Corporacion Favorita Grocery Sales Forecasting

Milestone Report 2 – Machine Learning

George Tang

1 Introduction

The importance of sales forecasting cannot be overstated for any brick and mortar retail business. Managers leveraging accurate forecasts can ensure their stores have the right products in stock at the right time. This in turn provides a positive shopping experience which is critical in maintaining customer satisfaction and their retention. Also, it helps businesses to control item inventory and improve shelf space usage, reduce perishable goods wastage, lower labor and transportation costs, and ultimately increase revenue and profit.

These benefits motivated Corporacion Favorita to host Kaggle's Corporacion Favorita Grocery Sales Forecasting competition in 2017. Corporacion Favorita is an Ecuadorian-based grocery retailer that operates hundreds of supermarkets with over 200,000 different products. At the time, the business relied on subjective forecasting methods backed by little data analytics. They used the competition to challenge data scientists to develop innovative, data-driven solutions that can ultimately improves their sales forecasts.

1.1 Objectives

The overall objectives of this project are:

- To Identify features that affect sales, both short-term and long term
- To develop sales forecasting models and evaluate their performances
- To provide actionable recommendations

In this Report, we will present the results of on the development and evaluation of sales forecasting machine learning models.

1.2 Programming Code

The programming code can be found here.

2 Dataset

The data used in this project are obtained from the Kaggle competition's website, which contains 7 data tables. We will provide a description of each data table in this section.

2.1 Sales Data

The train dataset (train.csv) consists of the sales data for each item sold in each store. It consists of 125,497,040 rows and 6 columns namely, Id number, Date, Store Number, Item Number, Unit Sales and Promotion Status.

The test dataset (test.csv) is also provided for competition. Nonetheless, we will not use it in this project because the true target values (daily unit sales for each item at each store) are not provided, which makes model validation impossible.

2.2 Transaction Data

This dataset (transactions.csv) consists of the transaction information for each store. It contains 83,488 rows, one row for the number of transactions of the date at that store. The three columns are Date, Store Number and Number of Transactions. The information from this dataset is not used in the development of machine learning models.

2.3 Store Information

This dataset (stores.csv) consists of 54 rows, one for each of the stores owned and operated by the grocery store chain. The 5 columns are: Store Number, City, State, Type, and Cluster.

2.4 Item Information

This dataset (items.csv) consists of 4,000 rows, with each row consisting of information of one grocery item. Note that it is a subset of all products carried by this company. The 4 columns are: Item Number, Family, Class and Perishable Status.

2.5 Holiday Events

This dataset (holiday_events.csv) consists of holidays and events information of the country.

2.6 Oil Price

Since Ecuador is an oil-dependent country, Ecuadorians' spending can be tied to changes in oil prices. This dataset (oil.csv) contains two columns, namely Date and Oil Price.

3 Sales Forecasting with Machine Learning

3.1 Test Dataset

We define the target dataset as the the sales of each item for each store for the 15-day period from 8/1/2017 (Tuesday) to 8/15/2017 (Tuesday). It is the last 15-day period in the sales data (train.csv) with sales record available. We choose this time period so that its length is close to that of Kaggle competition, which goes from 8/16/2017 to 8/31/2017 (16 days).

We evaluate model performance with the metric provided by the Kaggle competition, known as the Normalized Weighted Root Mean Squared Logarithmic Error (NWRMSLE):

$$NWRMSLE = \sqrt{\frac{\sum_{i=1}^{n} w_i[\ln(\hat{y}_i + 1) - \ln(y_i + 1)]^2}{\sum_{i=1}^{n} w_i}}$$

Where:
n is the total number of items across all stores in the 15-day period;
w_i is the item weight, which is 1.25 and 1 for perishable and non-perishable items, respectively;
$\hat{y}_i$ is the predicted unit sales; and
y_i is the actual unit sales.

Note that the metric is calculated is based on the logarithmic transformation of the unit sales. The purpose is to account for the difference in magnitude of unit sale values across all items, which range from 0 to > 1,000.

This metric also penalize error on predictions for perishable items. Perishable products have limited shelf lives and must be disposed of after they are passed. As such, errors in predicting sales of those products will lead to higher financial loss.

3.2 Modeling Approach

We adopt the modeling approach from the winner of this Competition with modification.

1. Create a training, validation and test dataset which consists of features as discussed below;
2. Use the training dataset to fit machine learning models. In total, we create 15 separate models, one for each day of the 15-day period, with the sales data of each day as target variables;
3. Use the validation dataset to determine early stopping
4. Make predictions with the test dataset for the 15-day sales forecasting period

3.2.1 Feature Engineering

We created the following three type of features as the input variables: past sales data, promotion status, and store and item information.

Historical sales data

Sales forecasting is a type of time series problem where the unit sales of items in a store is related to their sales in a previous time. We created the following features for each item sales for each store:

- Unit sales 1 to 16 days before the reference date
- Mean daily difference for the last 3, 7, 14, and 30 days before the reference date

- Mean, median, minimum and maximum sales and standard deviation of sales 3, 7, 14 and 30 days before the reference date
- Total number of days with sales (i.e. unit sales > 0) 7, 14 and 30 days before the reference date
- Mean sales on days with promotion 3, 7, 14 and 30 days before the reference date
- Mean sales on days without promotion 3, 7, 14 and 30 days before the reference date
- Mean daily sales for each date of week (i.e. Sunday, Monday etc.) 4 weeks and 20 weeks before the reference date

Note that the reference date is the first day of the 15-day sales forecasting period. For the test dataset, the reference day is 8/1/2017, which is a Tuesday. As such, we set the reference dates for both the training and validation datasets to be the same day of the week. Also, since it is a time series prediction, the time period for the training dataset must precede the validation dataset, which in turn must precede the test dataset to avoid using future data to predict past data.

The chosen reference dates for the training, validation and test datasets along with the corresponding sales forecasting periods are shown in Table 3-1. The sales data during the 15-day sales forecasting periods are the target variables for the training and validation datasets.

Table 3-1 Data time periods for training, validation and test datasets

Dataset	Reference Date	Sales Forecasting Periods (Target variables)
Training	5/30/2017 (Tues) 6/6/2017 (Tues) 6/13/2017 (Tues) 6/20/2017 (Tues)	5/30/2017 (Tues) – 6/13/2017 (Tues) 6/6/2017 (Tues) – 6/20/2017 (Tues) 6/13/2017 (Tues) – 6/27/2017 (Tues) 6/20/2017 (Tues) – 7/4/2017 (Tues)
Validation	7/11/2017 (Tues)	7/11/2017 (Tues) – 7/25/2017 (Tues)
Test	8/1/2017 (Tues)	8/1/2017 (Tues) – 8/15/2017 (Tues)

For the training data, we have chosen four reference dates to increase the size of the training dataset. We concatenate vertically the dataset from the four reference dates and the corresponding target variables to result in one training dataset.

Promotion Status

As discussed in Milestone Report #1, item sales generally increase with promotion status. Intuitively, item sales may decrease after, or before a pending promotion event. We create the following features base on items' promotion status:

- Promotion status from 14 days before to 14 days after the reference date
- Sum of promotion days with 7, 14 and 30 days before the reference date
- Sum of promotion days with 3, 7 and 14 days after the reference date
- First and last promotion days 14 days before the reference date
- First and last promotion days 14 days after the reference date

Store and Item Information

We create dummy variables for the following store and item information:

- Province of the store
- City of the store
- Store type

- Store cluster
- Item family
- Item class
-

3.3 Machine Learning Approach

We will use the following machine learning approach:

1. Use gradient boosting to establish a 'baseline' performance
2. Use neural network for additional gain in accuracy

Gradient boosting is an advanced machine learning algorithm that has been shown to predict with high accuracy for regression problems. For this project, we have chosen Microsoft's LGBM which is one of the state-of-the-art GB packages.

Since the algorithm of neural network (NN) is considerably more complex, it is possible that it can produce even more accurate predictions. We will test if it is the case.

3.4 Model Performance

The validation and test errors for the LBGM and NN models are summarized in Table 3-2. For the two models that we tested, LGBM has a better performance than NN.

Table 3-2 Validation and Test Errors for LBGM Models (NWRMSLE)

Model	Validation	Test
LGBM	0.584	0.597
NN	0.609	0.631

4 Summary and Next Step

In this reporting period, we developed training, validation and test datasets with feature engineering, and evaluated two machine learning algorithms.

Next, we will expand our machine learning efforts by evaluating more LGBM and NN model configurations for improving performance. We will also take a closer look at the modeling results by comparing the actual sales and predicted sales for selected items and stores. Finally, we will develop recommendations for the client.