Forecasting Sales for Chain Stores

CZ 4032 Data Analytics & Mining - Group 7

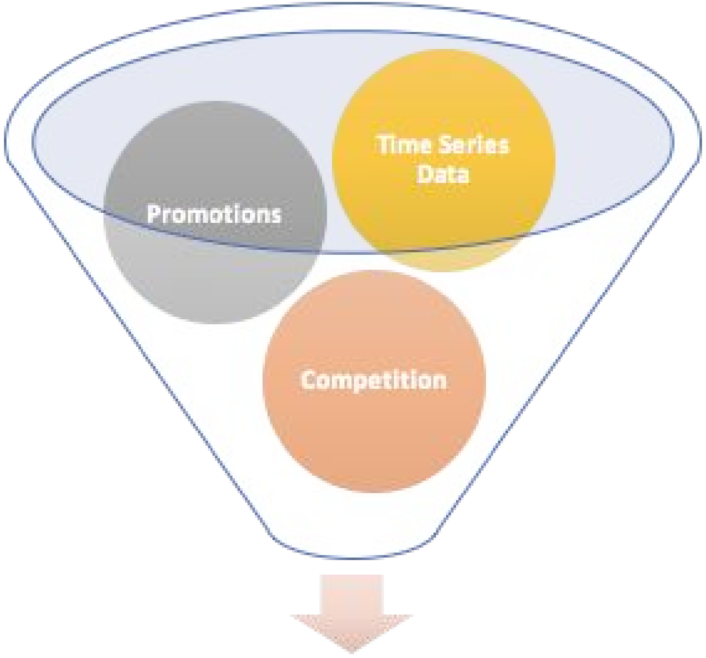
Outline

- ❑ Problem Description
- ❑ Approach
- ❑ Datasets
- ❑ Feature Selection
- ❑ Models
- ❑ Analysis
- ❑ Conclusion

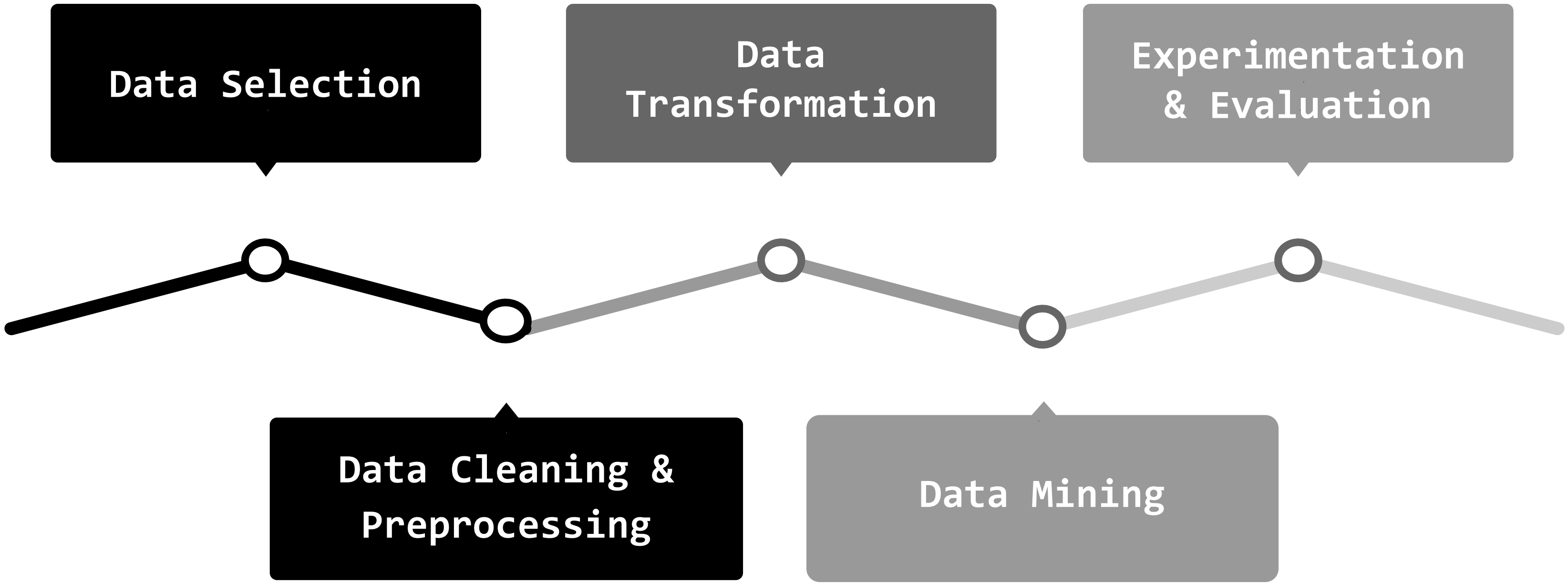
Problem Definition: Forecasting sales for chain stores based on historical data.

Rossmann Store Sales

Data Analysis Challenge: Predict **6 weeks of daily sales** for **1,115 Rossmann Stores** located across Germany to help Rossmann create ideal staff schedules and increase productivity.



High-Level Approach



Datasets

Dataset Summary

1. **train.csv**
   Contains historical data about past sales of each particular store.

2. **stores.csv**
   Contains supplementary information on the 1,115 Rossmann stores.

3. **test.csv**
   Dataset to test the models implemented and get the RMSPE score.

Preprocessing the Dataset

- Data Cleaning
  - Verifying that values for a feature are consistent (i.e. either all strings, all numbers, all characters etc.)
  - Substituting NaN values with the appropriate value based on certain assumptions for `CompetitionOpenSince[X]`, `CompetitionDistance` and `Open`.
- One Hot Encoding for Categorical Features
- Creating New Features (Extraction / Combination)
  - `DayOfMonth, Year, Month, YearMonth & WeekOfYear`
  - `AvgCustStore, AvgCustStoreMonth, AvgCustStoreYear`
  - etc.

Feature Analysis & Selection

Dataset Statistics - train.csv

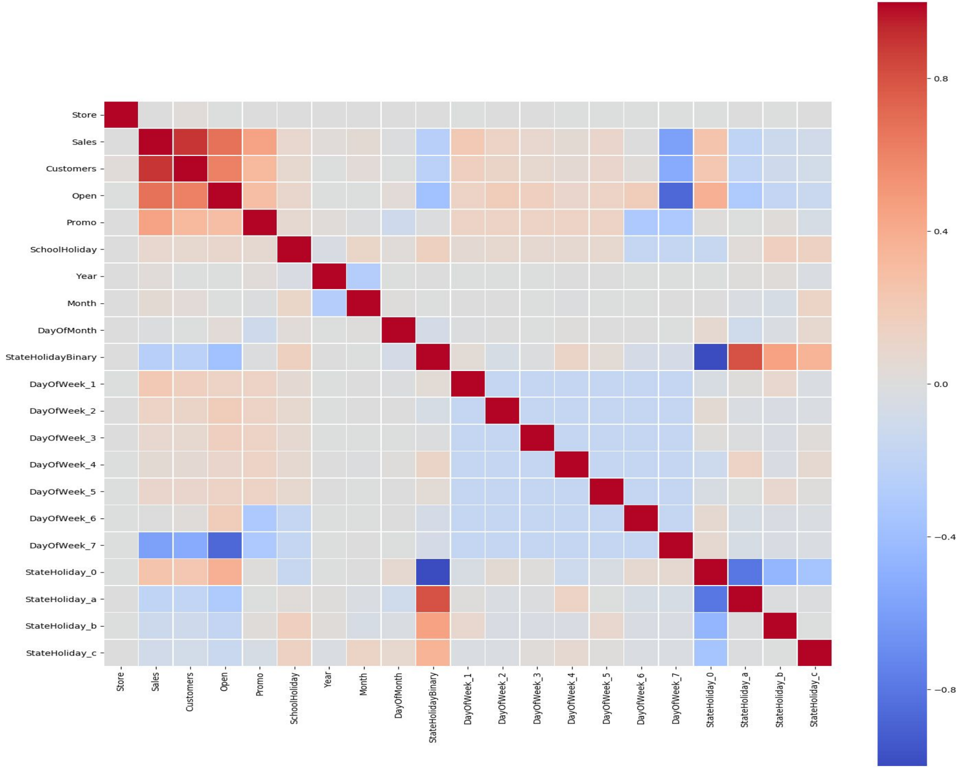
	Store	DayOfWeek	Sales	Customers	Open	Promo	SchoolHoliday
count	1.017209e+06	1.017209e+06	1.017209e+06	1.017209e+06	1.017209e+06	1.017209e+06	1.017209e+06
mean	5.584297e+02	3.998341e+00	5.773819e+03	6.331459e+02	8.301067e-01	3.815145e-01	1.786467e-01
std	3.219087e+02	1.997391e+00	3.849926e+03	4.644117e+02	3.755392e-01	4.857586e-01	3.830564e-01
min	1.000000e+00	1.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00
25%	2.800000e+02	2.000000e+00	3.727000e+03	4.050000e+02	1.000000e+00	0.000000e+00	0.000000e+00
50%	5.580000e+02	4.000000e+00	5.744000e+03	6.090000e+02	1.000000e+00	0.000000e+00	0.000000e+00
75%	8.380000e+02	6.000000e+00	7.856000e+03	8.370000e+02	1.000000e+00	1.000000e+00	0.000000e+00
max	1.115000e+03	7.000000e+00	4.155100e+04	7.388000e+03	1.000000e+00	1.000000e+00	1.000000e+00

Dataset Statistics - stores.csv

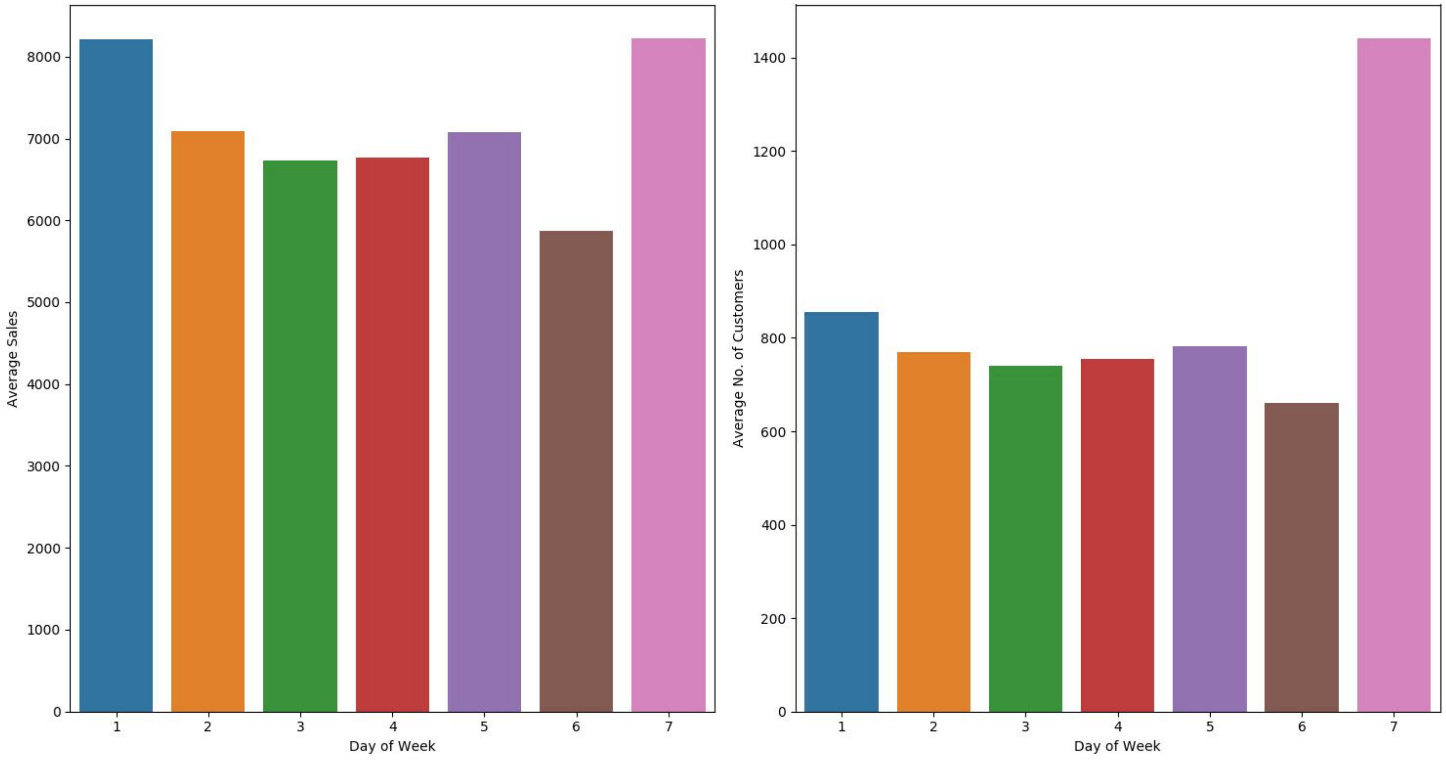
	Store	CompetitionDistance	CompetitionOpenSinceMonth
count	1115.00000	1112.000000	761.000000
mean	558.00000	5404.901079	7.224704
std	322.01708	7663.174720	3.212348
min	1.00000	20.000000	1.000000
25%	279.50000	717.500000	4.000000
50%	558.00000	2325.000000	8.000000
75%	836.50000	6882.500000	10.000000
max	1115.00000	75860.000000	12.000000

	CompetitionOpenSinceYear	Promo2	Promo2SinceWeek	Promo2SinceYear
count	761.000000	1115.000000	571.000000	571.000000
mean	2008.668857	0.512108	23.595447	2011.763573
std	6.195983	0.500078	14.141984	1.674935
min	1900.000000	0.000000	1.000000	2009.000000
25%	2006.000000	0.000000	13.000000	2011.000000
50%	2010.000000	1.000000	22.000000	2012.000000
75%	2013.000000	1.000000	37.000000	2013.000000
max	2015.000000	1.000000	50.000000	2015.000000

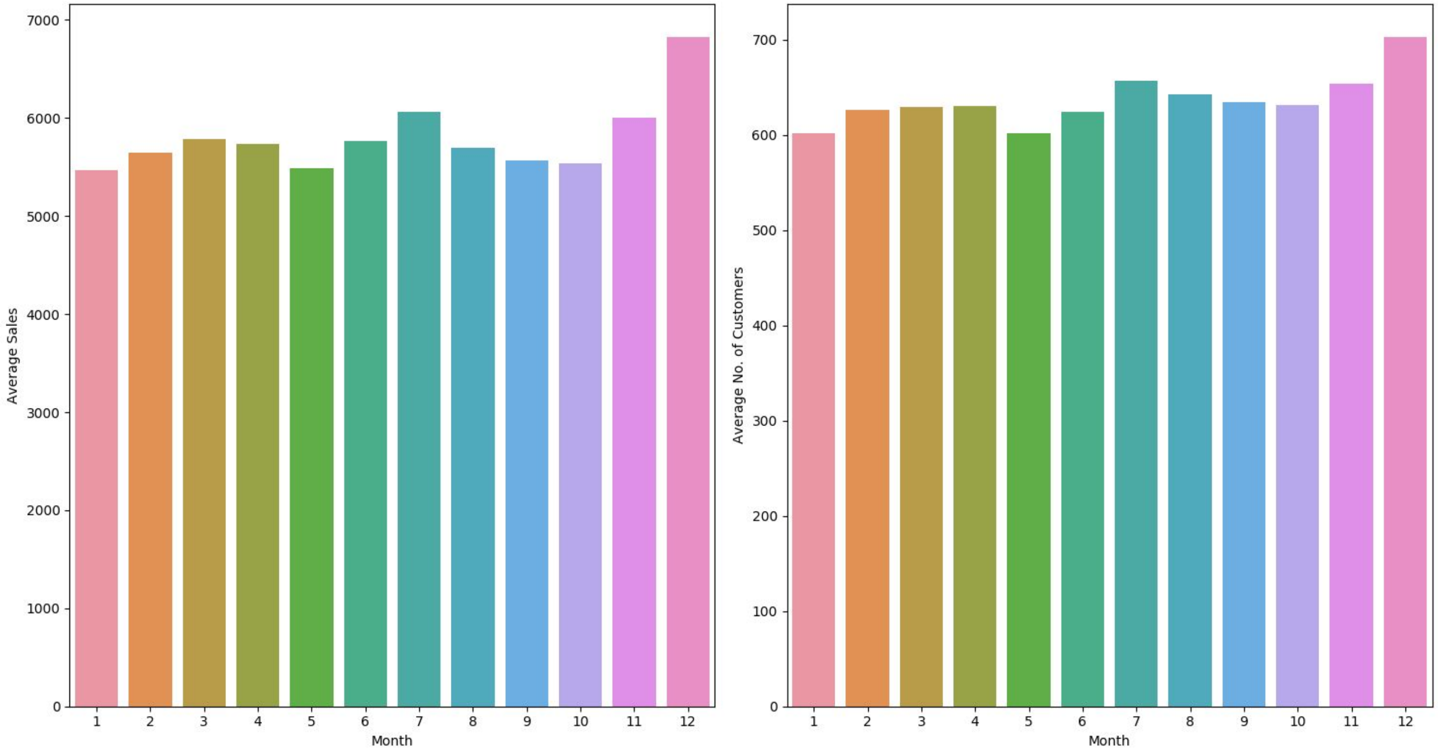
Correlation Matrix



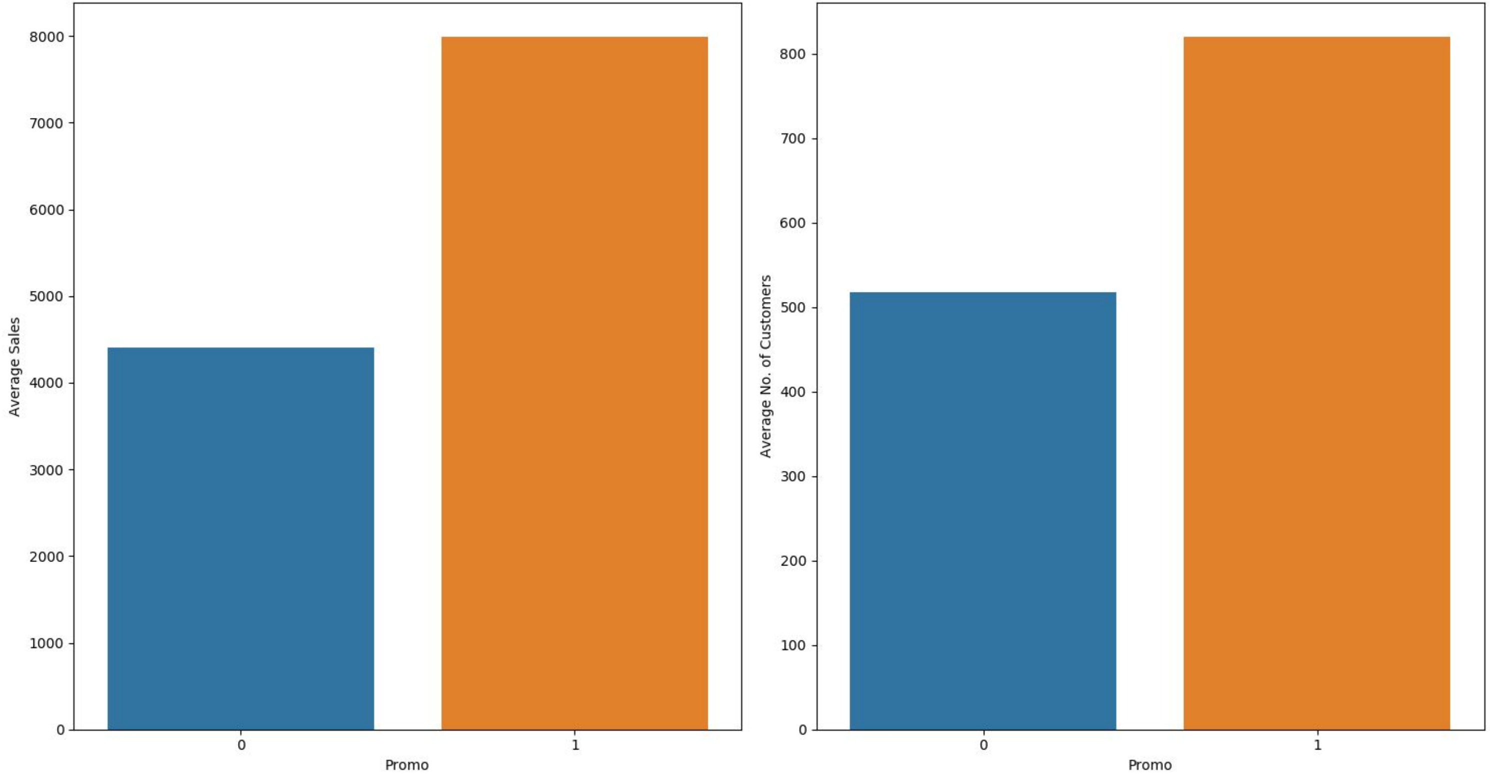
Average Sales & No. of Customers by Day of Week



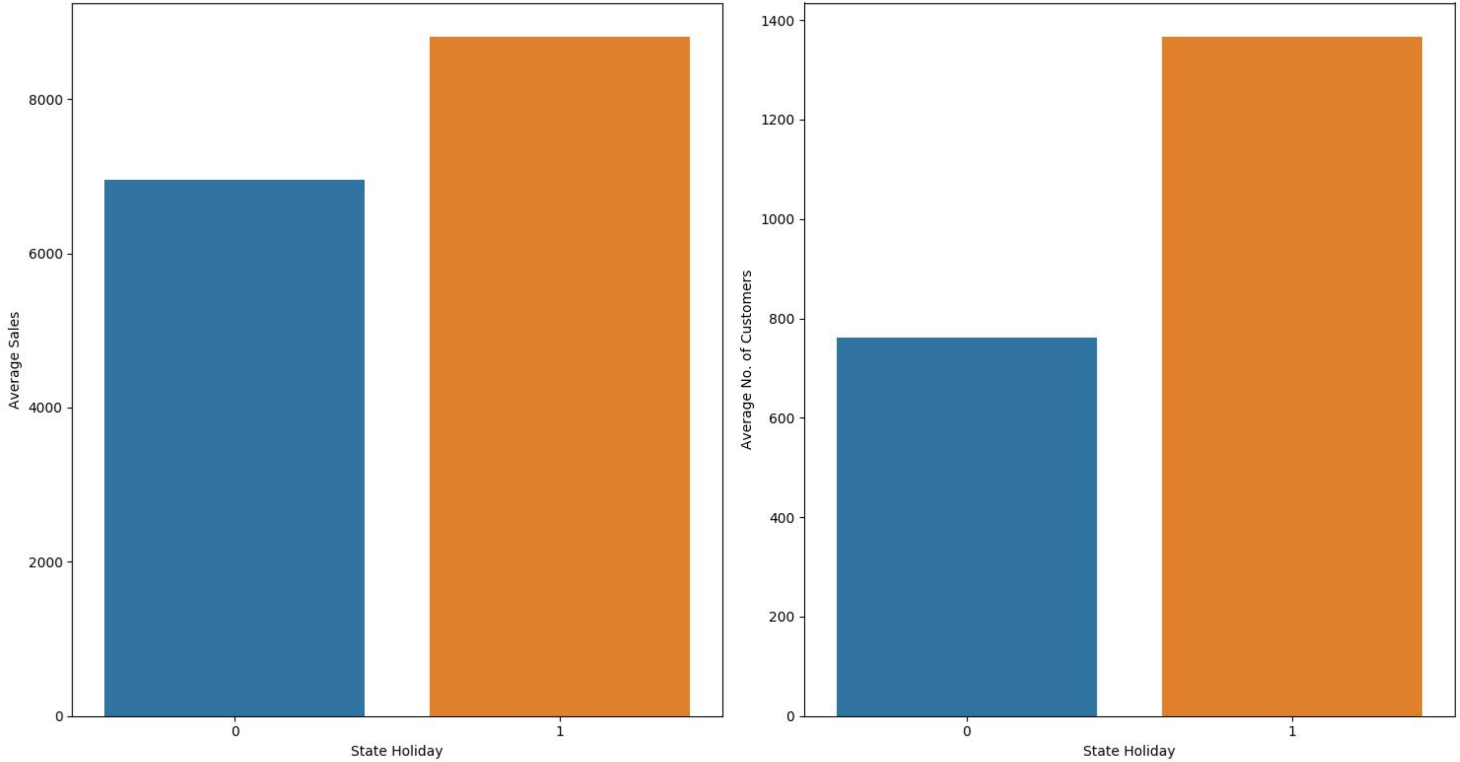
Average Sales & No. of Customers by Month



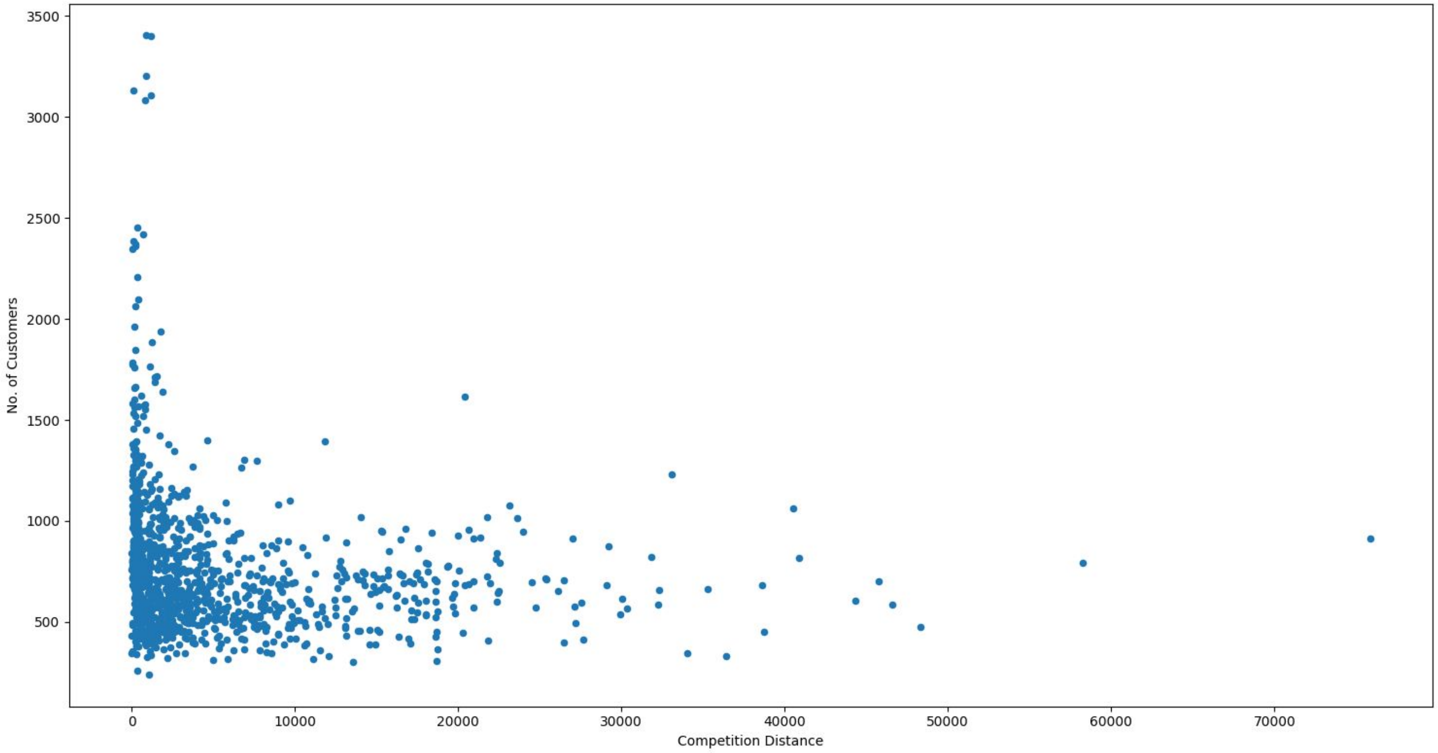
Average Sales & No. of Customers for Promo



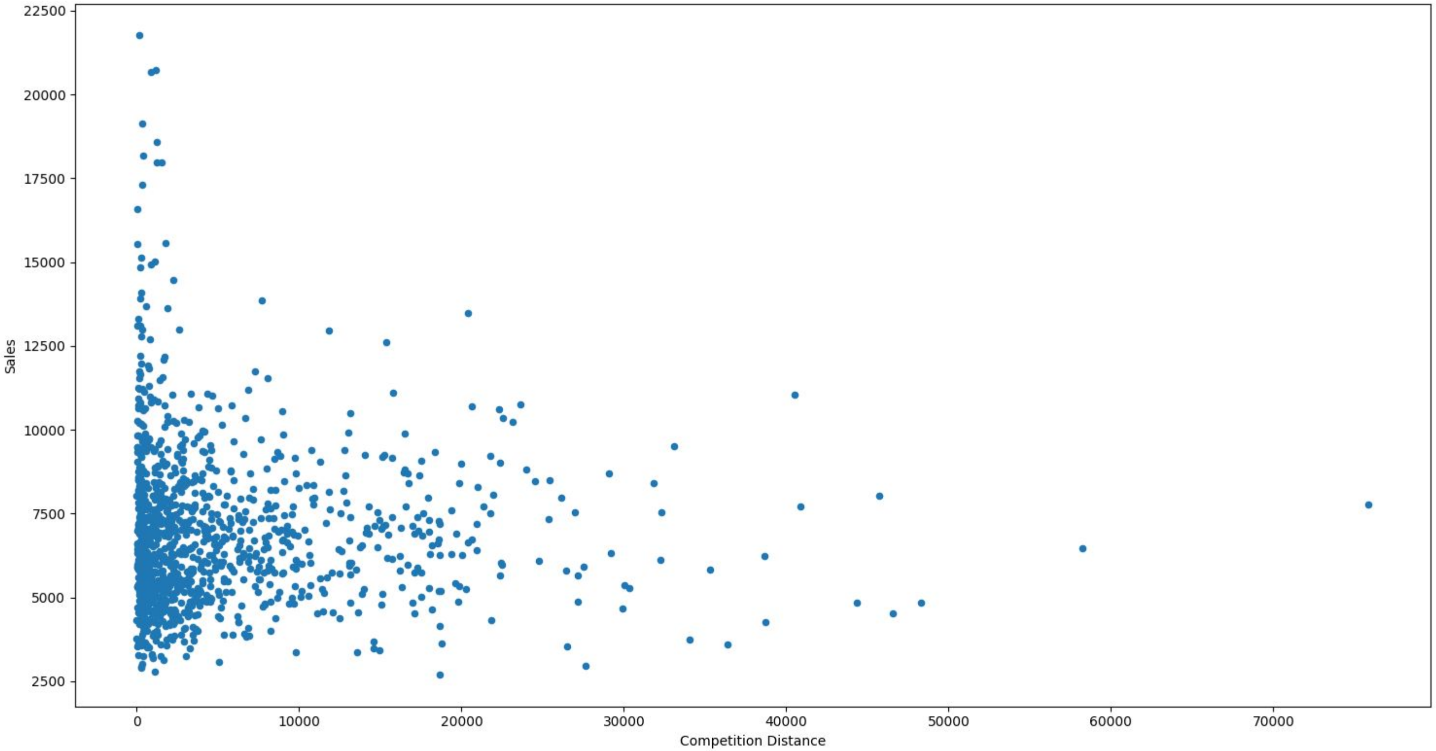
Average Sales & No. of Customers for State Holidays



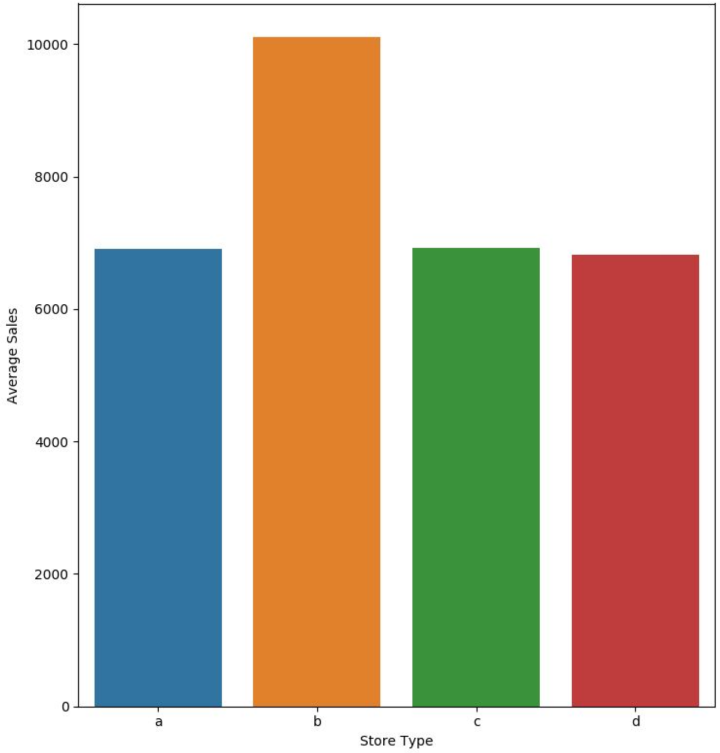
Effect of Competition on No. of Customers

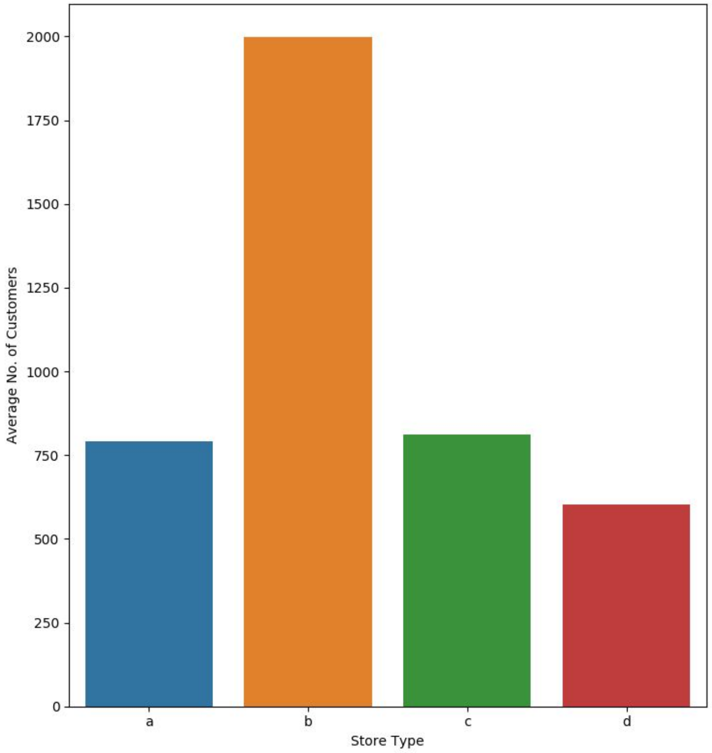


Effect of Competition on Sales

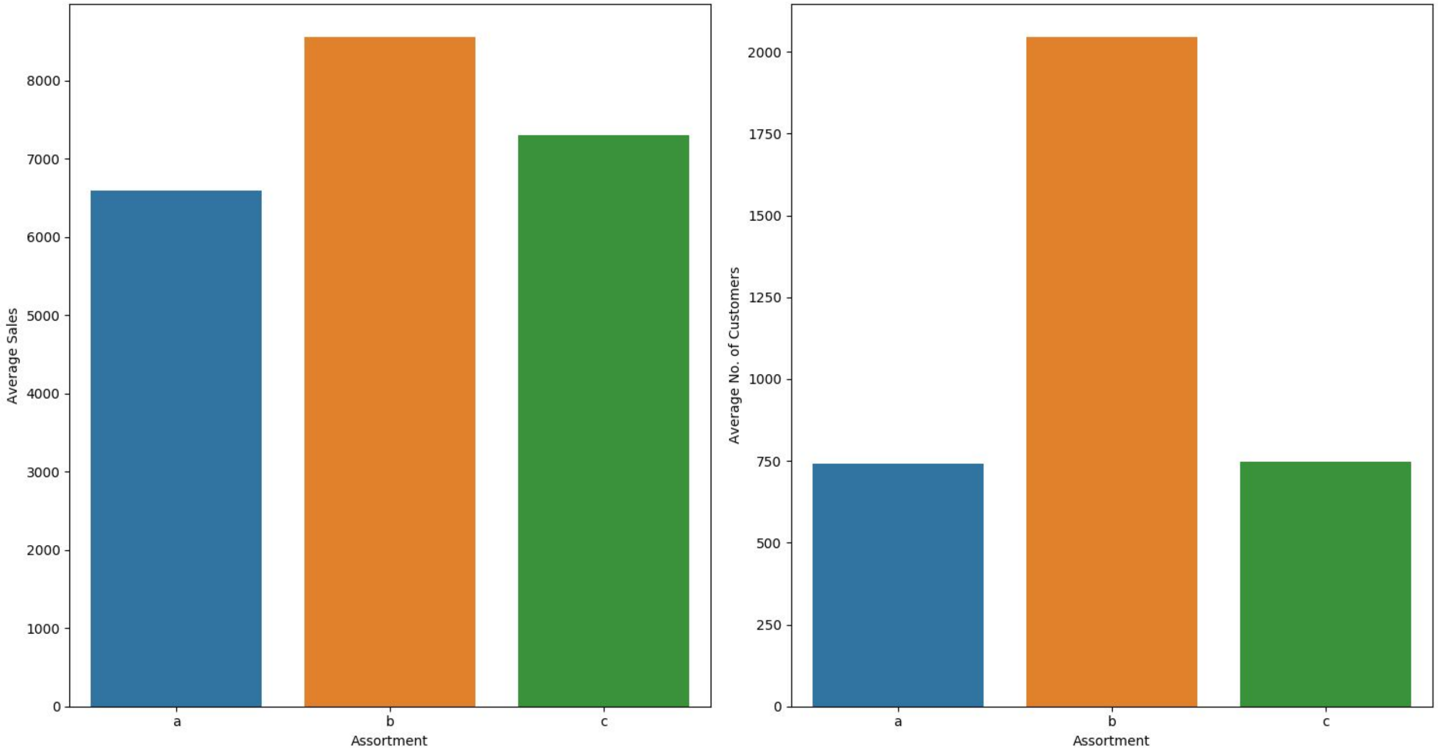


Effect of Store Type





Effect of Store Assortment



Models

Evaluation Metric

Evaluation is based on the Root Mean Square Percentage Error (RMSPE):

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

The overall goal is to minimize the RMSPE. The scores listed in this presentation is the Kaggle private score (i.e. the RMSPE value for ~70% of the test dataset).

Benchmark Models

Simple Geometric Mean Model

Simply calculates the geometric mean value for every `(Store, DayOfWeek, Promo)` combination and assigns that value as the prediction for the same combination in the test dataset.

Kaggle Private Score : 0.15996

Simple Median Model

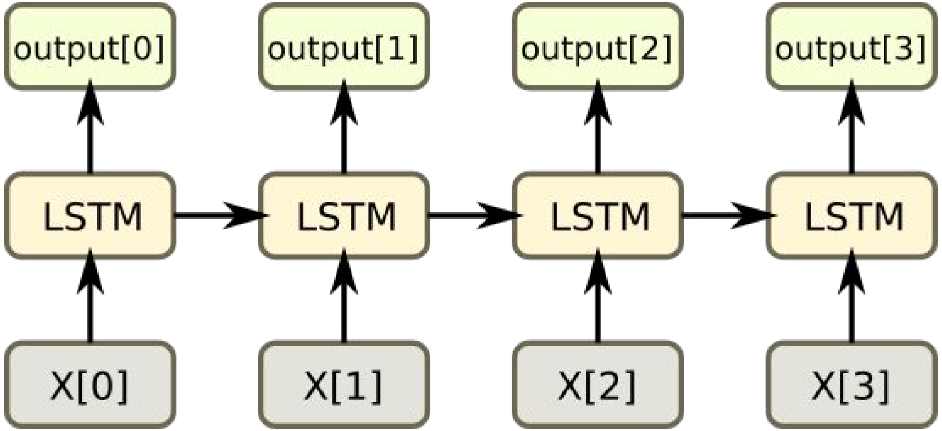
Calculates the median value instead of the geometric mean for every `(Store, DayOfWeek, Promo)` combination.

Kaggle Private Score : 0.15996

LSTM (Recurrent Neural Networks)

- Time Series Data
  - `sales[t-1], sales[t-2] ... sales[t-n-1], sales[t-n]`
  - `Where n = 7, 14, 21, 28`

- Supervised Training
  - RMSProp Optimizer
  - 500 LSTM Units
  - Batch Size of 64
  - 30% Dropout
  - Keras & TensorFlow APIs



Kaggle Private Score : 0.16041

Linear Regression

Entire Dataset as Single Regression:

Store, Promo, SchoolHoliday, Year, Month, DayOfWeek, StateHoliday, CompetitionDistance, StoreType, Assortment, AvgCustStore, AvgCustStoreMonth and AvgCustStoreYear

Kaggle Private Score : 0.26837

Each Store as Isolated Regression:

Promo, SchoolHoliday, Year, Month, DayOfWeek, StateHoliday, AvgCustStore, AvgCustStoreMonth

Kaggle Private Score : 0.16209

Using Log Normalised Values:

Kaggle Private Score : 0.15522

Inference :

Lacklustre performance of the first two implementations could be attributed to not using log-normalized values for Sales as those values exist in a Poisson-like distribution.

Other Regression Algorithms

S No.	Model Implemented	Private RMSPE Score
1	Ridge Regression (log-normalized data)	0.15482
2	Random Forest Regression (log-normalized data)	0.14502

XGBoost

Standard Set of Features

`Store, DayOfWeek, Year, Month, DayOfMonth, Open, Promo, StateHoliday, SchoolHoliday, StoreType, Assortment, CompetitionDistance and Promo2.`

Kaggle Private Score : 0.12727

Extended Set of Features

`Store, DayOfMonth, Week, Month, Year, DayOfYear, DayOfWeek, Open, Promo, SchoolHoliday, StateHoliday, StoreType, Assortment, CompetitionDistance, AvgSales, AvgCustomers and AvgSalesPerCustomer.`

Kaggle Private Score : 0.12305

Inference :

Gave a much higher performance as compared to both the linear regression models as well as the benchmark models due to ensemble learning methods.

XGBoost

- Static Combiner Model
    - Weighted combination (i.e. ensemble) of two XGBoost models.
    - Simply considers two models and applies a weighted average to get the final predictions.
    - `y_pred = y_pred1 * w1 + y_pred2 * w2`

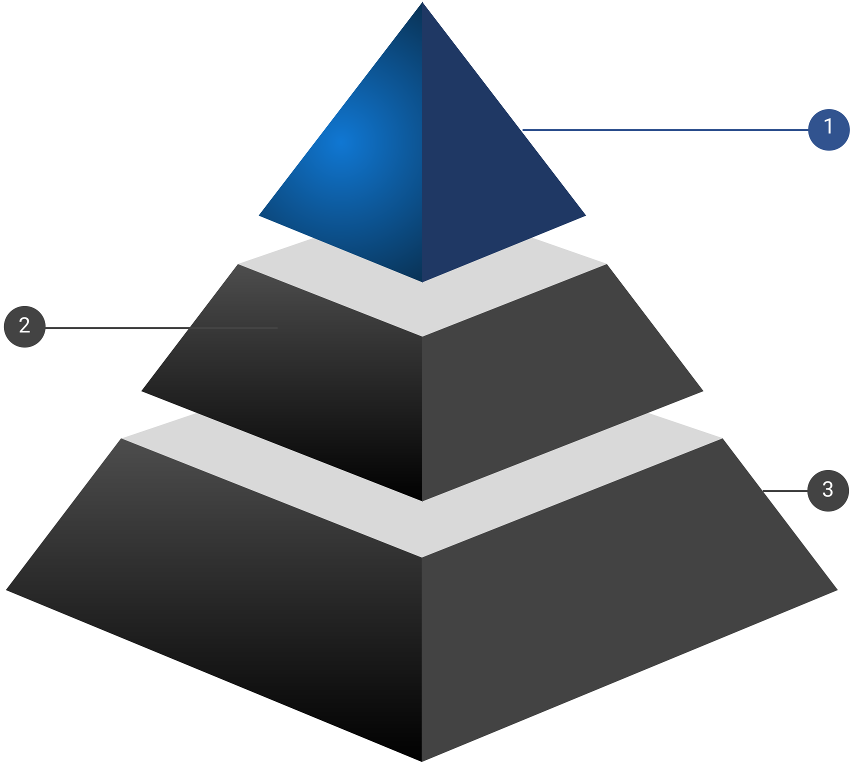
Kaggle Private Score : 0.11880

Inference :

The weighted predictions of several models together in an ensemble manner attain a much higher score. However, although this performs better for this dataset, the approach may lead to overfitting.

Analysis

Post Project Analysis



1

KDD APPROACH

Generic, adaptable and flexible for all knowledge discovery tasks.

2

TREND IDENTIFICATION

Important to select the right features by visualizing their effect on sales values.

3

MODEL SELECTION

Detailed experimentation and evaluation is required to identify the most effective models for the task at hand.

Conclusion

Conclusions

- The specific instance of the Rossmann Stores provides us with a good starting point to tackle the sales forecasting problem we identified.
- The focus is primarily on feature selection and engineering post which various models can be applied to those features.
- Our approach mainly revolved around analyzing the data and identifying interesting trends and patterns, which helped us select the right features to train our models on.
- We concluded that ensemble learning provides the best accuracy and boosted decision trees, in particular, work extremely well for sales forecasting problems.