PROJECT REPORT

Shiying Chen, Jianjun Du, Bo Huang, Yanyan Zhu

CS 6375 Machine Learning
University of Texas at Dallas
December 1st, 2017

Contents

1. Introduction

For traditional grocery stores, the forecast of customers purchase and product sales is very important. It’s a common problem that the grocers face with overstocked goods when they predict a little over or they suffer complains from customers when they predict a little under to cause the popular items are out of stock. The forecast of product sales becomes more complex and difficult when adding some factors such as new locations with unique needs, products with seasonal tastes and unknown product marketing.

Corporación Favorita is a large Ecuadorian-based grocery retailer, which operates hundreds of supermarkets with over 200,000 different products. They built a product sales forecast model which was based on very little data, used subjective forecasting methods and executed plans with very little automation. In order to improve the accuracy of product sales forecast and let customers can purchase right products at the right time, Corporación Favorita holds a prize competition.

The goal of our project is to use machine learning knowledge to accurately predict sales for a large grocery chain. This is a regression problem about predicting unit sales of more than 5000 products in 54 stores of Corporación Favorita.

2. Related Work

This project belongs to a category of predicting retail sales, which is a common scenario for data analysis and mining. Time series models, such as moving average and autoregression. Neural Network has also been successfully applied to this type of models. In Kaggle, there was also a similar competition, Rossmann Store Sales, forecasting sales of a drug store chains. But, most winning models of data science competitions for sales prediction, are boost trees, especially the popular Extreme Gradient Boost algorithm. In this project, we will explore time series models, SVM, random forest, and XGB.

3. Dataset and techniques description

3.1 Dataset Introduction

This Data files include train.csv, test.csv, store.csv, items.csv, transaction.csv, oil.csv, and holiday_events.csv.

train.csv: It includes the target *unit_sales* by *date*, *store_nbr*, and *item_nbr* and a unique id to label rows. The *onpromotion* column tells whether that *item_nbr* was on promotion for a specified *date* and *store_nbr*, and approximately 16% of it's value in this file are NaN.

test.csv: With the *date, store_nbr, item_nbr* combinations that are to be predicted, along with the *onpromotion* information.

stores.csv: It indicates the *type*, location that *city* and *state* of 54 *stores*, and *cluster* variable represents a grouping of similar stores.

items.csv: It indicates the type of *family, class,* and *perishable* for a specific *item_nbr*.

transactions.csv: The count of sales transaction for each *date* and *store_nbr* combination.

oil.csv: The price of oil corresponding to each *date* of both the train and test data.

holidays_events data: Holiday and events for each *date*. The *transferred* column is indicated whether a holiday that is transferred officially falls on that calendar days, but was moved to another date by the government. If transferred, that day is more like a normal day than a holiday.

3.2 Language Description

- Python
-

4. Data Pre-Processing

There are 120 million records in the train.csv, and the size is more than 4G, so training such a big data is impossible.

Steps for preprocessing:

1. Convert the type of data in the different columns. For example, a int64 can be changed to int32, and other categorical features can be changed to int8. This step shrinks the data size to 2 G.
2. There are 54 stores. The assumption is the sale of every stores are independent. As a result, the dataset can be separated to 54 sub datasets.
3. Extract information from the date column. This generates year, month, day of week, and the days from beginning. It is shown from the attached Python codes, all of these factors are impacting the sale.

4. It is said in the competition’s website that oil price has some impact on the economy of the country. Therefore, daily oil price is also added as feature.
5. Change the month, and day of week to categorical features, and create dummy variables.
6. Add feature of store type to the dataset
7. Add feature of promotion to the dataset
8. Add feature of Holidays to the dataset

5. Solutions and Methods

In this project, firstly, we did some simple analysis, such as descriptive data analysis and time series analysis. Then we merged the holiday and oil information to train the dataset. Next, we build three models: 1) Extreme Gradient Boost, 2) Random Forest, and 3) SVM. To compare these three models, we use the criterion of square errors. After we found out the best model, we tried to find out if it’s overfitted and then searched the best parameters for that model. Finally, we used the best model to make prediction and computed the accuracy.

Extreme Gradient Boost: A very efficient implementation of gradient boosting algorithm by utilizing the principles of machine learning data computer science. It is far better than the scikit learn boosting tree, and about the half of the winners of kaggle competition use models built with XGB.

Object function: $obj(\theta)=L(\theta)+\Omega(\theta)$

Loss function: $L(\theta)=\sum_i(y_i-\hat{y}_i)_2$

6. Results and Analysis

In the descriptive data analysis part, we counted there are 4036 items are sold by the company, and the company owns 54 stores. Also, we figured out the difference for each day of week, month, and year.

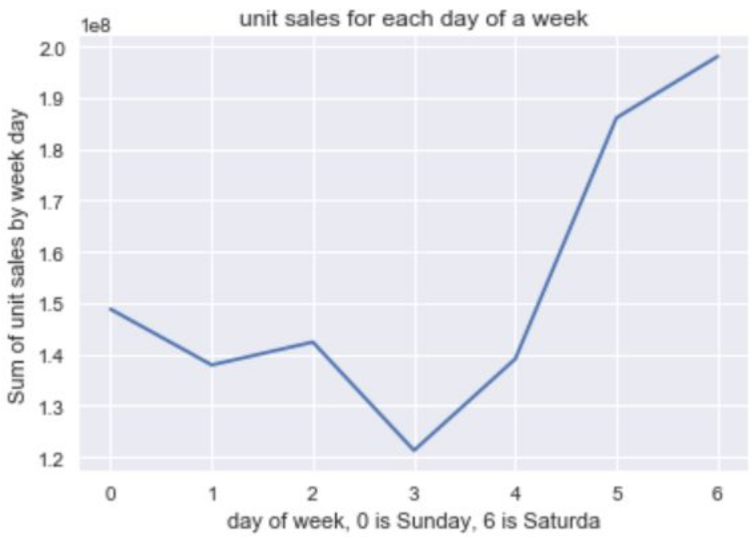


Fig 1. Difference for each day of week.

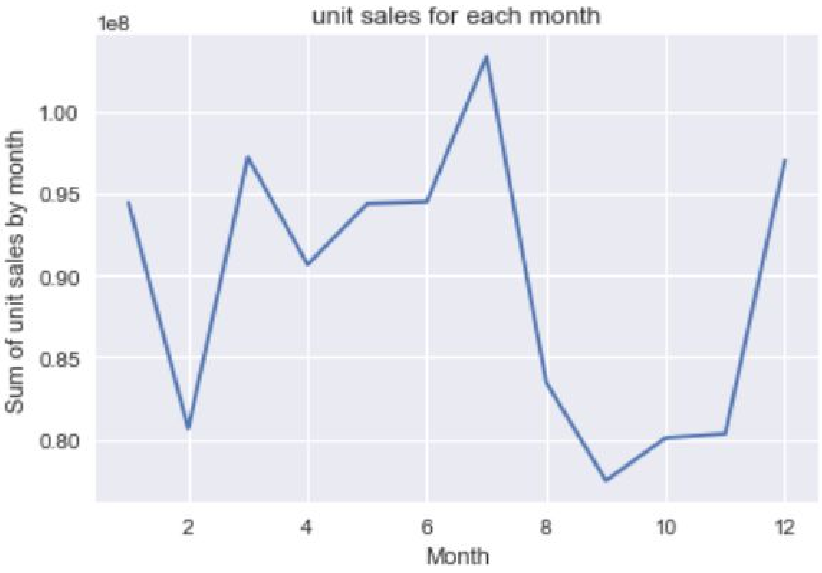


Fig 2. Difference for each month.

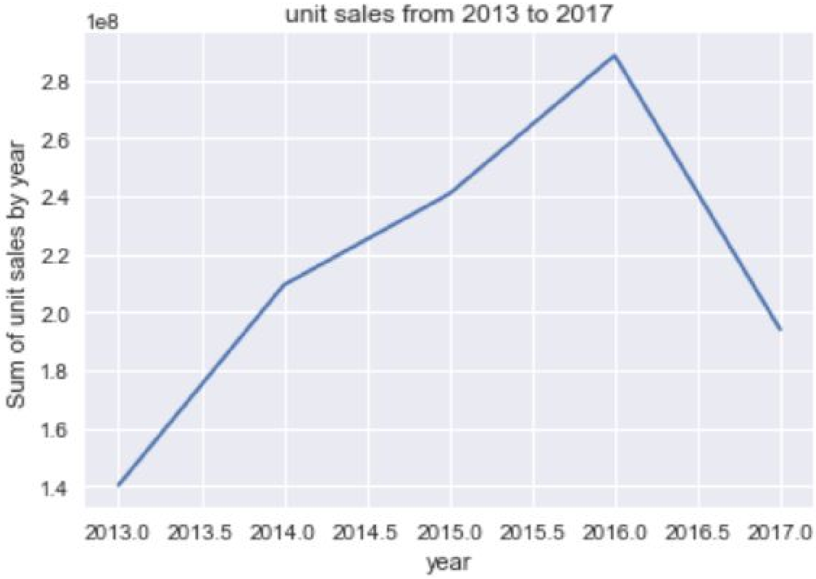


Fig 3. Difference for each year.

The figure 1 is about the difference for each day of week, as we can see, on Friday and Saturday, the sales are much higher than the other days, and Wednesday has the smallest sales. Figure 2 shows the differences between different months. In July, the sum of unit sales is highest, and in September, the sales is the lowest. Also, in the figure 3, we can find out the sales is increasing from 2013 to the beginning of 2016, and after that, it is decreasing until 2017 since the data only cover from 2013 to 2017.

The figure 2 shows that store type A and D always have higher sales compared to store type B, C, E.

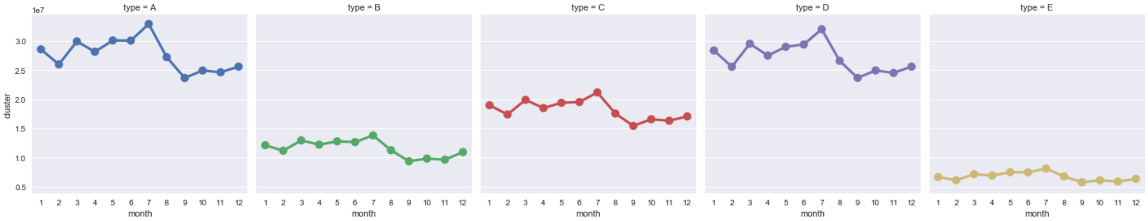


Figure 2: Month and store types

Time Series Models

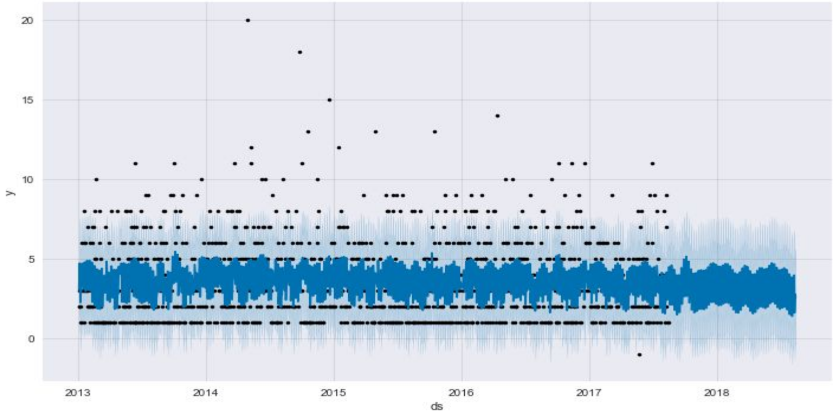
Time series data has some unique characteristics, such as trends, seasonalities. We will do a typical time series analysis first, and then develop a prediction model with prophet, which was open sourced by facebook.

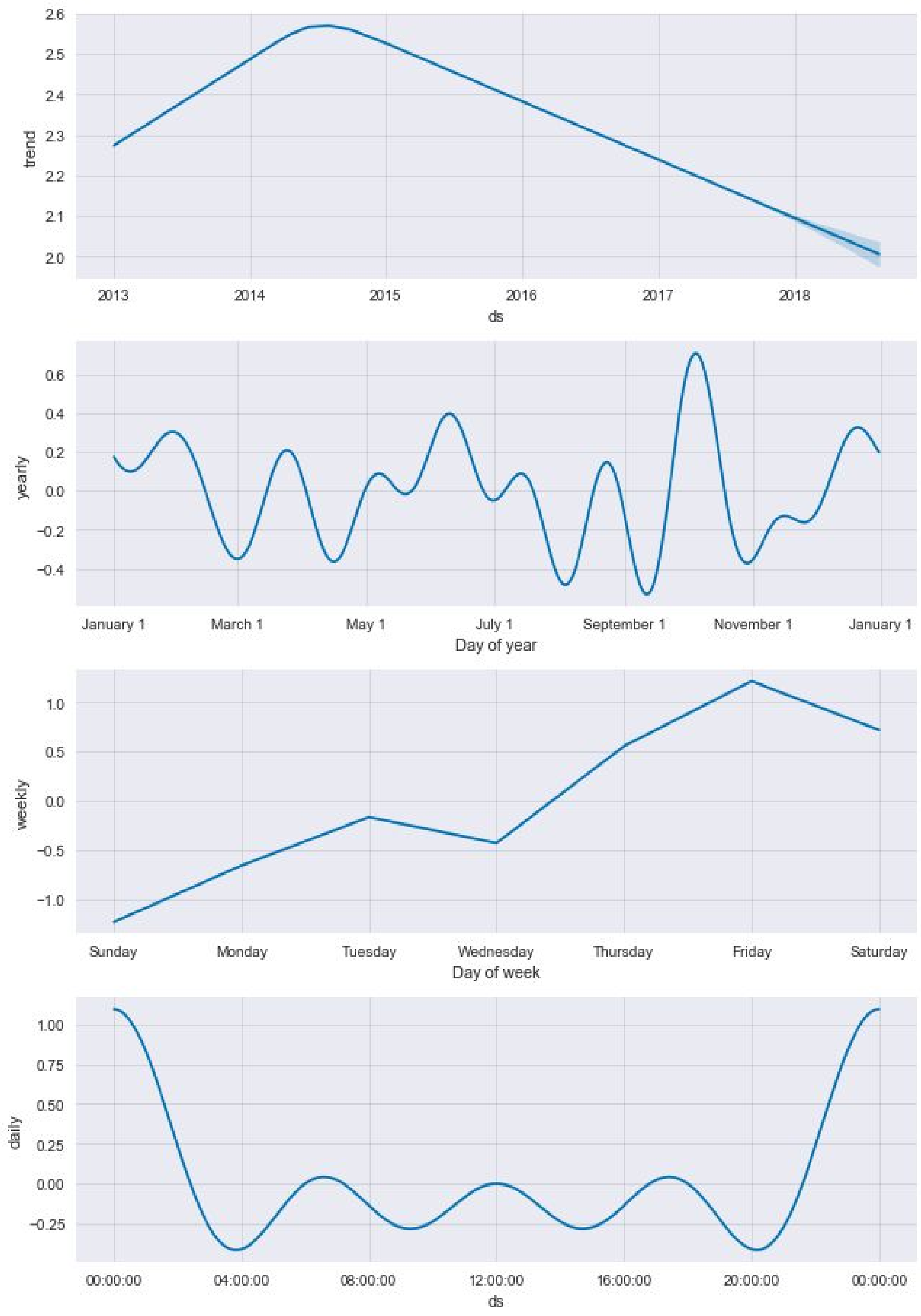
Autoregression (AR): the output at time t is a linear combination of past outputs

$$r_t = c + \epsilon_t + \sum_{i=1}^{p} \phi_{t-i}\, r_{t-i}$$

Moving average (MA): the output at time t is a linear combination of past shocks

$$r_t = \mu + \epsilon_t + \sum_{i=1}^{q} \theta_{t-i}\, \epsilon_{t-i}$$





Machine Learning Models:

SVM, Random Forest, XGB

model metrics: mean square error, mean absolute error

It has shown that XGB has better performance on both metrics. From then on, a grid search of best parameters was conducted, and the best parameters were found. Finally, the data was training with the best parameters and used to predict the test data.

7. Conclusion

This is a time series data forecasting problem. The datetime has big impact on the sales. To efficiently predict the future sales, the datetime needs to be splitted to year, month, and weekdays. The other information, such as holidays, oil prices, store types, also have some impact on the results. All of the above belong to the category of so-called feature engineering.

The open source package, fbprophet, is used for the time series model establishment. The time series model is different from machine learning models. It uses the concepts such as moving average, to do prediction. The limitation of this model is that it only considers the impact of date and if it is a holiday. All the other features were ignored. This reason has made it less accurate compared with the other predictive models.

In the last part, three models were used for prediction, SVM, Random Forest, and eXtreme Gradient Boosting. Two metrics were used. One is Mean square error, and the other one is Mean absolute error. For both metrics, XGB performed better than the other two models. The best parameters were figured out by doing a grid search within the hypotheses space of XGB.

8. Team member contribution

Four of us have evenly contributed to the project. We have regular weekly meeting on programming, feature engineering, and model selection.

9. Reference

http://xgboost.readthedocs.io/en/latest/python/python_api.html

https://pypi.python.org/pypi/fbprophet/

https://pandas.pydata.org/pandas-docs/stable/

http://scikit-learn.org/stable/documentation.html

https://docs.scipy.org/doc/