Time series forecasting for monthly sales prediction

Georgios Koumouridis
International Hellenic University
Thessaloniki, Greece
g.koumouridis@ihu.edu.gr

Orfanoudakis Dimitrios
International Hellenic University
Thessaloniki, Greece
d.orfanoudakis@ihu.edu.gr

Abstract

Managing efficiently a business requires being able to plan your next move. Predicting short-term or long-term sales is the key to understand the future needs of the market and adapt the business' strategy plan. Sales forecasting, especially for a product or a service, is based on carefully analyzing trends from the past in order to extract patterns that describe the customers' behavior. This paper proposes and compares some algorithms as an approach to predict future sales for a Kaggle competition, named "Predict Future Sales". More specifically, our research focuses on extracting the appropriate features from the provided dataset and applying different machine learning algorithms. Exploratory data analysis techniques are essential to the whole process in favor of selecting meaningful features.

I. Introduction (*Problem description*)

The goal of the problem is to predict the total sales, for every combination of product and store, for the next month. Since the project involves a time component, it is a typical example of time-series forecasting. Time series are widely used for non-stationary data, like economic, weather, stock price and retail sales, and can be cast as a supervised learning problem hence various Machine Learning methods can be applied. The basic concept is to build a model, based on previously observed values, able to predict as accurate as possible the amount of sales for the next. Considering that that value expected is a continuous quantity, the problem is transformed into a regression task. The evaluation of the model's performance is calculated by the root mean square error (RMSE) as indicated by the competition. RMSE represents the difference between the values predicted by the model and the values observed. The formula is given below:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{T} (\hat{y}_t - y_t)^2}{T}}$$

where yt are the predicted values and yt are the actual values for T different predictions.

II. Dataset description

The dataset consists of five files provided by a Russian software firm, called 1C Company. The training set contains 2.935.849 daily transactions within a period of 34 months, from January 2013 until the October of 2015. Specifically, for each transaction there are details about the date of the purchase, the product and the amount of products ordered, the shop from which the product was bought and the price that particular day. Three of the files share supplemental information about the items, the items categories and the shops respectively. The last file is the test set with 214.200 pairs of shop and product ids, on which our model will predict the number of sales for November of 2015..

III. Data preprocess

In order to make our predictions, the data should undergo some preprocess. First of all, items with extremely high price and number of sales are excluded from the dataset as they are considered noise for our model. By taking a closer look, we found out that there is an item with a negative price. We decided to fill that value with the median price of the product in respect to the month and the shop that the transaction took place. Next, we examined the name of the shops. Three of them had duplicate ids. Moreover, each shop name starts with the city name hence a column city was created. That process was followed for the item-category name since it contained the type and subtype that each product belongs to.

IV. Feature extraction

Below we summarize the features created from the information given in the datasets. The features are divided into five categories according to their content.

Time-based features: these features derive from the date of each observation. From the line graph below we can clearly see that the sales follow the same trend every year with a sudden increase in December.

The Time-based features are:
Month: The month in which the transaction took place (1-12).
Days: The number of days in each month. Since there is no leap year, every February Is mapped to 28 days.
Seasons: A four-bin feature with the seasons of the year.
Distance from December: The distance in months between the date of order and December, within the same year. Months since the first and the last sale for each item and for each pair of item and shop.

Lag features: the idea behind these features is to predict the value at time t, given the value at time t-1. In our paper we used the monthly sales for the previous three months as long as the same month from a year ago.

Rolling window statistics: For these features, we calculated the mean values of the sales for previous time steps. The different combinations and the time steps are displayed in the table:

Average items (per month)	Previous time steps (in months)
Average items sold	1
Average items sold by item	1,2,3,12
Average items sold by shop	1,2,3,12
Average items sold by item category	1
Average items sold by shop and by item category	1
Average items sold by city	1
Average items sold by item and by city	1

Trend features: Trend features represent a gradual change in a process, or a general tendency of a series of data points to move in a certain direction over time. Thus, we subtracted the difference between the items mean price within each month and the items average price, for the last six months. Moreover, since we want to predict the increase or decrease of the products total sales for the next month, based on the previous year we calculated the fluctuation of sales from the previous month.

Other features: city, item's type and subtype ids: in order to make the data ready for the model, we had to convert the categorical text data such as the city name, the item category and the item subcategory into model-understandable numerical data. For this task, we calculated the total sales according to the city, the type and subtype, and the value with the highest sales was mapped to the highest number as a try to capture their importance.

V. Description of models used in each submission

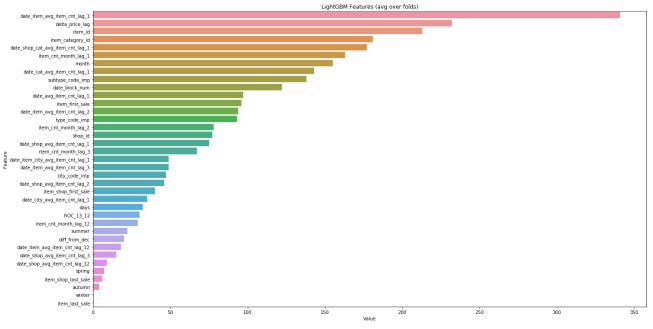
For this regression problem we tested the performance of three widely used regressors. Namely, the models used are lightGBM Regressor, CatBoost Regressor and Linear Regression. Finally, we constructed a stacking model from three regressors. We fed a linear regression with the predictions of the validation set in hope of making a better overall prediction.

VI. Comparative Experiments and Results

The performance of each regressor and of the stack model, are summarized in the table below.

Regressor	RMSE
CatBoost	0.93782
LightGBM	**0.89908**
Linear Regression	1.00981
Stack Model	0.91787

LightGBM outperformed the other two regressors as long as the ensemble model. The importance of the features according to the LightGBM are presented in the bar-chart.



VII. Conclusions

This paper was an approach to predict future sales using machine learning techniques. With feature engineering we transformed the time series dataset into a supervised learning dataset. More specifically, we created time-based, lag-based and sliding window summary statistics features. Finally, we trained three different regressors and the combination of them, and tested their performance on the unknown data.