Forecasting Sales
Understanding Data and their Environment

Luis Da Silva

March 28, 2019

1 Introduction

Forecasting sales suppose a major concern for business managers as it is a key component for profit maximization. Some of the benefits of an accurate prediction are storage cost reductions, better distribution dynamics, customer satisfaction and avoidance of waste, thus affecting profitability both in cost reduction and income increase. This is even truer when the business has a large number of stores, as in the case is being presented in this report.

By making use of data from a nationwide retailer in the U.S. with 45 different stores, the goal is to predict weekly sales for each department in each store for the period from 02/11/2012 to 26/07/2013. The case gets more interesting when the evaluation metric gives special emphasis on holidays that the retailer runs promotional activities on (namely the Super Bowl, Labor Day, Thanksgiving and Christmas).

The problem may be treated either as panel data multilevel modelling or as a series of individual time series problems, which predictions may be then combined together to represent a holistic view on the future of the retailer. Each approach has its own advantage.

Word count: 2787

Finally, before proceeding to the main body of the report, it is worth mentioning that, although everything presented here is written by me, I received deep insight from my team members Diep Do, Charlotte Appleton, Yanyao Cao and Jirui Wei.

2 Early insights about the data

Four datasets were given to describe the business. All the data had undergone some level of anonymization and thus, as little extra metadata was given, their exact meaning might be fuzzy.

The first dataset is called ”Stores”, and contains information about the size and type of each of the 45 stores. There were no missing data.

Most of the stores in the data are catalogued as type A, or ”Supercenters”, and this is even truer when considered as the total sum of square feet of each store (see figure 1). Type B, ”Superstores” come second and type C ”Supermarkets” last. Nevertheless, these names don't actually make direct reference to the size of the store. Actually, there is a wide region of overlap between store sizes and their classification. In fact, the two smallest stores are of type B and the third smallest store has type A.

Next comes ”train” dataset, which has information about the weekly sales for each store and department. In addition, an indicator variable for whether the week includes a holiday or not is given. This data runs from 05/02/2010 to 26/10/2012, thus a total of 10 weeks marked as holiday are present, corresponding to 3 Super Bowls, 3 Labor Day, 2 Thanksgiving and 2 Christmas.

As seen on figure 2, at least on an aggregate level, most holidays don't have an actual impact on sales (i.e. holiday and non-holiday distributions are very alike) with the exemption of Super Bowl and Labor Day, which show an increase in sales from about 16000$ to about 22000$.

Given the assumption that any department number always refers to the same kind of good across stores, one could zoom in and explore the effect of holidays in sales graphically for a given department across different stores. As an example, figure 3 shows the weekly sales for departments 1, 14 and 96 for store 10, 30 and the average across all stores. It is easy to see that

Figure 1: Distribution of stores

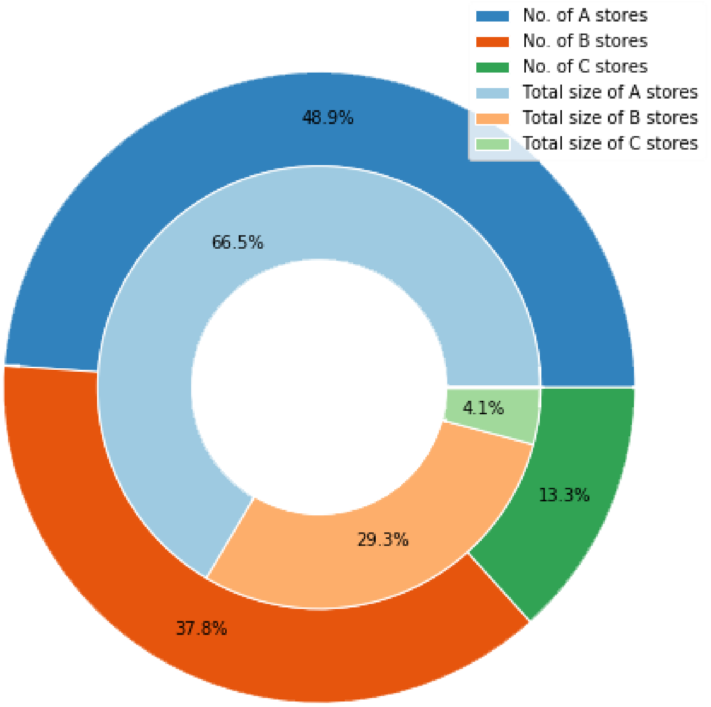


Figure 2: Sales over holidays

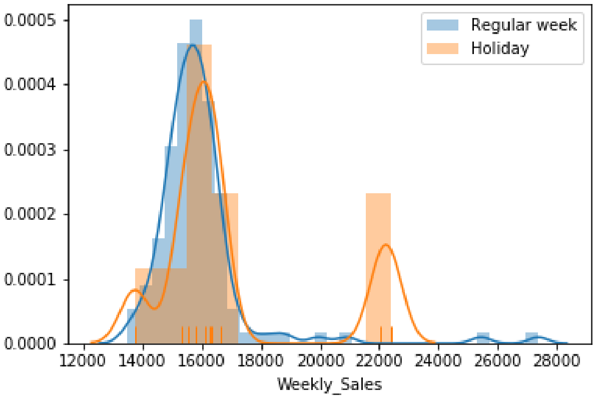
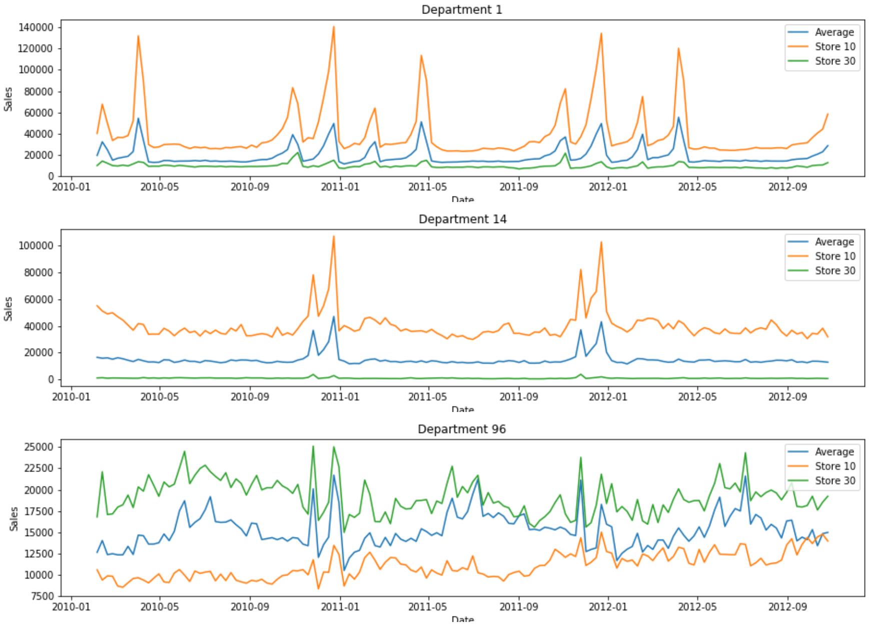


Figure 3: Weekly Sales on different departments

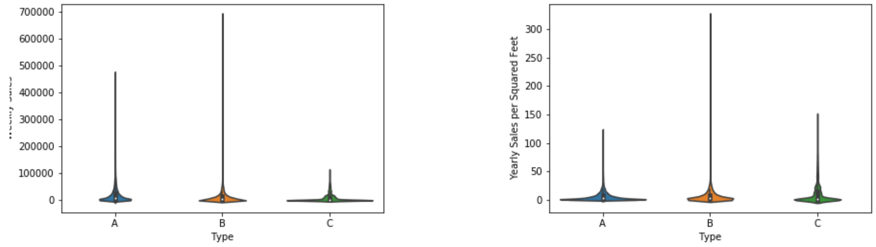


Department 1 seems to be very affected by all four holidays (data has four spikes per year), although the magnitude of the effect varies a lot between stores. Similarly, department 14 also presents spikes, but these are only related to Thanksgiving and Christmas, while the Superbowl and Labor day doesn't have any effect. Furthermore, these spikes are barely perceptible on store 30. Finally, for department 96, data is very noisy and thus no direct association between holidays and changes in sales can be made by eye.

A following natural question that arises when these two datasets are seen together is: what is the relationship between store type and sales?. Figure4 shows that B type stores (i.e. Superstores) are the most volatile in both mean yearly sales and mean yearly sales per square feet. On average, A type stores sell the most and C type sell the less, but when measured on sales per square feet, the opposite is true. Nevertheless, it is not easy to conclude that one type of store is more profitable or efficient than the other because there is a huge overlap is both metrics.

This train dataset also comes with a "test" dataset, which is identical in structure to train, but it has no observations for sales. Its dates range just

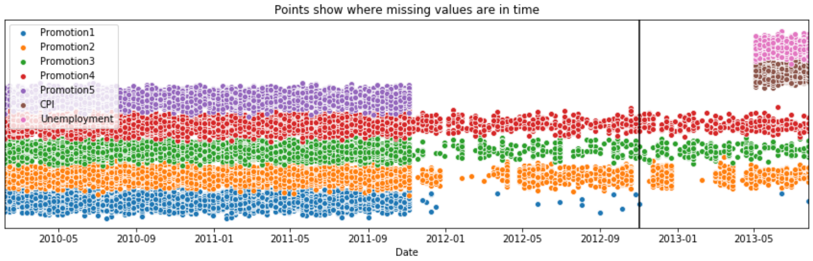
Figure 4: Sales per store type



(a) Yearly Sales

(b) Yearly Sales per Squared Feet

Figure 5: Missing data per variable in ”features” dataset. Data before black line is on the training period while data after is on test period.



from the week next to the end of train, 02/11/2012, to 26/07/2013, thus it contains 39 weeks to be predicted and it does not include Labor day.

Finally, the last dataset is called ”features”. This is the biggest dataset of all as it contains information about the temperature, fuel price, Consumer Price Index, unemployment rates and 5 magnitudes of promotions available on each store.

The biggest issue with this dataset is that it only recorded promotion’s data from November 2011 onwards. To make things worst, missing data is also found in the period that it got recorded. This situation is illustrated by figure 5, which plots the presence of missing data per variable across time. In the end, more than 55% of the promotions data is missing, so one could argue that it is mostly useless.

In summary, a transformed fuzzy meaning dataset is presented with high levels of missing values in the variables that seem to be intuitively more related to sales, huge overlaps in store classifiers and data on store level while predictions on store and department level are required. Fortunately, due to time dependence, time series forecasting may be done with a good level of success without external features. Thus, data is of regular quality and relevance.

3 Analysis process

Forecasting weekly data is challenging because the seasonality[1] periods are large and non-integer, with the average year having 52.18 weeks. Even when rounding to 52 periods, most methods can't capture efficiently the seasonality in such a large period [3].

The main question one needs to ask to time series data[2] is whether or not it is stationary[3]. That is because stationary data is often easier to model than non-stationary one [3]. Thus, data might need to be differentiated to achieve stationarity.

Figure 6 shows the probabilities of each time series to have a unit root (i.e. non-stationarity) measured by the Augmented Dickey-Fuller unit root test in each time series presented in data. Some of these time series have p-values over 5%, meaning that one can't statistically say that they are stationary with at least 95% confidence, but as 94% of them fall under this barrier, then it is assumed that the data is stationary.

Next, ideally one would like a time series to be highly correlated with its past (i.e. sales from some point in the past are proportional to sales today)[3]. Unfortunately, this is not always the case for sales registered in the training dataset. Figure 7 shows the autocorrelation plot for department 1 and 96 of store 10, while department 96 has significant autocorrelation until about lag 20, department 1 barely has one significant autocorrelated lag.

[1] Regular and predictable changes in a time series in a regular time period. It is usually a consequence of human actions.

[2] Remember that, although this is panel data, it may be treated as a set of time series data.

[3] Whether or not statistical properties are constant over time.

Figure 6: Augmented Dickey-Fuller p-values

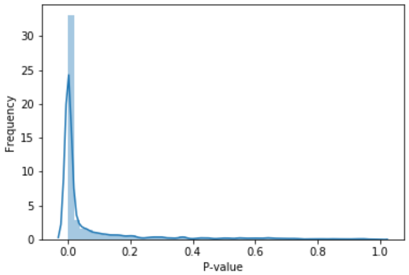
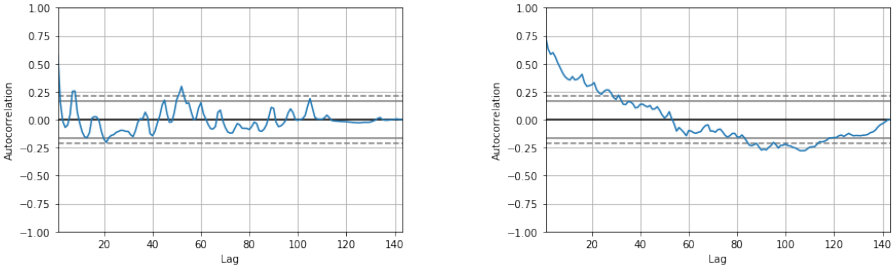


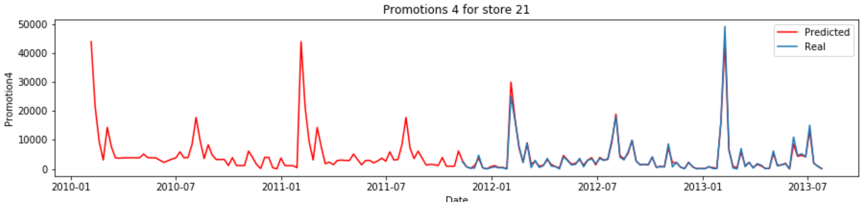
Figure 7: Autocorrelation plots for Store 10



(a) Department 1

(b) Department 96

Figure 8: Promotions 4, store 21. Predicted vs Real.



This is bad news for the case presented because weekly sales are the most informative and higher quality piece of data that is given. Thus, a need for avoiding the loss of promotions data because of the number of missing values arises.

In order to achieve this, promotions missing data (see figure 5) are filled. As promotions are being recorded since November 2011, all missings after that date are simply filled with zeros, this assumes that a missing value means that there were no promotions. Additionally, for consistency reasons, all below zero values were replaced by zeros.

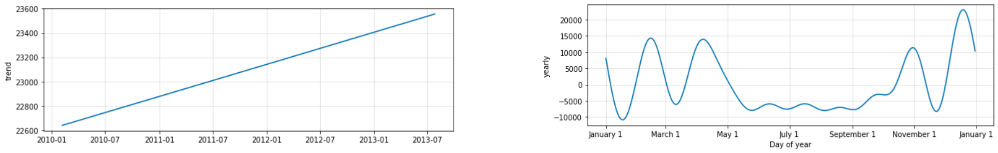
On the other hand, the data before November 2011 for each promotion variable is predicted backwards by a random forest model specially tuned via forward selection with 4-fold cross validation[4]. After fitting, selected variables discriminate between stores, weeks, months and some holidays to achieve promotions predictions. A sample result for ”Promotions 4” in store 21 is shown in figure 8.

Once promotions missing data has been filled, all datasets are linked together for the purpose of making the sales modelling easier. Outliers, in general, were not treated because even really low values of ”5” for weekly sales are repeated 289 times, and thus their removal or replacement will result in large losses of information; even though, negative values were not allowed and set to 0.

Another important bit of information for modelling is how does the trend and seasonality (if any) looks like in this dataset. Again, as different stores and especially different departments may have distinct behaviour across time, trend and seasonality modelling is done per department and store. Figure

[4] For a brief explanation on how time series cross-validation works, see appendix A.

Figure 9: Store 10 trend and seasonality



(a) Trend

(b) Seasonality

9 shows the modelled trend and seasonality for department 1 of store 10 by additive Seasonality and Trend decomposition with Loess (STL).

Given this particular panel data characteristics, a number of different approaches may be used to model the series. For example, some related are:

- Thomassey and Fiordaliso (2005) worked with clustering via K-means in combination with decision trees to predict sales of the Textile-Apparel-Distribution network [5].

- Doganis et al. (2006) used a radial basis function (RBF) neural network architecture and a specially designed genetic algorithm (GA) to predict sales for short shelf-life food products (by using milk sales data) and managed to outperform 8 other methods ranging from linear autoregressive to autoregressive moving average neural networks [4].

- In 2014, David Thaler used an average of 8 models, ranging from Seasonality Trend decomposition with Loess Function (STLF) and Exponential Smoothing (ETS) to Linearly seasonal, to win a kaggle held competition on sales predictions for Walmart [2].

- In 2017, Chen and Lu proposed 6 clustering based forecasting models that used self-organizing maps (SOM), growing hierarchical self-organizing maps (GHSOM) and K-means and support vector regression (SVR) and extreme learning machine (ELM) to forecast sales in the computer retailing industry. the GHSOM ELM combination turned out to be the best performing strategy [1]

Simplifications of these approaches are being implemented in order to forecast sales. Specifically, the models to be evaluated are:

- Linear ARMA. A simple linear model including autoregressive and moving average features. This is tuned by forward subset selection.

- Linear ARMA with L1 regularization (LASSO ARMA). Also a linear model but this time a regularization term is included which helps to perform feature selection more efficiently by selecting an α parameter. α is chosen via time series cross-validation (see appendix A)

- Random Forest ARMA. No special tuning is performed on this method due to computational cost. LASSO selected features are used as input.

- Prophet. This is a special forecasting package developed by Facebook, it automatically models seasonality, trend and holidays by applying exponential smoothing.

- Multi-Layer Perceptron (MLP). A 2 hidden layer with 200 and 10 hidden units architecture is used. Again, no further tuning is applied due to computational cost, but LASSO's selected features are used as inputs.

- LSTM Neural Network. 100 neurons LSTM network followed. All features were fed as input.

4 Modelling, results and validation

The evaluation metric required is a weighted mean absolute error (WMAE), which considers errors in holidays to be 5 times as important as errors in any other day. Mathematically, it is computed by:

$$WMAE = \frac{1}{\sum w_i} \sum_{i=1}^{N} w_i \mid y_i - \hat{y_i} \mid$$

To prevent overfitting, about 42.7% of the training data is being kept as a held-out dataset and used for validation. That is, training observations from 02/11/2011 are not used. This specific date is chosen because the main goal is to select the best model that aims to forecast test set (beginning in 02/11/2012).

Table 1: Results summary

Model	LASSO	Linear	MLP	Random Forest	Prophet	LSTM
WMAE	1441	1455	1629	1653	1708	3046
				All but LSTM average		1399

Figure 10: Predictions on Held-out Dataset for store 1 department 5

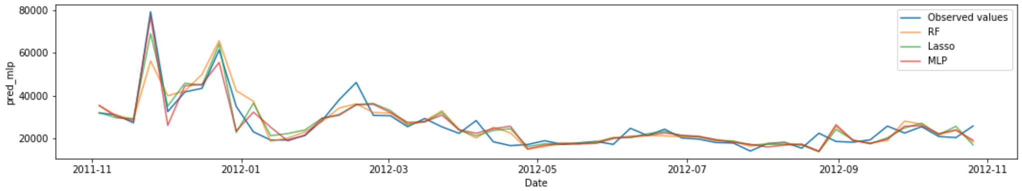


Table 1 summarises the validation WMSE scores on the held-out dataset and figure 10 shows a sample prediction for some of the models. The best performing model turns out to be LASSO, which used 17 features to predict future sales, among which there are 15 different autoregressions (including last year's sales), seasonality, and Promotion #3.

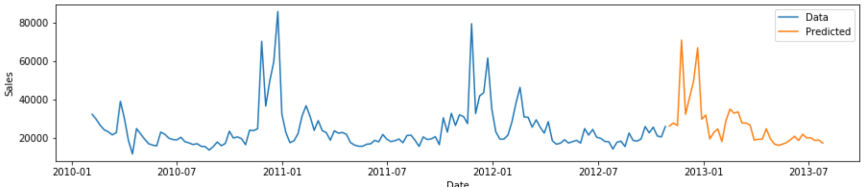
The closest scoring model, a forward-selected linear model chose a much simpler structure, with only 8 features: seasonality, 2 autoregressors (including last year sales) and 5 moving averages. It is worth noticing that this model is not using any information besides the sales data itself, so it is also easier to implement.

For the neural networks, the MLP model got to third place without further tuning, although its result might be quite volatile with this amount of data, so it could well have gotten the last place. A similar issue arises with LSTM, who got the last place, but it has proved to be useful when used with the right settings [6].

Finally, Facebook's package Prophet is surprisingly easy and fast to use, but it still manages to perform relatively well. One interesting fact is that by calculating the mean of the predictions of all models but LSTM, the validation error actually drops and manages to become the best estimate.

Once selected LASSO as the best performing model, future observations

Figure 11: Predictions on test Dataset for store 1 department 5



are predicted one date at a time. In this sense, the predictions y_t in the test dataset are used to predict y_{t+1}, which increases uncertainty while the model moves into the future.

5 Conclusions

After the research made, different models managed to forecast retail sales with a very close rate of accuracy mainly using no extra information than the time series itself. Although a good chunk of information is actually embedded into the history of the amount of sales itself, it is also possible that the true explaining factors of sales are not being recorded with enough quality (or just not being recorded at all) and thus cannot be used for modelling.

If data quality is hard to improve (and even if it's not), further modelling research could yield improvements in results. At a first level, as David Thaler suggests [2], SVD decomposition could improve the signal to noise ration in the data and thus make modelling easier. Then, better hyper-parameter optimization on random forest should yield directly to better results. Finally, as Phillip and Alex suggest [4], using GA to tune LSTM architectures (assuming one has sufficient computational power) should allow a better forecast.

References

[1] Chen, IF. & Lu, CJ. Neural Comput & Applic (2017) Sales forecasting by combining clustering and machine-learning techniques for com-

puter retailing. URL: https://doi.org/10.1007/s00521-016-2215-x (accessed: 07/02/2019)

[2] David Thaler (2014). First Place Entry. URL: https://www.kaggle.com/c/walmart-recruiting-store-sales-forecasting/discussion/8125 (accessed: 22/03/2019)

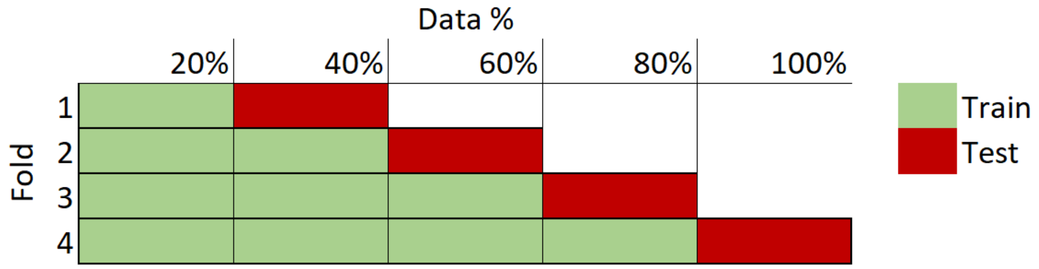
[3] Hyndman, R.J., & Athanasopoulos, G. (2018) Forecasting: principles and practice, 2nd edition, OTexts: Melbourne, Australia. OTexts.com/fpp2. Accessed on 15/03/2019.

[4] Philip D, Alex A, Panagiotis P, Haralambos S (2006) Time series sales forecasting for short shelf-life food products base on artificial neural networks and evolutionary computing. https://rsandstroem.github.io/predicting-retail-sales.html

[5] Thomassey S, Fiordaliso A (2005) A hybrid sales forecasting system based on clustering and decision trees.

[6] Pan B., Yuan D., Sun W., Liang C., Li D. (2018) A Novel LSTM-Based Daily Airline Demand Forecasting Method Using Vertical and Horizontal Time Series. In: Ganji M., Rashidi L., Fung B., Wang C. (eds) Trends and Applications in Knowledge Discovery and Data Mining. PAKDD 2018. Lecture Notes in Computer Science, vol 11154. Springer, Cham

Figure 12: 4 fold time series cross validation



Appendices

A Time Series Cross Validation

Given the sequential properties of time series data, regular random K-Fold Cross Validation is not valid. That is because, if one has information about the future and past state of a time series, then is much easier to predict its current state. To overcome this, cross validation is performed on a sequential basis.

As figure 12 explains graphically, a sequential four fold cross validation actually splits the data in 5 sets of 20% each. For the first validation, it uses the first split to train the model and then the second split to validate results on new data. Once it's finished, it retrains the model but this time it uses the first two splits and validates on the third, and keeps doing that until all data have been used. Final cross-validation score consists on an average of all scores calculated.