Forecasting hobby retail data of Walmart

Brecht Boskaljon 10821007
Allard Tonkens 13170651
Dat Nguyen 11053828
Group 27
University of Amsterdam

ABSTRACT

In this project paper several forecasting methods will be used to predict future sales data for a large retailer. Being able to predict sales is key for a good inventory stock management at a supermarket. For this reason we tried to predict the sales 28 days in future for 149 product in the 'hobbies' category sold at a Walmart in California. The historical sales data was analysed and decomposed to examine seasonality and trends. Afterwards the future sales were predicted using the following forecasting methods: Average, Exponential Smoothing, Prophet and Light Gradient Boosting Machine. The performance of the algorithms was evaluated by computing the root mean squared error. Exponential smoothing yields the best accuracy with a root mean squared error of: *0.89980.*

1 INTRODUCTION

This paper will deal with the 'Sales Time Series Forecasting' as found on Kaggle (https://www.kaggle.com/c/sales-time-series-forecasting-ca-afcs2020/overview). The main goal of this project was to predict the amount of sales per day for 149 types of products; for 28 days in future.

Forecasting sales data gives useful insights to the retailer. In order to maximize profit the retailer should work with an effective inventory management system. After all, the economic principle of supply and demand is vital for retailers. Knowing when, what and what amount products the consumer demands is key to this inventory management. Storing unsold products is expensive; products being out of stock will cause consumers to choose for competitors, all leading to loss of revenue. Moreover, having this information enables the shop owner to buy the goods at low-cost.

For this forecasting project we used a subset of the M5 Forecasting - Accuracy hierarchical sales data from Walmart at one store, CA3 in the State of California. The products used in this forecast project are all goods in the 'hobbies' category. We were provided with a data set containing 3 .csv files:

- $sales\_train\_evaluation\_afcs2020.csv$
- $sell\_prices\_afcs2020.csv$
- $calendar\_afcs2020.csv$

$sales\_train\_evaluation\_afcs2020.csv$ (and a similar validation set) contained the item_id opposed to d_1, d_2, d_3, ... , d_1941; the days from 29-01-2011 until 1941 days further. So for every item we know how many times it was sold and when.

$sell\_prices\_afcs2020.csv$ contained the store id (CA_3 for all), the item_id, the wm_yr_wk, containing the week and the sell_price, the average price of the item sold during the wm_yr_wk specific week.

$calendar\_afcs2020.csv$ which contained information about the days the products were sold. This included holidays like Christmas and Independence Day. Of course the holidays can have a potential effect on the sales data. For example, it's expected that people buy more good just before Christmas. During the analysis of the data we will discover if such trends are visible.

With these data we will proceed to eventually predict the expected sales for each of the 149 products in the 'hobbies' category for 28 days in future.

In the following paragraph the data will be examined in more detail. Hereafter, the used forecasting methods will be explained.

2 STATISTICAL METHODS

Decomposition to analyze trend and seasonality of sales

In order to analyze the data, multiple decomposition's of the data are performed. First, a classical multiplicative decomposition of the total sales is done [figure 1]. A multiplicative decomposition is done as the seasonality increases when the trend increases. Classical decomposition methods assume that the seasonal component repeats yearly [4]. For our data this is a reasonable assumption.

From the decomposition we can see that from July 2011 the trend of the total sales stayed relatively constant until October 2013. This means that the total sales stayed constant over time with the total sales per day varying due to seasonality. From October 2013 until April 2015, we can see that there was an upward trend in the total sales. After that, the trend of the sales stayed constant again. A possible explanation is that due to the long lasting effects of the financial crisis, that even though the GDP [3] of the United States increased annually in the period from 2011-2014, people were reluctant to spend their money on products related to hobbies. When the economy recovered even more, it is likely that people are willing to spend more money on products of the category hobbies when they are in a more solid financial position. We can see that there is quite a lot of seasonality in the total sales. The sales at the beginning of the years spike, then decrease slightly in February. In March there is a yearly spike in the total sales. We can see that the total sales then decay at an almost constant rate until June. After that total sales increase quite rapidly until September, at an almost exponential rate. Total sales are the highest in late September or early October for most of the years. This could be due to the fact that many gifts for thanksgiving are bought in this period. From half October until half November, sales are relatively low, but they recover to the level of sales in late September until that. That the sales are so high in that period is probably due to Christmas shopping. On December the 25th Walmart closes due to Christmas and sales are thus zero at that date. After Christmas we see that the total sales are at the lowest level in most of the years. But sales recover quite rapidly and are up to average total sales at the start of the new year again.

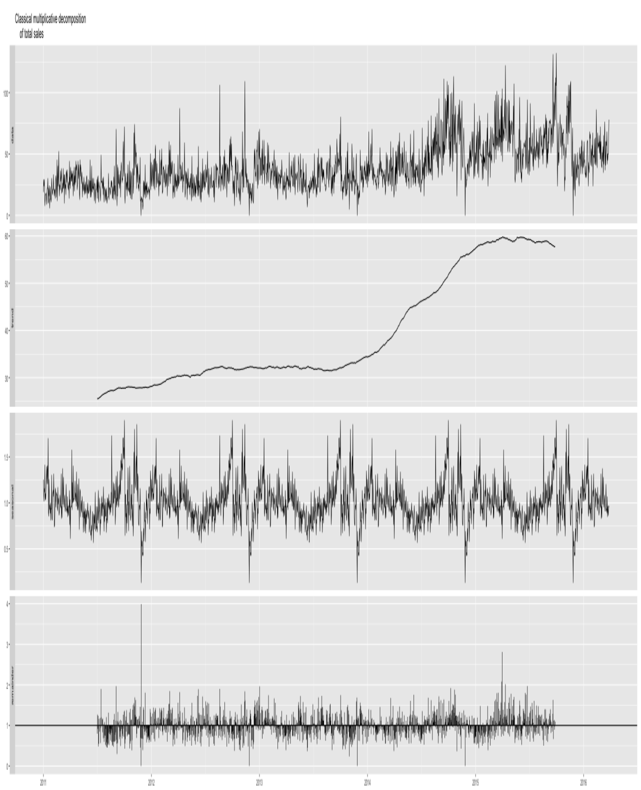

Figure 1: Multiplicative decomposition of total daily sales of Wall-Mart in a store in California in the hobby category

In figure 2 we can see a Seasonal and Trend decomposition using Loess of the total sales. A STL is a versatile and robust method for decomposing time series it was developed by R.B. Cleveland, Cleveland, McRae & Terpenning [1]. This method seeks to construct from an observed time series, a number of component series where each of these characteristics have a type of behavior. In our STL decomposition we have a trend component, which reflects the long-term progression of the series. A trend exists in the data when there is a persisting change, either upward or downward, in the data. In our STL decomposition, the window is set to 162 days, which is half a year. Additionally we have a seasonal component. This reflects a seasonal pattern that may exist in our data. Again, the same patterns are found as with the classical multiplicative decomposition.

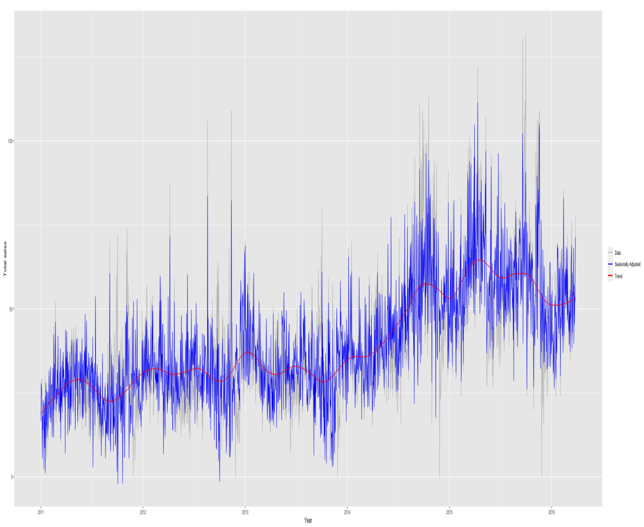

Figure 2: STL decomposition of total daily sales of Wall-Mart in a store in California in the hobby category

In order to investigate the seasonality even better. We looked at the monthly sales for four years and computed the average sales per month. As we can see from [figure 3], the sales follow the pattern described above, which is: a slow increase of total products sold from January until May, a decrease of sales during the summer months. A peak of total sales in October and a decrease in sales after that. As we can see, the total number of sales in a month are not very high. The month with the most average total sales, October, has 52 sales per day. This would mean that on average in October, 1560 products are sold. This means that each product is sold around 10.5 times on average. Therefore we conclude that these goods have a low turnover rate. This is probably because they are in the hobbies category. We would, for example, expect food products to have a higher turnover rate.

Additionally, we were not only interested if we were able to find seasonality in a year, but we were also interested if

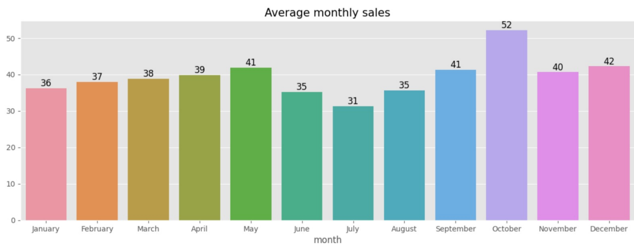

Figure 3: Average daily sales per month of hobby sales in a Walmart in California.

there might exists a daily pattern. In order to investigate this, we computed the average product sales for every day of the week [figure 4]. We can see that most sales happen on Saturday and Sunday. This is probably due to the fact that more people do not have to work in the weekend and therefore have more time to do shopping, which increases the daily average sales.

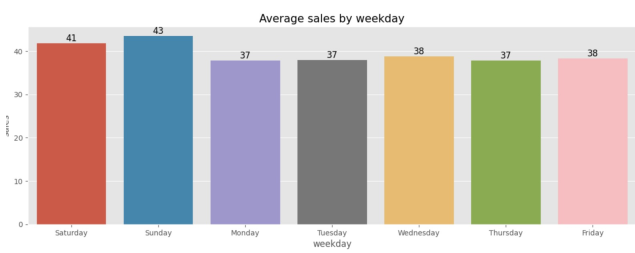

Figure 4: Average daily sales of hobby sales in a Walmart in California.

Since the fact that we are dealing with time series data of sold products, it can be of value to do further investigation to the potential effect of special event days on the amount of sold products. The offered data set included a calendar data set (as listed in the introduction). In this file all special event / holidays in The USA were mentioned. Since many of these days are about meeting family and friends and giving presents to each other, shop owners usually experience increased sales amounts during the public holidays. In general people will buy during the weekend preceding the event because of spare time and not being able to shop during working days. The graph in [figure 5] shows the average sales on the weekend preceding special event days.

This graph [figure 5] indicates that a holiday not necessarily means that there are more sales during the weekend prior to that specific day. However, it seems that Halloween and Easter lead to significant more sales in the preceding weekend. Although Christmas and Thanksgiving, two of the most well known holidays in The USA, show a preceding weekend with higher then average sales, we expected them to be top listed and stand out more. This could be because

Figure 5: Average sales at weekend preceding event days.

purchases for these festive days might be spread out over multiple weekends. The weekend preceding to New Year indicated the least sales compared to the average and other holidays. This could be because many shops are closed in the preceding weekend and all essential purchases are done before Christmas.

Additionally, we were interested if there is a relationship between the type of public holiday and the number of sales in the preceding weekend. In the calendar data set we could see that events were categorized based in four different categories: Cultural, i.e. Halloween, fathersday ; National: i.e. Christmas day, Thanksgiving ; Sporting: i.e. NBA finals, Superbowl ; Religious: i.e. Othordox Easter, Eid al-fitr. The average number of goods sold in weekends preceding these events is computed and the results are presented in [figure 6]. We can see that on a weekend preceding a cultural event sales where higher than the average sales on a single day. Before all other events (National, Sporting and Religious) sales in the preceding weekend were less high than the daily average. That sales are up in a weekend in advance of a cultural event can be explained by that people buy gifts. We note that it is remarkable that sales in a weekend before a national event are lower than the daily average, but are not able to provide a logical explanation for this.

We also investigate the price of the products sold. A histogram of the prices of the 149 products is given in [figure 7]. We can see that most of the products are around the price of \$1.00 or at a price of around \$2.50. The highest price of a product is \$ 10.00.

Subsequently, the relationship between the price of a product and the total units sold of that product is of interest. This allows us to analyze wether there is a relationship between the price and the total number of sales. In order to do this, we divided products into equal price intervals of 1\$. We count the number of products sold in that price category and divide it by the total number of products sold, which yields us a percentage. The percentages for all 10 price intervals are given in table [figure 1]. The total number of products sold is 63.078.618

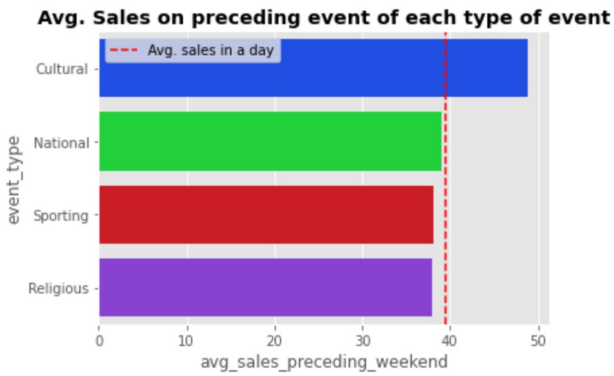

Figure 6: Average number of sales in weekend before categorized event types

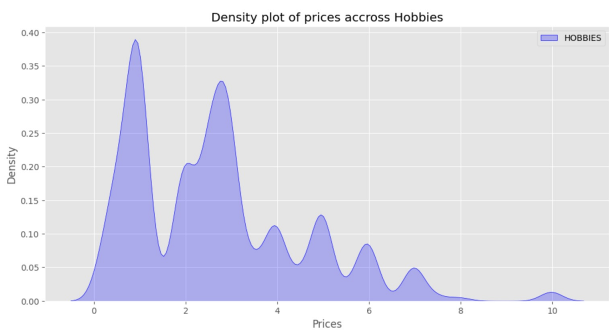

Figure 7: Distribution of the price of products sold.

Price range ($)	Total units sold
0-1	32.2%
1-2	11.9%
2-3	27.2%
3-4	9.2%
4-5	9.2%
5-6	5.7%
6-7	3.1%
7-8	0.7%
8-9	0%
9-10	0.8 %

Table 1: Sales per price category as percentage of total units sold

From table 1, we can see that there is a strong relationship between the price of a product and the total number of units sold of a product in that price range. We see that the cheapest products, from $0 to $3, make up more than 70% of the total products sold. This is not a surprising result, as most of the 149 products that were sold, were also in this price range. The 0% of total units sold in price range $8-$9, is due to the fact that there are no products within that price range, hence no sales are done in that price range.

3 METHODS

The goal of this report is to forecast the expected sales of 149 different products in the category hobbies for the coming 28 days. In this section of the report we present different forecasting methods that are used to forecast the expected sales per product. The performance of the algorithm is determined by posting a submission on Kaggle, where the root mean square error (RMSE) on an unseen test set is calculated. The following algorithms were used to do the prediction:

- Average method
- Exponential Smoothing
- Prophet
- LGBM

Average method

As baseline we have chosen a simple forecasting method, the Average method [4]. This method produces forecasts as following: $\hat{y}_{T+h|T} = \frac{1}{T} * y_1 + y_2 + ... + y_T$, where h is the forecast horizon and T is the amount of days of which the average is computed. We selected T to be 28. Because the month we need to forecast is April, and there is little difference between the average sales in March and April, the previous 28 days are used to forecast the next 28 days. This method is preferred as baseline over the naive method, as the data has a lot of zero values and this would result in the forecasts from the naive method to be 0 for the 28 days for which we forecast. The average method is used to compute forecasts for all 149 products.

Exponential smoothing

Exponential smoothing are weighted averages of past observations with the weights decaying exponentially as the observations get older. Per item that needs to be forecasted, exponential smoothing is applied. As there are a vast amount of exponential smoothing methods available, and it is likely that the best exponential smoothing method might vary per product, we select the best method for each product. We quantify best by selecting the model that results into the lowest AICc score. The AICc score gives a measure of goodness of fit. For most of the cases smoothing trend and smoothing seasonal were set to 0.

Prophet

Aside from using well known classical forecasting methods, such as exponential smoothing, the average method and ARIMA, we also investigated how well, more state of the art forecasting methods score. One of these methods we used is Prophet. Prophet is a tool developed by Facebook that is capable of generating forecasts of a reasonable quality at scales. It is used in many applications across Facebook for producing reliable forecasts, and it performs better than any other approach in the majority of the cases [5], [6] . Prophet is an open-source software released by Facebook's Core Data Science team, and is a procedure developed for forecasting time series data based on an additive model where nonlinear trends are fit with yearly, weekly, and daily seasonality, plus the effects of public holidays. Prophet is able to handle outliers well and is robust to missing data and shifts in the trend. In this paper, Prophet is used to model the dynamics of the sales of one item of the 149 products. The aim is to produce daily forecasts for each of the 149 products.

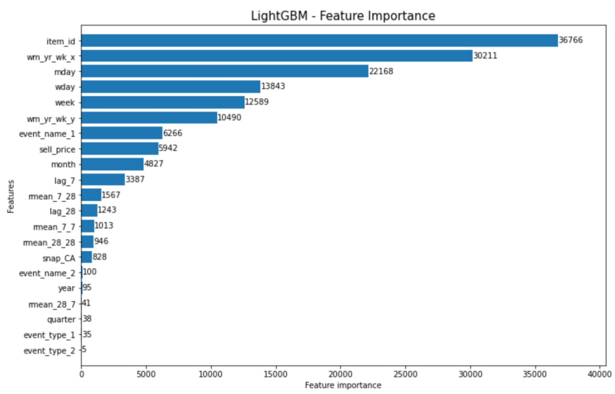

Figure 8: Feature importance LGBM

Light Gradient Boosting Machine

Light Gradient Boosting Machine (LGBM) is another state of the art method which is often used for forecasting. LGBM is a type of gradient boosting which is a type of ensemble learning. It integrates a series of weak regressors into a strong regressor. Each weak regressor that is trained represents different features of the data set, and we can evaluate the effect of each regressor on the forecasted results. LGBM is a gradient boosting machine framework based on a decision tree algorithm published by Microsoft Research Institute for Asia in 2016. LGBM constructs the algorithm based on feature importance. As shown in [figure 8], we can see that the features are divided into many small bins, that are stored in a histogram and split on these bins. This method is used by LGBM to reduce the cost of storage and calculation. We added the lags and rolling means of the lags as features to get a better accuracy in this model: lag of 7 and lag of 28. We can see that the most important feature is the id of the item sold, which is in line with expectations, as the total number of products sold differs quite a lot per product, see Statistical Methods. Additionally, we find that the week number and year also have high feature importance. This is also in line with expectations, as there is quite some seasonality in the data along with a trend, thus sales will depend a lot on what week of the year it is.

Algorithm	RMSE
Average method	1.09894
Exponential smoothing	0.89980
Prophet	0.93983
LGBM	0.91137

Table 2: Root mean squared error on the test set for the implemented forecasting algorithms

4 RESULTS

Forecasts are done using all four methods presented above. The score of these algorithms is measured against the test set on Kaggle. The metric that we decided to use to compare performance is root mean squared error (RMSE). Which is a metric that penalizes outliers more and can be easily interpreted. RMSE is computed as follows: $RMSE(\hat{\theta}) = \sqrt{E[(\hat{\theta} - \theta)^2]}$ with $\hat{\theta}$ being the forecasted value and θ the true number of sales, which is in the test set. The performance of the different algorithms on the test set is given in [table 2].

To analyze the different forecasts that are produced by our algorithms, we investigate the forecasts of these algorithms for a single product. As there are some items that do not sell quite often, we selected a item with a high turnover rate as forecasts for these items. This is done to give more insight to the effect of using different algorithms. In [figure 9], we can see the sales for product 28 in the previous 28 days, and the forecasts from the different algorithms for the next 28 days.

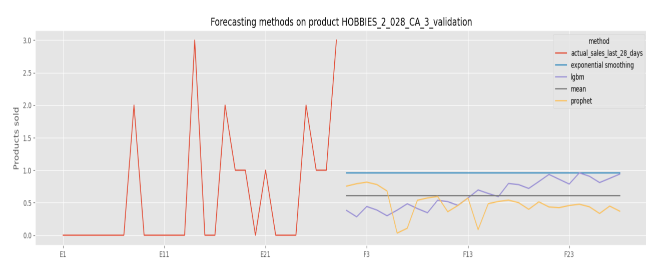

Figure 9: Forecasts produced with four different methods for 28 days

5 CONCLUSION

From the results we can see that, in line with expectations, the average method performs the worst. This is because this is a simple forecasting method and therefore does not exploit all possible information in the data. The algorithm that had the second worse accuracy is Prophet. As Prophet is reported to perform well on time series data in previous research [5], the result of the forecast was surprising. We expect the bad performance of the Prophet algorithm to come from a low turnover rate of the products. Additionally the low performance could be improved when feature selection was done manually instead of automatically. Light gradient boosting performed as second best algorithm. The best performing algorithm was exponential smoothing. This is remarkable as this algorithm produces a straight line as output. We expect that because of the low turn over rate of these products, it seems that forecasting a constant value for the coming 28 days outperforms the algorithms that try and capture a daily pattern.

By comparing the forecasts in [figure 9] we can see that the Prophet method and LGBM method try to capture a daily pattern. It is likely that due to the low turnover rate of the products, the daily pattern is hard to capture and hence it is hard to produce accurate forecasts that take this pattern into account. The other two methods, The average method and the exponential smoothing method, do not capture any of that daily pattern but produce a constant forecast for the following 28 days. The exponential method does take the yearly seasonality into account when producing forecasts, which is probably the reason that the exponential method outperforms the average method.

6 DISCUSSION

To asses if the presented algorithms are robust and thus also produces accurate forecasts for different products, we would like to investigate the accuracy of the algorithms on products that have higher average daily sales. Because the average daily sales of the product category investigated in this paper are quite low, around 10 units sold per product per month, we are not sure if the algorithms would have the same accuracy when we would produce forecasts for goods with a high turnover rate. In order to investigate this, we would investigate what the performance is of our algorithms when we train and test them on i.e. products of the food category.

In this report we mainly focused on the total number of sales and the price of the products that were sold. In future research we would like to investigate if there is a relationship between price change and daily products sold. We would do this by, for each time series, analyzing at which times the price went either up or down, and analyze whether this effect is visible in the daily sales of this product. This would be of interest as it could help managers set the correct price of a product based on the expected returns.

Autoregressive integrated moving average (ARIMA) is another widely used forecasting method [4]. While exponential smoothing models are based on a description of trend and seasonality in data, ARIMA models attempt to describe the autocorrelations in the data. Previous research has proved that ARIMA models perform well when forecasting demand, and even outperform artificial neural networks [2]. In order to do this we would produce different ARIMA models for each product and measure there performance with the AICc score. The best performing model would be used to produce forecasts for each product.

Additionally, as all presented methods have their own weaknesses and strength, future research could be done in examining whether we can use ensembling to increase the forecasting performance. In order to do this we would use a neural network ensemble method. This uses the forecasts of the different algorithms and their performance is combined. Neural Networks ensemble have proved to produce more accurate forecasts in many areas and thus are promising if future research would be done [5].

REFERENCES

[1] McRae Terpenning Cleveland, Cleveland. 1990. STL: A Seasonal-Trend Decomposition Procedure Based on Loess. , 3-73 pages.

[2] Jamal Fattah, Latifa Ezzine, Zineb Aman, Haj Moussami, and Abdeslam Lachhab. 2018. Forecasting of demand using ARIMA model. *International Journal of Engineering Business Management* 10 (10 2018), 184797901880867. https://doi.org/10.1177/1847979018808673

[3] United States government. [n.d.]. *Gross Domestic Product.* https://www.bea.gov/data/gdp/gross-domestic-product#gdp (accessed: 15.12.2020).

[4] R.J. Hyndman and G. Althanasopoulos. 2018. Forecasting: Principlesnbsp;andnbsp;Practice. https://otexts.com/fpp2/

[5] Sean J. Taylor and Benjamin Letham. 2018. Forecasting at Scale. *The American Statistician* 72, 1 (2018), 37–45. https://doi.org/10.1080/00031305.2017.1380080

[6] 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