Sales Time Series Forecasting

Iris Pijning (12509035), Judit Győrfi (13209647), Will Chien (13236490)
University of Amsterdam

ABSTRACT

In this study, an effort is made to make a forecast as accurate as possible for the sales of a category of several hobby items at a California Walmart store. In doing so, there is a comparison between traditional forecasting methods and machine learning methods. The compared methods are Moving Average, ARIMA, Linear Regression, Random Forest, Gradient Boosting, and Prophet. All models are outperformed by a multiple linear regression that takes historical sales data along with engineered features as the input, demonstrating both a relatively good performance by machine learning models compared to traditional models, as well as the success in using multiple explanatory variables to give the forecasting model more predictive power.

1 INTRODUCTION

For large supermarkets like Walmart stores, forecasting future sales of products is crucial for keeping stock such that consumer demand can be met. This forecasting study focuses on the demand for a subcategory of hobby products in a Walmart store in California, USA. To be exact, we will try to forecast the need for 149 hobby products for 28 consecutive days. Along with previous demand for the products, we have access to data about the sales prices of the products and special events on the calendar for the sampled time series data. We hope to give insight into the accuracy of models in product sales forecasting by comparing the performance of traditional time series forecasting and machine learning methods.

Dataset Files

To train the forecasting models, we have access to three datasets. The description of which can be found in table 1.

Table 1: Data files

File names	Description
sales_train_evaluation _afcs2020.csv	This is the main training data. It is composed of a column for each of the 1914 days starting on 2011-01-29. Column IDs are a combination of item ID, product category, store region, and store number. As all items are from the same type and the same store, this information is not of great importance for this study. This dataset does not include the 28 day validation period.
sell_prices_afcs2020.csv	This file contains the IDs for the items and the store they are in, along with the products' sell prices and the corresponding dates for when the products were these specific prices.
calendar_afcs2020.csv	This is a calendar of the dates corresponding to the sales dates, days of the week, the month, the year dates, and related features like day-of-the-week, month, year, and special events on certain days with the types of events.

Literature Review

In a summary article of M5 sales forecasting competition results and strategies, Makridakis, Spoliotis, and Assimakopoulos [6] note that an approach that has resulted in good results in this forecasting scenario is to make use of simple machine learning methods. According to this analysis, a model like LightGBM is effective for sales forecasts, partly because of its ability to consider multiple explanatory variables. This also emphasizes the importance of using explanatory variables in past successful M5 sales forecasts, like the information about promotions and special events and about the sales prices.

Another method that has shown promising results in sales forecasting is Prophet, an additive model introduced by Facebook. Recent studies, like that of Žunić, Korjenić, Hodžić, Đonko [8] show the usefulness of the Prophet model for forecasting on real-world retail data. Facebook itself claims that Prophet forecasts provide quick and easy forecasting because 'Prophet's default settings [...] produce forecasts that are often as accurate as those produced by skilled forecasters, with much less effort' [2]. While Prophet forecasts are mainly based on historical time-series data of the sales themselves rather than extra explanatory variables, its robustness in dealing with missing data and shifts in trends may make it a very interesting model to try on the Walmart sales data.

Figure 1: Demand over time

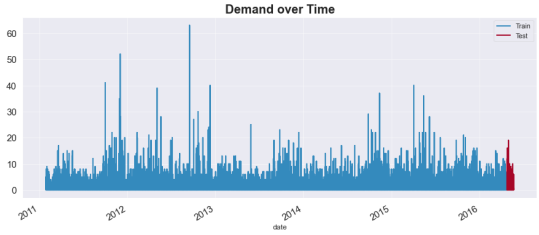
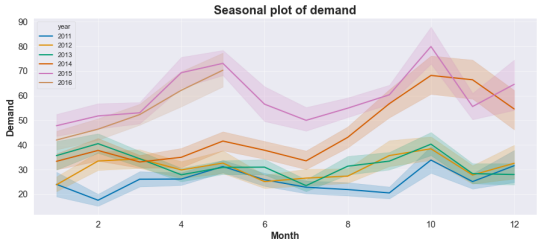


Figure 2: Seasonal plot of demand



Hypothesis

However curious we are to see the Prophet forecast results, based on previous studies, we expect to get the best forecasting results from a model that incorporates explanatory variables alongside the historical sales data, such as the data about special events that may affect the product sales. Therefore we expect that machine learning methods can include these variables to outperform traditional univariate forecasting methods, such as the Moving Average or ARIMA method.

2 RESEARCH METHOD

Exploratory Data Analysis

We start by taking a closer look at the available data. Figure 1 shows the plot of all 149 products' total added up demands to assess any consistent seasonality in the overall market. No clear seasonal patterns can be seen in this visualization.

A seasonal plot of all products' total demand over the years (Figure 2) shows some mild peaks at the start of May and October. This seasonality is more clear in 2015 and 2016 than in earlier years.

The following boxplot (Figure 3) on the left shows the increasing demand for total hobby products in this store in California for consecutive 5 years. The boxplot on the right provides us with an insight into the seasonality of the data, which peaked in May and October.

The time series decomposition (Figure 4) proves no distinctive seasonal pattern (third plot) across all items. It also shows that the residuals have a wider variance towards the end of the evaluation period.

Figure 3: Year-wise boxplot for the trend and month-wise boxplot for the seasonality

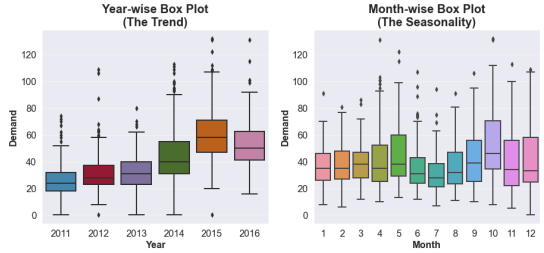


Figure 4: Demand decomposition

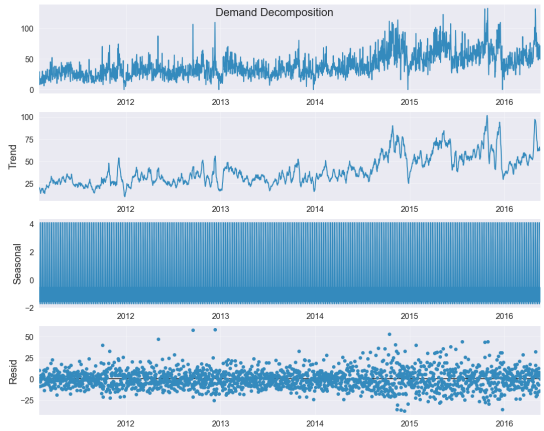
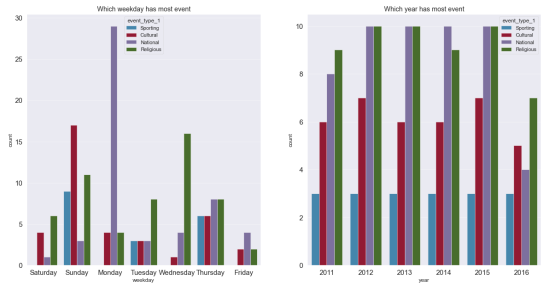


Figure 5: Events



On the left plot in Figure 5, we can see the frequency of the events on different days: Tuesday was the most frequent for national events, Wednesday for religious ones, and Sunday for cultural fests. Almost every year had the same amount of federal, spiritual, and sports events; meanwhile, the number of cultural events differed.

This density plot in Figure 6 that the sale price ranged between 0 and 10 dollars, but the majority of the products were under 5 dollars.

Figure 6: Distribution of sell price

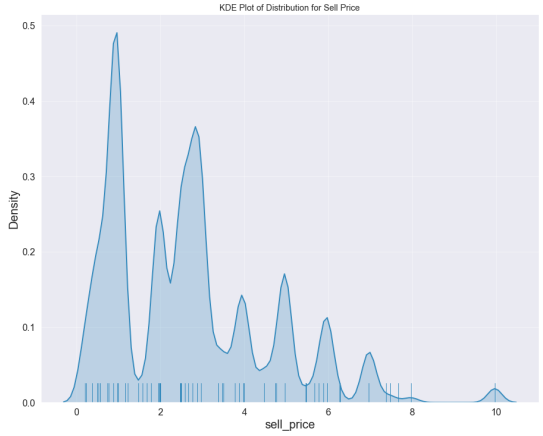
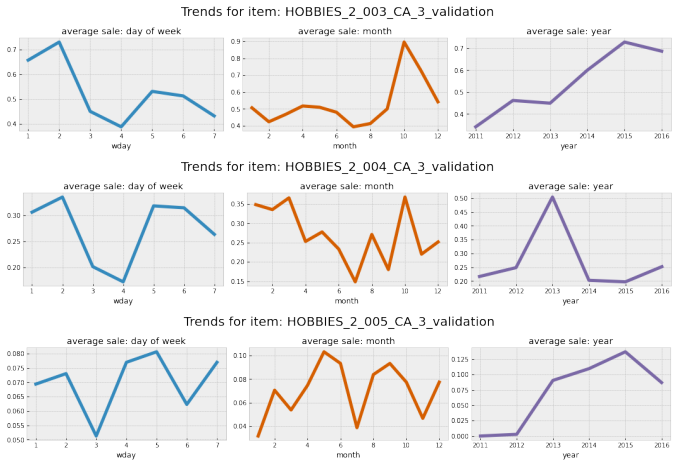


Figure 7: Seasonal Trend for a Single Item



Feature Engineering

Introducing appropriate variables can increase models' complexity and ameliorate underfitting. In that sense, feature engineering is the critical process of selecting relevant features and applying a transformation to these data to construct a robust predictive model [4].

In the competition, two major features regarding time series are applied to enrich the dataset. First, based on the insight acquired from data visualization, we assume the demand for each item is autocorrelated to seven days ago. Therefore, a lag of seven days in demand is introduced. The second assumption is that a similar sales pattern could appear on both an annual and weekly basis. We consequently utilize a groupby method to add descriptive statistics for each month and day of the week. Several new features are generated as ['lag_7', 'rmean_7_7', 'demand_month_mean', 'demand_month_max', 'demand_month_max_to_min_diff', 'demand_dayofweek_mean', 'demand_dayofweek_median', 'demand_dayofweek_max']. Figure 8 illustrates the above-mentioned feature transformation for a single item.

Figure 8: Feature engineering for time series

	id	lag_7	rmean_7_7	demand_month_mean	demand_month_max	demand_month_min	demandmonth_max_to_min_diff
2330	HOBBIES_2_148_CA_3_validation	1.0	0.142857	0.11828	5.0	0.0	5.0
2828	HOBBIES_2_148_CA_3_validation	0.0	0.142857	0.11828	5.0	0.0	5.0
2829	HOBBIES_2_148_CA_3_validation	0.0	0.142857	0.11828	5.0	0.0	5.0
2830	HOBBIES_2_148_CA_3_validation	0.0	0.142857	0.11828	5.0	0.0	5.0
2831	HOBBIES_2_148_CA_3_validation	0.0	0.142857	0.11828	5.0	0.0	5.0

Figure 9: Categorical features after label encoding

	id	event_name_1	event_type_1	event_name_2	event_type_2
2330	HOBBIES_2_148_CA_3_validation	19	2	3	1
2828	HOBBIES_2_148_CA_3_validation	19	2	3	1
2829	HOBBIES_2_148_CA_3_validation	19	2	3	1
2830	HOBBIES_2_148_CA_3_validation	19	2	3	1
2831	HOBBIES_2_148_CA_3_validation	19	2	3	1

Besides, as categorical features require additional encoding to fit into regression models, several approaches are also experimented with, including one-hot encoding, label encoding, mean encoding, and group-by encoding. In our final practice, the categorical factors, including event names and event types, are processed with label encoder as it outperforms others. Figure 9 demonstrates the categorical factors after label encoding.

Forecasting

Traditional Methods.

Moving Average. The Moving Average is set to be the baseline model. Even though this is relatively a simple method for time series forecasting analysis, we shall not underestimate its power. The summary table shows that it also provided us with a good forecast for the dataset (testing error of 0.9284) where the n was 5. It could have such a good result because it captures quite well the short-term fluctuations, and it highlights long-term trends.

ARIMA. For the ARIMA method, only historical sales data is required to generalize the forecast. However, like other traditional time series forecasting methods, it also assumes a linear relationship between the past and future demand for each sales item, which might not be the case for most of the real-world sales forecasting problems.

Machine Learning Methods. In recent time series forecasting competitions, novel machine learning approaches have attracted much attention. Several popular methods are experimented, including linear regression, random forest, gradient boosting, and Prophet.

Linear Regression . A multiple linear regression model is built in our practice with several predictor variables. Taking

demand as a forecast variable, we assume that a linear relationship between the demand and other predictor variables. After several rounds of experiments, we choose six major features including [‘sell_price’, ’year’, ’month’, ’wday’,’lag_7’, ’rmean_7_7’] to fit in our model.

Random Forest. Random Forest is a bagging algorithm for Decision Trees, and it is believed to be a powerful technique as it can effectively reduce variances. Random Forest parallel develops multiple Decision Trees by randomly resampling training data[5]. In our setting, features other than [’id’, ’item_id’, ’date’, ’weekday’, ’demand_month_min’, ’day’] are chosen for algorithms training.

Gradient Boosting. Gradient boosting is an ensemble method that improves the model performance by learning the errors from each training. As this algorithm trains a subsequent model from residuals of the previous model, it can effectively reduce both bias and variance [7]. In a regression problem, it fits on the negative gradient of the given loss function. Similar to our approach for Random Forest, features besides [’id’, ’item_id’, ’date’, ’weekday’, ’demand_month_min’, ’day’] are also chosen for algorithm training.

Prophet. The Prophet is a forecasting package launched by Facebook to predict time series data based on an additive model. While it does not take attributes other than the prior sales data, it fits with non-linear trends and various seasonality, including holiday effects. It also demonstrates remarkable forecasting power even with missing values and outliers [1].

Metrics

The Root Mean Squared Scaled Error (RMSSE) is used to assess the predictions in the competition. This metric measures the prediction error relative to a naive forecast, assuming that step t equals step t-1 (Figure XX). The RMSSE metric is a variant of the MASE (mean absolute scaled error) metric, where the sum of squared value replaces the absolute value. Different from MASE, the scaling is introduced to provide a scale-free error regardless of the data [3]. Therefore, the measurement is scale-free which allows us to compare different items’ demand of time series. In addition, the RMSSE metric includes weights so that it gives more importance to items with high demands, as the weights are proportional to the sales volume of the product.

3 RESULTS

During the training, the dataset is split into a training set and a validation set. While the latter is formed of the last twenty-eight days’ values, the former comprises the rest of the data. All the models are applied to respective items iteratively and evaluated based on the validation set score.

Figure 10: Root Mean Squared Scaled Error(RMSSE)

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

Table 2: Results of Different Methods

Models	Test RMSSE
Moving Average (Baseline)	0.928
ARIMA	0.988
Random Forest	0.981
Gradient Boosting	0.957
Prophet	0.912

For each model, the RMSSE on the final test set can be found in table 2. The worst performing model is the model with the highest error score, namely ARIMA, whereas the linear regression with feature engineering is the best performing model.

4 DISCUSSION AND CONCLUSION

The model that generates the most accurate forecasts is a straightforward machine learning method that based its predictions on a combination of historical sales and multiple predictors, which was the linear regression with feature engineering. The worst-performing model is the ARIMA model. This outcome of the best and worst-performing models suggest that our hypothesis that simple machine learning models will outperform classic forecasting models is correct. However, we have to conclude that the traditional models’ performance ranks and the machine learning models are mixed. As a result, we do not have conclusive evidence that machine learning models outperform traditional forecasting models for this case study.

However, we think that the significant limitations of traditional forecasting models are assuming that future sales are only associated with the past sales volume. Each item is independent of the other, whereas many other reasons might contribute to the sales pattern in a real-world setting. On the contrary, machine learning approaches take other factors into account, which enrich and empower the predictive models.

Furthermore, it cannot go unnoticed that although the Prophet model did not score the lowest error in this forecasting study, it did perform relatively well. Facebook makes strong claims about the ease in use and the relatively low effort involved in making decent forecasts using Prophet and in this study we learned that this is indeed the case.

Another notable performance is that of the moving average model. As mentioned before, while it is a relatively simple model, it can capture the short-term fluctuations and consequently have good predictive power. We also believe that the reason for this method to work considerably well is that we had 149 different products and the amount of sold items were moving on a narrow scale.

Future studies on a sales forecasting case like this one may be interesting to try more deep learning approaches, like convolutional neural networks. In our study we did have a go at trying a deep learning LSTM, but it did not perform well. More study into the performance of deep learning for this case may help explain why this approach failed and what can be done to improve the forecasts via these methods.

Finally, although it came highly recommended in the literature, we could not implement the LightGBM model for this forecast. Therefore, we could not compare this method to our other methods, which leaves a bit of a gap in the study. In future research, it would be interesting to see how the models that performed well in this study compare to the LightGBM model that performed well in the bigger M5 competition.

REFERENCES

[1] [n.d.]. *Forecasting at scale.* Retrieved December 20, 2020 from https://facebook.github.io/prophet/

[2] 2017. *Prophet: forecasting at scale.* Retrieved December 20, 2020 from https://research.fb.com/blog/2017/02/prophet-forecasting-at-scale/

[3] Rob Hyndman. 2006. Another Look at Forecast Accuracy Metrics for Intermittent Demand. *Foresight: The International Journal of Applied Forecasting* 4 (01 2006), 43–46.

[4] M Kuhn and K Johnson. 2019. Feature engineering and selection: A practical approach for predictive models. *CRC Press* (2019).

[5] Gilles Louppe. 2015. Understanding Random Forests: From Theory to Practice. arXiv:stat.ML/1407.7502

[6] Spyros Makridakis, Evangelos Spiliotis, and Vassilis Assimakopoulos. 2020. The M5 Accuracy competition: Results, findings and conclusions. (10 2020).

[7] Alexey Natekin and Alois Knoll. 2013. Gradient Boosting Machines, A Tutorial. *Frontiers in neurorobotics* 7 (12 2013), 21. https://doi.org/10.3389/fnbot.2013.00021

[8] Emir Žunić, Kemal Korjenić, Kerim Hodžić, and Dženana Đonko. 2020. Application of Facebook's Prophet Algorithm for Successful Sales Forecasting Based on Real-world Data. *International Journal of Computer Science and Information Technology* 12, 2 (Apr 2020), 23–36. https://doi.org/10.5121/ijcsit.2020.12203