

# Data Science Capstone Project 1

## Modeling report

### Introduction

The problem I chose is to forecast the weekly sales, department wise, of 45 Walmart stores located in the US. The data set consists of around 450,000 observations of training data spanned over 2.5 years and 115,000 observations of test data spanned over 10 months.

This is fundamentally a time series problem, however, my attempt to solve the problem is through machine learning models. The approach that I have adopted has put me at the 4th spot on the Kaggle leaderboard.

### Modeling strategy:

I tried to solve the problem using an ensemble of 3 models. The final results were the average of the prediction from each model.

Model 1: XGBoost Model trained all of the training data.

Model 2: XGBoost Models trained on each Store-Dept pair

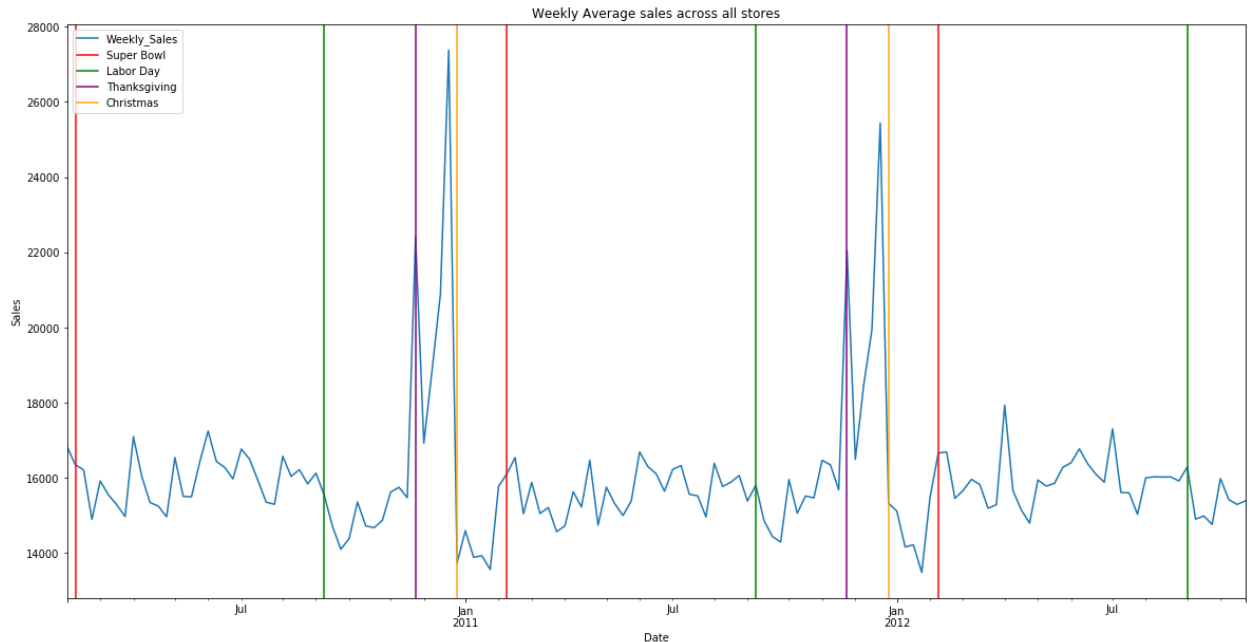
Model 3: RandomForest Models trained on each Store-Dept pair.

### Model 1 - Single XGB Model:

This model is an Xtreme Gradient Boosting regressor trained on all of the training data. Following is a list and description of the features that were engineered to generate optimal results.

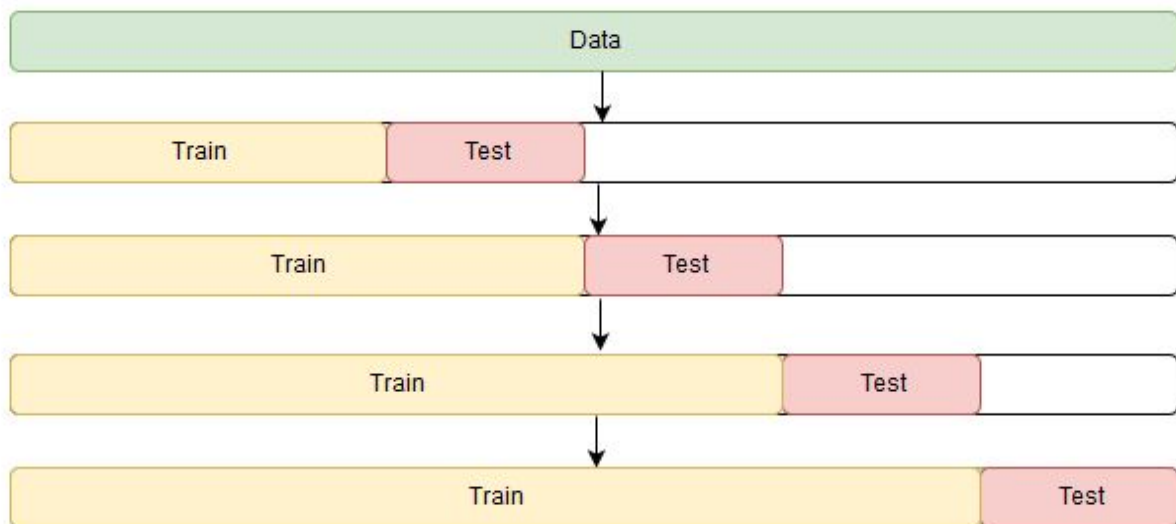
#### Features Engineered:

1. **CPI\_Cat and Size\_Cat:** CPI (Customer Propensity Index) and Size (of Store) were available in the training data as continuous values. These were discretized following along the rational that came from the Exploratory Data analysis.
2. **Which\_holiday:** The training data had a column for 'IsHoliday'. It was realized during EDA that each holiday had a different trend on the weekly sales. Some holidays had virtually no effect on the sales trends e.g. Labor Day as shown below. As a result feature was created for each holiday

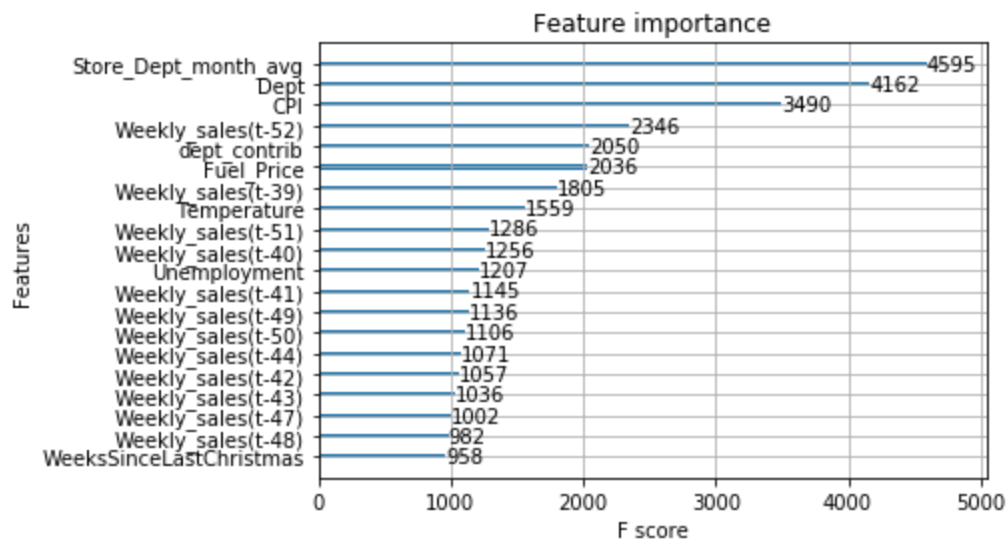


3. **TillNext{Holiday}, SinceNext{Holiday}**: These features were created to depict the number of weeks since a particular holiday has passed and the number of weeks until the next holiday comes. These were created for 4 holidays thus a total of 8 features
4. **DateLagFeatures**: This feature depicts the Weekly Sales for each Store-Dept pair 'x' weeks from the current week.
5. **DateFeatures**: These features were derived from the date of the observation. These include, Quarter, Month, Year, WeekOfMonth, WeekOfYear
6. **Department\_Contribution**: This feature is the ratio of each Department's average monthly sales with the Store's average monthly sales. The rationale here is that the ratio would be a measure of how much the department's sales contribute to the sales of the store for each month.
7. **Store\_dept\_month\_avg**: This is the average sale for that store, dept pair for that month

The model was trained using the XGBRegressor API. Sample weights were included in the training according to whether the observation fell on a Holiday or not. The model was tuned using GridSearchCV using a 4-fold Time series train, test scheme as shown below:



The following is a plot of the top 20 most important features from the final model.



## Results:

The models were evaluated using the Weighted Mean Absolute Error where the weight is 5 for Holiday dates

Model	Mean Training WMSE	Mean CV WMSE	Test WMSE	Kaggle Leaderboard
1	1293.429	2249.225	2767.456	43rd

## Model 2 - Granular XGB Models:

This approach is more granular than Model 1. A different model was trained for each Store-Dept pair. If less than 10 observations were available in the data for any pair than all the data for that Department was used. The rationale for this approach is that according to EDA the sale trends for all Departments were similar across all Stores. This observation implies that every department sells the same things, irrespective of the Store. Thus, it would make sense to train a model for each Store-Dept pair.

The target was transformed to  $\log(\text{Weekly\_Sales})$  so that a near normal distribution could be simulated. It proved faster to train with better results.

Lesser amount of features were engineered for this approach since now each model will have considerably less samples to train so adding more features would have caused problems in the line of curse of dimensionality.

1. CPI\_cat and Size\_cat: Similar to the ones in Model 1
2. Date\_features
3. LagSales for 5 weeks

3 Fold - TimeSeries cross validation was used to tune the models.

### Results:

Model	Mean Training WMSE	Mean CV WMSE	Test WMSE	Kaggle Leaderboard
2	10.139	242.798	2545.701	13th

### Model 3: Granular RF Models:

Same as Model 2 but used RandomForest instead of XGB model.

#### Results:

Model	Mean Training WMSE	Mean CV WMSE	Test WMSE	Kaggle Leaderboard
3	85.399	266.958	2606.899	20th

### Post Adjustment:

I averaged the results of these models to make the final prediction.




One observation that was noted in data; In the first year of the training data, Christmas occurs on a Saturday (with weeks ending on Friday). That causes all of its sales bulge to fall into the week before. In the second year of the training data, it occurs on a Sunday, so there is one pre-Christmas shopping day in week 52. The test set has Christmas for 2012 which puts it on a Tuesday, with 3 pre-Christmas shopping days in its week.

I implemented a post-forecast adjustment that said that if, in a given department, the average sales for weeks 49, 50 and 51 were at least 10% higher than for weeks 48 and 52, than I would circularly shift a particular fraction of the sales from weeks 48 through 52 into the next week (and from 52 back to 48). Since the underlying model is based on 2 years of data, I shifted  $2.5/7$ . This is because the test year shifts 2 days with respect to the second year of the training data, and 3 days with respect to the first year.

Final Results:

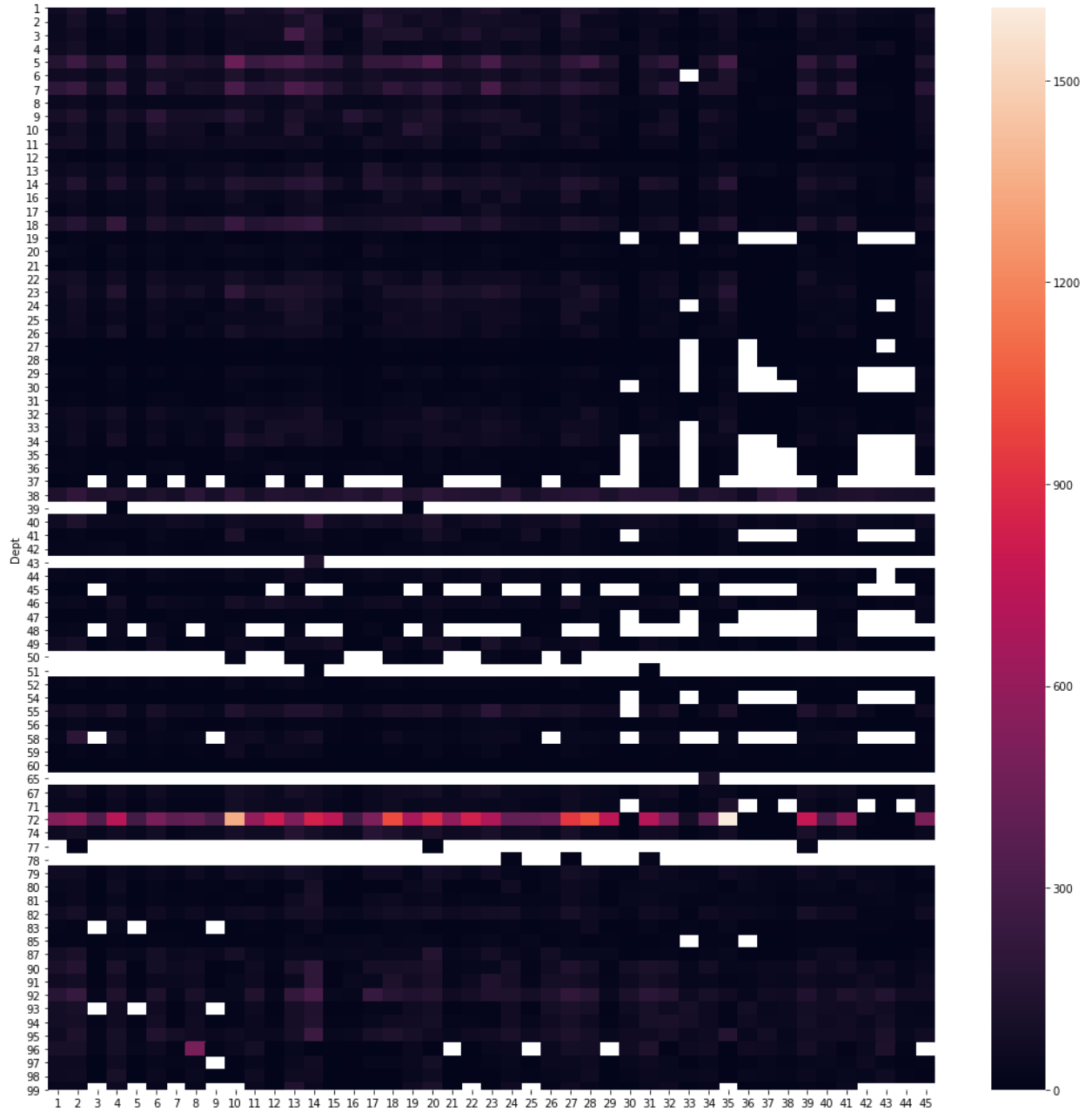
Model	Mean Training WMSE	Mean CV WMSE	Test WMSE	Kaggle Leaderboard
1	1293.429	2249.225	2767.456	43rd
2	10.139	242.798	2545.701	13th
3	85.399	266.958	2606.899	20th
Ensemble Average	53.572	N/A	2504.917	10th
Post Adjustment	N/A	N/A	2424.208	4th

Submission and Description		
<a href="#">final_preds.csv</a>		2432.23112
3 days ago by <a href="#">ShahbazMasood</a>		
<a href="#">add submission details</a>		
<a href="#">weighted_sub.csv</a>		2424.20837
3 days ago by <a href="#">ShahbazMasood</a>		
<a href="#">add submission details</a>		
<a href="#">weighted_sub.csv</a>		2453.72112
7 days ago by <a href="#">ShahbazMasood</a>		
<a href="#">add submission details</a>		

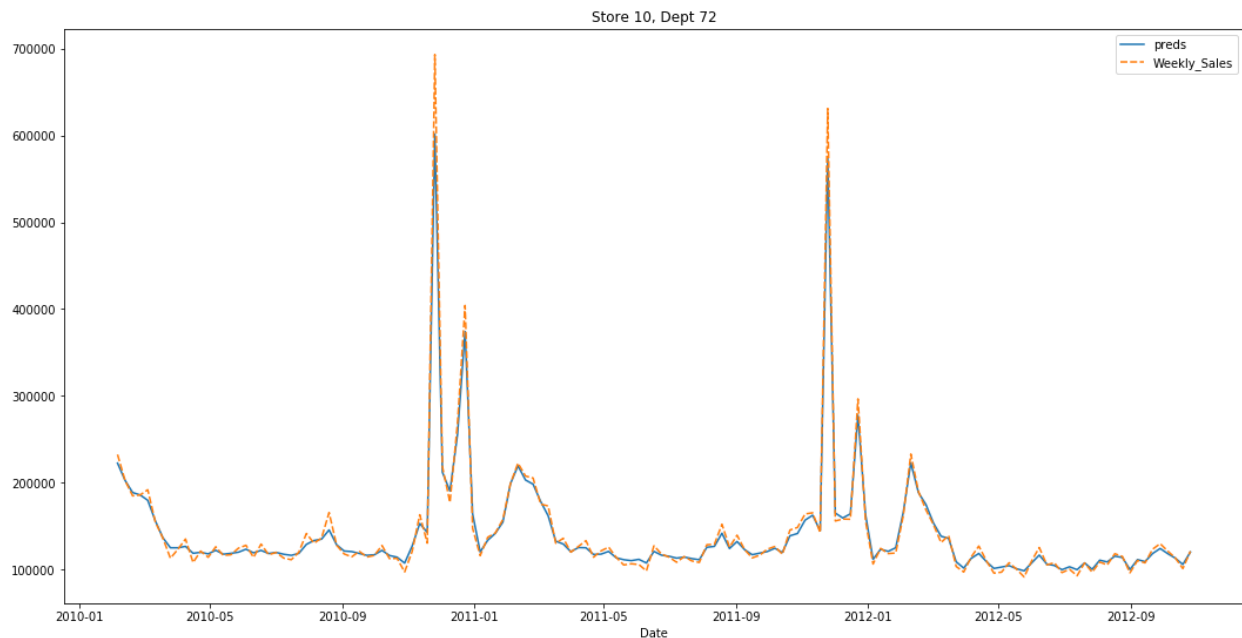
#	Δpub	Team Name	Notebook	Team Members	Score ?	Entries	Last
1	—	David Thaler			2301.48792	223	6y
2	▲1	Srihari Jaganathan			2371.42364	211	6y
3	▼1	James King			2394.70198	170	6y
4	▲1	Giulio			2424.24309	104	6y
5	▲1	Domcastro			2427.05483	216	6y

## Error Analysis

The following is a heatmap of the errors department-Store errors



We can see that the models are not performing well for department 72. The maximum error is on store 10, department 72.



On first sight it seems that the predictions are really good. The problem here is that it is a very high Sales pair. The average sales for this store is more than 10 times the average of all the stores. Thus, even a slight deviation from the actual values results in a high error. One remedy could be to train separate models for such stores.

## Conclusion:

The above approach to forecast sales of store is successful. The results have put me on the 4th position on Kaggle. This approach can be used to forecast the sales of any company for which salesforecasting is crucial for their operations, especially during the Holiday season.

## Further improvements:

Predicting the sales of department 72 can be done separately. If a model can be trained for just this department and be fine tuning accordingly the results can be further improved.