Forecasting sales using store, promotion, and competitor data

CHRIS JEE (CJEE),
TEJINDER SINGH (TEJINDER)

Motivation and Problem Description

Motivation

Many businesses that maintain physical stores must predict their sales figure in advance, in order to strategically plan and respond to the market. The sales figure for a given stores is a combination of many underlying factors such as time, promotions, competitions, type and size of stores, etc. Such complex setting provides a good opportunity to apply machine learning based, to create models for predicting the desired features.

Problem Description

Rossmann is a company that operate over 3000 drug stores in 7 European countries. The problem provides sales and related store data from 1115 stores located across Germany.

The main goal of the problem is to create a machine learning based model that can predict 6 weeks of daily sales for each store.

Description of Data

Time series data

The time series data contains each store's time-dependent data for a given day from 1/1/2013 to 7/31/2015. A sample of the data is shown below.

Store	DayOfWeek	Date	Sales	Customers	Open	Promo	StateHoliday	SchoolHoliday
1	5	7/31/2015	5263	555	1	1	0	1
2	5	7/31/2015	6064	625	1	1	0	1
3	5	7/31/2015	8314	821	1	1	0	1

Static Data

The static data includes information about each store, including distance to closest competitor, store type, promo, etc.

Store	Store Type	Assortment	CompetitionDistance	CompetitionOpenSince Month	CompetitionOpenSince Year	Promo2	Promo2 SinceWeek	Promo2 SinceYear	Promo Interval
1	c	a	1270	9	2008	0			
2	a	a	570	11	2007	1	13	2010	Jan,Apr,Jul,Oct
3	a	a	14130	12	2006	1	14	2011	Jan,Apr,Jul,Oct
4	c	c	620	9	2009	0			

Summary of Design

Preprocess Time Series Data:
- Organize the data into 3 dimensional matrix indexed by (StoreId, Date, DynamicFeatureId)

Preprocess Static Data:
- Organize the data into 2 dimensional matrix indexed by (StoreId, StaticFeatureId)

Preprocess Time Series Test Data:
- Organize the test set as same as 2 dimensional matrix indexed by (EntryId, FeatureId)

Hybrid Multivariate Gradient Ascent and SoftMax Algorithm:

1) Dividing into Sets:

- Divide the time series data into 4 sets to in order to take advantage of distributed computing power.

2) Adaptive Binning:

- For each store, quantize the given data set into 40+ quantization levels (bins). The width of bins is controlled. This models the sales data to belong to one of the known discrete values.

3) SoftMaxCostAscent:

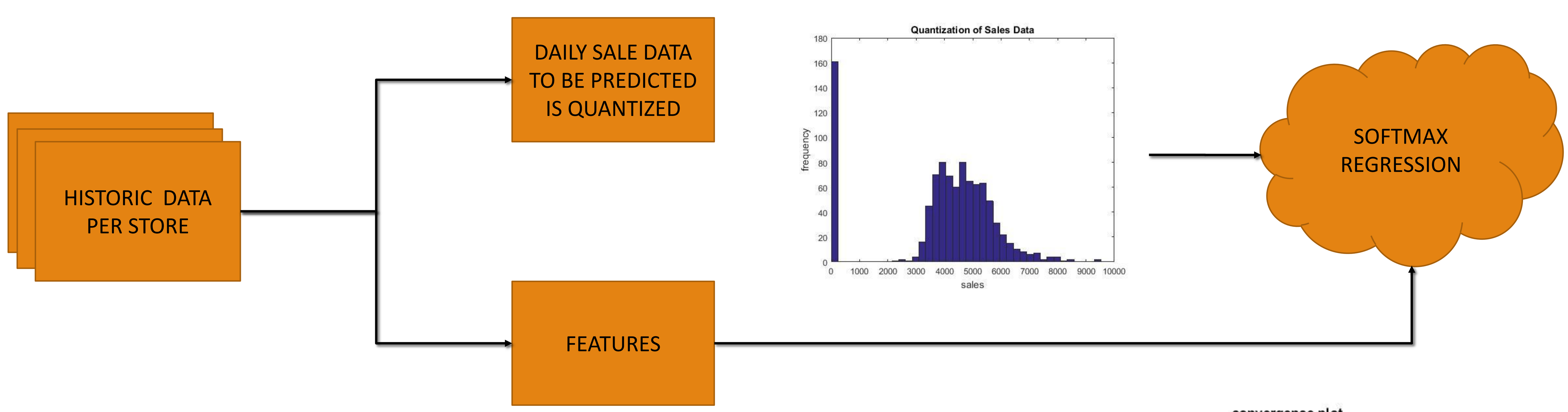
- Use SoftMax method with adaptive learning rate α to maximize the likelihood of the training set and estimate the associated parameters.

- Select the bin with maximum likelihood.

Prediction:

- For each store, predict the sales figure for next 6 weeks based on the estimated parameters.

Detailed Description of Algorithm:

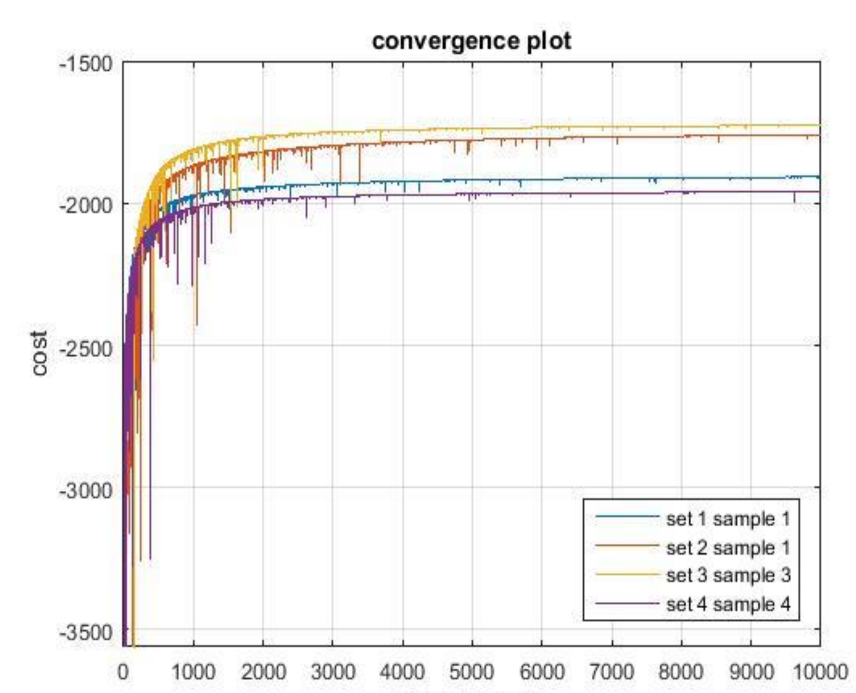


$$p(sale = l|x;\theta) = \frac{e^{\theta_l^T x}}{\sum_{j=1}^{k} e^{\theta_j^T x}}$$

where
x represents features
l is quantization level
k is number of quantization levels
θ_j is parameter to be learned for j^{th} quantization level

$$l(\theta) = \sum_{i=1}^{days} \log \prod_{l=1}^{k} \left(\frac{e^{\theta_l^T x}}{\sum_{j=1}^{k} e^{\theta_j^T x}} \right)^{1\{sale^{(i)}=l\}}$$

The equation shown above is the cost function to be maximized. We used steepest ascent with adaptive learning rate. Every step that results in a lower cost than the current is reversed and learning rate is reduced. The cringes in the convergence plot show this.



Results

RMSE of Predicted Sales Forecast vs Test Data = 0.17305

RMSE of Median Benchmark (www.kaggle.com) = 0.19255

Scope for Improvements

1) We can introduce a heuristic that computes probability of both the single maximum probability bin and the maximum transition probability bin, and dynamically select which ever method that give the higher probability.

2) The current algorithm estimates parameters per each store, and ignores features that apply to different groups of stores. It may be possible to improve the speed of training the model by grouping the stores into clusters and generating models for each groups instead.