horizontal line

**Pankaj Acharya**

Capstone Project 1 - Milestone Report

Springboard Data Science Career Track

**Predicting Future Sales**

Introduction

To maintain product stocks in stores is challenging for companies making those products. Having the product in stock for too long costs the companies a lot and on the other hand short supply will also lead to lost revenue, let alone customer trust. Maintaining a balance between overstock and understock is a major problem for companies in present cut-throat competition.

1C Company, one of the largest Russian software firms has provided a challenging time-series dataset consisting of daily sales for data scientists to build a model to predict future sales for every product and store in the next month. This will help the company make decision on maintaining stocks for particular items in particular shops optimally.

Goal

The main goal of this project is to predict next month’s sales of items based on historical data to help optimize the stocks.

The Dataset

All the data for my project are downloaded from Kaggle website (<https://bit.ly/2RsJ8AV>). Following files are provided:

1. **sales\_train.csv** - the training set. Daily historical data from January 2013 to October 2015
2. **test.csv** - the test set. I need to forecast the sales for these shops and products for November 2015.
3. **sample\_submission.csv** - a sample submission file in the correct format.
4. **items.csv** - supplemental information about the items/products.
5. **item\_categories.csv** - supplemental information about the items categories.
6. **shops.csv** - supplemental information about the shops.

The dataset is a real-world data set and is fairly large. The training dataset has 2.9 million rows. There were no missing values. It is a time-series data, but due to multiple transactions on the same day makes it challenging.

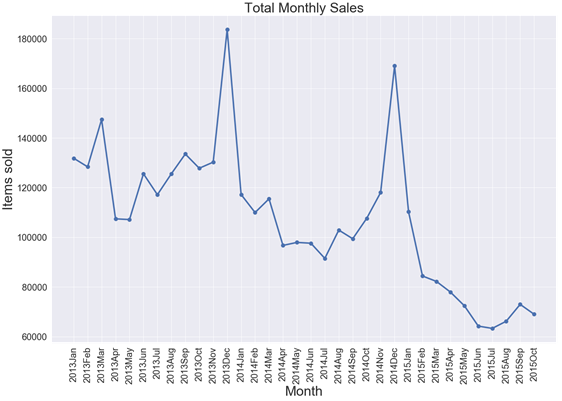
Data Wrangling

The test file contains shop id and item id. Each shop and item combination is given a unique ID and we have to predict the sales of that unique ID (combination of shop and item) next month. The other four files (sales\_train, items, item\_categories and shops) each have information that needs to be combined before proceeding. Figure 1 shows a screenshot of the dataframe after these files were combined.



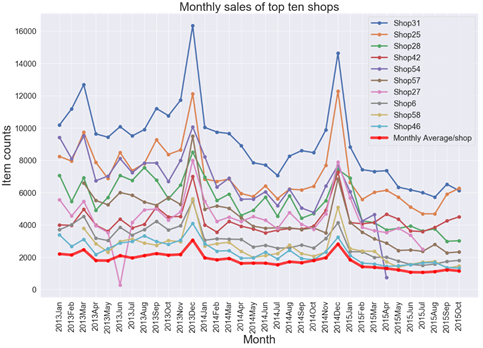
**Figure 1. Screenshot of merged data frames showing all the columns**

Exploratory Data Analysis

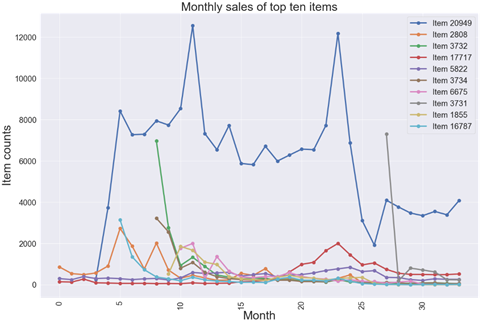
To get an idea of how the items sold throughout the year, I combined the daily sales into monthly sales. The figure 2 below shows that the total sales peaked every December. This is probably due to Christmas and New Year.

**Figure 2: line plot showing distribution of total monthly sales over the given period.**

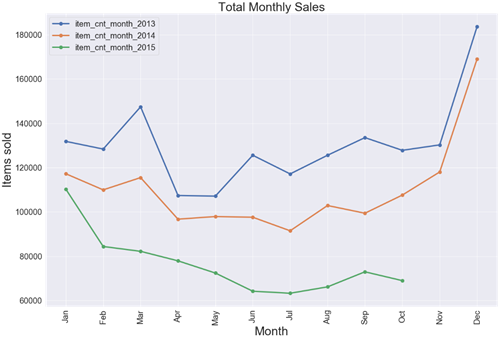
To find out if the trend is similar for the top selling shops or it dominated by few shops, I plotted the total monthly sales for top ten performing shops. The figure 3 shows that in general all the shops have similar pattern over the given period. To further explore if the sales is dominated by a few items, I plotted the total monthly sales of top ten selling items and as shown in figure 4, one item dominates the total sales throughout the 33 month period.



**Figure 3: line plot showing distribution of total monthly sales over the given period for top performing shops.**

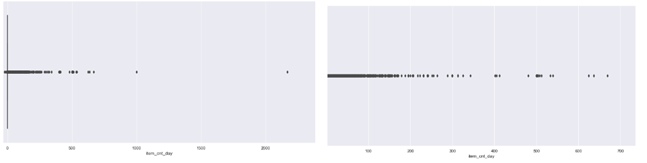


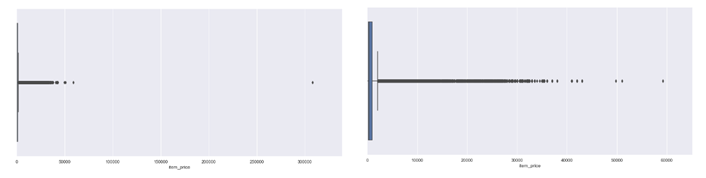
**Figure 4: line plot showing distribution of total monthly sales over the given period for top selling items.**

Finally, there is a clear trend that the sales of items is declining over time. This is shown by figure 1, and figure 5 below.

**Figure 5: line plot showing monthly sales trend each year.**

There are certain outliers in this dataset. The first one is negative value in column item\_cnt\_day (as seen in the figure above). We can also see another outlier in the same column as seen in the figure below:-

**Figure 6: Boxplot showing distribution of the item counts per day before (left panel) and after (right panel) removing the outliers.**

In addition to the outliers in item counts per day, there is also an outlier in price (one item costs over 300,000 while rest are below 70,000). That one is also removed as shown in figure 7 below.

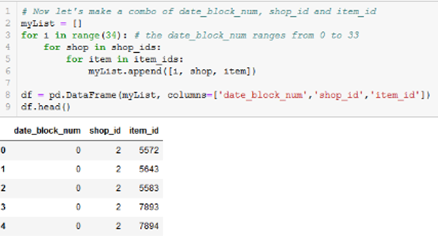
**Figure 7: Boxplot showing distribution of the item price before (left panel) and after (right panel) removing the outlier.**

There were no missing values in the data provided, however, after merging all the dataframes, there were some missing values. This is because some shop IDs, item IDs and item categories did not have any data. In other words, although the list has all the shops, items or item categories, they did not have sales in the given time frame. I removed those and moved to feature selection.

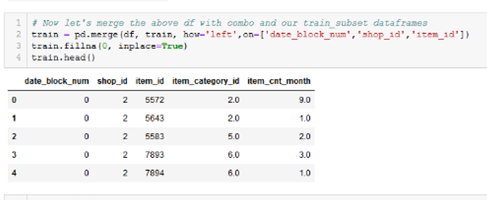
Feature selection

There are following columns in the merged data: data, date\_block\_num, shop\_id, item\_id, item\_price, item\_cnt\_day, item\_name, item\_category\_id, shop\_name and item\_category\_name. Because we are predicting sales for next month and we have date\_block\_num, all the data were pooled for a month and got rid of date column. Furthermore, for the prediction the basic columns needed are date\_block\_num, shop\_id and item\_id, and item\_cnt\_day, all the other features were removed from further analysis because they don’t add any value.

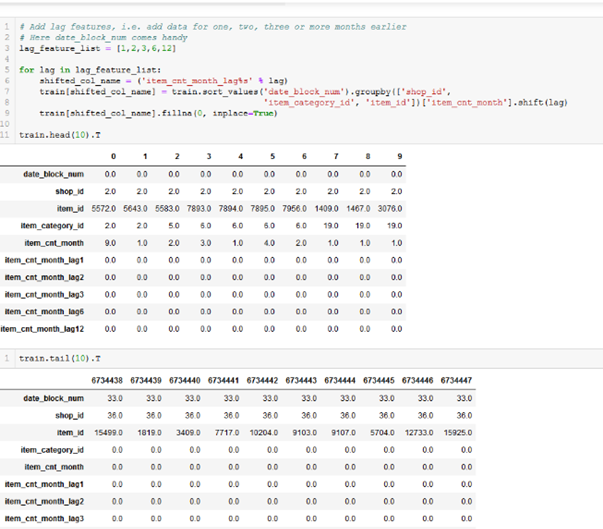
Feature engineering

There are a total of 42 unique shops and 4716 unique items in the test set. Next I made a data frame consisting of all the possible combinations of those shop and items combined with the date block number as there are 34 data blocks provided (0 through 33). After that I merged this data frame with the training dataset and filled the missing values with 0. This is because, certain combination might not exist, suggesting that certain shop-item combination might not have been sold yet. Figures 8 and 9 below show screenshots of the codes used to generate the data.

**Figure 8: Screenshot of the code used to generate data frame of date block num, shop id and item id combinations**



**Figure 9: Screenshot of the code used to merge the data frames**

Next I added lag features for 1, 2, 3, 6 and 12 months in separate columns. This means the sales data for certain combination for one, two, three, six and twelve months ago are added to the existing data for better prediction.

**Figure 10: Screenshot of code used to generate lag features.**

Finally, the first 12 months (date block number 0 through 11) of data were removed because it won’t add much information. Then I took data from month number 12 through 28 for training set and month number 29 through 33 for validation set. For the final prediction I have to process the test data and make it have same number of columns as training and validation sets. Since we need to predict sales for the next month, so I added a column data\_block\_num = 34 in the test set. Then I added lag features as with the training data set. Lastly, I saved these processed files and will use these processed files for modelling.