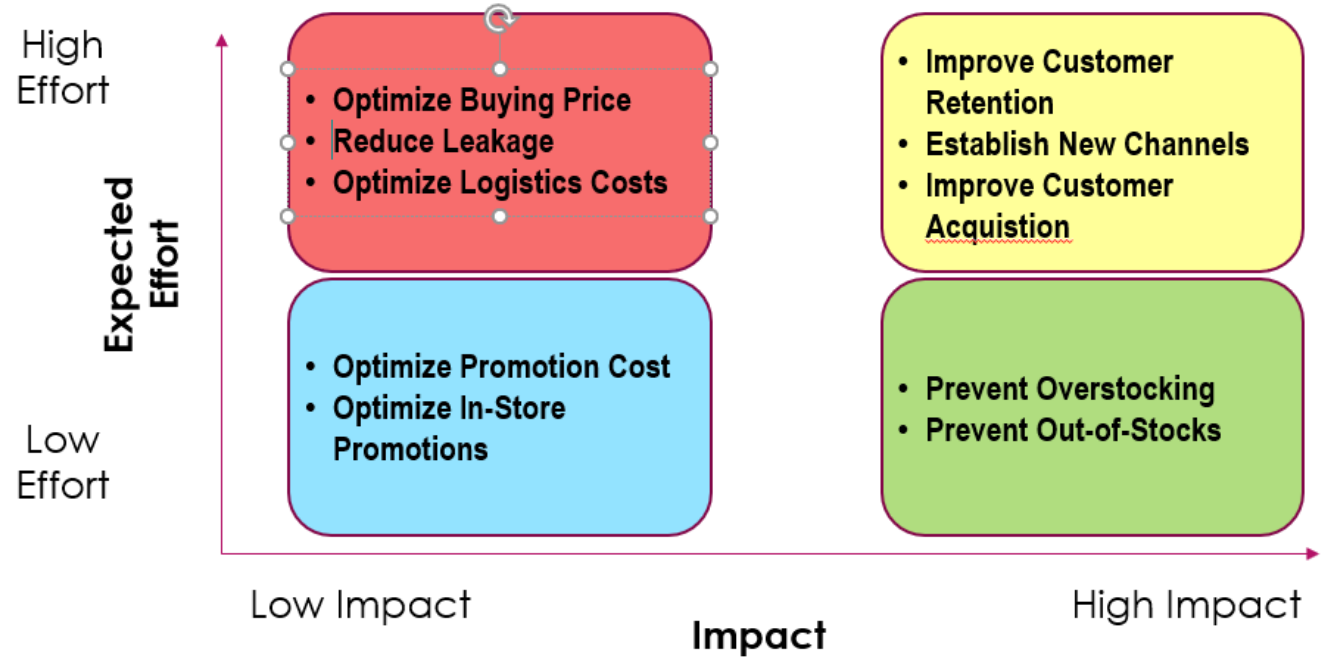
Demand Prediction

CAPSTONE 2 – SCOTT WARD

Introduction to Problem

- Company Management identified decline in profitability. After management reviewed the situation, there were several issues identified for the company to look at addressing first.
- The issues were broken down into potential impact on the business as well as effort to fix.

Introduction to Problem



Since Overstocking and Understocking of products has High Impact potential with relatively low company effort, the decision was made to create a project to come up with potential solutions.

Project Overview

- The company has not done any demand planning. Products are ordered based on periodic inventory and sales reports, or when suppliers have specials. Although individual markups are done initially, these are not pro-actively nor continually measured and evaluated. The order “system” is to more or less let store managers place ad-hoc orders for products they feel are running low. If the products are in the company warehouse, the orders get shipped. If not, the order more from the supplier.

Project Overview

This project will analyze current and projected demand for 30 of the company's products in its 76 retail locations as the first step in implementing an overall supply chain management program. Specifically, the main objective of this project will be**: solve the problem of over-stocking and under-stocking of products, by developing a predication model that can predict the demand of products for each store for the next week**. This will be completed as follows:

- Step 1 - Exploratory Data Analysis
- Step 2 – Data Pre-Processing
- Step 3 – Baseline Model & Validation Strategy
- Step 4 – Optimize Prediction Model

Step 1 – Exploratory Data Analysis

Data Summary

- The data used in this project was acquired from the company. They provided three datasets:
- sales.csv – contains 232,287 sales records with the UPC code of the product sold, the sales date, the store ID where the product was sold, the sales price, the base price, whether the product was on promotion for the week of the sale (1 or 0), whether the product was in the in-store circular (1 or 0) and the number of units sold for each week;
- product_data.csv – contains the product description, manufacturer, product category and sub-category, product size and UPC for 30 products; and
- store_data.csv – contains the store ID, store name, city, state, MSA code, market segment type of store, number of store parking spaces, store sales area square footage and average weekly baskets, for each of the 76 store locations.

Step 1 – Exploratory Data Analysis

Data Summary

- The data used in this project was acquired from the company. They provided three datasets:
- sales.csv – contains 232,287 sales records with the UPC code of the product sold, the sales date, the store ID where the product was sold, the sales price, the base price, whether the product was on promotion for the week of the sale (1 or 0), whether the product was in the in-store circular (1 or 0) and the number of units sold for each week;
- product_data.csv – contains the product description, manufacturer, product category and sub-category, product size and UPC for 30 products; and
- store_data.csv – contains the store ID, store name, city, state, MSA code, market segment type of store, number of store parking spaces, store sales area square footage and average weekly baskets, for each of the 76 store locations.

Step 1 – Exploratory Data Analysis

Sales Data: ('sales.csv' with 232,287 rows of data)

- **WEEK_END_DATE** - week ending date of sales report
- **STORE_NUM** - store number where sale was made
- **UPC** - (Universal Product Code) product specific identifier
- **BASE_PRICE** - base price of item
- **DISPLAY** - whether product was a part of in-store promotional display (1-Yes, 0-No)
- **FEATURE** - whether product was in in-store circular (1-Yes, 0-No)
- **UNITS** - units sold (target)

Step 1 – Exploratory Data Analysis

Sales Data: ('sales.csv' with 232,287 rows of data)

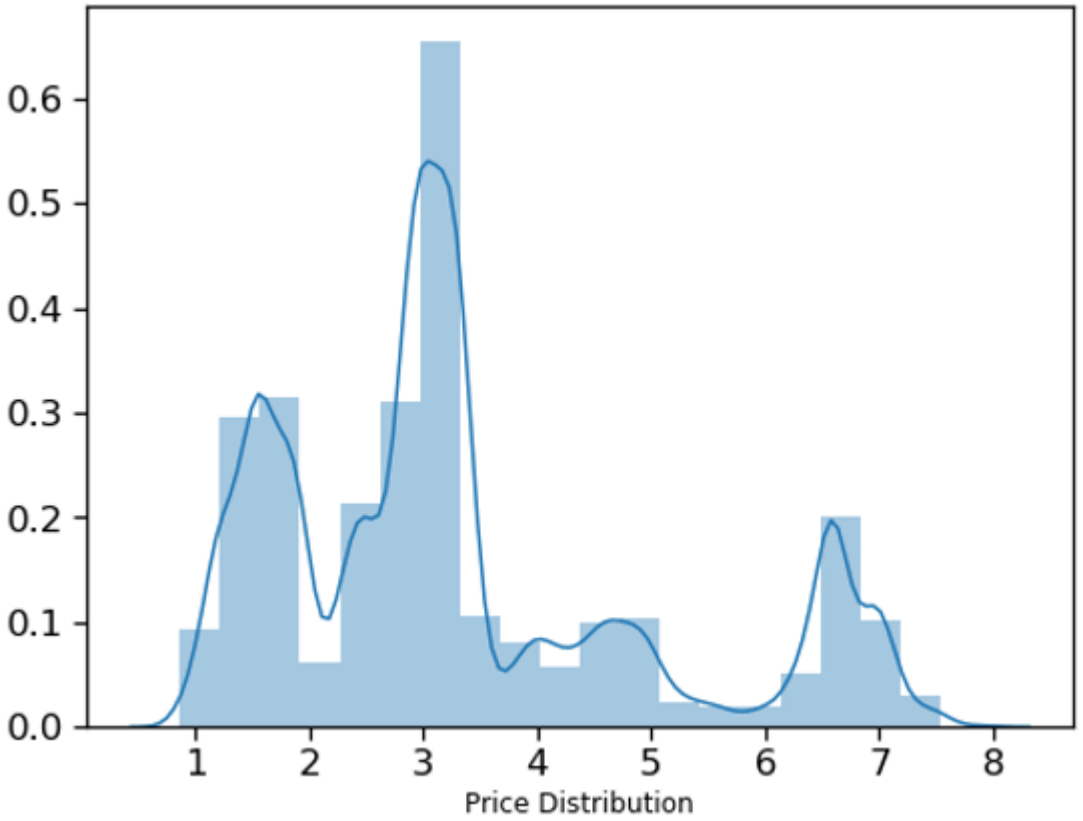
- **WEEK_END_DATE** - week ending date of sales report
- **STORE_NUM** - store number where sale was made
- **UPC** - (Universal Product Code) product specific identifier
- **BASE_PRICE** - base price of item
- **DISPLAY** - whether product was a part of in-store promotional display (1-Yes, 0-No)
- **FEATURE** - whether product was in in-store circular (1-Yes, 0-No)
- **UNITS** - units sold (target)

Step 1 – Exploratory Data Analysis

Sales Data:

BASE_PRICE Distribution

- No extreme values.
- Range $0.86 - $7.89
- Mean of $3.35

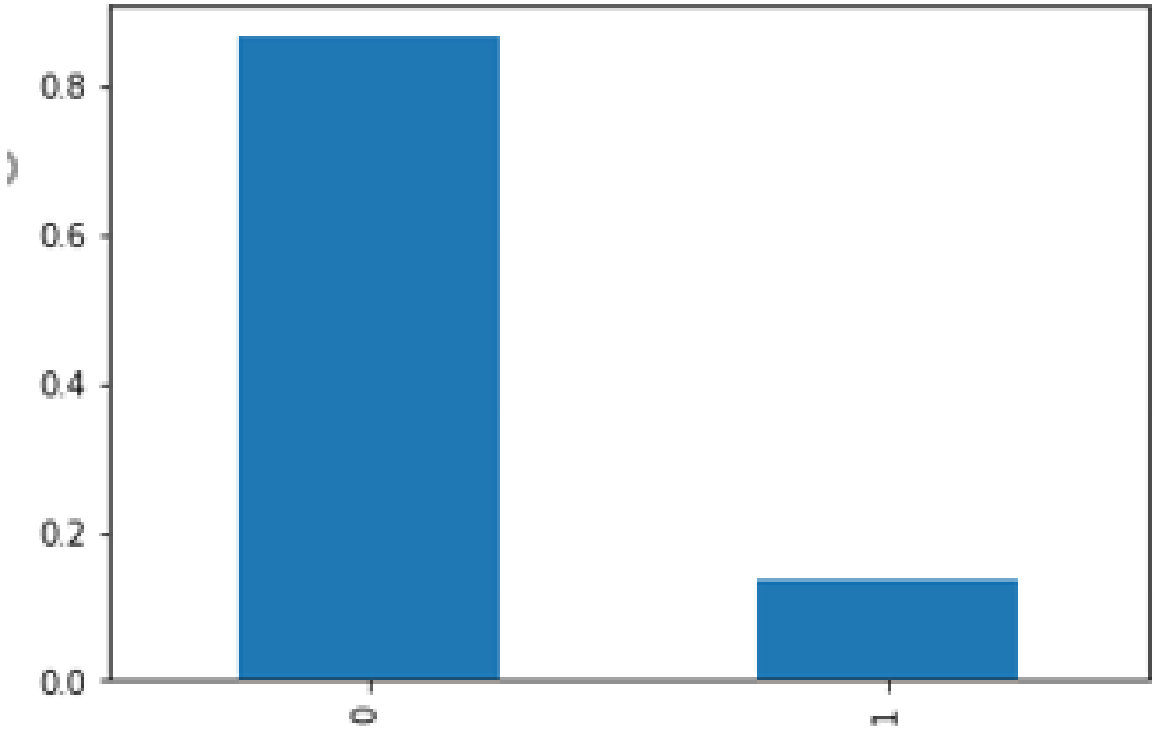


Step 1 – Exploratory Data Analysis

Sales Data:

DISPLAY Distribution

- 86.5% not 'Displayed'
- 13.5% are Displayed each week

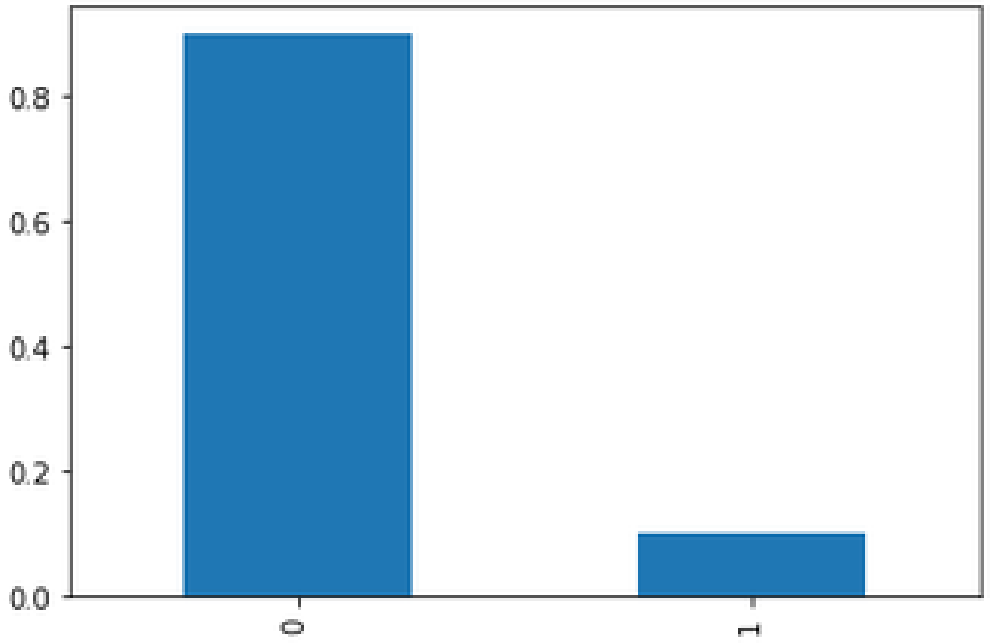


Step 1 – Exploratory Data Analysis

Sales Data:

FEATURE Distribution

- 90.0% not 'Featured'
- 10.0% are Featured each week

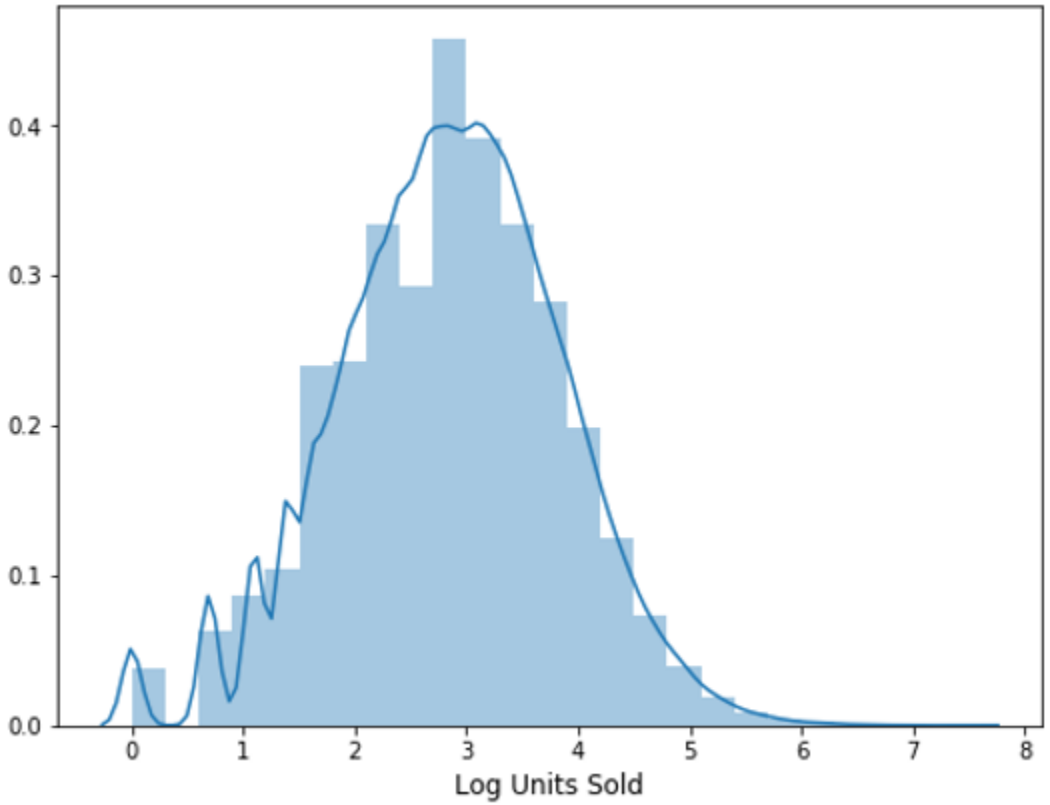


Step 1 – Exploratory Data Analysis

Sales Data:

UNITS Distribution

- Took log of distribution
- Close to normal distribution



Step 1 – Exploratory Data Analysis

Products Data: ('product_data.csv' with 30 rows of data)

- **UPC** - (Universal Product Code) product specific identifier
- **DESCRIPTION** - product description
- **MANUFACTURER** - product manufacturer/supplier
- **CATEGORY** - product category
- **SUB_CATEGORY** - product sub-category
- **PRODUCT_SIZE** - package size/quantity

Step 1 – Exploratory Data Analysis

Products Data:

- **CATEGORY** – four product categories
  - Bag Snacks (9)
  - Oral Hygiene Products (8)
  - Cold Cereal (7)
  - Frozen Pizza (6)

Step 1 – Exploratory Data Analysis

Products Data:

	CATEGORY	SUB_CATEGORY
0	BAG SNACKS	PRETZELS
5	COLD CEREAL	ALL FAMILY CEREAL
6	COLD CEREAL	ADULT CEREAL
19	COLD CEREAL	KIDS CEREAL
8	FROZEN PIZZA	PIZZA/PREMIUM
3	ORAL HYGIENE PRODUCTS	MOUTHWASHES (ANTISEPTIC)
16	ORAL HYGIENE PRODUCTS	MOUTHWASH/RINSES AND SPRAYS

The sub-categories give additional detail about the products.

- Cereal has 3 sub categories, differentiating on the age group.
- Oral hygiene products have 2 sub categories, antiseptic and rinse/spray.
- Bag Snacks & Frozen Pizza have just 1 sub category.

Step 1 – Exploratory Data Analysis

Products Data:

	CATEGORY	SUB_CATEGORY	PRODUCT_SIZE
0	BAG SNACKS	PRETZELS	15 OZ
14	BAG SNACKS	PRETZELS	16 OZ
25	BAG SNACKS	PRETZELS	10 OZ
6	COLD CEREAL	ADULT CEREAL	20 OZ
7	COLD CEREAL	ALL FAMILY CEREAL	18 OZ
19	COLD CEREAL	KIDS CEREAL	15 OZ
20	COLD CEREAL	KIDS CEREAL	12.2 OZ
5	COLD CEREAL	ALL FAMILY CEREAL	12.25 OZ
13	COLD CEREAL	ALL FAMILY CEREAL	12 OZ
8	FROZEN PIZZA	PIZZA/PREMIUM	32.7 OZ
9	FROZEN PIZZA	PIZZA/PREMIUM	30.5 OZ
10	FROZEN PIZZA	PIZZA/PREMIUM	29.6 OZ
24	FROZEN PIZZA	PIZZA/PREMIUM	22.7 OZ
21	FROZEN PIZZA	PIZZA/PREMIUM	29.8 OZ
23	FROZEN PIZZA	PIZZA/PREMIUM	28.3 OZ
3	ORAL HYGIENE PRODUCTS	MOUTHWASHES (ANTISEPTIC)	500 ML
16	ORAL HYGIENE PRODUCTS	MOUTHWASH/RINSES AND SPRAYS	1 LT
17	ORAL HYGIENE PRODUCTS	MOUTHWASHES (ANTISEPTIC)	1 LT

Step 1 – Exploratory Data Analysis

Products Data:

MANUFACTURER / CATEGORY	FRITO LAY	GENERAL MI	KELLOGG	P & G	PRIVATE LABEL	SNYDER S	TOMBSTONE	TONYS	WARNER
BAG SNACKS	1	0	0	0	1	1	0	0	0
COLD CEREAL	0	1	1	0	1	0	0	0	0
FROZEN PIZZA	0	0	0	0	1	0	1	1	0
ORAL HYGIENE PRODUCTS	0	0	0	1	1	0	0	0	1

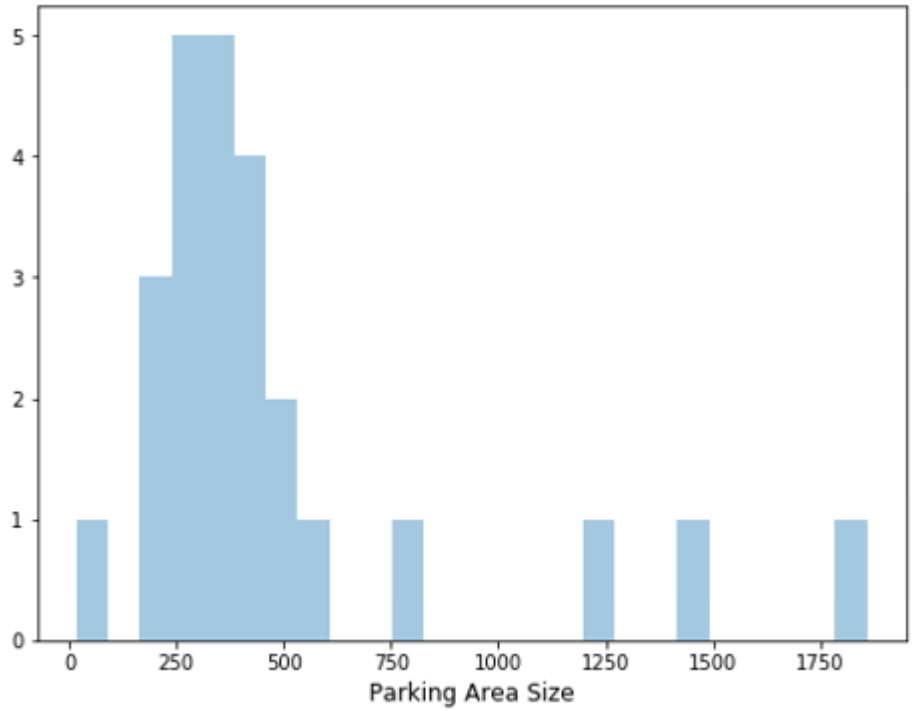
Step 1 – Exploratory Data Analysis

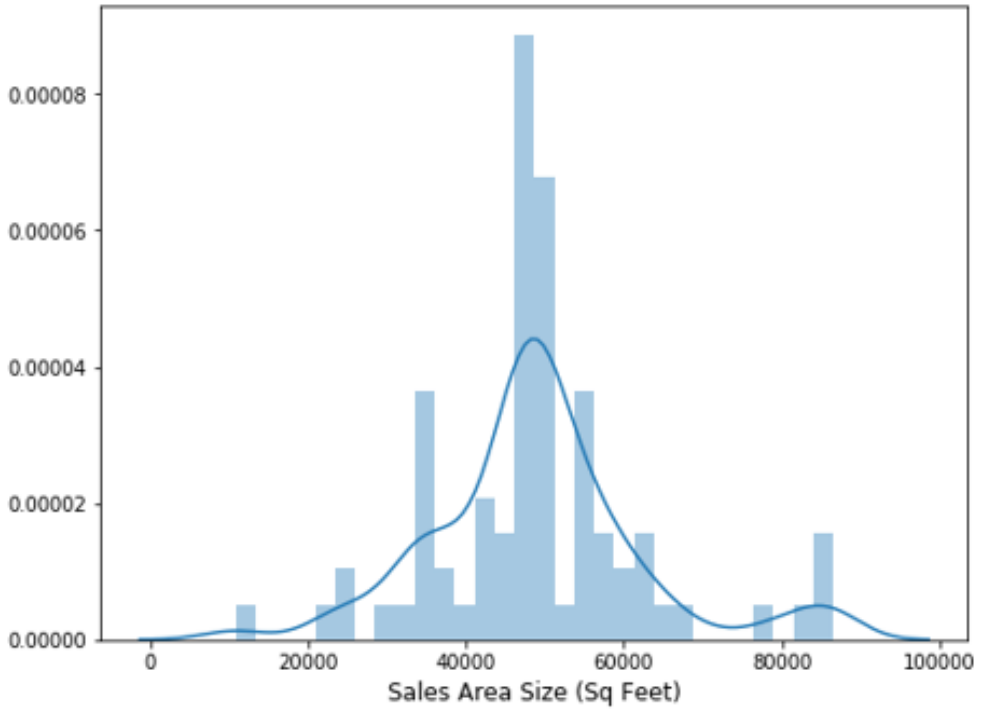
Store Data: ('stor_data.csv' with 76 rows of data)

- **STORE_ID** - store number
- **STORE_NAME** - Name of store
- **ADDRESS_CITY_NAME** - city
- **ADDRESS_STATE_PROV_CODE** - state
- **MSA_CODE** - (Metropolitan Statistical Area) Based on geographic region and population density
- **SEG_VALUE_NAME** - Store Segment Name
- **PARKING_SPACE_QTY** - number of parking spaces in the store parking lot
- **SALES_AREA_SIZE_NUM** - square footage of store
- **AVG_WEEKLY_BASKETS** - average weekly baskets sold in the store

Step 1 – Exploratory Data Analysis

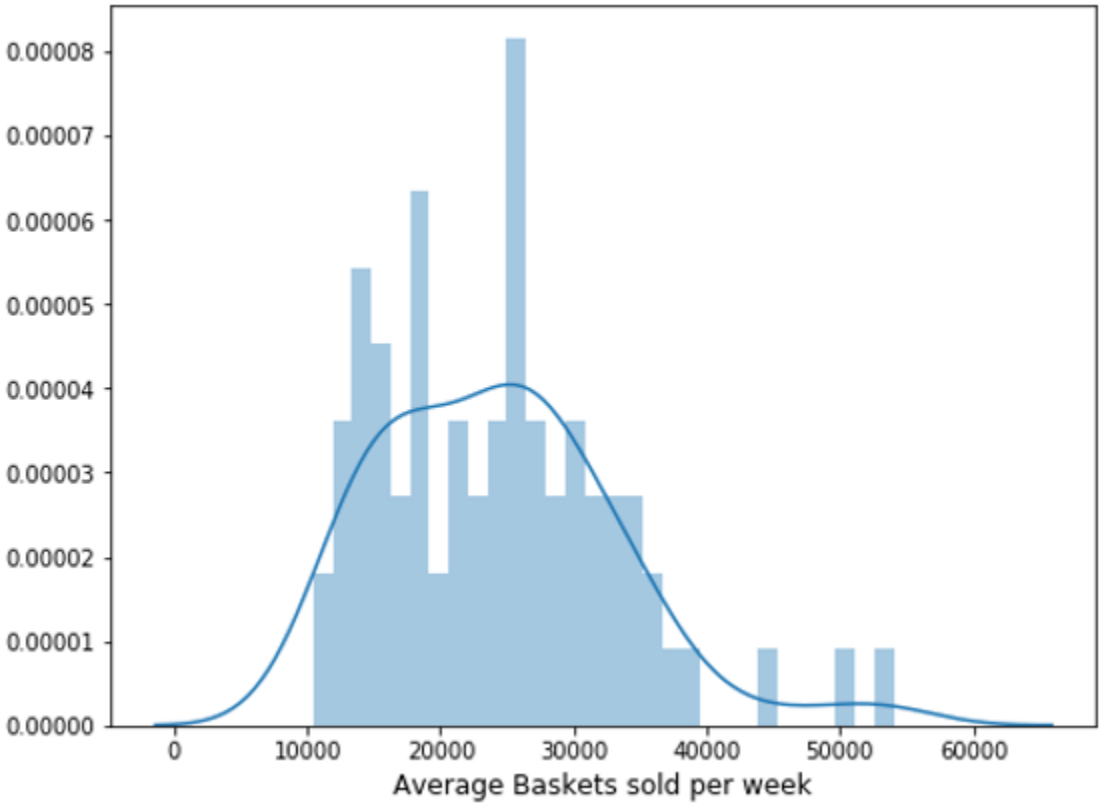
Store Data:





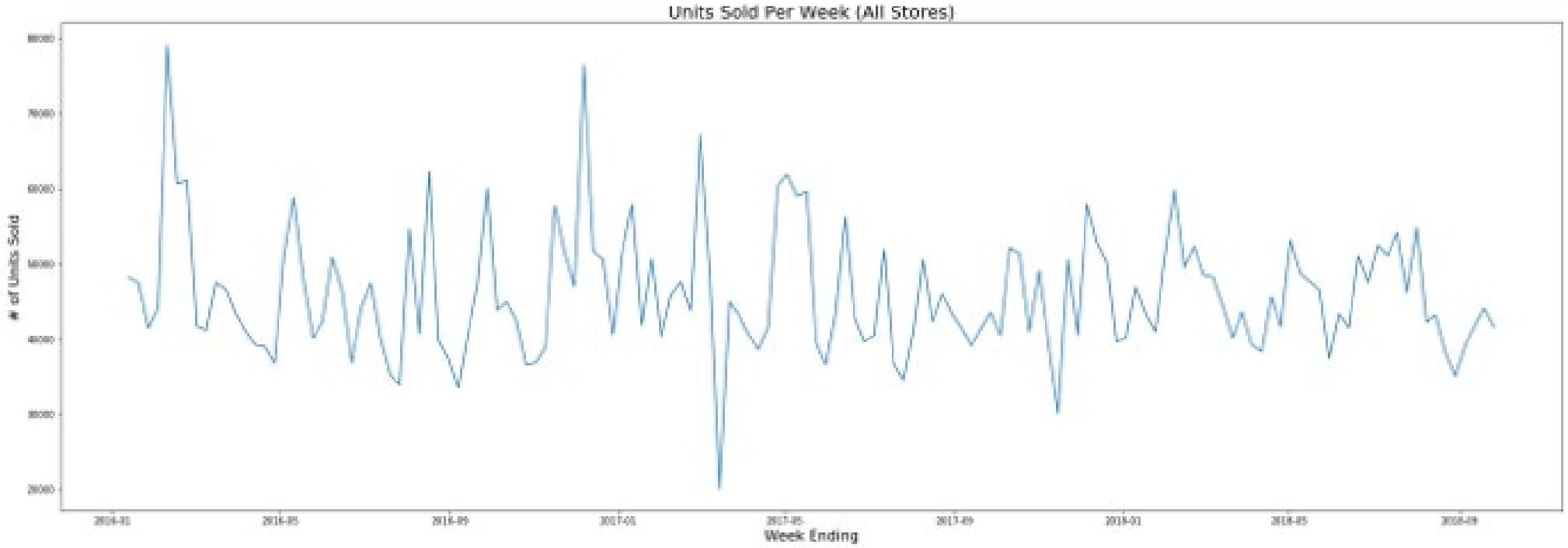
Step 1 – Exploratory Data Analysis

Store Data:



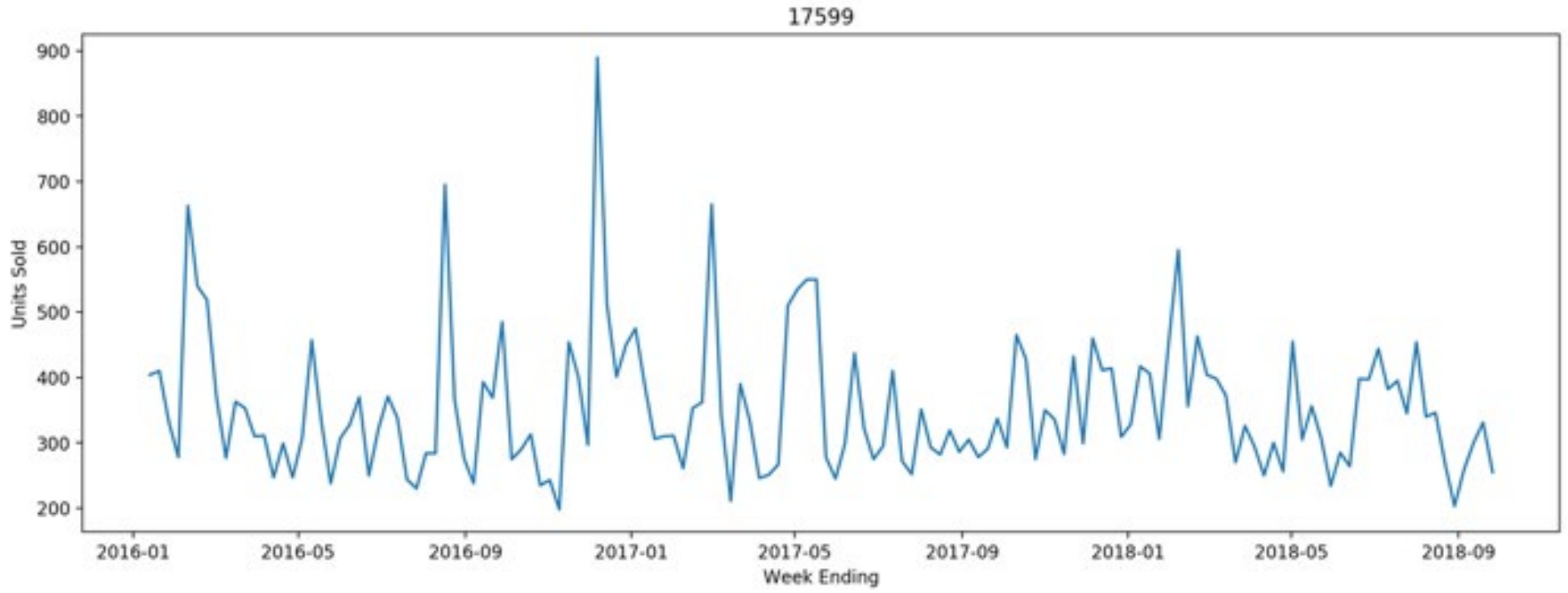
Step 1 – Exploratory Data Analysis

Store Data:



Step 1 – Exploratory Data Analysis

Store Data:



Step 2 – Data Wrangling

Now that we understand the data a bit better, we are going to take a closer look and wrangle (or pre-process) the data in a way that will allow us to use it in a prediction model. This will involve:

- converting categorical features to a numeric representation
- making sure that there are no missing values in our categorical features (we already checked numeric ones)
- removing features that don't add value to the predication model, and
- choosing an encoding scheme by which to covert the categorical features to numeric features.

Step 2 – Data Wrangling

CATEGORICAL FEATURES

DATASET 1: 'SALES' Dataframe

- DISPLAY – already coded as '1' or '0' for on display or not.
- FEATURE - already coded as '1' or '0' for featured in sales flyer or not.

DATASET 2: 'PRODUCTS' Dataframe - All categorical features.

- 'MANUFACTURER', 'CATEGORY' and 'SUB_CATEGORY' do not have an order or sequence. Thus, they will be converted to numerical using 'One Hot Encoder'.
- The 'PRODUCT_SIZE' feature has different unit sizes for each category. As such, we will create numerical bins for this to be relevant in the model.

Step 2 – Data Wrangling

CATEGORICAL FEATURES

DATASET 3: 'STORES' Dataframe

- STORE_ID – Same as before—key value for merging dataframes.
- STORE_NAME & ADDRESS_CITY_NAME – Since there are 72/51 unique names out of 76 different unique stores, we will drop these features due to high cardinality.
- ADDRESS_STATE_PROV_CODE and MSA_CODE – No order in these categories, so we will use One Hot Encoder on these variables.
- SEG_VALUE_NAME—Store segments are divided into 3 categories: upscale, mainstream and value.

Step 2 – Data Wrangling

CONTINUOUS FEATURES

DATASET 1: 'SALES' Dataframe

- BASE_PRICE - base price of item
  - 12 missing values. Imputed based on average store price for same product UPC.
- UNITS - units sold (target)
  - No missing values, but 21 values with 750 or more units sold.
  - Removed these rows as outliers.

Step 2 – Data Wrangling

CONTINUOUS FEATURES

DATASET 2: 'PRODUCTS' Dataframe

- **No continuous features.**

DATASET 3: 'STORES' Dataframe

- In checking for null values, we found 51 in the PARKING_SPACE_QTY variable. Since it is reasonable to assume that the number of parking spaces would be somewhat related to the store size, we checked the correlation between PARKING_SPACE_QTY and SALES_AREA_SIZE_NUM.
- Since the correlation is high, drop PARKING_SPACE_QTY

	PARKING_SPACE_QTY	SALES_AREA_SIZE_NUM
PARKING_SPACE_QTY	1.000000	0.763274
SALES_AREA_SIZE_NUM	0.763274	1.000000

Step 3 – Baseline Model Development

Validation Strategy

- Prior to doing baseline model development, a discussion of the evaluation metrics that were considered are in order. Since the problem we are analyzing is demand forecast, there are two possible problems in store operations—too much product, resulting in higher than necessary stocking costs, or not enough product, resulting in lost sales from empty shelf space when customers want to purchase a product. In business terms, lost revenue hurts the business (has a greater negative effect on profit) than too much product on the shelf.

Step 3 – Baseline Model Development

Mean Absolute Error

Store/ Product	Forecast	Actual	Absolute Error
A	22	27	5
B	34	56	22
C	8	11	3
D	39	32	7
E	15	8	7

MAE: 8.8

Step 3 – Baseline Model Development

Root Mean Squared Error

Store/ Product	Forecast	Actual	Absolute Error	Squared Error
A	22	27	5	25
B	34	56	22	484
C	8	11	3	9
D	39	32	7	49
E	15	8	7	49
			MAE: 8.8	MSE: 123.2

RMSE: 11.9

Step 3 – Baseline Model Development

Root Mean Log Squared Error

Forecast	Demand	Status	RMSE	RMSLE*100
50	40	Overstocking	5	4.8455
30	40	Understocking	5	6.2469

Step 3 – Baseline Model Development

Steps

The steps for this process will consist of the following:

- Merge datasets from Step 2
- Create train and validation sets
- Perform Mean Prediction using different models
- Evaluate results
- Select model for further development

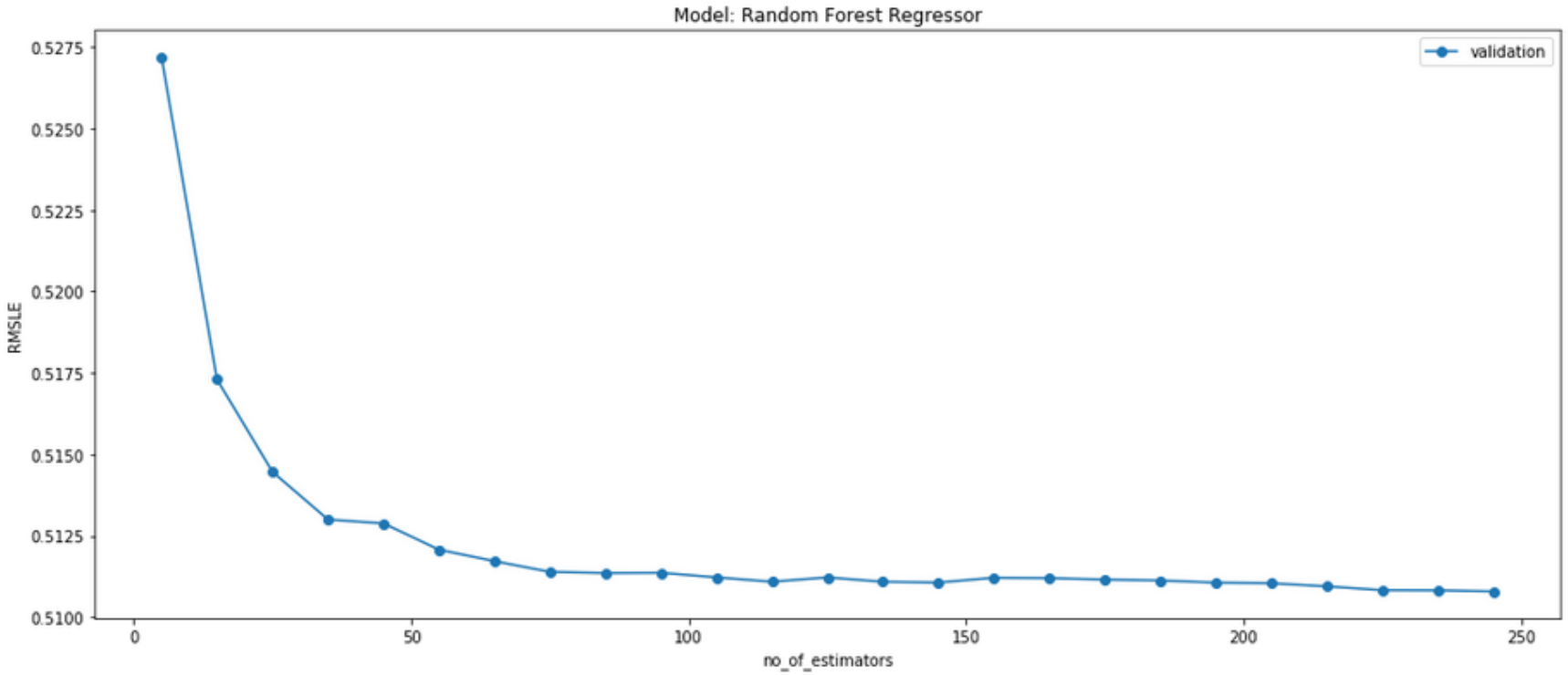
Step 3 – Baseline Model Development

Basic Prediction Models

- Regression Models
  - Basic mean prediction – **RMSLE: 0.5986**
  - Simple moving average – **RMSLE: 0.5988**
  - Linear regression – **RMSLE: 0.9433**
- Decision Tree Models
  - Basic Decision Tree – **RMSLE: 0.5035**
  - **RandomForest – RMSLE: 0.4878**

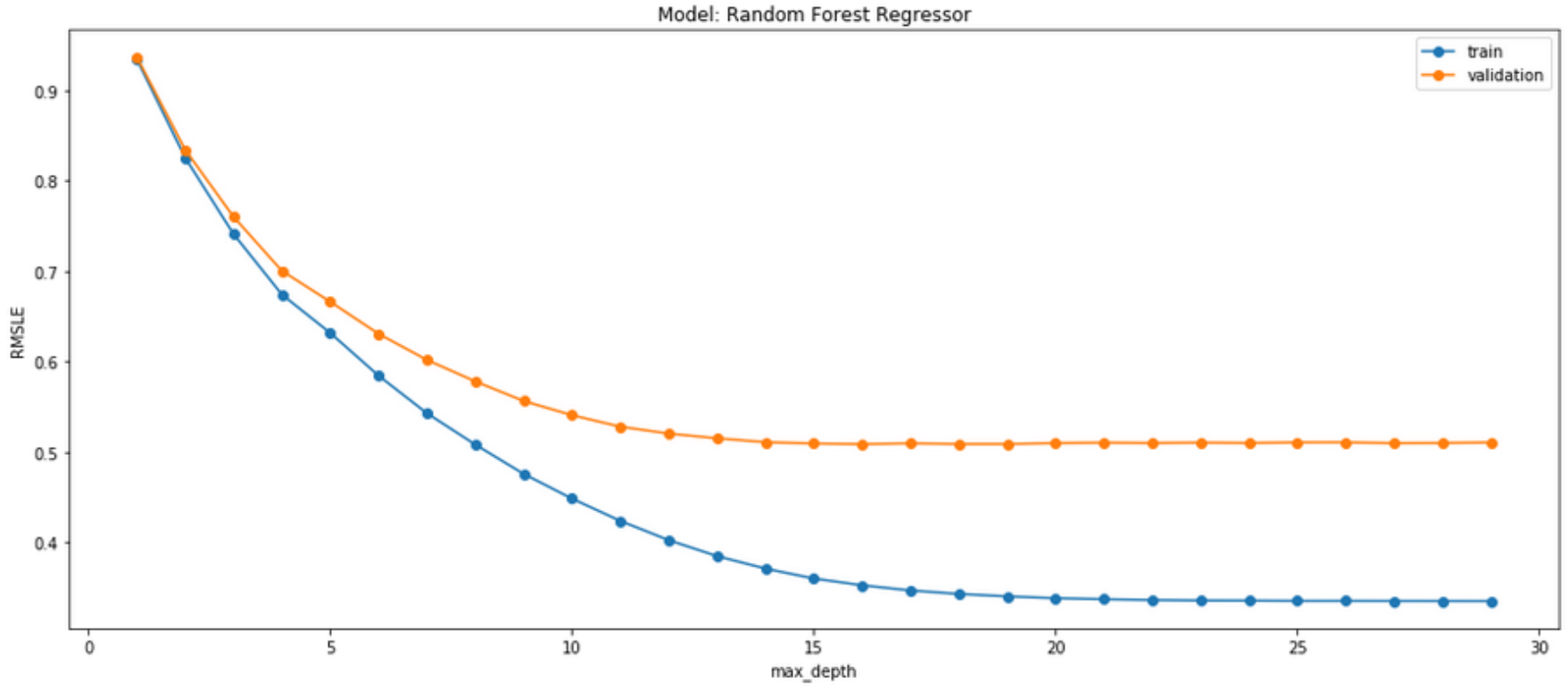
Step 4 – Model Optimization

Finding Optimal Number of Estimators with Baseline Features



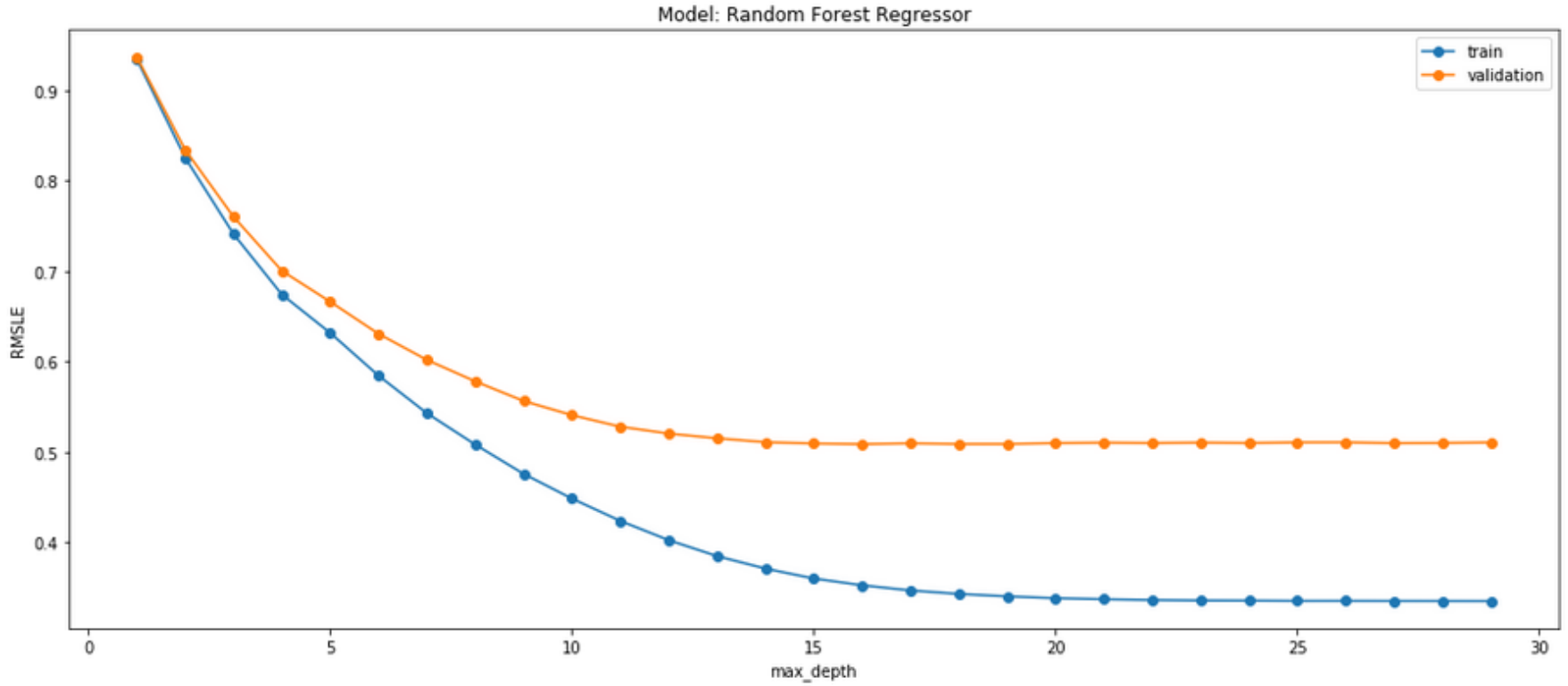
Step 4 – Model Optimization

Finding Optimal Max Depth with Baseline Features



Step 4 – Model Optimization

Finding Optimal Max Depth with Baseline Features



Step 4 – Model Optimization

- Create validation datasets
- 14 total w/2 months of data

	train_start_1	train_end_1	train_start_2	validate_week	test_week	no_days_train_1	no_days_train_2	set_no
0	2011-07-13	2011-08-31	2011-07-27	2011-09-14	2011-09-28	56 days	56 days	set1
1	2011-07-06	2011-08-24	2011-07-20	2011-09-07	2011-09-21	56 days	56 days	set2
2	2011-06-29	2011-08-17	2011-07-13	2011-08-31	2011-09-14	56 days	56 days	set3
3	2011-06-22	2011-08-10	2011-07-06	2011-08-24	2011-09-07	56 days	56 days	set4
4	2011-06-15	2011-08-03	2011-06-29	2011-08-17	2011-08-31	56 days	56 days	set5
5	2011-06-08	2011-07-27	2011-06-22	2011-08-10	2011-08-24	56 days	56 days	set6
6	2011-06-01	2011-07-20	2011-06-15	2011-08-03	2011-08-17	56 days	56 days	set7
7	2011-05-25	2011-07-13	2011-06-08	2011-07-27	2011-08-10	56 days	56 days	set8
8	2011-05-18	2011-07-06	2011-06-01	2011-07-20	2011-08-03	56 days	56 days	set9
9	2011-05-11	2011-06-29	2011-05-25	2011-07-13	2011-07-27	56 days	56 days	set10
10	2011-05-04	2011-06-22	2011-05-18	2011-07-06	2011-07-20	56 days	56 days	set11
11	2011-04-27	2011-06-15	2011-05-11	2011-06-29	2011-07-13	56 days	56 days	set12
12	2011-04-20	2011-06-08	2011-05-04	2011-06-22	2011-07-06	56 days	56 days	set13
13	2011-04-13	2011-06-01	2011-04-27	2011-06-15	2011-06-29	56 days	56 days	set14

Step 4 – Model Optimization

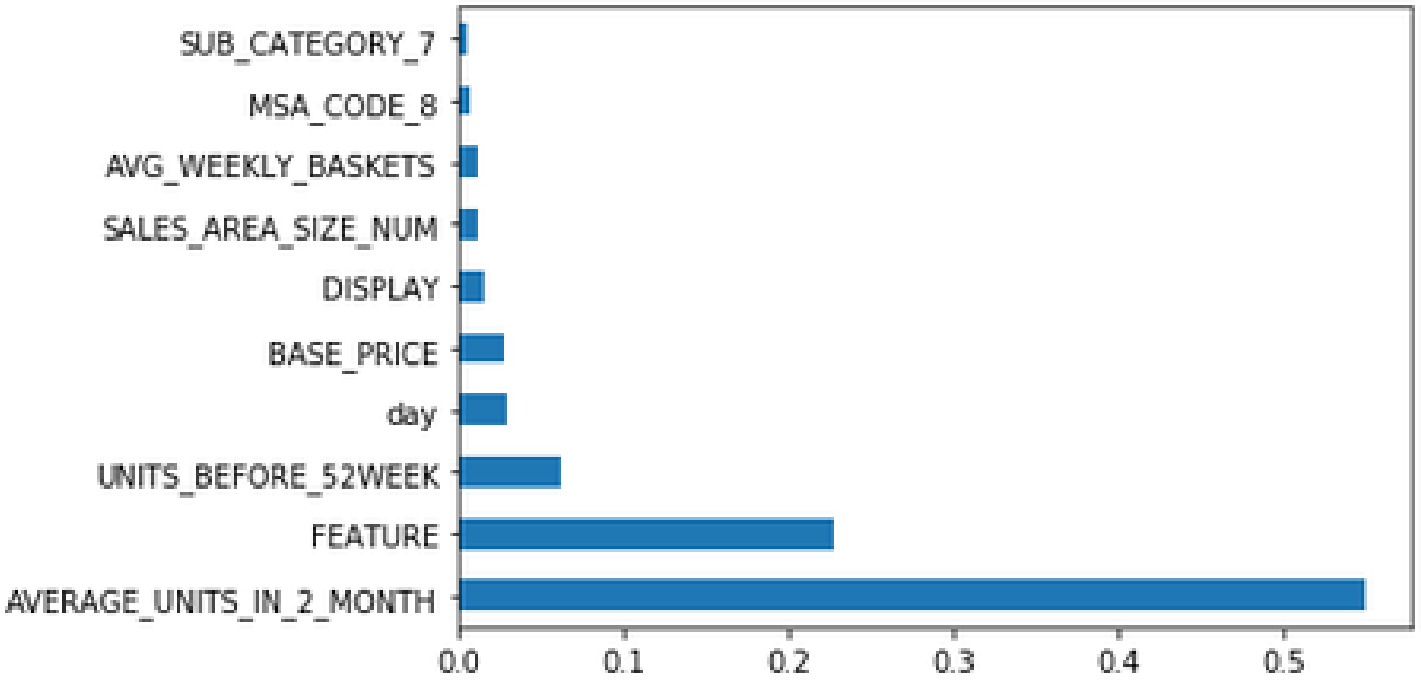
- Run RandomForest on validation datasets

	rmsle_train_2	rmsle_test
0	0.353240	0.416301
1	0.355415	0.493602
2	0.363767	0.401761
3	0.350412	0.463373
4	0.344208	0.558591
5	0.349586	0.479080
6	0.339725	0.465425
7	0.334268	0.534997
8	0.328488	0.432694
9	0.332360	0.470960
10	0.346669	0.427036
11	0.348837	0.436376
12	0.348793	0.453843
13	0.344507	0.423164

```
rmsle_train_2    0.345734
rmsle_test       0.461229
dtype: float64
```

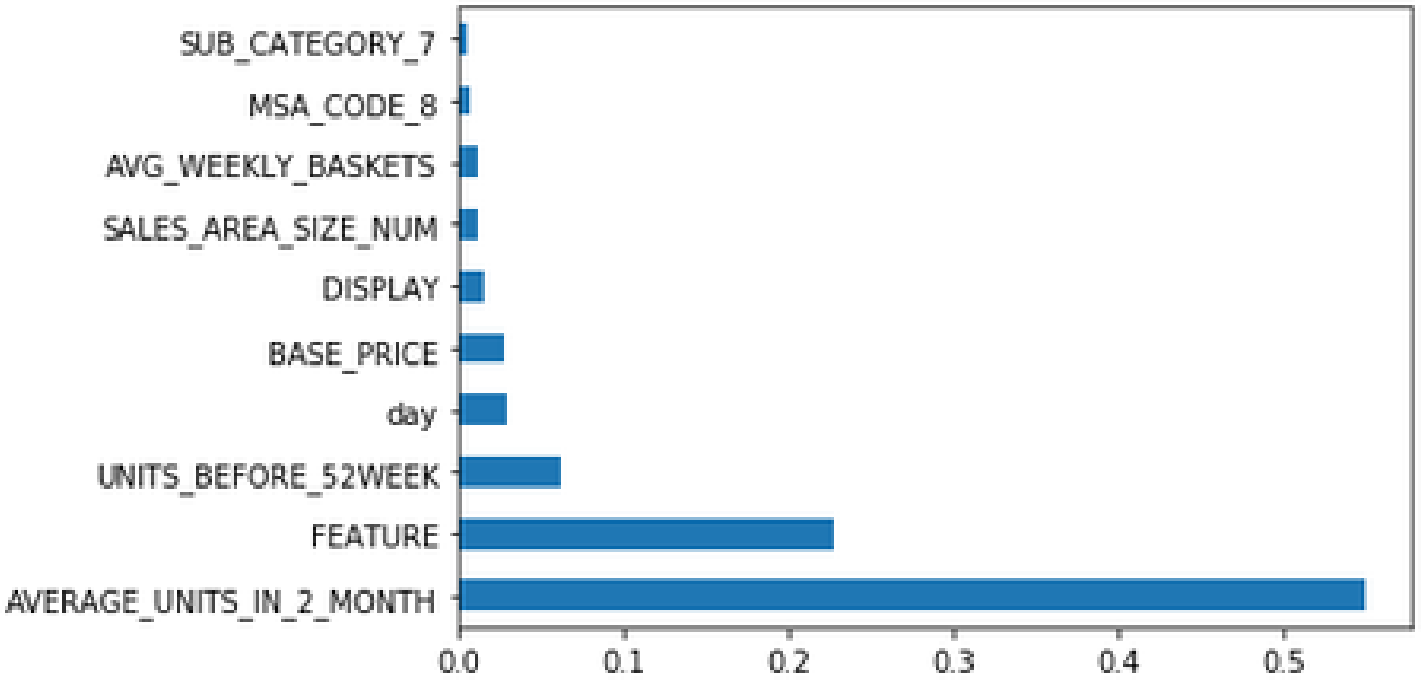
Step 4 – Model Optimization

- Feature Importance



Step 4 – Model Optimization

- Feature Importance



Recommendations & Next Steps

- Continue Collection of Company Data to Produce Inputs into Forecast Prediction Model
- Run Prediction Model to Generate Product Demand Forecast for Coming Week
- Share Results with Inventory Management Team
- Bonus: Create Production Dashboard to Monitor on On-Going Basis