Forecasting sales using store, promotion, and competitor data

Qianren Zhou
Computer Science and Engineering
University of California, San Diego
qiz122@ucsd.edu

Kun Huang
Computer Science and Engineering
University of California, San Diego
kuh004@ucsd.edu

Di Huang
Computer Science and Engineering
University of California, San Diego
dih024@ucsd.edu

Abstract

Sales forecasting is a common topic in business. Our task is predicting a famous drug company daily sales for 1,115 stores located across Germany for six weeks in advance. Store sales are influenced by many factors. Our project aims to create a robust prediction model. Based on Gradient Boosting and Random Forest, our model performs well in this sales forecasting competition with resulting in ranking at 977th/3066 and it is found that Competition had opened days, the day, competition distance affect drugs sales the most.

Keywords

Sales Forecasting; Random Forest; Gradient Boosting; Time Series Analysis

Introduction

Rossmann operates over 3,000 drug stores in 7 European countries. The task is to predict 6 weeks of daily sales for 1,115 stores located across Germany. Store sales are influenced by many factors, including promotions, competition, school and state holidays, seasonality, and locality. Reliable sales forecasts enable store managers to create effective staff schedules that increase productivity and motivation.

As the given feature is only store related and there is no customer related data or item related data, recommendation system is hard be applied in the problem. Besides, as the target is continuous, the problem could be a regression problem based on both categorical feature(e.g, Store Type) and continuous feature(e.g, Days). Besides, some feature could be considered as categorical as well as continuous, for example, Day in week could be considered as continuous by [1,2,3,4,5,6,7], assuming there is a relationship for adjacent days or as categorical [Mon, Tue, Wed, Thu, Fri, Sat, Sun], assuming there is not relationship between them.

In the following sections, the selection and construction of features, and how previous sales forecast problems are solved, and the models fits best into this problem will be discussed.

Table 1
Dataset statistic

STATISTICS	NUMBERS
Dataset size	1017209
Testing data size	41088
Total stores number	1115
Training data Time ranges	2013-01-01 to 2015-07-31
Testing data Time ranges	2015-08-01 to 2015-09-17

Table 2
Sales statistic

STATISTICS	VALUES
Global store sales average	5773.82
Max daily sales	41551
Min daily sales	46

Dataset Identification and Analysis

Our dataset comes from kaggle competition "Rossmann Store Sales". The training data includes "Store", "Day Of Week", "Date", "Sales", "Customers", "Open", "Promo", "State Holiday" and "School Holiday".

The following is a sample of training data.

1,5,2015-07-31,5263,555,1,1,"0","1"

The testing data doesn't include the fields of "Sales" and "Customers". One is the field we need to forecast and the other is the field which we can't know in advance.

The following is a sample of testing data.

1,1,4,2015-09-17,1,1,"0","0"

The Table 1 shows some general statistics for the data.

The Table 2 shows statistics related to sales.

Previous Work

A lot of work has been done related to sales prediction, as it's one of the most important concern by the retailer. Sales prediction could be done by customer related features, store related features, and item related feature. For example, (Chen, Lee, Kuo, Chen, & Chen, 2010) has been working on forecasting sales model on fresh food. The prediction is based on both item and customer, as when consumers are making purchases of food products, they would first consider if the foods are fresh and if they are expired. But the methodology can't feed into our problem, as only store related features are provided and customer-item prediction could not be predicted at all. Another retail sale prediction problem has been described in (Giering, 2008). It's also a sales prediction problem based on customer related feature and item related feature where SVD and recommendation system is applied. Although the methodology can't be well applied in our problem, there is still inspiration from their work: using log(Sales) as the target in prediction as it might normalized the distribution.

(Chang, Liu, & Lai, 2008) has described a way to make sales prediction only based on item related feature which is more similar to our problem where only store related feature is provided. In (Chang et al., 2008), it obtained *case-based reasoning model* and *k-nearest neighbors algorithm* to find the most similar item with sale history, given an item without sale history. We tried *k-nearest neighbors algorithm* in our dataset to find the most similar store and time information. We got some result, however, its performance is not as good as expected. The model and the result could be find in the following section.

(Thiesing, Middelberg, & Vornberger, 1995) has adapted *Back-Propagation* as a neural network method to make sales prediction on Transputer system. The article has also described how they applied parallel computing into the model to improve the efficiency of computing.

Data Exploratory

The following subsections are trying to analyse dataset and figure out useful features that can be used to forecast sales. At first, we will attempt to extract features from training dataset. Then, in order to get more useful features, the store information will be reviewed. At last, we will try to get more information from what we have now based on store information.

Store ID

It is customary to think the store ID as one feature because sales may change from store to store. However, if we just use the Store ID as one feature, we will find that the correlation coefficient between Store ID and Sales is only 0.005.

One idea to make it better is using store daily sales average as the feature instead of store ID. The coefficient between store daily sales average and Sales is 0.53, which is pretty high in analysis. We can see the store daily sales average in Figure 1

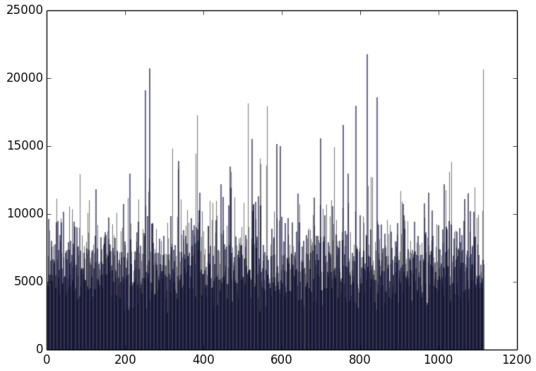

Figure 1: Store Daily Sales Average

Day of Week

It's also easy to think that in different day of week, every store will have different sales since people get used to shop in differert days. The day of week seems to have large effect on sales as we can see that the correlation coefficient is -0.46.

Also, for day of week, we calculated the daily sales average of each store in each day of week and the correlation coefficient grows up to 0.85.

Numbers of Customers

In the training dataset which represents the past information includes the numbers of customers. However, in the test dataset which provides the future information doesn't have the numbers of customers. It's logical. Therefore, in order to try to use numbers of customers as one feature, we calculated the monthly average numbers of customers for each store as one feature. However, the correlation coefficient is only 0.06.

Open

"Open" shows whether this store is open or not in a specified day. Because the sales must be 0 when the store is closed, we removed the data point with "Open = 0" and after prediction, we would set the value of sales as 0 for the data point with "Open = 0" in testing data.

State Holiday

People will have different demands for drugs during holidays. We have the information of state holiday for each store in each day. The correlation coefficient is -0.23. It shows some correlation between state holiday and sales.

School Holiday

We do the same thing with school holiday. However, it shows a weak correlation with sales. This correlation came out to be 0.09.

Store Type

There are 4 different store models. We encode them as 1,2,3,4. There are 1: 602, 2: 17, 3: 148, 4: 348 stores in each type store. However, The correlation coefficient is -0.01.

Then we attempted to calculate the average sales for each type of stores.

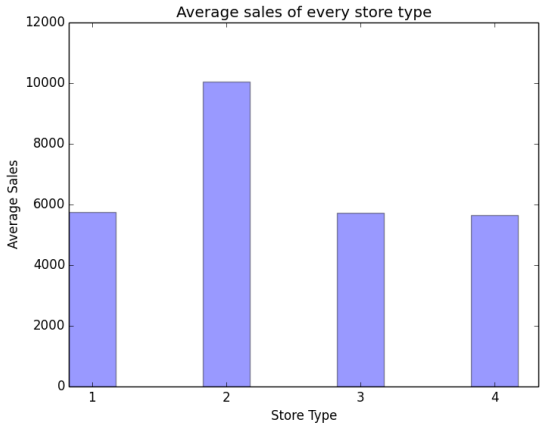

Figure 2: Average sales of every store type

The correlation coefficient goes up to 0.14, which still shows a weak correlation with sales.

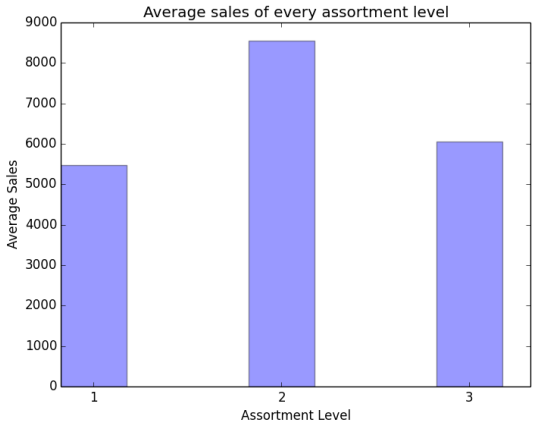

Figure 3: Average sales of every assortment level

Assortment

Because different stores have different assortment level, we also take assortment as a feature. The correlation coefficient is 0.07. The average sales for each kind of assortment model stores is showed below.

The correlation coefficient goes up to 0.10.

Year

The sales could have relationship with years, since the brand influence of this company may change from year to year, which could affect the sales.

Month

Because people may tend to have a cold in winter and easy to get sunstroke in summer, people would have different demands for durgs during different month. So, the sales may be affected by month.

Day

Each single day could affect the sales. For instance, people may tend to buy drugs in the first day of month or they might go to stores when today is payday.

Competition Distance

It's customary to think that if there is a competition store near us, our sales could be affected since some customers would choose the competition store instead of us.

Competition had opened days

As we all know, when a competition store open, our sales must be affected since customers may be attracted by the new stores.

Promo

It indicates whether a store is running a promo on that day. As promotion would be a attracting thing for customer, it might have effect on the sales. But, we could say that because drugs are not luxuries or daily supplies, the promo wouldn't affect the sales too much.

Promo2

Promo2 is another promotion option which need to be calculate from store information. By combining *Promo2*(continuing and consecutive promotion for some stores: 0 = store is not participating, 1 = store is participating), *Promo2Since[Year/Week]*(describes the year and calendar week when the store started participating in Promo2), and *PromoInterval*(describes the consecutive intervals Promo2 is started), we could get *Promo2* which indicates whether the store has promotional 2 in specific day. Similarly to *Promo*, it might have affect on the sales as well, but not too much.

Store Continuously Opened Days

For instance, customers and Sales may increase in the first open day. So sales might have relationship with store continuously opened days.

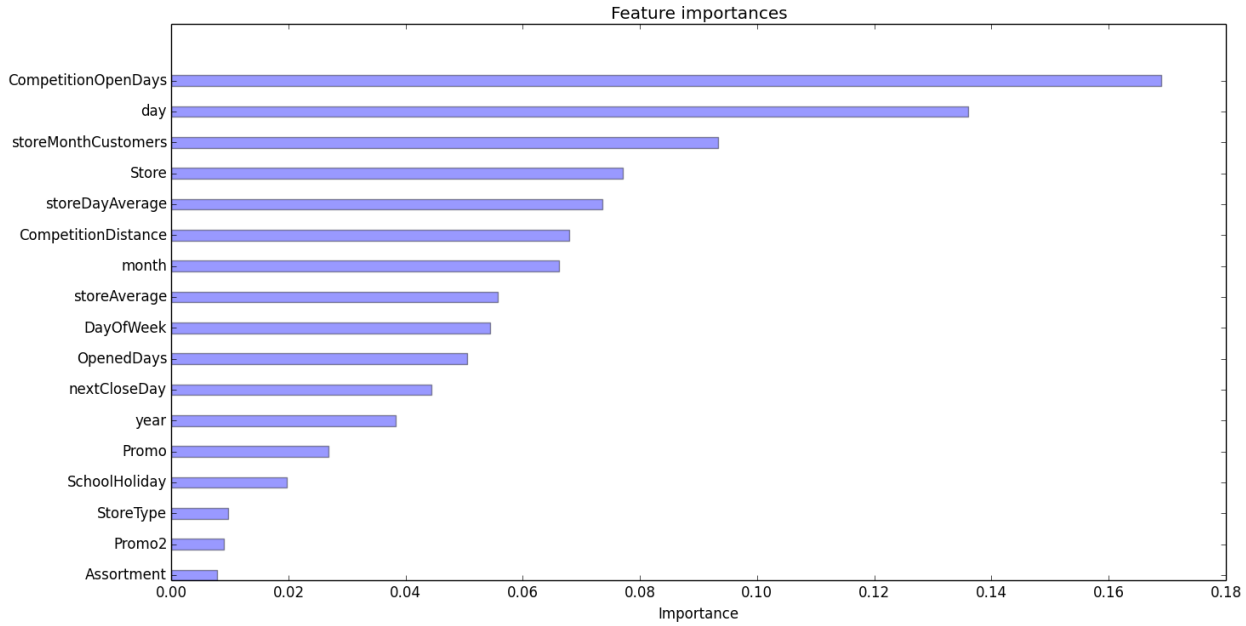

Figure 4. feature importance

Store next close day

Customers may go to the store before it's closed to buy some drugs for backup. So, the store next close day may also affect the sales.

Model Selection

We tried several models in our experiment. The first model we tried is the linear regression. The prediction score is 0.38971 and we set it as the prediction baseline. We skipped other prediction method like logistic regression, SVC regression and directly forwarded to ensemble learning methods, which has been proved to be the best among all other approaches in our assignment1.

The basic idea behind ensemble learning is like this: given the training data, ensemble learning uses multiple learning algorithms to find the best representative model for the specific amount of data. So the actual regressor method may be different for different bulk of data. Two of the most representative ensemble learning method are the random forest tree and gradient boosting tree.

Random Forest

Random Forest Tree tries is to construct a multitude of decision trees. Then it classifies the data into the decision tree node and for each node it calculate the mean value and use this value for prediction. Random forest tree uses random amount of data for training. With this randomized data, it is hard for random forest tree to overfit. Therefore, it is much more easier to tune the data compared to Gradient Boosting Tree.

Gradient Boosting

Like other boosting method, Gradient Boosting Tree also generate decision trees. However, the key idea behind Gradient Boosting Tree is that it can generate the tree as an optimization algorithm on a suitable cost function. With this idea, we can define our own loss function for optimization, which is vital for optimizing the final parameters. In random forest tree, it is hard to define our own loss function. The default loss function is to optimize the mean absolute error while the competition score is calculated by root mean square percentage error. False loss function will bias the prediction parameters. Finally we decided to use the Gradient Boosting Tree.

Unfortunately, Gradient Boosting Tree has its own drawbacks. First, It is more likely to overfit than random forest. It trusts every data point and tries to find optimal linear combination of trees given the train data. Second, there are more parameters in Gradient Boosting Tree. We have to tune the number of trees, maximum depth and the learning rate at the same time. For the first drawback, we can try to avoid overfitting by using the cross validation provided by sklearn. For the second method, we have to spend more time to tune it, which is tradeoff between time and accuracy.

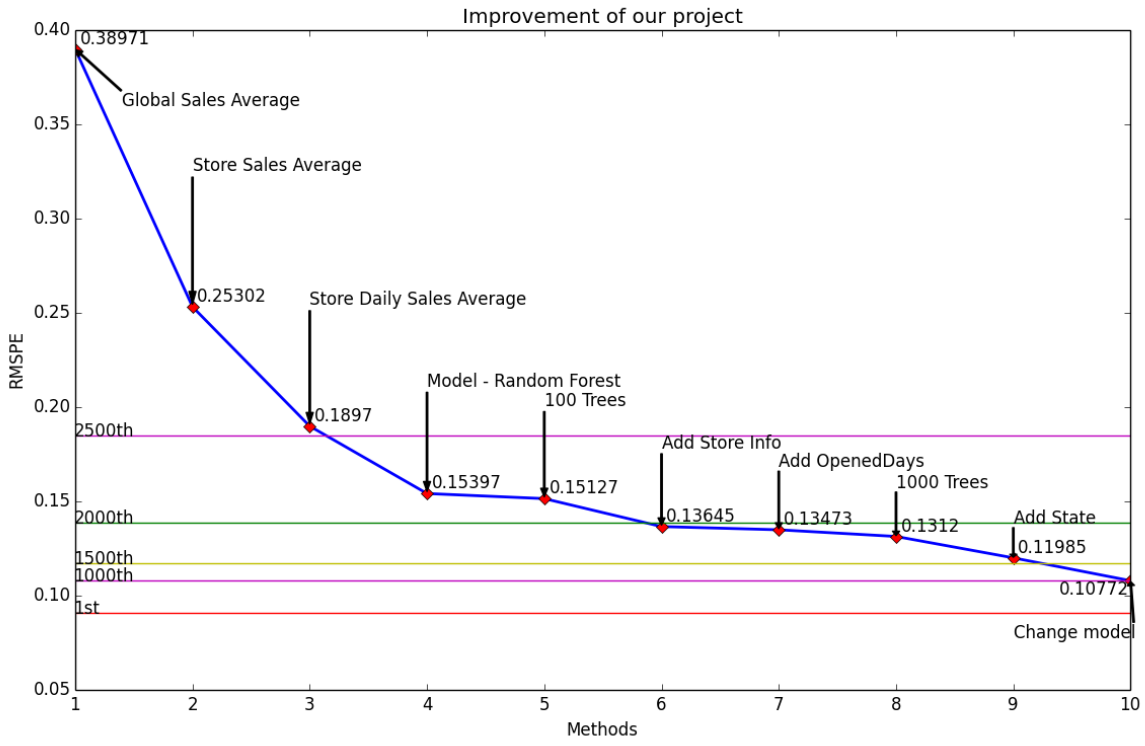

Figure 5. Improvement of our project

Implementation

We learned some lessons from the previous project on assignment1. The previous implementation simply applied the sklearn packages for training and prediction, which is super slow. This time we use the xgboosting, which is faster since it is implemented in C++. Another important feature of this package is that it supports parallel computing so we can start multi-core optimization for our program.

We first use all the features in our analysis with tree size = 10000, max_depth = 13 and learning rate = 0.1. We then output the feature importance in this model as shown in Figure 4

After that, we choose the 8 most influential feature for further fine-grained training and tuning. We finally get our parameters set as: tree size = 3000, max_depth = 10 and learning rate = 0.2.

Result

Progress curve

The improvement of our project can be seen in Figure 5

Scores

The best score is around 0.09059.
Our final score is 0.10772.

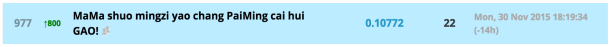

Figure 6: Score

Rank

There are 3066 teams in total. We rank at 977th / 3066.

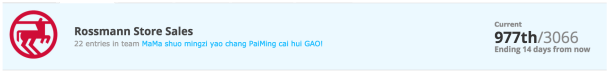

Figure 7: Rank

Result Analysis

The top 5 important features are: CompetitionOpenDays, day, store, StoreMonthCustomers, StoreDayAverage. It is easy to understand that the days since the competitor opening has a negative influence on the Sales. Usually there would be a lot of discounts on the first several days of new competitor opening. The Sales of the previous store will have a drastic

Table 3
Model Comparison

Model Selection	Score
Linear Regression	0.3897
Logistic Regression	0.3328
KNN(20 neighbors)	0.3058
Random Forest(1000 trees)	0.1199
Gradient Boosting(1000 trees)	0.1077

decrease on Sales. But as the time goes, the influence would decrease. The day and store are the original information from the dataset. It is obvious that different stores would have different sales trend. What surprises us most is that the day in a month has such a big influence on the prediction. We can not find a reasonable explanations on it. We guess that the sales have a seasonal fluctuation. StoreMonthCustomers and StoreDayAverage are the features that we mined in our analysis. We use the average month sales to represent the store's monthly fluctuations. Intuitively, in different day of week, the sales would be different. But for the same day of week, they might have the same trend. For example, weekends are more likely to have more sales compared to weekdays.

Besides feature selection, we also compare with several baseline methods to prove that our model works well. As shown in Table 3, it is obvious that two of the ensemble regressor outperforms other regressor, just like what we have analyzed in the model selection phase. Gradient Boosting is more descriptive for our data so we use it as our main model. We tuned the model and finally get the parameters of tree size = 3000, max_depth = 10. We have to note here that we cannot totally avoid over-fitting even when we use the cross validation. When we set the tree size = 10000, the model is definitely over-fitted since our training data contains only 1 million data points.

Future Work

Since time limited, we have no time to do too much further analysis on the dataset. However, we did some preliminary work which may be useful for future work.

Among all the features, it seems that StateHoliday and SchoolHoliday are the least important features in the model. However, these kind of information may be useful for latent feature mining. According to some description in the Kaggle Forum, different states have different public holiday. Thus we can infer the state of each store based on the public holidays provided by the Calendar. Figure 8 shows the state statistics. These stores distribute in 12 states and a quarter of them are located in Nordrhein-Westfalen. For one aspect, we could use the state as a feature dimension because different states could have different economic growth rate, which will have influence on the sales. What's more, we could infer the weather information based on the state information of each store. For example, bad weather is likely to cause flue and fever, which will stimulate the sales growth.

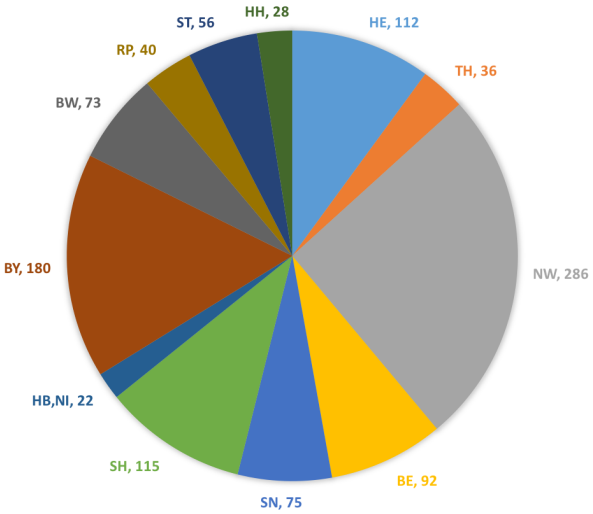

Figure 8: States Distribution

References

Chang, P.-C., Liu, C.-H., & Lai, R. K. (2008). A fuzzy case-based reasoning model for sales forecasting in print circuit board industries. *Expert Systems with Applications*, *34*(3), 2049–2058.

Chen, C.-Y., Lee, W.-I., Kuo, H.-M., Chen, C.-W., & Chen, K.-H. (2010). The study of a forecasting sales model for fresh food. *Expert Systems with Applications*, *37*(12), 7696–7702.

Giering, M. (2008). Retail sales prediction and item recommendations using customer demographics at store level. *ACM SIGKDD Explorations Newsletter*, *10*(2), 84–89.

Thiesing, F. M., Middelberg, U., & Vornberger, O. (1995). Parallel back-propagation for sales prediction on transputer systems. In *Proc. of proceedings world transputer congress' 95* (pp. 1995–318).