Machine Learning Engineer Nanodegree

Capstone Proposal

Sasa Pavkovic February 14th, 2019

Proposal

(approx. 2-3 pages) Predicting future sales from Time Series data using machine learning approaches. It is a part of a bigger domain of predicting a continous variable, hence the idea can be generalized to other types of similar challenges in predicting customer moetary value in the future.

The data for the project will be taken from the Predict Future Sales competition on Kaggle. The link is provided here:

- https://www.kaggle.com/c/competitive-data-science-predict-future-sales

Domain Background

(approx. 1-2 paragraphs)

The domain that i chose comes from business where usually one of the relevant problems is to be able to predict buying behaviour of the customers and being able to predict sales. The additional component here is the temporal component where we can use some feature engineering approaches to map the temporal relations. Also, i would like to try out if a deep learning approach can be used sucessfully for this particular problem.

The need in business for solving this type of problem is very high as it can be generalized relatively easily. Some similar, but different questions can then be answered:

- How much revenue can we expect from diffferent segments?,
- In which product segments to invest most of marketing budget? or how to balance the marketing budget to maxmize CLV. Trying out a one of the approaches with neural networks might be a challenge but also very interesting.

Papers: https://arxiv.org/pdf/1708.05123.pdf

Problem Statement

(approx. 1 paragraph)

The problem that will be solved is predicting future sales based on the historical observed temporal data (time series). Main idea for solving the problem is to do feature engineering such that the temporal structure is transformed into features. Then standard regression algorithms can be used to make a prediction for the next period. We want to predict total sales in number of items for each store in the next period.

Datasets and Inputs

(approx. 2-3 paragraphs)

The datasets will be taken from the related Kaggle competition mentioned at the top of the proposal.

We have daily historical sales data. Note that the list of shops and products slightly changes every month.

File descriptions

sales_train.csv - the training set. Daily historical data from January 2013 to October 2015. test.csv - the test set. The test that we will use for predictions sample_submission.csv - a sample submission file in the correct format (related to Kaggle submissions) items.csv - supplemental information about the items/products. item_categories.csv - supplemental information about the items categories. shops.csv- supplemental information about the shops.

Data fields

ID - an Id that represents a (Shop, Item) tuple within the test set shop_id - unique identifier of a shop item_id - unique identifier of a product item_category_id - unique identifier of item category item_cnt_day - number of products sold. You are predicting a monthly amount of this measure item_price - current price of an item date - date in format dd/mm/yyyy date_block_num - a consecutive month number, used for convenience. January 2013 is 0, February 2013 is 1,..., October 2015 is 33 item_name - name of item shop_name - name of shop item_category_name - name of item category

The data prepared for the training is representing daily observations of shop and item sales with their respective price and total number of items sold. Only the sold items are actually in the list. Data was kindly provided by one of the largest Russian software firms - 1C Company. This is the main relevant data that came along with the Kaggle competition.

More info on 1C company can be found here https://en.wikipedia.org/wiki/1C_Company

Solution Statement

(approx. 1 paragraph)

After initial exploratory data analysis we will be applying feature engineering to create the needed temporal features and in combination with additional data available we will perform several machine learning regression algorthims to see which one of them is the best to use with the problem. We will find out which one is the best by looking to minimizie the Root Mean Squared Error rate or RMSE. Several iterations will be made so that the best parameters for the algorithms are found. Then the model that is the best will be applied to the test data in order to make predictions based on the test data.

Benchmark Model

(approximately 1-2 paragraphs)

I would like to use the Linear regression as a benchmark model as it is so widely recognized as a simple yet powerfull tool. The value of the scorer function (RMSE) can be inspected on the related Kaggle competition. At the time of the writing the best score is RMSE=0.79215, so that would be the current known upper limit. Still, i would expect the results to be worse then that. So, the RMSE will be used as a metric, and linear regression algorithm as benchmark model.

Evaluation Metrics

(approx. 1-2 paragraphs)

RMSE or Roor Mean Squared Error will be used as an evaluation metric for the performance of the models and to pick the one that will become the solution. Also others methods are popular like simply MSE, R squared, Adjusted R suared. The RMSE is chosen because of its relative simplicity. The RMSE is developed as the squared root of the average of squared difference between values predicted by the model and values observed.

An explanation of the RMSE and its related formulae can be found at:

- https://en.wikipedia.org/wiki/Root-mean-square_deviation\
- https://medium.com/human-in-a-machine-world/mae-and-rmse-which-metric-is-better-e60ac3bde13d

Project Design

(approx. 1 page)

The data will be loaded in the python environment and then some explanatory data analysis will be performed in order to understand some of the relationships in the data better. Outliers will be looked into and if there are outliers that are also high influence points then i will consider removing them from the dataset. Next the transformation of the temporal component of the data from 1 datetime column to several different features will be considered. The infulence of each of the features will be checked by inspecting feature importance for each of the models. Scaling the data will help with increasing precision in calculating distances between datapoints. Once data is prepared it can be used in modeliing.

Several algorithms will be considered for implementation Linear regression, Ridge & Lasso Regression, XGBoost and NN for regression.

Once the algortithms are parametrized i expect several iterations for each of them, as well as going back and forth between feature importance, feature selection and feature engineering until the best set of features is used in the data and best results are achieved. Linear regression is expected to behave worse then other algorithms but simplicty of the implementatin makes a good choice for the baseline model.
