Business Summary

February 10, 2023

1 I. Business Context & Project Background

1.1 1.1 What Is the Case?

Forecasts are indispensable for many of the daily decisions that we make, such as time to get up in the morning in order to not be late for work, or when is the best time to fill up the tank before the gas price goes up. Supermarkets require forecasts to support their strategic development, make tactical decisions, and manage their demand and supply planning processes to avoid customer service issues and high inventory costs (Fildes et al., 2019). Therefore, it is important for supermarkets like Walmart to forecast as accurately as possible because stocking too many products incurs extra costs, whereas stocking insufficient would lead to lost sales and low profits. The M competitions have been conducted for almost 40 years. They aim to identify ways to improve forecasting accuracy by empirically evaluating several methods and identifying the one with the best accuracy. The findings obtained in these competitions have significantly influenced the theory and practice of forecasting by providing valuable insights into how forecasting accuracy can be improved (Hyndman, 2020).

The M5 competition extended the objectives of the previous four competitions by focusing on a retail sales forecasting application and using real-life, hierarchically structured sales data with intermittent and erratic characteristics (Syntetos and Boylan, 2005, Syntetos et al., 2005). The competition attracted many eager participants to experiment with effective forecasting solutions in real-life situations faced by numerous retail companies on a daily basis.

1.2 1.2 What Is Success?

The objective of the M5 "Accuracy" competition is to forecast Walmart's daily sales in 10 stores from 3 states for the next 28 days (4 weeks ahead) for each 42,840 units. For prediction performance measure, the case competition utilized a variant of the MASE originally proposed by Hyndman and Koehler (2006) called the Root Mean Squared Scaled Error (RMSSE) to measure "Point forecasts". The measure is calculated for each series as follows:

$$RMSSE = \sqrt{\frac{1}{h}\frac{\sum_{t=n+1}^{n+h}(Y_t - \hat{Y}_t)^2}{\frac{1}{n-1}\sum_{t=2}^{n}(Y_t - Y_{t-1})^2}}$$

After estimating the RMSSE for all the 42,840 time series of the competition, the participating methods will be ranked using the Weighted RMSSE (WRMSSE), using the following formula to determine the top 5 of most accurate point forecasts participants:

$$WRMSSE = \sum_{i=1}^{42,840} w_i * RMSSE$$

From the business's perspective, if we can develop a predictive algorithm that could predict daily sales with high accuracy in the real-world retail industry setting, supermarkets like Walmart will be able to make better decisions on its ordering from suppliers, inventory level, transportation planning, as well as staffing schedules in the future and that could not only maximize its profit but also improve both customers and employees satisfaction.

2 II. Preparations

2.1 2.1 Which Tools are Used?

The main machine learning algorithm used in this project is LightGBM. It is a gradient boosting framework based on decision trees to increase the model's efficiency and reduce memory usage.

The other packages used for data cleaning, model building, training, testing, and validation are as follow: * Pandas (v 0.24.0) * Numpy (v 1.16.2) * Tensorflow (v 2.0.0) * Sklearn (v 0.20.3)

2.2 2.2 Which Datasets are Used

All data used for this analysis were provided by the Kaggle M5 competition in CSV format. The dataset involves the unit sales of 3,049 products, classified into 3 product categories (Hobbies, Foods, and Household) and 7 product departments. The products are sold across ten stores, located in three States (CA, TX, and WI) in the USA. For the time series, the historical data range from 2011-01-29 to 2016-06-19. Thus, the products have a (maximum) selling history of 1,941 days (5.4 years). The three main data files are described below.

- calendar.csv - Contains the dates on which products are sold. The dates are in a yyyy/dd/mm format.
- sales_train_validation.csv - Contains the historical daily unit sales data per product and store [d_1 - d_1913].
- sales_train_evaluation.csv - Available one month before the competition deadline. It will include sales for [d_1 - d_1941].
- submission.csv - Demonstrates the correct format for submission to the competition.
- sell_prices.csv - Contains information about the price of the products sold per store and date.

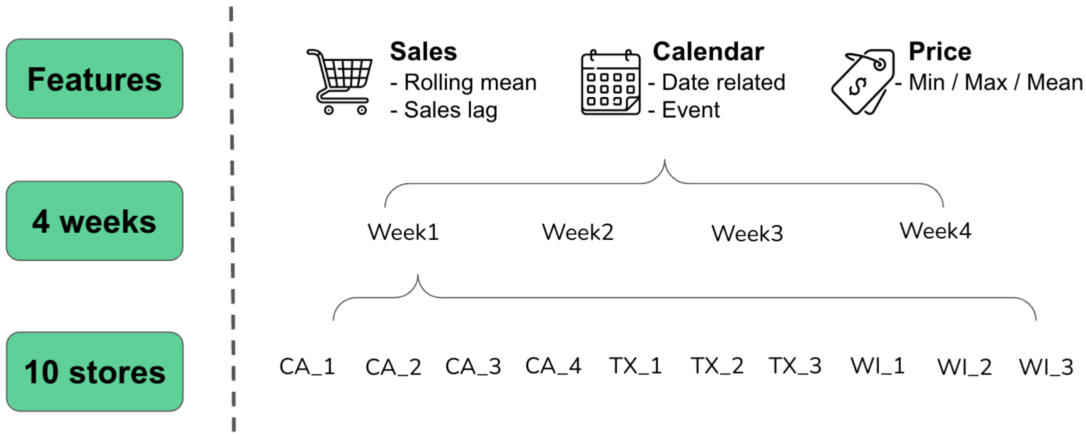
3 III. Predictive Model

To solve the forecasting problem, we first try the neural network (LSTM) and the gradient boosting tree technique: (lightGBM). However, the performance of LSTM was not as good as we expected; besides, due to the computation resources limitation and the implementation complexity, we decided to focus only on lightGBM.

Solution struction:

[ ]:
```
from IPython.display import Image
Image('/content/solution.png',width=950, height=500)
```

[ ]:



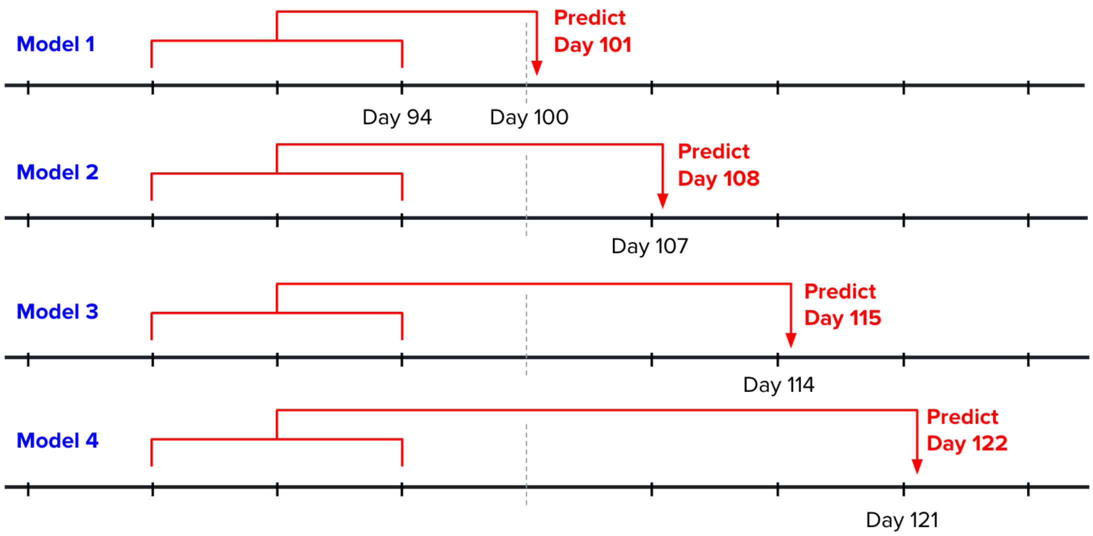
Our features can be categorized into the following 3 different parts: > 1. Sales related features, like lag sales for different time period, and rolling sales for different time period. This type of features reflect trend component. > 2. Price related features, like max, min standard deviation of prices for that product. > 3. Calendar related features. Different weekdays like Monday, Tuesday, Wednesday; different month like January, February and so on and so forth. The reason why we created those features is that we blieved based on seasonality, each different time zone has has a certain characteristic or type of behavior.

Based on those features, we will use different models for different weeks. We originally treated 4 weeks as a whole unit and directly predicted the entire month. But the performance isn't good enough, so after a couple of thoughts, we realized that each week has its own characteristics, intuitively saying, when we take about time schedule, human beings will treat a whole week independently, so four weeks will have four different situations inherently.

Each week we will have 10 different models for different stores, so we will have 4 weeks* 10 stores = 40 models in total. The reason why we use different models for different stores is that, according to our exploratory analysis, we observed that different stores have their own patterns, which intuitively makes sense since people in the different area has their own buying habits, like stores in California (CA) sells more items in general, while Wisconsin (WI) was slowly catching up to Texas (TX)

[ ]:
```
Image('/content/4week.png',width=950, height=500)
```

[ ]:



To further explain how a four-week prediction is implemented, we will assume the current day is day 100. This indicates we have all the sales, price, and calendar information before this day.

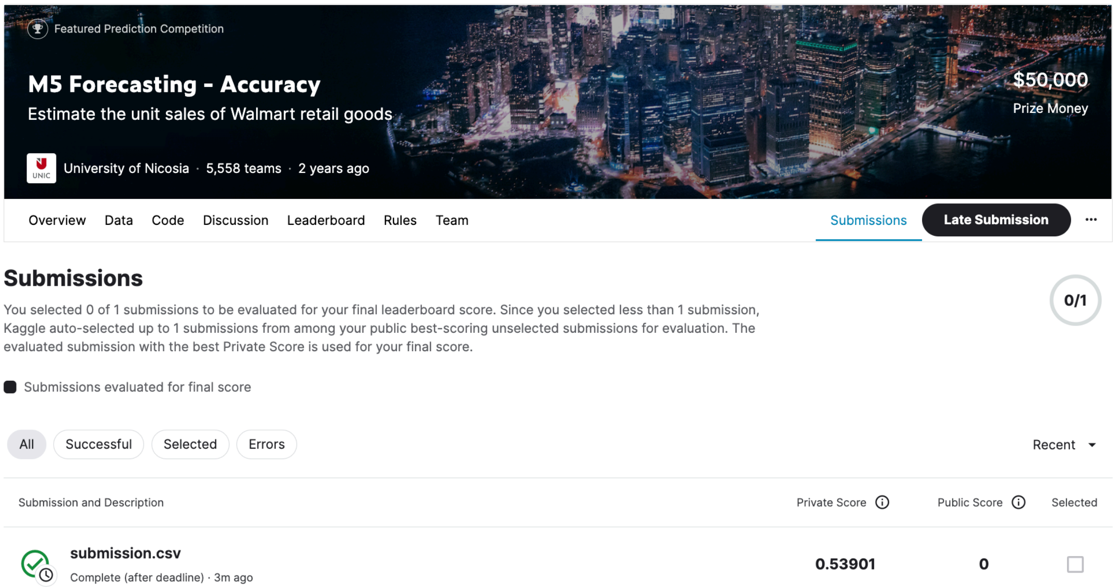
- For model 1, the task is to predict the next week in the future (so day 101 to day 107 in our scenario). More specifically, if we want to predict day 101, what we actually leverage is the information before day 94, and for day 102 we will utilize the information we have before day 95, so on and so forth until we reach day 107. In other words, there is a consistent 7 days gap between the information we utilize and the day we want to predict. You might wonder why we need this 7 days gap. This question is corresponding to how we gonna use this model, and we will further explain this at the end of the slide.
- Next, let's look at model 2. The task of this model is to predict the second next week in the future (in our scenario, it will be day 108 to day 114). The only difference right here is the prediction gap now becomes 14 days, which means if we want to predict day 108, what we will count on is the information before day 94 since 108-14=94.
- As for model 3 and model 4, we will follow a similar architecture but change the prediction gap to 21 and 28, respectively.
- Now, let's go back to the question at the beginning - why the prediction gap? It is because the target of this project is to predict 28 days in advance. If there is no prediction gap, our models will only be capable of predicting tomorrow; in our case, it will be day 101. If we want to predict day 102, then all the information from sales, price, and calendar on day 101 is necessary. Unfortunately, this is the information we will never have when we implement the models in a real world. As a result, the prediction gap serves as an essential component when training these models.

4 IV. Results

We have submitted our model's prediction result 23 times so far, including results from models which made predictions in higher granularity. We also tried using different start dates in training data to see the difference in performance. The best performance we could get is 0.53901 for the private score.

[ ]:
```
Image('/content/submission.png', width=950, height=500)
```

[ ]:



5 V. Forward Thinking

Apart from tuning the model to get better performance, having additional data could also make the prediction more accurate. Here are some examples of the data that the company you provide would be beneficial.

1. data about inventory, for products that have no sales, we don't know whether that's because they were out of stock or no one actually buying them.
2. Information about promotional campaigns, such as knowing whether or not there is a discount for each product on a particular day would be helpful.
3. Having data about customer demography could help identify different buying patterns from a diverse group of customers.