store models: a, b, c, d 3. Assortment - describes an assortment level: a = basic, b = extra, c = extended 4. CompetitionDistance - distance in meters to the nearest competitor store 5. CompetitionOpenSince   [Month/Year] - gives the approximate year and month of the time the nearest competitor was opened   1. Promo2 - is a continuing and consecutive promotion for some stores: 0 = store is not participating, 1 = store is participating 2. Promo2Since[Week/   Year] - describes the year and calendar week when the store started participating in Promo2   1. PromoInterval - describes the consecutive intervals Promo2 is started, naming the months the promotion is started anew. E.g. "Feb,May,Aug,Nov" means each round starts in February, May, August, November of any given year for that store | 1. Same as train set excluding sales and customer. |

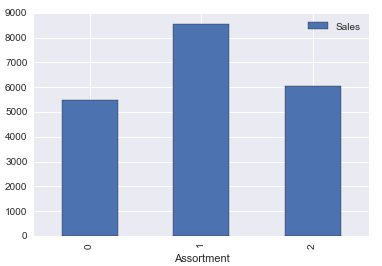
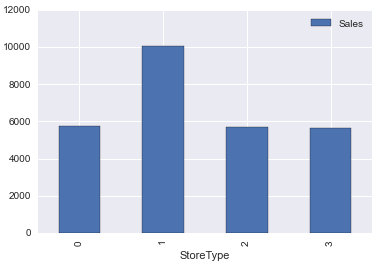
**3. Data Exploration and Visualization**

I first merged the data sets of training and sales. I explored the data based on various plots using matplotlib. I visualized the store sales based on the stores, assortment levels, days of the week, competition, promotion and holidays. I performed basics statistics such as mean and median calculations for data exploration. Some of the plots are depicted below-

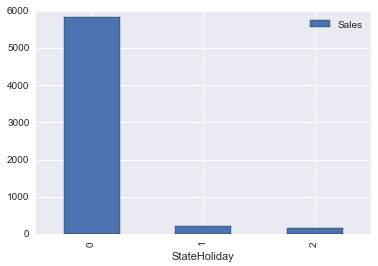
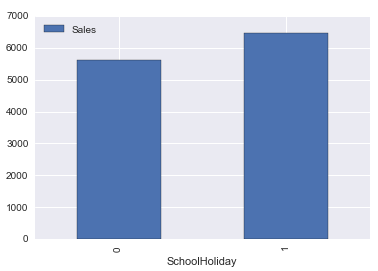
 

Sales

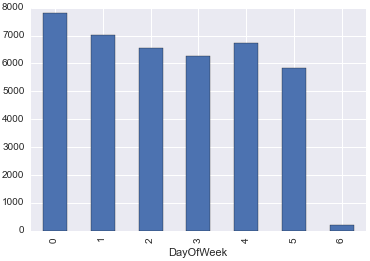
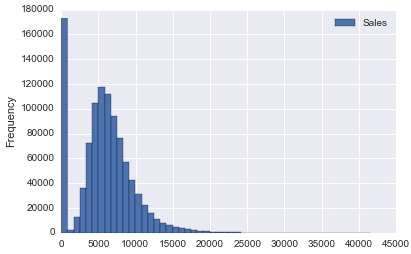
3.a. Sales vs Frequency plot 3.b. StoreType, Assortment vs Sales plot

3.b. Assortment vs Sales plot 3.c. StoreType vs Sales plot

3.d. StateHoliday vs Sales plot 3.e. StateHoliday vs Sales plot

3.f. DayOfWeek vs Sales plot CompetitionDistance

3.d. CompetitionDistance vs Sales plot

From the above plots, it can be inferred that store types, assortment level, holidays, day of the week and competition distance influence the sales.

**4. Data pre-processing**

I checked for null values in the train and store data sets using ‘isnull().sum()’ command. The train data set had no missing values, however there were 0 sales and customers for 52 open stores. I assumed that all the stores are open and sales are greater than 0, hence filled 1 in place of null values in the Open and Sales columns. In the store data set, there were 3 missing values in CompetitionDistance column, 354 in CompetitionOpenSince[Year/Month] columns and 544 in Promo2Since[Year/Week] and PromoInterval columns. I considered the most recent values since the start of competition and promotion2 periods from the data set and filled in the missing values with them. I filled in the CompetitionOpenSince[Year/Month] with 2015/8 and Promo2Since[Year/Week] with 2015/22 as these were the most recent. I filled in the PromoInterval with maximum occurring interval i.e., Jan,Apr,Jul,Oct. I assumed these values as the data sets might not have been properly updated on the recent changes. I filled the missing values in CompetitionDistance column with its mean. I extracted date, month, year, week of year, day of week from Date column; and dummy values for categorical variables – assortment, store type, stateholiday and promo months. I used feature engineering to find the number of months since the start of competition and number of weeks since the start of Promo2.

**5. Model creation**

Sales is a continuously changing process, hence, I used linear regression to model the features. The null RMSE is 3849.9242. I used train\_test\_split and cross validation on training set to obtain Root Mean Square Errors (RMSE). Based on the above exploratory analysis and visualizations, I used different sets of features and computed their RMSE scores. Some of the features and their corresponding RMSE values are shown below-

|  |  |
| --- | --- |
| Features | RMSE |
| 1. 'CompetitionDistance', 'StoreType', 'Assortment','StateHoliday','Promo', 'Promo2', 'SchoolHoliday', 'DayOfWeek', 'month', 'WeekOfYear', 'CompetitionOpen', 'PromoOpen' | 3120.393 |
| 1. 'CompetitionDistance', 'StoreType', 'Assortment','StateHoliday','Promo', 'SchoolHoliday', 'CompetitionOpen', 'PromoOpen' | 4123.681 |
| 1. 'CompetitionDistance', 'StoreType', 'Assortment', 'Promo', 'SchoolHoliday', 'DayOfWeek', 'month', 'WeekOfYear', 'CompetitionOpen', 'PromoOpen' | 3138.671 |

From the above table, it is evident that minimum RMSE value corresponds to the best set features. Furthemore, I wanted to eliminate the over-fitting of linear regression model. I used regularization models to overcome over-fitting due to high variance. I used the features listed in the first row in the above table to implement Ridge and Lasso Regression. I varied the alpha values to determine minimum RMSE values.

1. Ridge Regression:

|  |  |
| --- | --- |
| Alpha value | RMSE |
| 0.1 | 3130.72527851 |
| 0.01 | 3125.74220601 |

1. Lasso Regression:

|  |  |
| --- | --- |
| Alpha value | RMSE |
| 0.001 | 3125.64803694 |
| 0.01 | 3125.86080324 |

**6. Picking the best model:**

From the above results, although the RMSE values are not highly distinct for the three models, it is evident that linear regression is the best model. The order of precedence of models is as follows-

Linear regression > Lasso Regression > Ridge Regression

**7. Conclusion and next steps:**

Based on the sales forecast through my analysis, Rossmann can plan and create effective staff schedules that can boost their sales productivity and motivation. This can help the store managers manage their staff during peak hours and downtime; and stay focused on what’s most important to them: their customers and their teams!

I faced a number of challenges due to the large data sets. The regression algorithms took hours to execute. I initially divided the sales data into various classes with an idea to implement logistic regression, however, it wasn’t feasible due to system failure. I tried implementing decision trees and random forests so as to gain a better understanding of the features, but I encountered memory issues. I would like to implement the models using Apache Spark and hope to gain a better understanding of the problem.