Pankaj Acharya
Capstone Project01 – In Depth Analysis
May 01, 2019

Project: 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 decide how to maintain stocks for particular items in particular shops optimally.

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 in same day makes it challenging.

Data Modelling:

To build a model to predict future sales, all the necessary data wrangling steps were performed. As observed during exploratory data analysis, the first 12 months were

removed from the training set as removing them would provide better prediction. This was confirmed later when I tried modeling on both with or without first 12 months. Then the data for last five months were taken for validation and saved as validationset.csv. The data without first 12 and last 5 months was saved as trainset.csv. The test data was also wrangled similar to the train set and saved as testset.csv.

This is a regression problem of predicting future sales of shop-item pair. So I tried to build different models and validate. The following regression algorithms were used to model my data.

1. **Linear Regression**:- Somebody had said that “Linear Regression is the `Hello World` of Machine Learning”. I think he/she has correctly said it. This is the first basic algorithm that is used for prediction.
2. **Ridge Regression**:- Ridge Regression is a technique for analyzing multiple regression data that suffer from multicollinearity.
3. **Lasso Regression**:- Lasso regression is a type of linear regression that uses shrinkage. Shrinkage is where data values are shrunk towards a central point.
4. **Decision Tree**:- Decision tree builds regression models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed.
5. **Bagging Regressor**:- Bootstrap aggregation or bagging is a simple and very powerful ensamble method. An ensamble method is a technique that combines the predictions from multiple machine learning algorithms together to make more accurate predictions than any individual model. Bagging construct ‘n’ trees using bootstrap sampling of the training data and then combines their predictions to produce a final prediction.
6. **Random Forest**:- Random forest is one of the most popular and most powerful machine learning algorithms. It is a type of ensamble machine learning algorithm called Bootstrap Aggregation or Bagging.
7. **Adaptive Boost Regressor**:- Adaptive Boosting or Adaboost is a machine learning meta-algorithm that can be used in conjunction with many other types of learning algorithms to improve performance. The output of the other learning algorithms (‘weak learners’) is combined into a weighted sum that represents the final output.
8. **Gradient Boost**:- Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees.
9. **XGBoost**: - XGBoost is a library which provides a gradient boosting framework.
10. **LightGBM**:- LightGBM is a gradient boosting framework that uses tree based learning algorithms.

At first, I used all the above algorithms with default parameter for 5-fold cross validation. Next, I tuned Random Forest, Bagging Tree, Gradient Boosting, XGBoost and LightGBM using RandomizedSearchCV. Finally using the best parameters, I performed 5-fold cross validation of the tuned algorithms.

Table 1: Performances of algorithms with default parameters

Algorithm	r2	rmse
Gradient Boost	0.677656	1.505827
XGBoosting	0.674566	1.500516
Linear Regression	0.661112	1.349141
LightGBM	0.653470	1.546350
Ridge Regression	0.629179	1.351676
Random Forest	0.626711	1.531725
Bagging Tree	0.625435	1.542834
Decision Tree	0.502483	1.934200
Lasso Regression	-0.000437	2.453177
Adaptive Boost	-3.260495	2.512668

Table 2: Performances of algorithms with tuned parameters

Algorithm	r2	rmse
Gradient Boost	0.675227	1.461838
XGBoosting	0.669124	1.468157
Random Forest	0.664986	1.457858
Bagging Tree	0.661905	1.425319
LightGBM	0.656737	1.533018

Conclusion:-

Gradient Boosting seemed to work best among other algorithms with or without tuning.

Link to the document:

https://github.com/leukemia/Capstone_Projects/blob/master/05_Capstone_Project01_InDepth_Analysis.ipynb