Forecasting Rossmann Store Leading 6-month Sales

CS 229 Fall 2015

-
Sen Lin, Eric Yu, Xiuzhen Guo

Abstract

We investigated the comparative performance of Frequency Domain Regression (FDR) and Support Vector Regression (SVR) for time-series prediction of Rossman Store Sales. Due to the extent of the data variables provided, SVR clearly outperformed FDR. Within SVR, our results reviewed that a polynomial kernel with regularization is most effective.

Introduction

Sales forecasting is critical for inventory management in the retail industries. Ideally, store managers can use accurate predictions to meet demand while minimizing inventory footprint and therefore operational costs. Further, discrete factors such as holidays, opening of competitors and promotions all have a significant level of demand at any given day. We seek to analyze the impact of these factors with the aid of time series analysis and machine learning techniques.

We used data from the Kaggle competition Rossmann Store Sales - Forecast sales using store, promotion, and competitor data. Rossmann GmbH is a major pharmaceutical chain with over 3,000 stores across Europe, including stores in Poland, Hungary, Czech Republic, Albania, and Turkey. Rossmann is very similar to the pharmacy company Walgreens in the U.S. The data contains a rich set of features, including both boolean and continuous variables.

We investigated both the Frequency Domain Regression and the SVR method. We found that time-series methods underperformed more powerful machine learning techniques. Upon further scrutiny, we realized that this is because sales variations were mostly driven by these discrete events, while the time-series trends of seasonal or inter-year trends were minimal. We believe that this finding is generalized to many forecasting problems, where more granular day-to-day predictions are required on a short span of 1-3 years.

Related Work

One standard approach in dealing with time-series data was the use of frequency domain regression, with band-spectrum selection (Harvey, 1978). This model assumes that the disturbances from the mean are periodic and can effectively capture features like seasonal sales fluctuations in weather-related equipment sales (Wilson, Reale and Laywood, 2015).

Other papers have also investigated the use of support vector machines for time series forecasting (Muller and Vapnik, 1999) . Specific examples include electricity load prediction, as conducted by researchers from National Taiwan University (Hu, Bao and Xiong, 2013)

Lastly, groups have explored the use of neural networks for the same purpose (Connor, Martin and Atlas, 1994). We did not investigate this owing to the lack of resources, but this is a promising area for further research.

Data Set

We were provided with data on 1115 stores located across Germany. The data included sales records for each store over the course of 942 days, giving us a total of about 1 million data points. A second set of data included additional information on the model of the store, assortment of goods sold and presence of competitors in the area. We believe that this is sufficient data for our purposes. A summary of the raw data is shown in Table 1 below:

Cross Validation: We divided 70% of the training examples into the training set, and used the remaining 30% as the test set. We chose the first 70% of training examples in chronological order, since we wanted to test our models on their ability to extrapolate on dates outside of their given range.

Data Preprocessing: There we some general steps to take, including numerizing all data, calculating the day of the week, month of the year and so on. We also had to clean up the data for routine store closures. Further data processing was done differently for Frequency Domain Regression and SVR.

1. We observed that some of the shops were closed for ex-

Table 1: Raw Data Fields

Field	Value Range
Store ID	1 – 1115
Date	1 Jan 2013 – 31 July 2015
Sales	\$0 – \$41,551
Customers	0 – 7388
Open	0,1
School Holidays	0,1
State Holidays	0,1,2,3
Store Type	a,b,c,d
Product Assortment	a,b,c
Competitor Distance	100 – 76,000 meters
Date Since Competitor Open	1 Jan 2013 – 31 July 2015
Promo1	0,1
Existence of Promo2	0,1
Date Since Promo2 Began	1 Jan 2013 – 31 July 2015
Months with Promo2	Jan – Dec

tended periods of time and that these data-points with 0 sales affected our predictive algorithms. Since we only want to assess the algorithm's predictive power on days when the stores were open, we opted to remove the days when a store was closed from the data.

2. For SVR: We first normalized the sales by subtracting the mean sales for each store from the store's sales numbers. The mean sales number was retained as a parameter. This allows us to focus the SVR on impact of events on disturbances from the mean. Then, we made numeral/boolean the variables of the store type and product assortment, whether there was a promotion, a specific holiday, a competitor opening, and how long it has been open for.

3. For FDR: We treated the event variables as a blackbox and simply worked with sales versus time for each store.

Methodology

We used three different approaches in order to predict store sales with respect to time. Each method was trained on data from a single store, and then used to predict the sales for that particular store. This process was subsequently repeated for every store.

First we used linear regression to obtain a baseline for the prediction and to capture any inter-year trends which we may want to use to further normalize the data for SVR and FDR. The parameters for linear regression were found by using MATLAB to solve the normal equations

$$\theta = (X^T X)^{-1} X^T y$$

We then ran a discrete Fourier Domain Regression to construct a regression model for the periodic time series behavior. In short, this method attempts to model the sales y on a particular day x over the time period N by choosing the top k frequencies as shown in the following equation:

$$y(x) = \mu + \sum_k (A_k \cos \frac{2\pi k x}{N} + B_k \cos \frac{2\pi k x}{N}) + \epsilon(x)$$

We also used Support Vector Regression, which we felt was better suited to predict the effects of events, such as promotions and holidays, on store sales. For the training set $\{(x^{(1)}, y^{(1)}), ..., (x^{(m)}, y^{(m)})\}$, where $y^{(i)}$ is the sales for a particular point in time $x^{(i)}$, we seek to find a hypothesis of the form $h_{w,b}(x) = w^T x + b$ with a small value of w. Our optimization problem is

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2}||w||^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) \\ \text{s.t.} \quad & y^{(i)} - w^T x^{(i)} - b \leq \epsilon + \xi_i \quad i = 1, ..., m \\ & w^T x^{(i)} + b - y^{(i)} \leq \epsilon + \xi_i^* \quad i = 1, ..., m \\ & \xi_i, \xi_i^* \geq 0 \end{aligned}$$

Where $\epsilon > 0$ is a given fixed value. We solved this problem with the aid of of the support vector regression function in the scikit-learn package. We can then avoid underfitting or overfitting of the training data via conventional-validation on the variable C to control the slack allowance through the term $C\sum_{i=1}^{l}(\xi_i + \xi_i^*)$, as well as a choice between linear, polynomial or gaussian kernels.

Error Metric

The metrics that we used for our analysis was the root-mean-squared-percentage-error (RMSPE), which is calculated as

$$\text{RMSPE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{y_i}\right)^2}$$

where n is the number of days,y_i is the sales of a store on a single day and $\hat{y}_i$ is the corresponding prediction. This is a number used operationally in inventory planning and is hence pertinent to the problem at hand.

Experiment Results and Discussion

We only managed to train/test on a limited number of stores owing to computation resource constraints. Our observations for linear regression, FDR and SVR are summarized in Table2.

Table 2: Test errors for training methods

Method	Test Error
Linear Regression	30.2%
Frequency Domain Linear Regression	29.1%
Support Vector Regression	17.4%

Perhaps unsurprisingly, the more powerful SVR, with a degree-2 polynomial kernel and $C = 1000$ significantly outperformed LR. However, it was a surprising that FDR did so poorly - in many cases even worse than linear regression on the test data. We analyze this result below.

Linear Regression

This model is a rudimentary first look into the large scale trends. We did not expect it to capture the granular movements of the sales numbers and indeed it didn't, reporting an average RMSPE of 30.2%. Shown in Figure 1 below is the trend observations for a single store over the relevant time period.

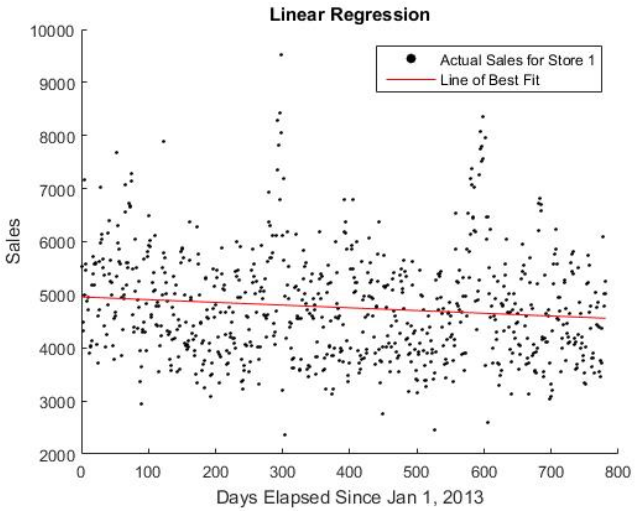

Figure 1: Linear Regression

We observe that there are minimal inter-year trends and therefore can safely disregard them in future considerations.

Frequency Domain Regression

The results of FDR are as follows:

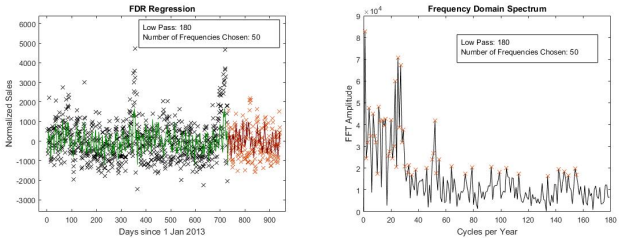

Figure 2: Frequency Domain Regression Results

The green line and points correspond to the training data and FDR trendline. The red line and points are the extrapolated prediction plotted against the test set. In the Frequency Domain graph, the black line shows the amplitude of each frequency, plotted as Cycles per Year. The red crosses denote the frequencies chosen for the regression model.

The use of FDR was motivated by the periodic movement of sales data over 2-3 weeks, which becomes evident when we plot over a period of 2-3 months. Shown in Figure 3a at the top is periodicity observed and the corresponding fitted trend over *training data*. The plot in 3b at the bottom shows the performance of the model with a chosen set of extrapolated test data.

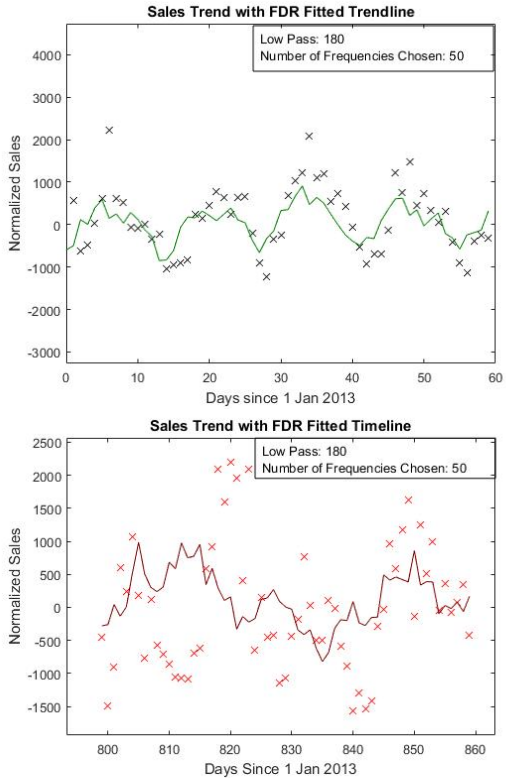

Figure 3: 2 Month Plot with FDR Line

Clearly, the model seemed to be almost shifted by half a phase from the actual periodicity of the data. Although it fitted the training data extremely well, its extrapolated performance was unacceptable. We believe that it is due to the following factors:

1. The periodicity was *not due* to unknown weekly or fortnightly factors but due to company driven actions - Promo1. Later, when plotting Promo1 against the sales, we realized that the sales increases when the boolean Promo1 is true. In this sense, there is no true periodic factors with consistent phase and frequency. The company has introduced periodicity on its own schedule.

2. The model is numerically unstable. Even if there is a natural periodicity, a small error in frequency is equivalent to a beats phenomenon in harmonic analysis, causing the model to be eventually π out of phase with the original trendline after some time. Given the low resolution (14 data points per cycle) of each period, the inherent imprecision is too big given our extrapolation time span.

With the knowledge that the periodicity is event-driven, we then approached the problem with SVR.

Support Vector Regression

From LR and FDR, we now know that the sales is unexpectedly lacking in event-agnostic time-series behavior. Virtually all movements in the sales volume are event-driven. Thus, we stripped the data of all time-based information - save for holidays, which we store as boolean variables - and plugged the entire dataset into an SVR.

Here, a stumbling block was the computational complexity of SVR. As a rule of thumb, the big-O for SVR algorithms are given (Chapelle, 2007)

$$\text{Gaussian Kernel: } T(n,k) = O(n^2 k)$$

$$\text{Linear Kernel: } T(n,k) = O(nk^2)$$

Where n is the size of the training set and k is the number of features. By this estimate, with 700,000 training data points, this was clearly unfeasible. An approximate SVM algorithm $T(n,k) = O(k^2)$ exists (Claesen et al, 2014), but we did not have time to try this.

We did however analyze a single store of type b and product assortment b, which is the most representative among the population of stores. The results are summarized in Figure 4, 5 and 6.

As we increase C, we are forcing the SVR algorithm to work with smaller slack variables. Thus, as C increases, we expect the training errors to fall.

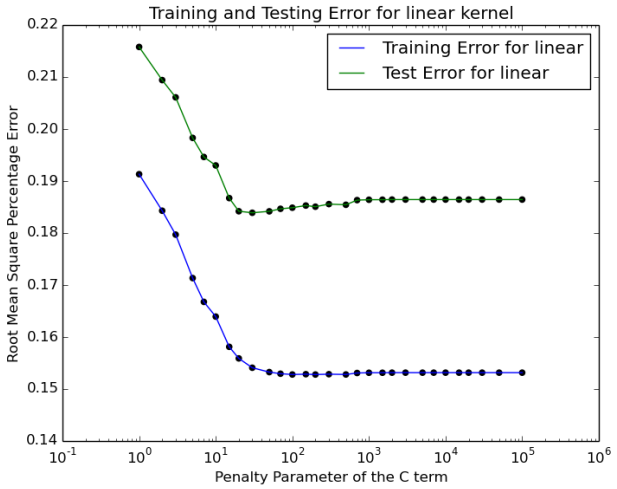

Figure 4: SVM with Linear Kernel Error Plots

The Linear Kernel is seen to be unable to increase its accuracy, despite our increasing C. Our results suggest that after $C \geq 10$, all the SVR is able to achieve is a great cost function without changing the underlying regression line. The prediction error thus plateaued in a high-bias situation.

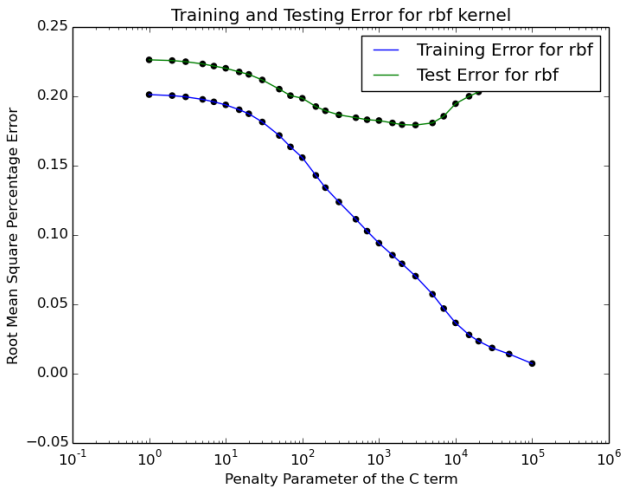

Figure 5: SVM with Gaussian Kernel Error Plots

The Gaussian Kernel on the other hand exhibited clear high-variance behavior. It was able to incrementally improve its performance on the training data as we increase C. Indeed, it has almost perfected fitted the training set at $C = 10^5$. Unfortunately, the test error did not follow and only reached an optimal at around $C = 500$.

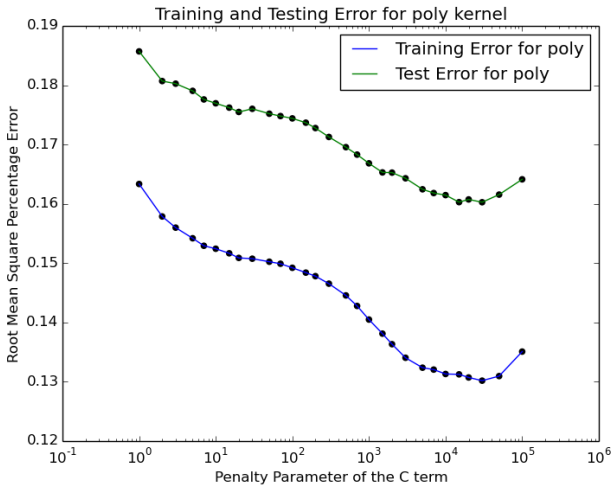

Figure 6: SVM with Linear Kernel Error Plots

Clearly, Polynomial Kernels outperformed both Gaussian and Linear Kernels. It seems to be the optimal model, with $C = 1000$, in the bias-variance trade-off. Here we note an anomaly in the Polynomial Kernel, where the training error briefly rose at $C = 10^5$. This is cause for further investigation beyond this report.

We are also concerned that the Gaussian kernel underperformed Polynomial kernels and would like to look deeper into this phenomenon.

Conclusion

Our highest performing method was the support vector machine, which displayed the lowest amount of testing error, and the worst performing method was linear regression. For the support vector machine we achieved the best results with the polynomial kernel, which achieved the best balance between overfitting and underfitting. As expected, linear regression did not perform very well due to the non-linearity of the data. Unfortunately, the frequency domain linear regression model failed to work as well as we had hoped, due to the fact that the factors driving the sales were not really periodic in nature, but rather due to company-driven promotions.

There are numerous areas for future work. We may include additional features in relation to some of the boolean variables in order to improve our models. For example, a more careful treatment of the promotional data would have possibly improved the prediction power of our algorithms, since the presence of Promo1 seems to be more closely related to the first derivative of sales.

We could have also implemented the $O(k^2)$ approximation scheme for the Gaussian kernel, so that our model can scale to the full 700,000 data set.

Additionally, it would be good to investigate the anomaly observed. A more thorough investigation of the tradeoffs between using a polynomial kernel and a Gaussian kernel would possibly have allowed us to optimize the accuracy of our model.

In the future, we also hope to explore the usage of neural networks in time series prediction, since they are also a widely used method for time-series prediction. Given the amount of data we have, this can be very promising. It will also be interesting to compare their performance with the other methods at hand.

Acknowledgments

We would like to thank our mentor Bryan McCann for his helpful input on this project.

References

Chapelle, Olivier. "Training a support vector machine in the primal." Neural Computation 19.5 (2007): 1155-1178.

Chen, Bo-Juen, Ming-Wei Chang, and Chih-Jen Lin. "Load forecasting using support vector machines: A study on EUNITE competition 2001." Power Systems, IEEE Transactions on 19.4 (2004): 1821-1830.

Claesen, Marc, et al. "Fast prediction with SVM models containing RBF kernels." arXiv preprint arXiv:1403.0736 (2014).

Connor, Jerome T., R. Douglas Martin, and Les E. Atlas. "Recurrent neural networks and robust time series prediction." Neural Networks, IEEE Transactions on 5.2 (1994): 240-254.

Harvey, Andrew C. "Linear regression in the frequency domain." International Economic Review (1978): 507-512.

Kaggle, 2015. https://www.kaggle.com/

Hu, Zhongyi, Yukun Bao, and Tao Xiong. "Electricity load forecasting using support vector regression with memetic algorithms." The Scientific World Journal 2013 (2013).

Muller, Klaus Robert, et al. "Using support vector machines for time series prediction." Advances in kernel methodssupport vector learning, MIT Press, Cambridge, MA (1999): 243-254.

Smola, Alex J., and Bernhard Schlkopf. "A tutorial on support vector regression." Statistics and computing 14.3 (2004): 199-222.

Wilson, Granville Tunnicliffe, Marco Reale, and John Haywood. Models for dependent time series. Vol. 139. CRC Press, 2015.