Machine Learning Engineer Nanodegree

Capstone Proposal

Olle Green

olleg.green@gmail.com

July 5th, 2020

Proposal

Domain Background

Demand predictions in supply chain managagemnt (SCM) and logistics have historically been a constant pressure to make them more precise (Thomas & Griffin, 1996). The reason for this is due to the fact that inaccurate forecasting results in either too low supply to fulfill the current market demand, or too much, which in turn results in increased holding costs from inventory. Common ways to predict demand in SCM have been statistical models such as the ARIMA model (Jaipuria & Mahapatra, 2014). Academic articles have tested different modern machine learning algorithms such as XGboost and compared them to more classical linear regressions (Vanichrujee, Horanont, Pattara-atikom, Theeramunkong, & Shinozaki, 2018). This is still a new concept of using more advanced machine learning algorithms to predict demand over classical statistical models such as linear regressions, which is a domain we would like to explore further in this project.

The key area of interest is: As an organisation grow larger, the more vital the precision in these predicitons become. Therefore, we will explore the possibility to utilise Machine Learning algorithms to predict the demand of certain products, in order for the SCM-team to make better planning for instance purchasing product components for an upcoming season.

Problem Statement

The problem organisations SCM-departments face is that inaccurate forecasting for future demand cost organisations a lot of either missed sales or increased costs due to handling and storage of inventory (Thomas & Griffin, 1996). Therefore, we will test the hypothesis of comapring classic and more common statistical demand forecasting models to the more recent neural networks and machine learning algorithms, to assess if they produce better results. This will be used for predicting the department-wide sales for each store for the following year given our dataset of 45 retail stores. With this set-up, we can see that it is quantifiable, measurable and replicable.

Datasets and Inputs

We will use the dataset provided by Walmart on Kaggle.com "Walmart Recruiting - Store Sales Forecasting"you find the link to the dataset here. We are provided with historical sales data for 45 stores located in different regions - each store contains a number of departments. The company also runs several promotional markdown events throughout the year. These markdowns precede prominent holidays, the four largest of which are the Super Bowl, Labor Day, Thanksgiving, and Christmas. The weeks including these holidays are weighted five times higher in the evaluation than non-holiday weeks.

In the dataset we find three separate files: features, stores, test, train, sample submission.

1. features.csv

Contains data related to the store, department, and regional activity for the given dates.

- Store - the store number
- Date - the week
- Temperature - average temperature in the region
- Fuel_Price - cost of fuel in the region
- MarkDown1-5 - anonymized data related to promotional markdowns. MarkDown data is only available after * Nov 2011, and is not available for all stores all the time. Any missing value is marked with an NA
- CPI - the consumer price index
- Unemployment - the unemployment rate
- IsHoliday - whether the week is a special holiday week
- Sales

2. train.csv

Historical sales data used for training. Within this tab you will find the following fields:

- Store - the store number
- Dept - the department number
- Date - the week
- Weekly_Sales - sales for the given department in the given store
- IsHoliday - whether the week is a special holiday week

3. test.csv

Same as train.csv in terms of columns, but instead we don't find the "Weekly_Sales", as this will be our dependent variable we will try to predict future sales in the final submission for the kaggle competition.

4. stores data-set.csv

Anonymized information about the 45 stores, indicating the type and size of store.

- The store number (anonomised)

- Type
- Size

Size of total dataset

The different files containts the following amount of data: * Features data set.csv - 8190 rows, 12 columns * train.csv - 421 570 rows, 5 columns * test.csv - 115 064 rows, 4 columns

How will we work with these files

From my inital analysis, we will try to at least merge the features and sales data-set to find potentially find ways to utilise the data to improve our forecast. Some dates seems to be missing for the sales data vs the features data, but that will address before we move on to use the data to analyse and finding the best forecast method.

Solution Statement

How we would solve this problem is to try out some of the more popular Machine Learning algortihms to find the one providing us with the best predicitons, given this data from Walmart.

Benchmark Model

The interesting part will be that the benchmark we will use to assess our score will not be based on anything else than it's final kaggle leaderboards score in the kaggle competition. In this care that will be the with "Hari Khanal" with the score of ***3985.79966.***, which is around the median score for the whole competition. Getting somewhere close to this indicates that we have a functional model that can be improved upon later on if the time is available.

Evaluation Metrics

Ways to evalute the different models will be the following statistical measures of how accuracy and robustness;

Accuracy: * Mean Absolute Error (MAE) - Measure the mean absolute errors * mean squared error (MSE) - measures the average of the squares of the errors * Root Mean Squared Error (RMSE) - the square root of the variance, known as the standard error

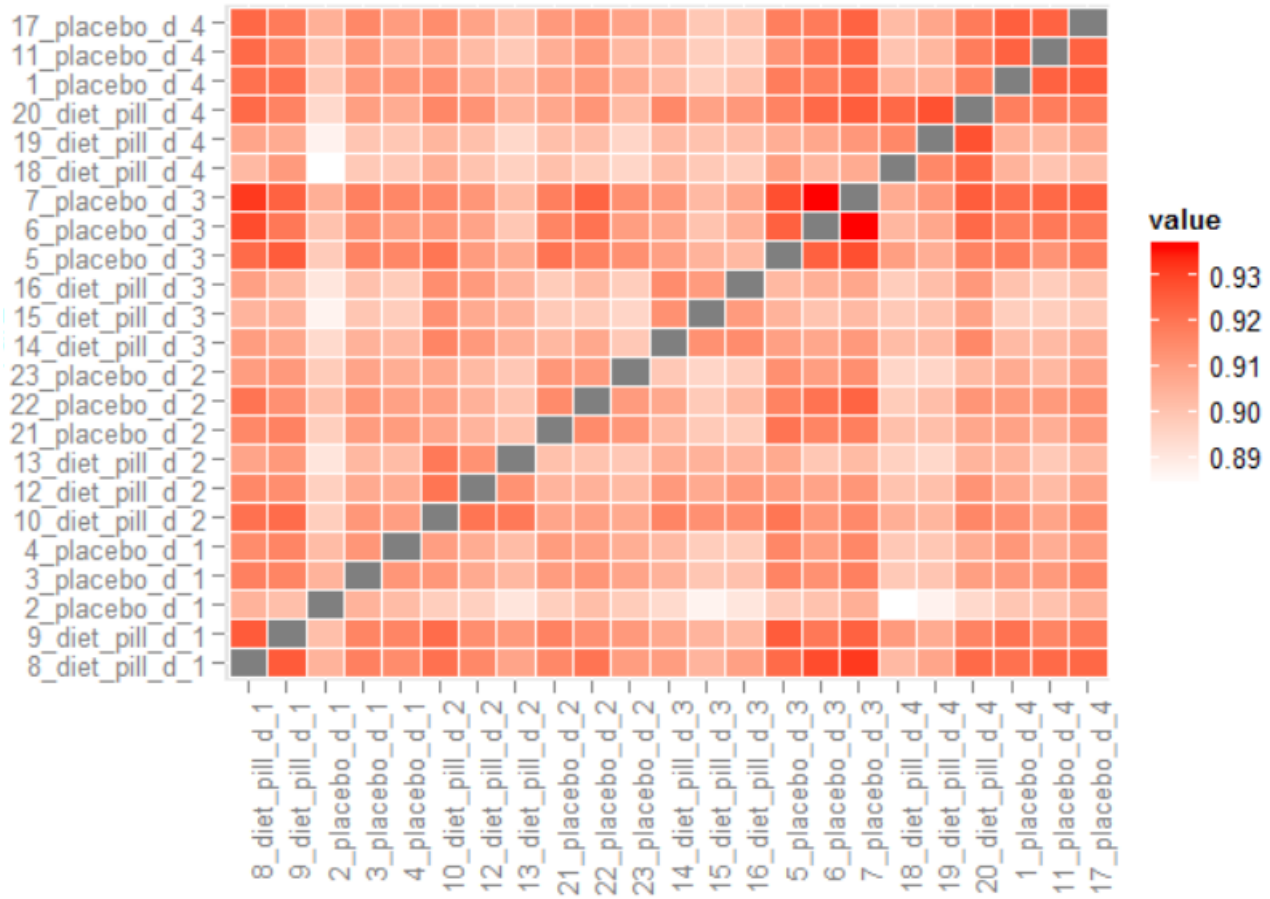
Mean squared error	$\mathrm{MSE} = \frac{1}{n}\sum_{t=1}^{n} e_t^2$
Root mean squared error	$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n} e_t^2}$
Mean absolute error	$\mathrm{MAE} = \frac{1}{n}\sum_{t=1}^{n} \lvert e_t \rvert$
Mean absolute percentage error	$\mathrm{MAPE} = \frac{100\%}{n}\sum_{t=1}^{n} \left\lvert \frac{e_t}{y_t} \right\rvert$

Robustness: * R-square (R^2)

All of which will help us to assess which models are providing us with the better demand forecast model. The smaller the number is (closer to 0), the better.

Initial Project Design

1. To start we will set up a notebook instance and set up the training, validation and test data.
2. Then we will proceed to visualise the data to get a better grip of what it looks like and how the different categorical data could be helping in the predictive demand forecasting model. In this part, we will include a correlation heat map, to understand how the data affect each other. Similar to the one attached:



3. Then we will build our models -- In this case it seems fitting to test it with a moving average and a neural network to find some good predictions for the sales of the company.
4. Then compare the scores -- Comparing the different models form step 3, might be other considered when we get into the data if needed.
5. Finalise the results in a summary of what each model gave and what some notes in regards to the use of each model.

Before submitting your proposal, ask yourself. . .

- Does the proposal you have written follow a well-organized structure similar to that of the project template? **Answer**: Yes.
- Is each section (particularly **Solution Statement** and **Project Design**) written in a clear, concise and specific fashion? Are there any ambiguous terms or phrases that need clarification? **Answer:** Yes
- Would the intended audience of your project be able to understand your proposal?
- Have you properly proofread your proposal to assure there are minimal grammatical and spelling mistakes?
- Are all the resources used for this project correctly cited and referenced? **Answer:** Yes

References

Jaipuria, S., & Mahapatra, S. S. (2014). An improved demand forecasting method to reduce bullwhip effect in supply chains. Expert Systems with Applications, 41(5), 2395–2408. https://doi.org/10.1016/j.eswa.2013.09.038

Thomas, D. J., & Griffin, P. M. (1996). Coordinated supply chain management. European Journal of Operational Research, 94(1), 1–15. https://doi.org/10.1016/0377-2217(96)00098-7