Lambton College

Walmart

SALES FORECASTING

USING MACHINE LEARNING

PRESENTATION

In this project, the team is comparing traditional time series methods with regression models for sales forecasting at Walmart Inc. Using the Walmart Dataset from Kaggle, the team assess the accuracy of each method and select the best-performing model to predict future sales accurately.

Presenter : **Team Everest**

Team Members:
Bharat Dhungana, Satish Kandel, Aljesh Basnet, Bijay Adhikari, Shishir Dhakal

BDM 2053: Big Data Algorithms and Statistic
Lambton College
18th April, 2024

Introduction to Walmart's Regression Models and Sales Prediction

- Due to hidden factors influencing consumer behaviour, Walmart, an international retail giant, finds it **difficult to estimate sales** with any degree of accuracy.
- A **variety of prediction techniques and models** have been developed to support strategic management choices.
- Precise sales forecasts are **essential** for allocating resources and developing successful marketing plans.
- The project investigates if regression models may improve sales prediction accuracy using the Walmart Dataset from Kaggle as a case study. Time series techniques, regression techniques, and assessment metrics like MAE, MSE, and RMSE are all used in this project.

Aim of Project

The aim of this project is designed to improve operational efficiency, accuracy, and effectiveness in Walmart's sales forecasting processes.

Objectives of Project

- Increase Sales Forecasting Accuracy
- Optimize Inventory Management
- Enhance Pricing Strategies
- Improve Operational Efficiency
- Enhance Customer Satisfaction

Rationale of the Project

Issue

Walmart encounters difficulties in precisely projecting sales because of imperceptible elements that affect pricing policies, inventory control, and operational effectiveness, so impeding profitability and competitiveness in the market.

Relevance

Increasing complexity of retail environments, changing consumer behavior, economic fluctuations, and technological advancements.

Significance

Essential for resource allocation, marketing strategies, and overall business decision-making.

Project's Contribution

Project compares traditional time series methods and regression models to determine the most accurate sales forecasting approach, aiding Walmart in informed decision-making.

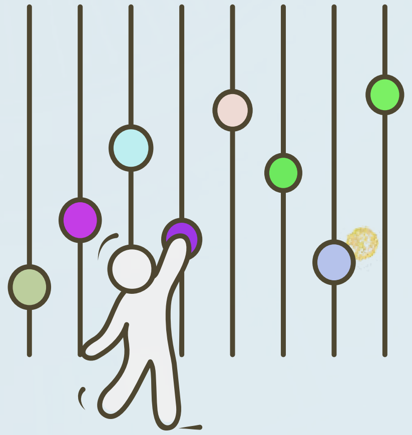
Scope of the Project

The scope of the project involves comparing the accuracy of sales predictions between traditional time series methods and regression models using the Walmart Dataset from Kaggle. Methods such as AR, MA, ARIMA, SARIMA, HW, Decision Tree, Random Forest, and XGBoost will be employed, with evaluation based on metrics like MAE, MSE, and RMSE. Analysis will be conducted in Python using Jupyter Notebook, acknowledging limitations such as time constraints and dataset scope, with the goal of providing insights and recommendations for retail management decision-making.

Dataset Overview

- *Source: The dataset utilized in this research was sourced from the well-known dataset site Kaggle (https://www.kaggle.com/c/walmart-recruiting-store-sales-forecasting). The data's dependability is increased by Kaggle's trustworthiness as a source. The Walmart is the source of this dataset. In a recruiting competition, job-seekers project sales for 45 Walmart stores and select holiday markdown events. This competition showcases modeling mettle and counts towards rankings and achievements.*

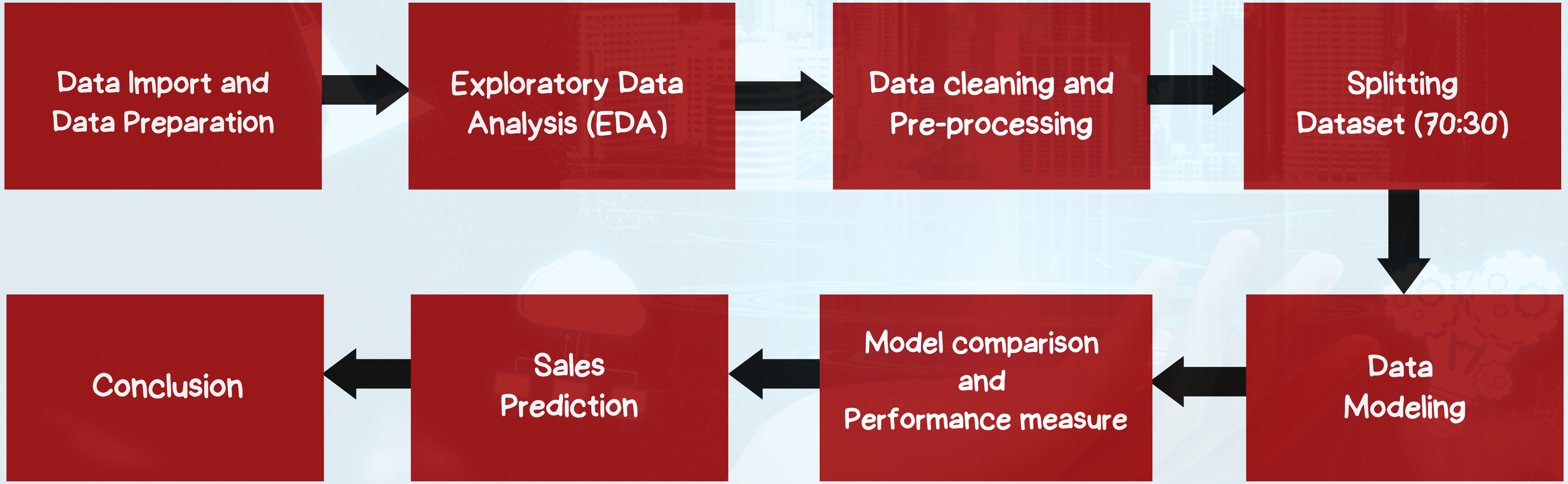
- *Dataset: Walmart has provided historical weekly sales data from 45 stores from 2010 to 2012, totaling 421,570 instances. Each store has around 90 departments. The dataset includes five CSV files: Features, Stores, Train, Test, and SampleSubmission, which will be used for Project submission.*

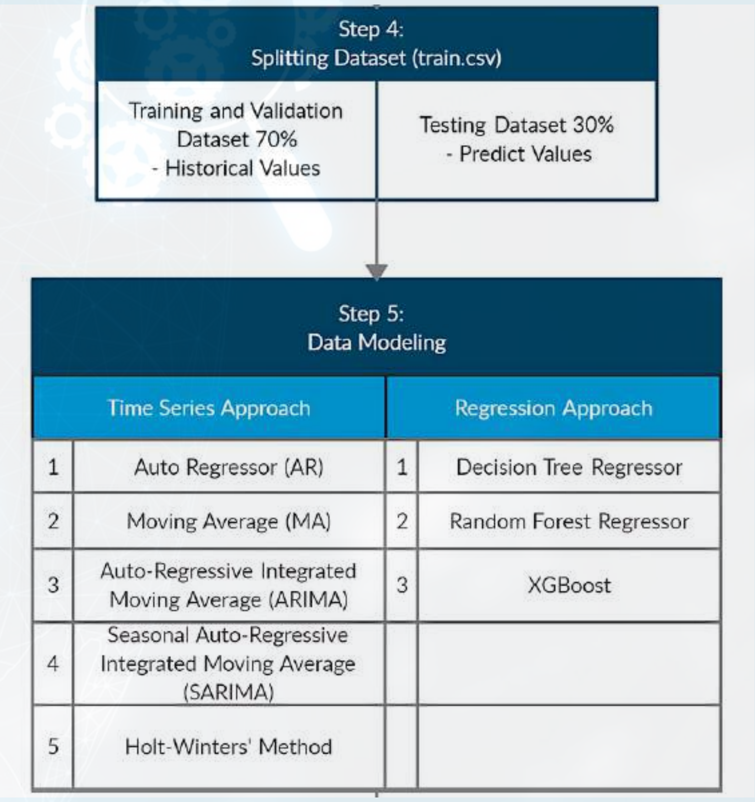


Parameters Taken

- *Store:* *This is the store number, and it ranged from 1-45*
- *Dept:* *It is the department number, it ranged from 1-99, for different categories of items.*
- *Date:* *The Week*
- *IsHoliday:* *whether that specific week has a special holiday in it*
- *Weekly_Sales:* *Store weekly total amount in USD*
- *Temperature:* *The average weekly temperature of the particular store region in Fahrenheit*
- *Fuel_Price:* *Cost of Fuel of the particular store region in USD*
- *MarkDown1-5:* *Anonymized Data related to Walmart's promotional markdown is only available after November 2011, and not all stores have it*
- *CPI:* *Consumer Price Index of specific store region for the week*
- *Unemployment:* *The unemployment rate of particular store region for the month*
- *Type:* *Type of the store, A, B or C*
- *Size:* *Size of the specific store measured in square feet*

METHODOLOGIES APPLIED


Data Import and Data Preparation
Exploratory Data Analysis (EDA)
Data cleaning and Pre-processing
Splitting Dataset (70:30)
Data Modeling
Model comparison and Performance measure
Sales Prediction
Conclusion



Step 4:
Splitting Dataset (train.csv)

Training and Validation Dataset 70% - Historical Values	Testing Dataset 30% - Predict Values

Step 5:
Data Modeling

	Time Series Approach		Regression Approach
1	Auto Regressor (AR)	1	Decision Tree Regressor
2	Moving Average (MA)	2	Random Forest Regressor
3	Auto-Regressive Integrated Moving Average (ARIMA)	3	XGBoost
4	Seasonal Auto-Regressive Integrated Moving Average (SARIMA)		
5	Holt-Winters' Method		

LIBRARIES AND MODULES USED

NumPy

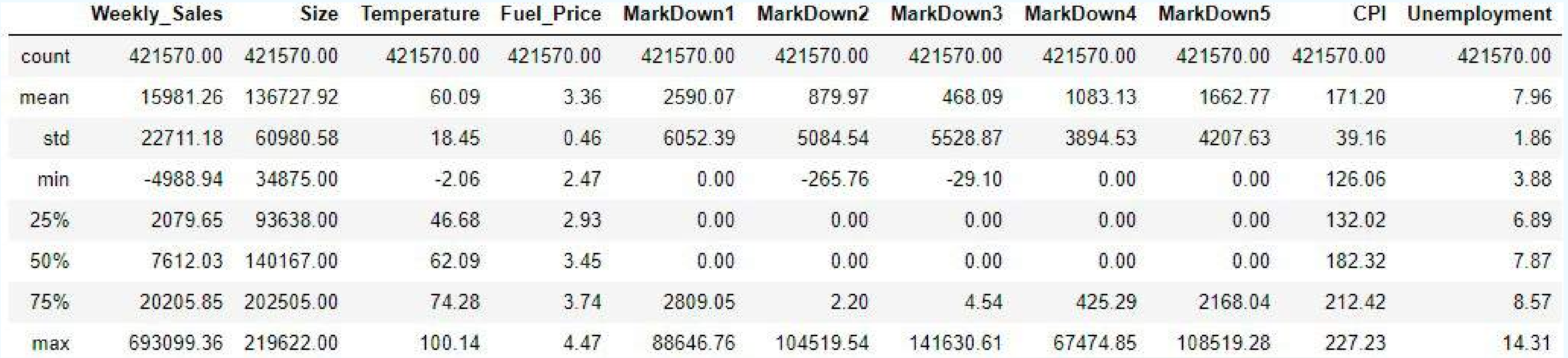
matplotlib

Pandas

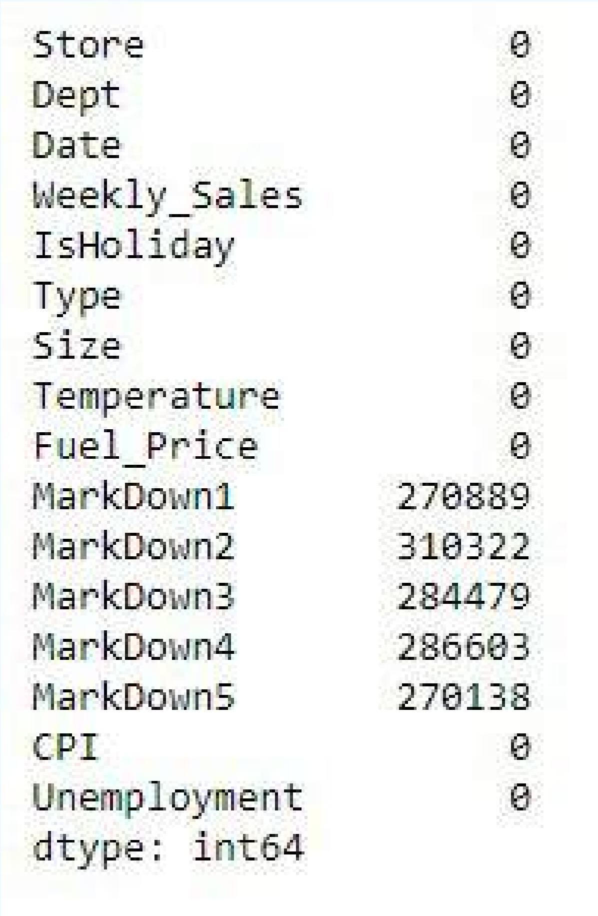
scikit learn

Seaborn

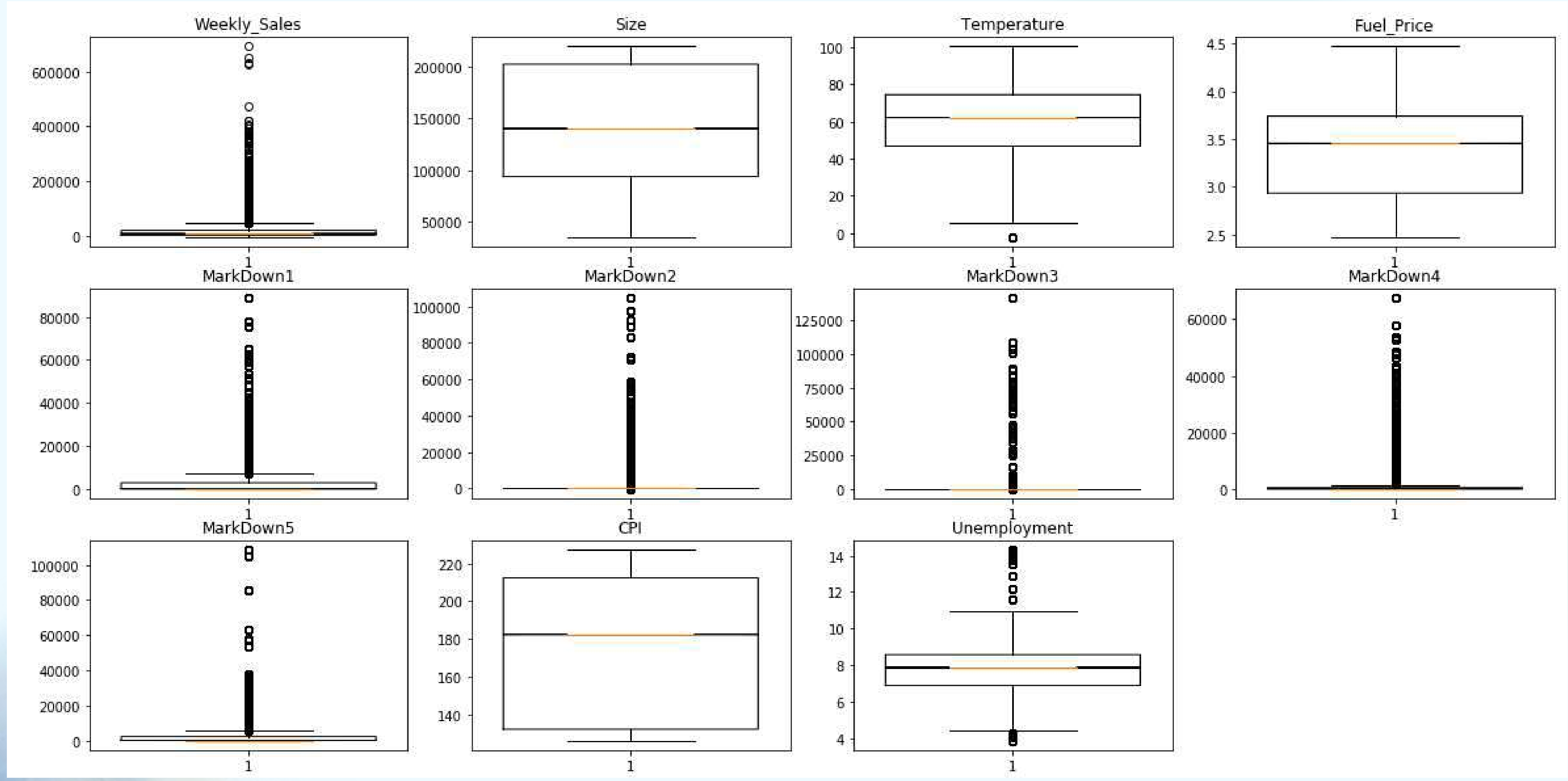
Exploratory Data Analysis (EDA) - I



	Weekly_Sales	Size	Temperature	Fuel_Price	MarkDown1	MarkDown2	MarkDown3	MarkDown4	MarkDown5	CPI	Unemployment
count	421570.00	421570.00	421570.00	421570.00	421570.00	421570.00	421570.00	421570.00	421570.00	421570.00	421570.00
mean	15981.26	136727.92	60.09	3.36	2590.07	879.97	468.09	1083.13	1662.77	171.20	7.96
std	22711.18	60980.58	18.45	0.46	6052.39	5084.54	5528.87	3894.53	4207.63	39.16	1.86
min	-4988.94	34875.00	-2.06	2.47	0.00	-265.76	-29.10	0.00	0.00	126.06	3.88
25%	2079.65	93638.00	46.68	2.93	0.00	0.00	0.00	0.00	0.00	132.02	6.89
50%	7612.03	140167.00	62.09	3.45	0.00	0.00	0.00	0.00	0.00	182.32	7.87
75%	20205.85	202505.00	74.28	3.74	2809.05	2.20	4.54	425.29	2168.04	212.42	8.57
max	693099.36	219622.00	100.14	4.47	88646.76	104519.54	141630.61	67474.85	108519.28	227.23	14.31

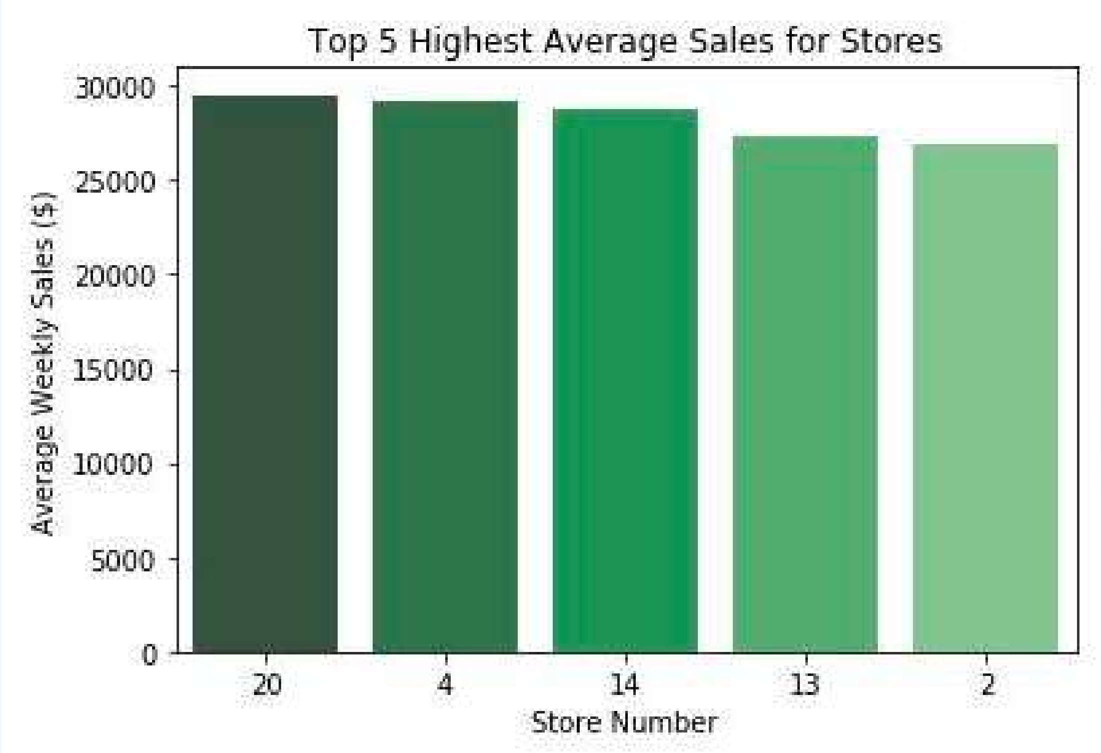


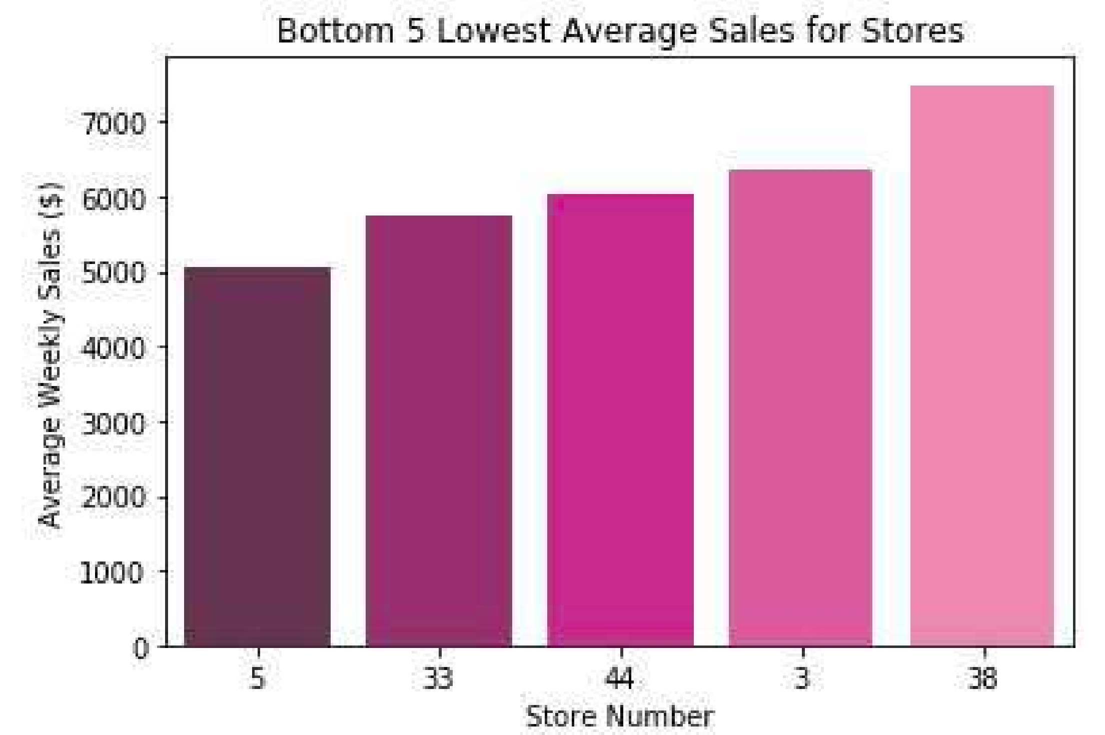
```
Store                0
Dept                 0
Date                 0
Weekly_Sales         0
IsHoliday            0
Type                 0
Size                 0
Temperature          0
Fuel_Price           0
MarkDown1       270889
MarkDown2       310322
MarkDown3       284479
MarkDown4       286603
MarkDown5       270138
CPI                  0
Unemployment         0
dtype: int64
```

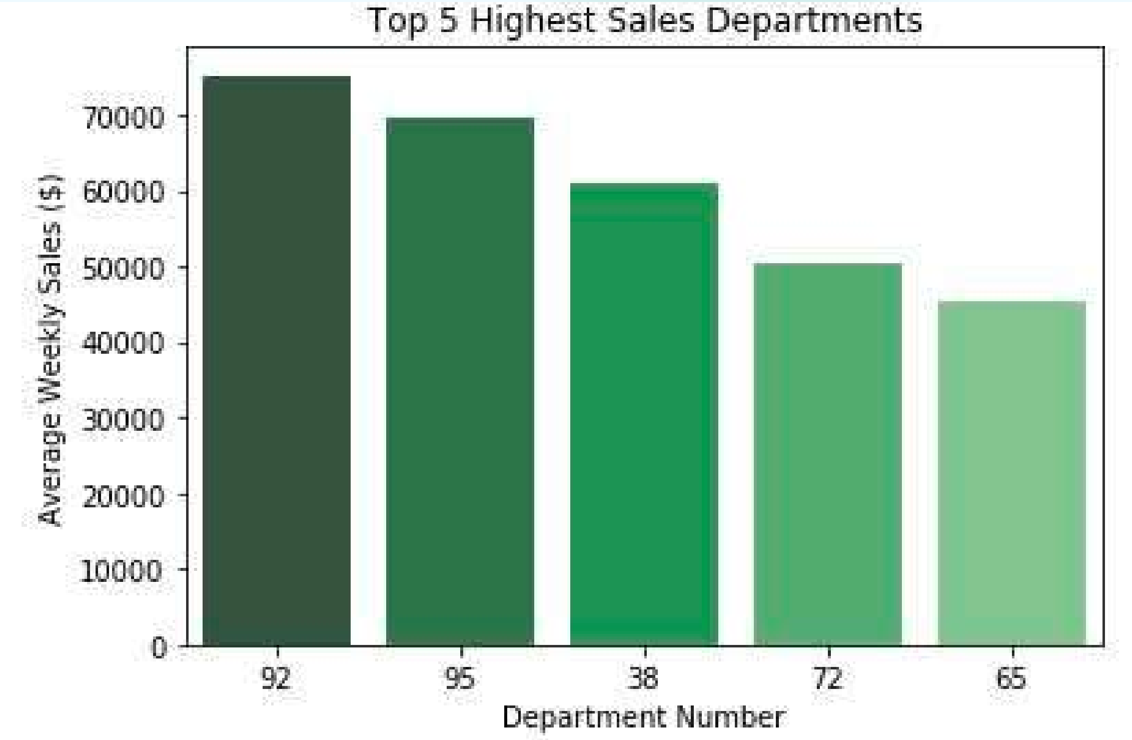


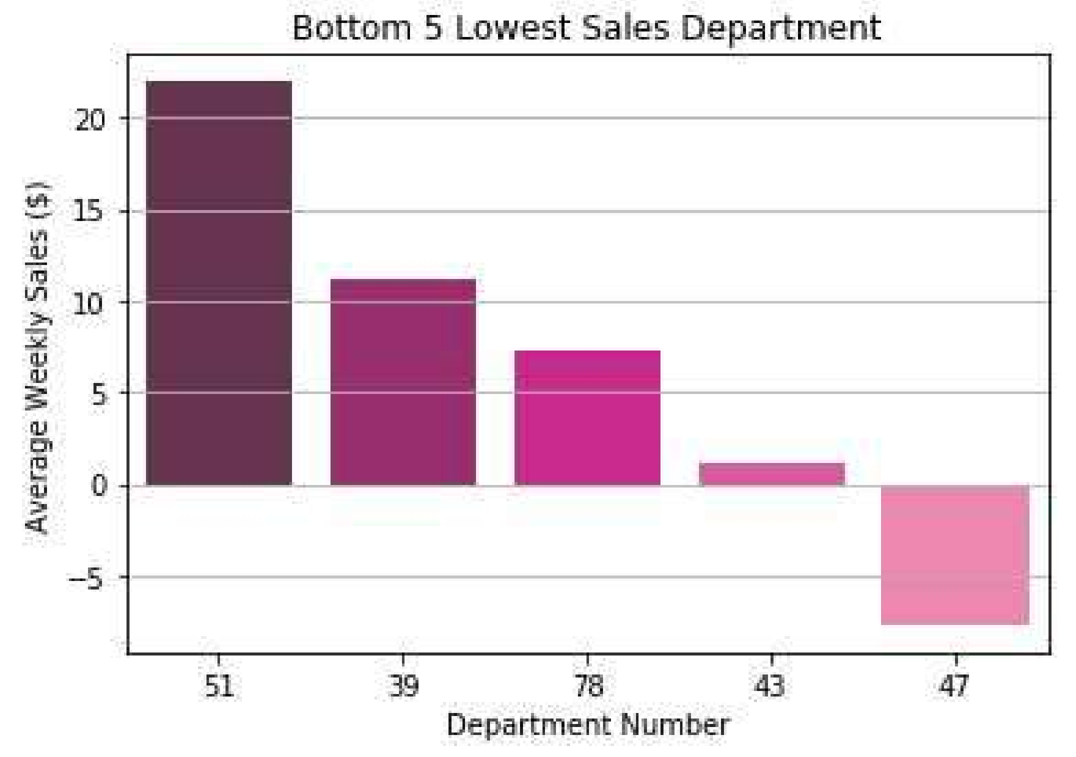
- *Descriptive statistic for the numeric attributes* (Left top)
- *NA's in the merged dataset* (Right top)
- *Outliers* (Left down)

Exploratory Data Analysis (EDA) - II


Top 5 Highest Average Sales for Stores

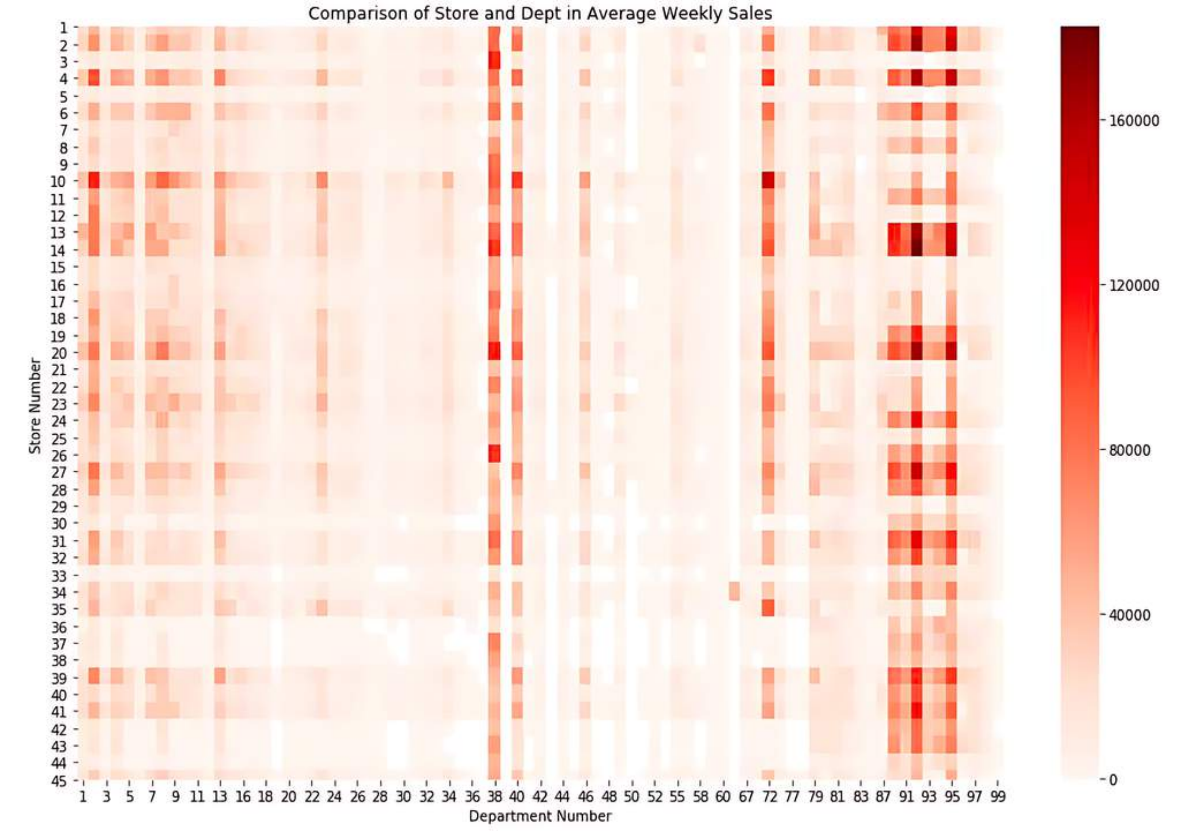

Bottom 5 Lowest Average Sales for Stores

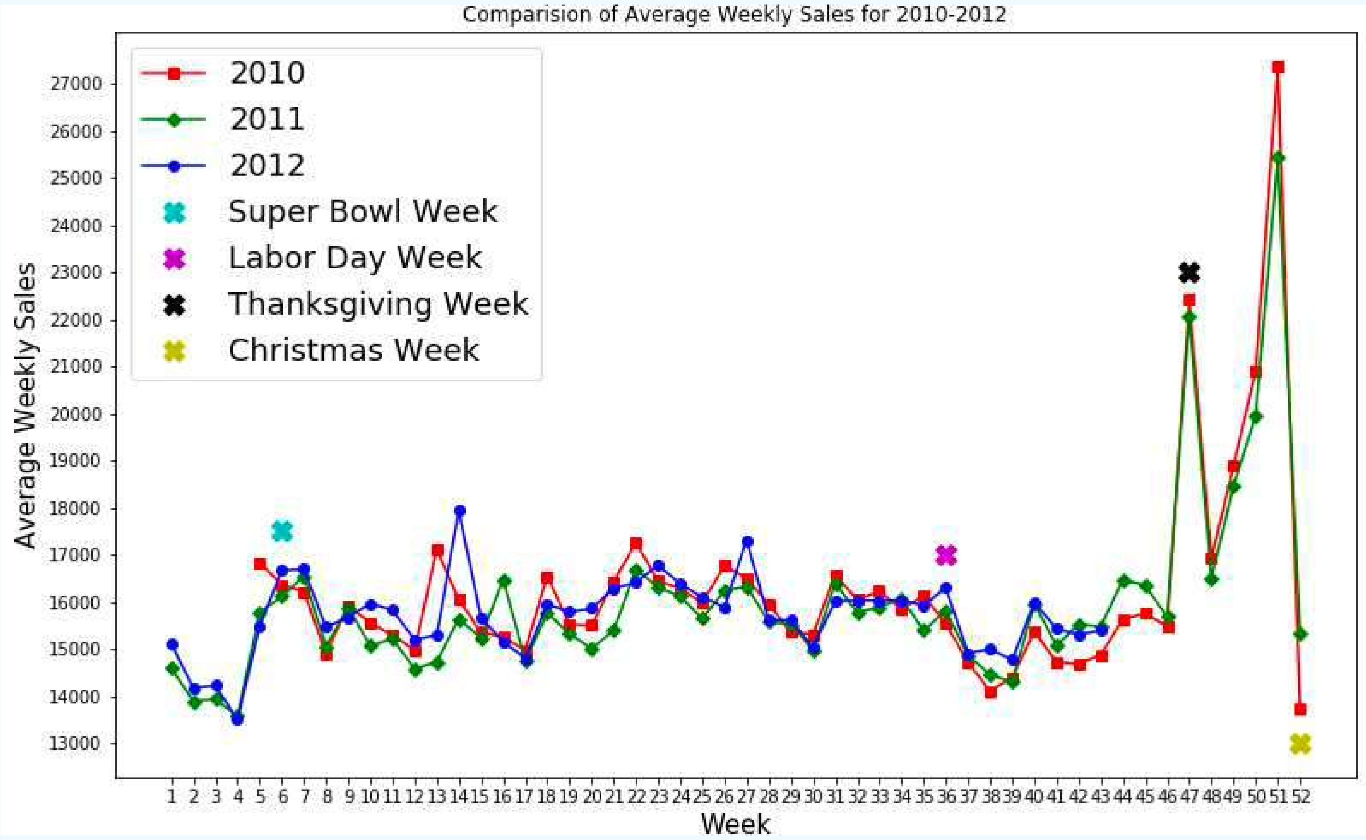

Top 5 Highest Sales Departments

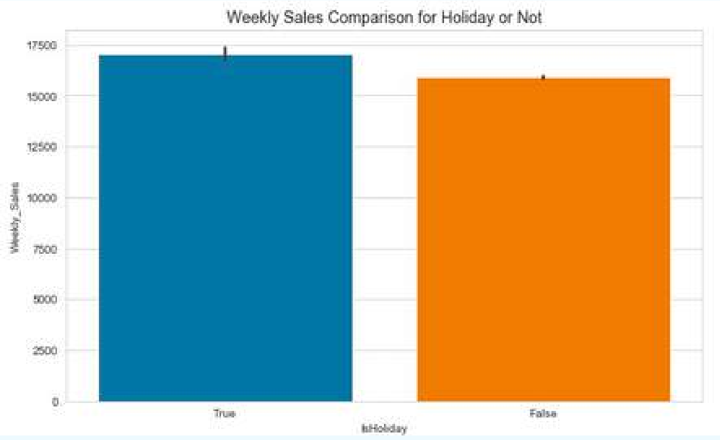

Bottom 5 Lowest Sales Department

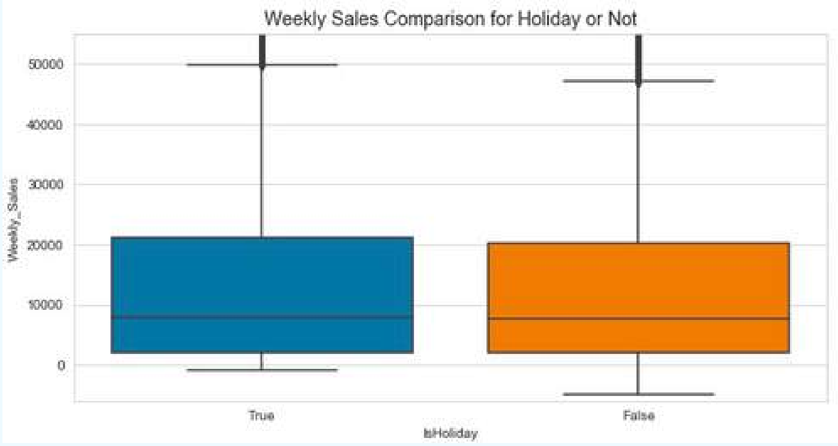
- *Top and bottom five stores in sales (top 2)*
- *Top and Bottom Five Departments in Average Sales (down 2)*

Exploratory Data Analysis (EDA) - III



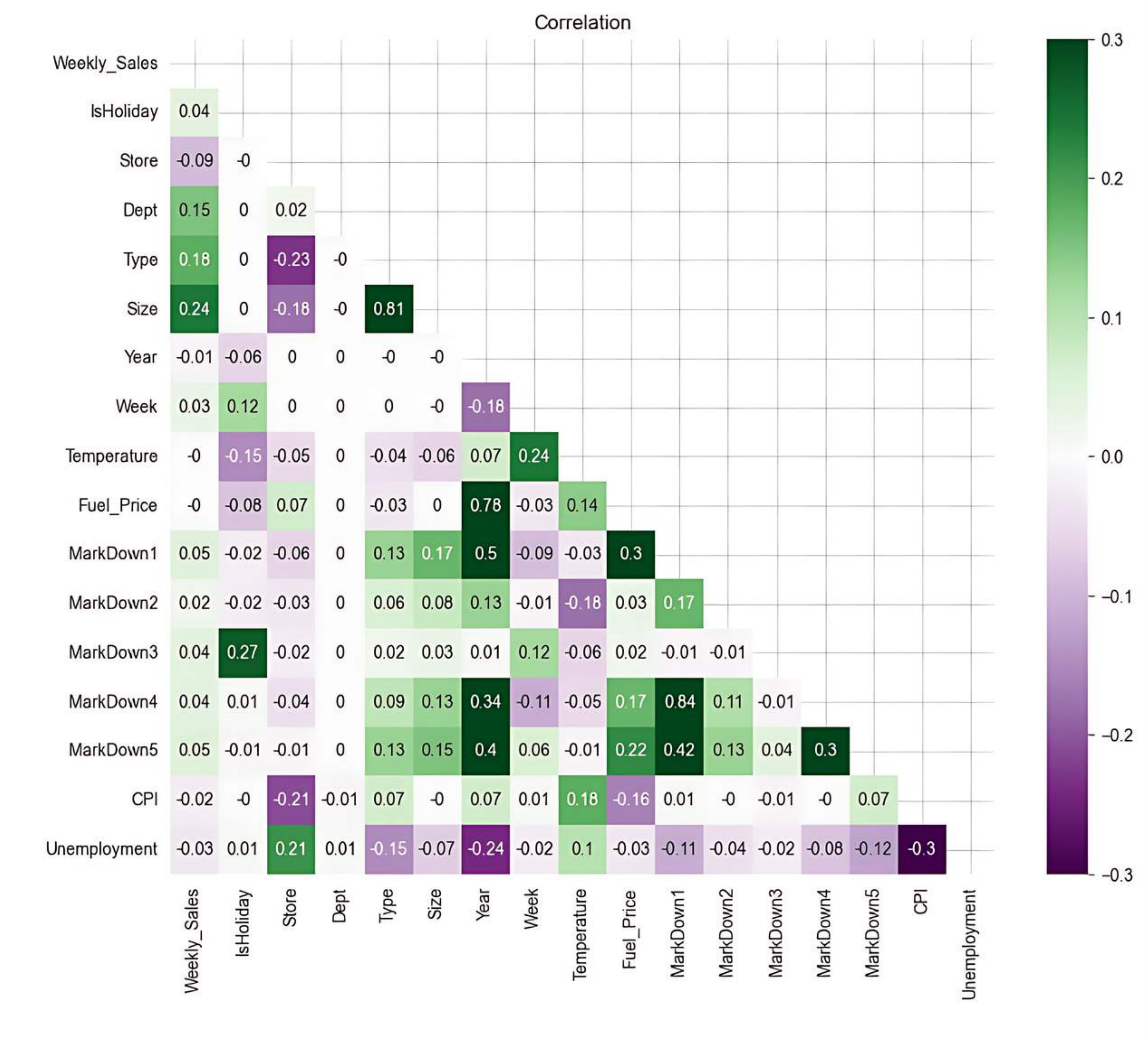






- *Comparison of stores and departments on average weekly sales (top left)*
- *Comparison of average weekly sales for 2010-2012 (top right)*
- *Weekly comparison for IsHoliday or not (down)*

Exploratory Data Analysis (EDA) - IV

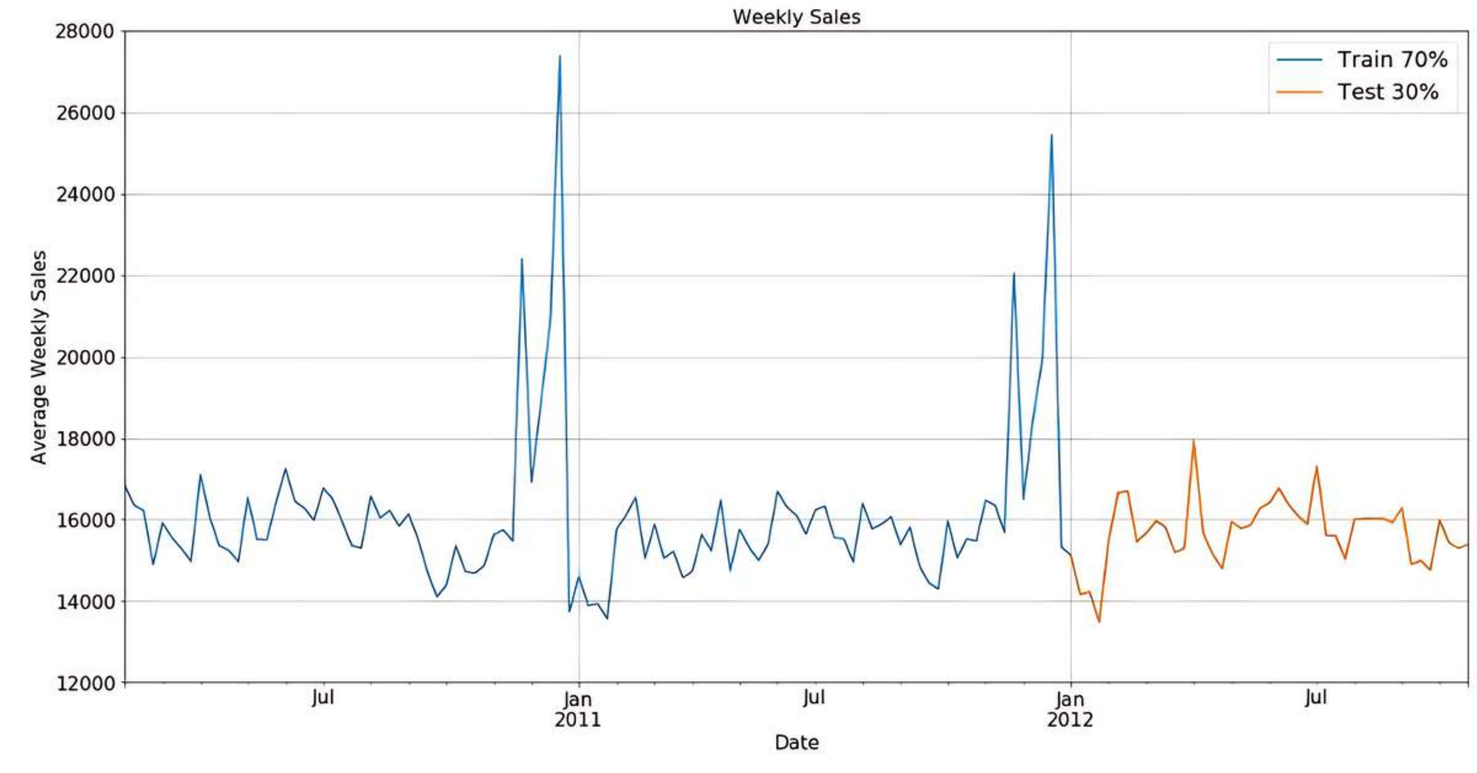


Correlation

	Weekly_Sales	IsHoliday	Store	Dept	Type	Size	Year	Week	Temperature	Fuel_Price	MarkDown1	MarkDown2	MarkDown3	MarkDown4	MarkDown5	CPI	Unemployment
Weekly_Sales																	
IsHoliday	0.04																
Store	-0.09	-0															
Dept	0.15	0	0.02														
Type	0.18	0	-0.23	-0													
Size	0.24	0	-0.18	-0	0.81												
Year	-0.01	-0.06	0	0	-0	-0											
Week	0.03	0.12	0	0	0	-0	-0.18										
Temperature	-0	-0.15	-0.05	0	-0.04	-0.06	0.07	0.24									
Fuel_Price	-0	-0.08	0.07	0	-0.03	0	0.78	-0.03	0.14								
MarkDown1	0.05	-0.02	-0.06	0	0.13	0.17	0.5	-0.09	-0.03	0.3							
MarkDown2	0.02	-0.02	-0.03	0	0.06	0.08	0.13	-0.01	-0.18	0.03	0.17						
MarkDown3	0.04	0.27	-0.02	0	0.02	0.03	0.01	0.12	-0.06	0.02	-0.01	-0.01					
MarkDown4	0.04	0.01	-0.04	0	0.09	0.13	0.34	-0.11	-0.05	0.17	0.84	0.11	-0.01				
MarkDown5	0.05	-0.01	-0.01	0	0.13	0.15	0.4	0.06	-0.01	0.22	0.42	0.13	0.04	0.3			
CPI	-0.02	-0	-0.21	-0.01	0.07	-0	0.07	0.01	0.18	-0.16	0.01	-0	-0.01	-0	0.07		
Unemployment	-0.03	0.01	0.21	0.01	-0.15	-0.07	-0.24	-0.02	0.1	-0.03	-0.11	-0.04	-0.02	-0.08	-0.12	-0.3	

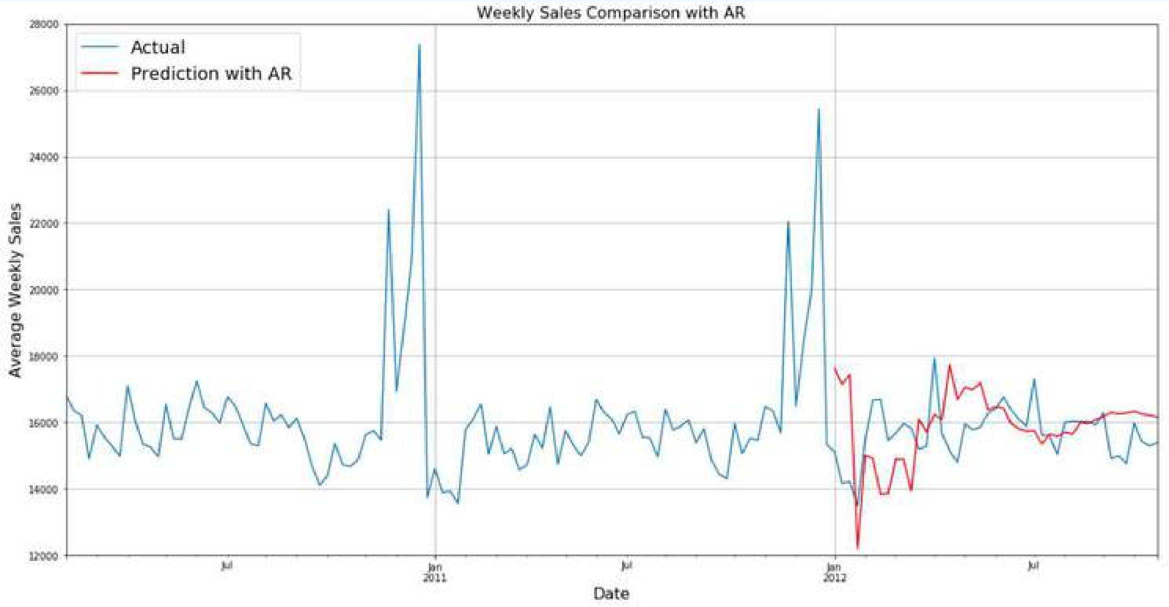
- *Correlation Heatmap* *(top left)*

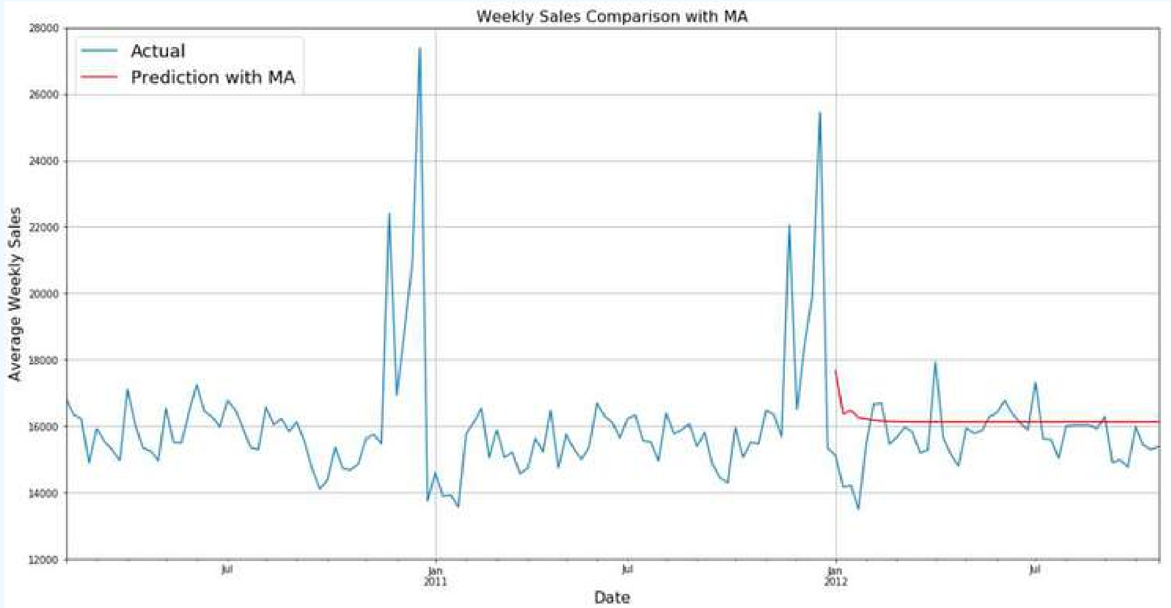
The figure compares attributes against the "Weekly_Sales" attribute, revealing low correlations between independent variables like temperature, fuel_price, CPI, unemployment, and Markdown 1-5. However, "IsHoliday" and "Store", "Dept", "Year", and "Week" are not removed due to their essential role in classifying holidays. "Type" and "Size" have weak positive linear relationships.

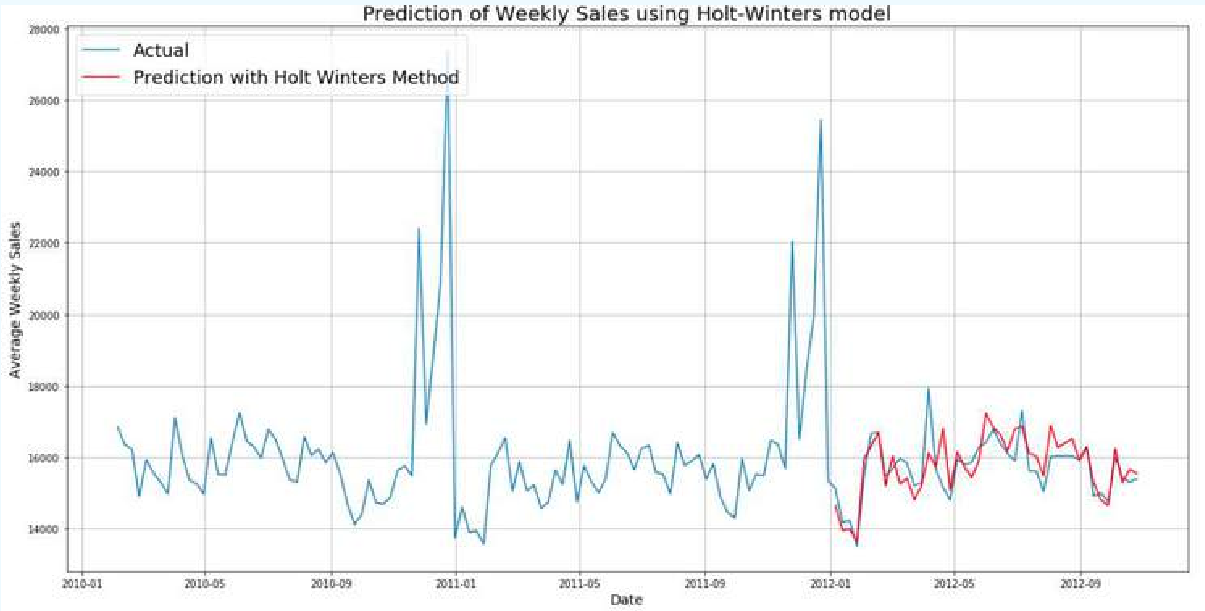


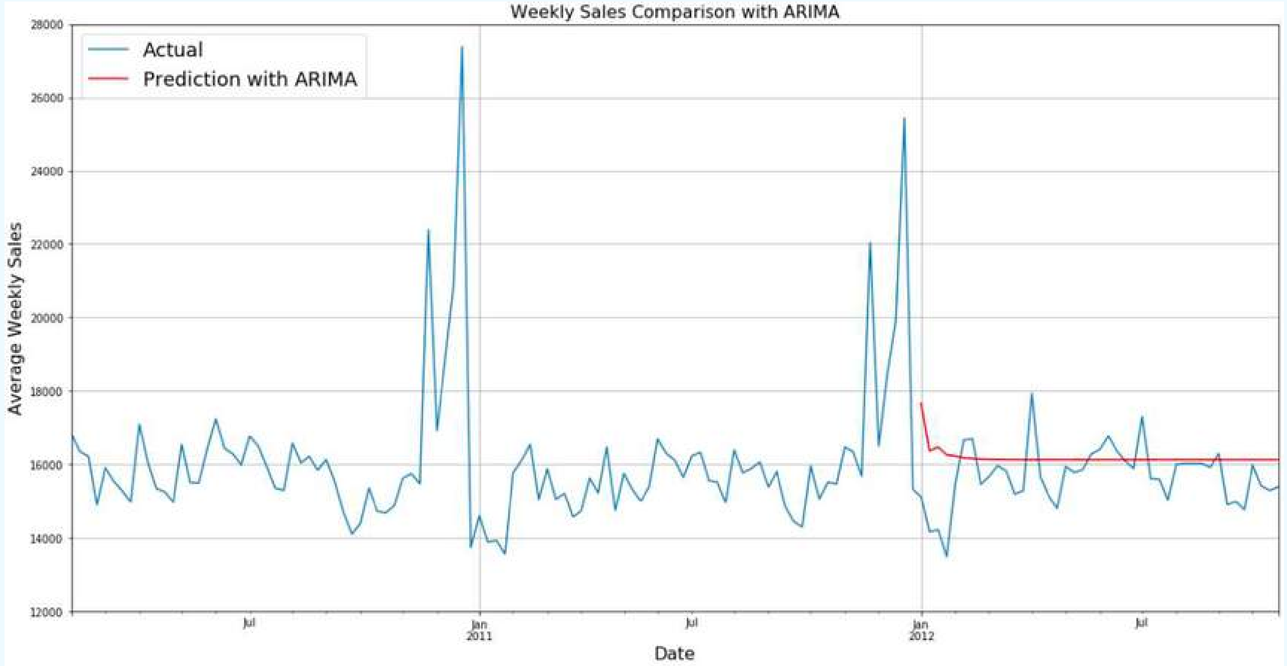
Split Training and Test set *(down right)*

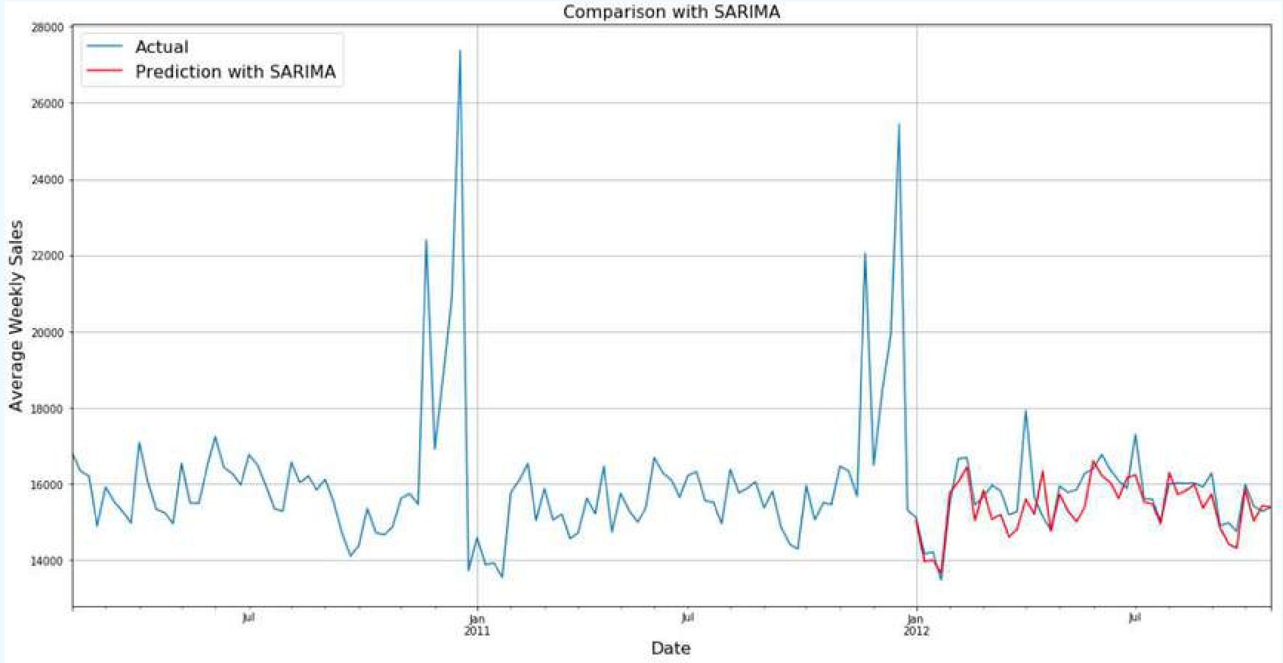
Data Modeling – Time Series Approach





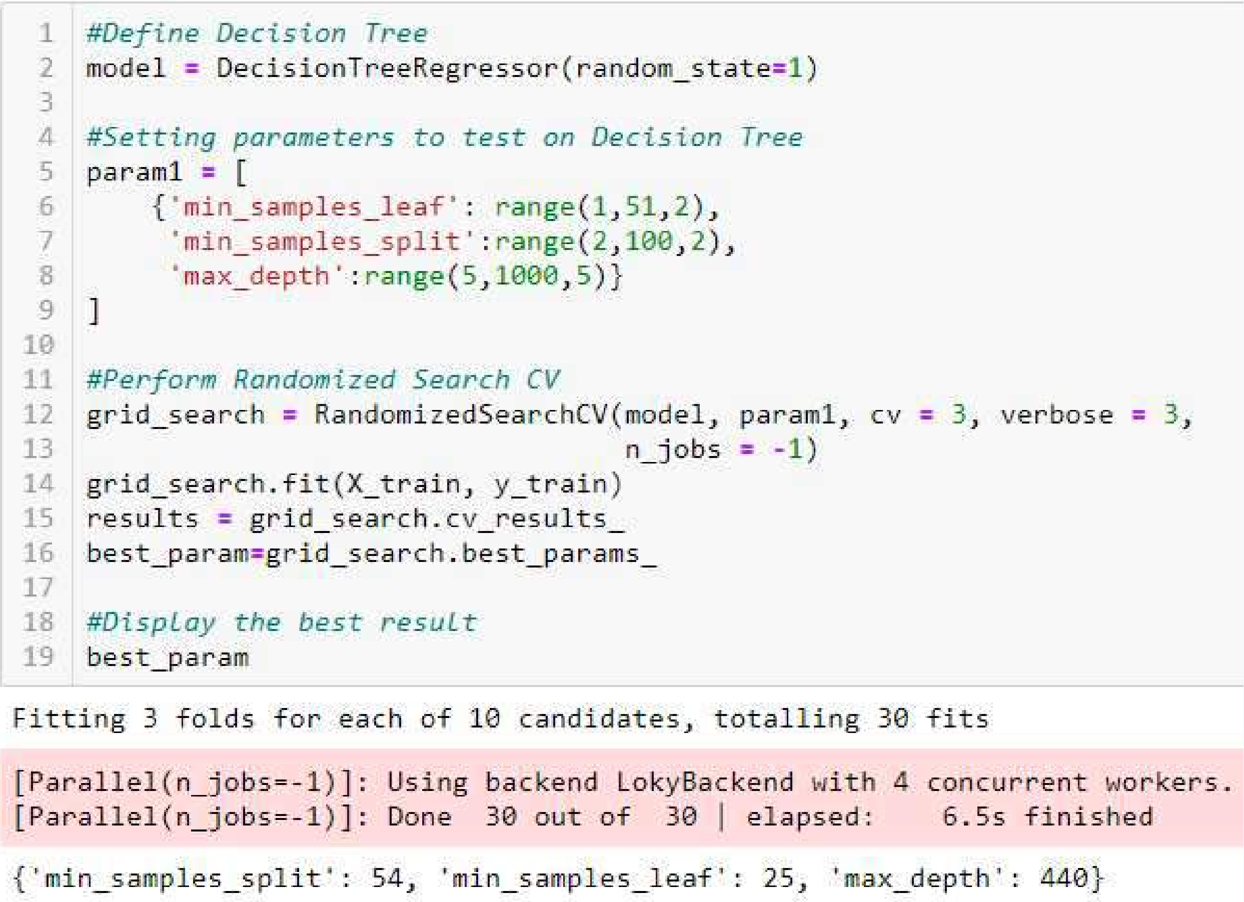






- *AR prediction compared to the actual data (top left)*
- *MA prediction compared to the actual data (top middle)*
- *AR prediction compared to the actual data (down left)*
- *SARIMA prediction compared to the actual data (down right)*
- *HW prediction compared to the actual data (top right)*

Data Modeling – Regression Approach



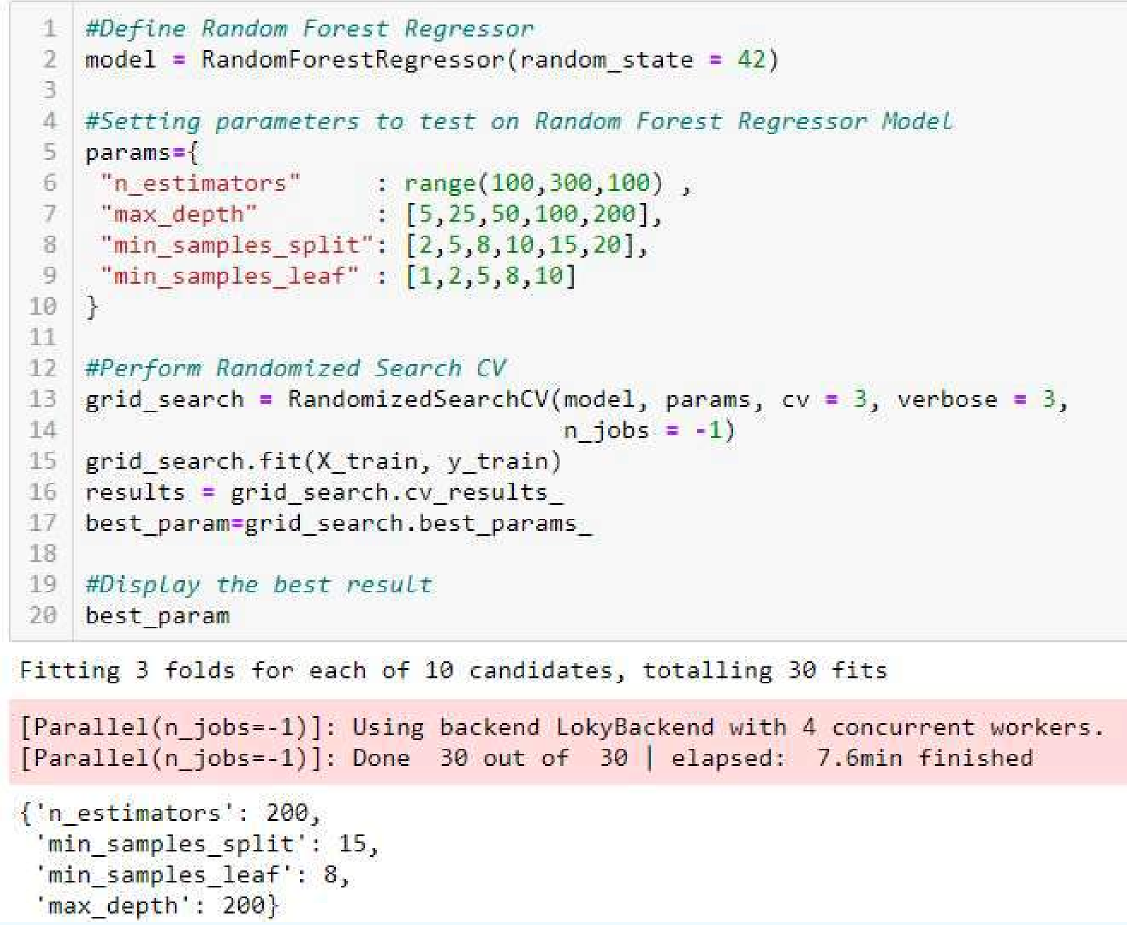
```
#Define Decision Tree
model = DecisionTreeRegressor(random_state=1)

#Setting parameters to test on Decision Tree
param1 = [
    {'min_samples_leaf': range(1,51,2),
     'min_samples_split':range(2,100,2),
     'max_depth':range(5,1000,5)}
]

#Perform Randomized Search CV
grid_search = RandomizedSearchCV(model, param1, cv = 3, verbose = 3,
                                 n_jobs = -1)
grid_search.fit(X_train, y_train)
results = grid_search.cv_results_
best_param=grid_search.best_params_

#Display the best result
best_param
```

```
Fitting 3 folds for each of 10 candidates, totalling 30 fits
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 4 concurrent workers.
[Parallel(n_jobs=-1)]: Done  30 out of  30 | elapsed:    6.5s finished
{'min_samples_split': 54, 'min_samples_leaf': 25, 'max_depth': 440}
```



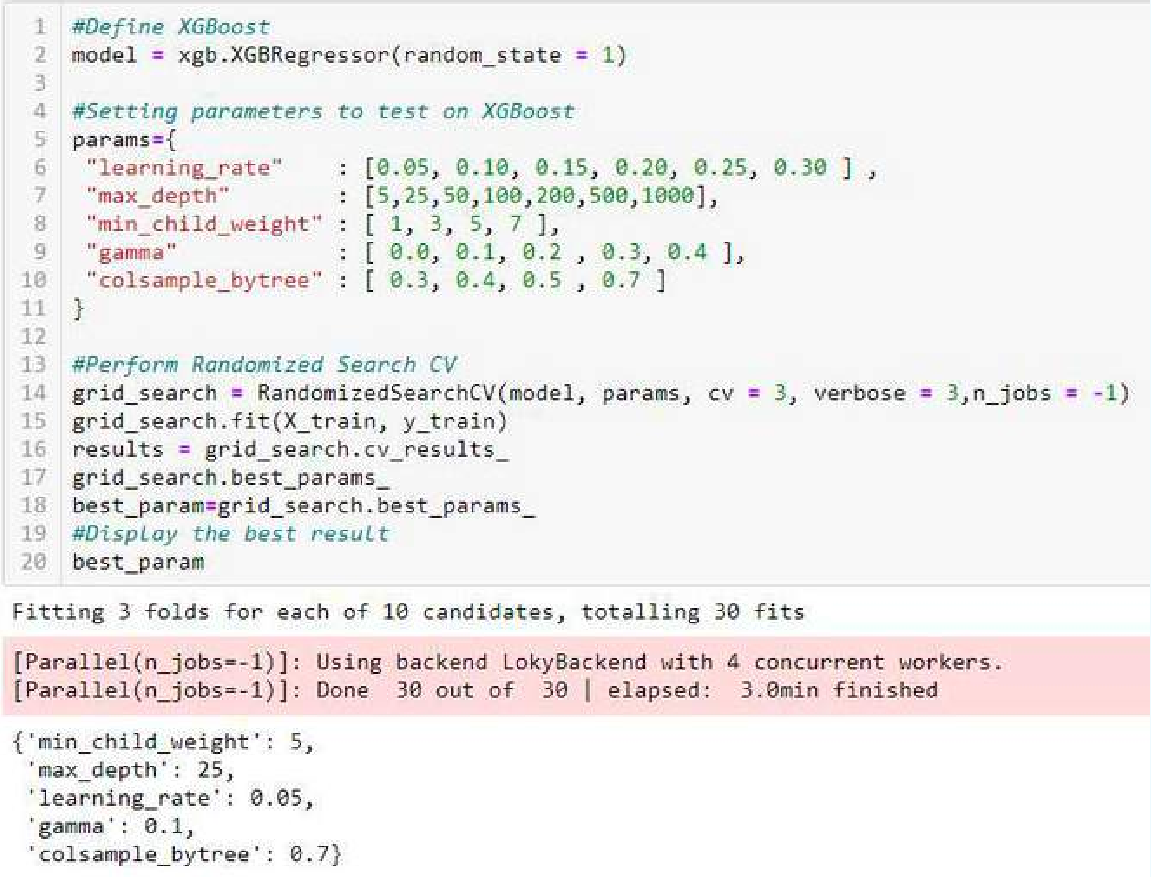
```
#Define Random Forest Regressor
model = RandomForestRegressor(random_state = 42)

#Setting parameters to test on Random Forest Regressor Model
params={
 "n_estimators"     : range(100,300,100) ,
 "max_depth"        : [5,25,50,100,200],
 "min_samples_split": [2,5,8,10,15,20],
 "min_samples_leaf" : [1,2,5,8,10]
}

#Perform Randomized Search CV
grid_search = RandomizedSearchCV(model, params, cv = 3, verbose = 3,
                                 n_jobs = -1)
grid_search.fit(X_train, y_train)
results = grid_search.cv_results_
best_param=grid_search.best_params_

#Display the best result
best_param
```

```
Fitting 3 folds for each of 10 candidates, totalling 30 fits
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 4 concurrent workers.
[Parallel(n_jobs=-1)]: Done  30 out of  30 | elapsed:  7.6min finished
{'n_estimators': 200,
 'min_samples_split': 15,
 'min_samples_leaf': 8,
 'max_depth': 200}
```



```
#Define XGBoost
model = xgb.XGBRegressor(random_state = 1)

#Setting parameters to test on XGBoost
params={
 "learning_rate"    : [0.05, 0.10, 0.15, 0.20, 0.25, 0.30 ] ,
 "max_depth"        : [5,25,50,100,200,500,1000],
 "min_child_weight" : [ 1, 3, 5, 7 ],
 "gamma"            : [ 0.0, 0.1, 0.2 , 0.3, 0.4 ],
 "colsample_bytree" : [ 0.3, 0.4, 0.5 , 0.7 ]
}

#Perform Randomized Search CV
grid_search = RandomizedSearchCV(model, params, cv = 3, verbose = 3,n_jobs = -1)
grid_search.fit(X_train, y_train)
results = grid_search.cv_results_
grid_search.best_params_
best_param=grid_search.best_params_
#Display the best result
best_param
```

```
Fitting 3 folds for each of 10 candidates, totalling 30 fits
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 4 concurrent workers.
[Parallel(n_jobs=-1)]: Done  30 out of  30 | elapsed:  3.0min finished
{'min_child_weight': 5,
 'max_depth': 25,
 'learning_rate': 0.05,
 'gamma': 0.1,
 'colsample_bytree': 0.7}
```

- *RandomizedSearchCV for decision tree regressor (top left)*
- *RandomizedSearchCV for random forest regressor(top middle)*
- *RandomizedSearchCV for XGBoost (top right)*

Model Comparison and Performance Measure

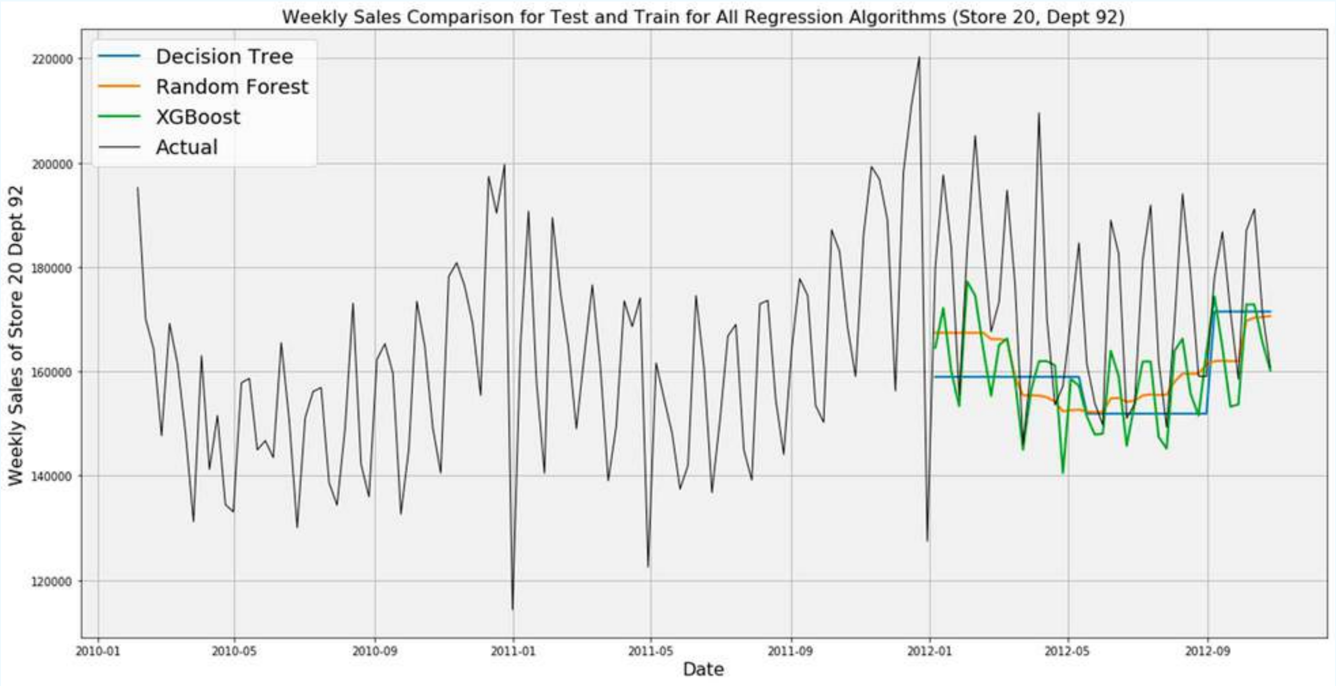
Methods\Metics	MAE	MSE	RMSE	Rank
AR	1,019.58	1,786,083.79	1,336.44	5
MA	747.95	1,027,941.54	1,013.87	3/4
ARIMA	747.95	1,027,941.54	1,013.87	3/4
SARIMA	422.64	343,466.02	586.06	2
Holt Winters' Method	383.26	282,655.10	531.65	1
Decision Tree Regressor	5,753.38	127,988,545.76	11,313.20	8
Random Forest Regressor	5,126.75	110,149,330.12	10,495.21	6/7
XGBoost	5,343.32	95,199,340.15	9,757.01	6/7

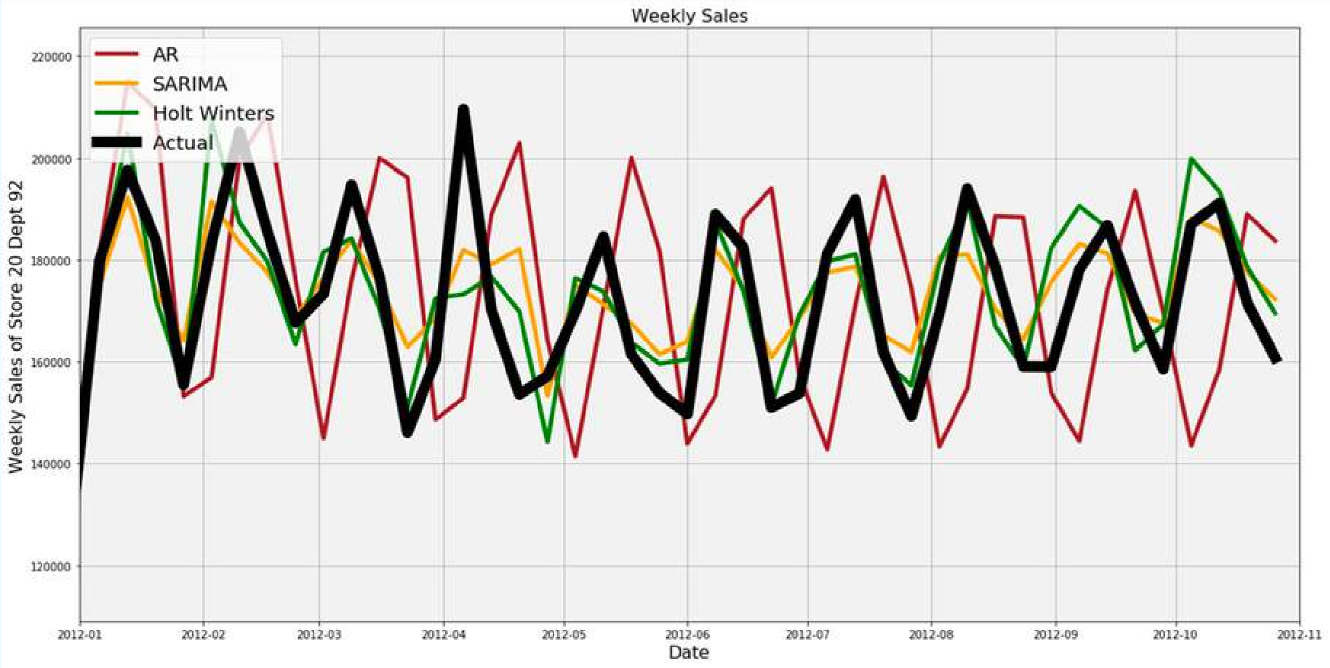
Time Series and Regression method comparison

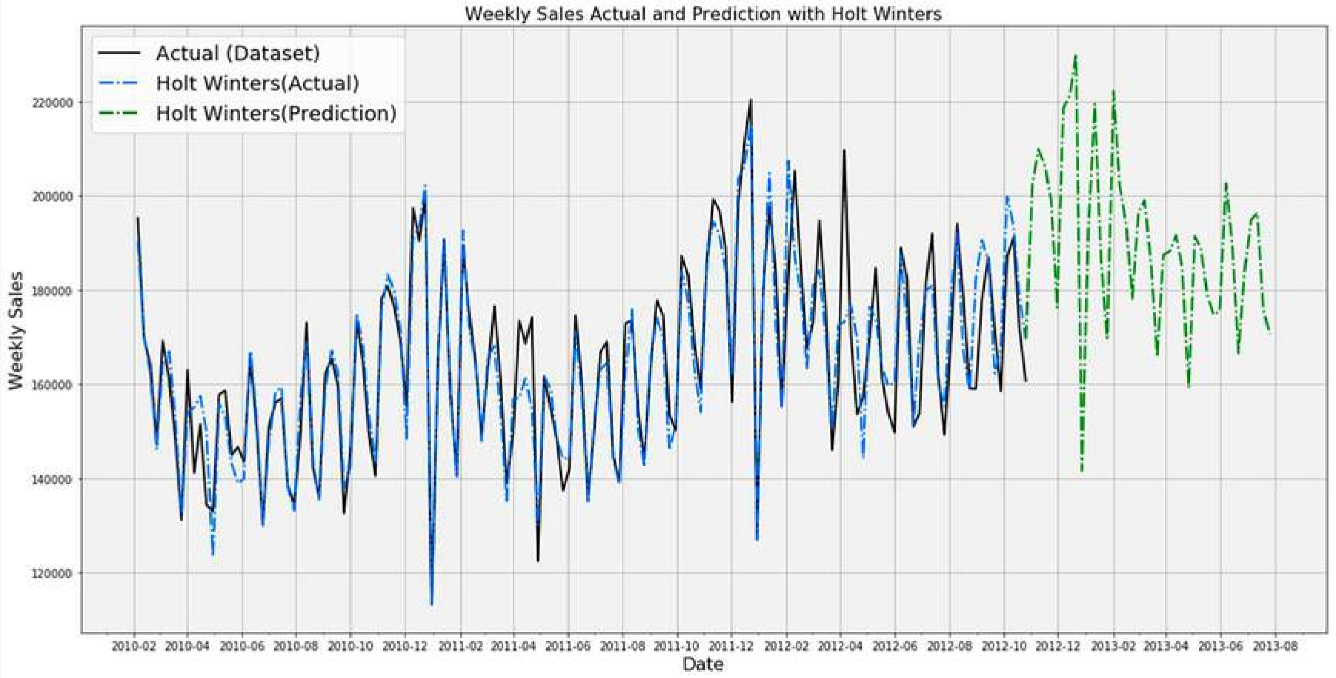
As the table, the models are compared against each other using performance measure of MSE, MAE, RMSE. HW has the lowest error in MAE, MSE, and RMSE, and it is the best model to predict the test set from the train set. Surprisingly, the time series models perform much better than regression models

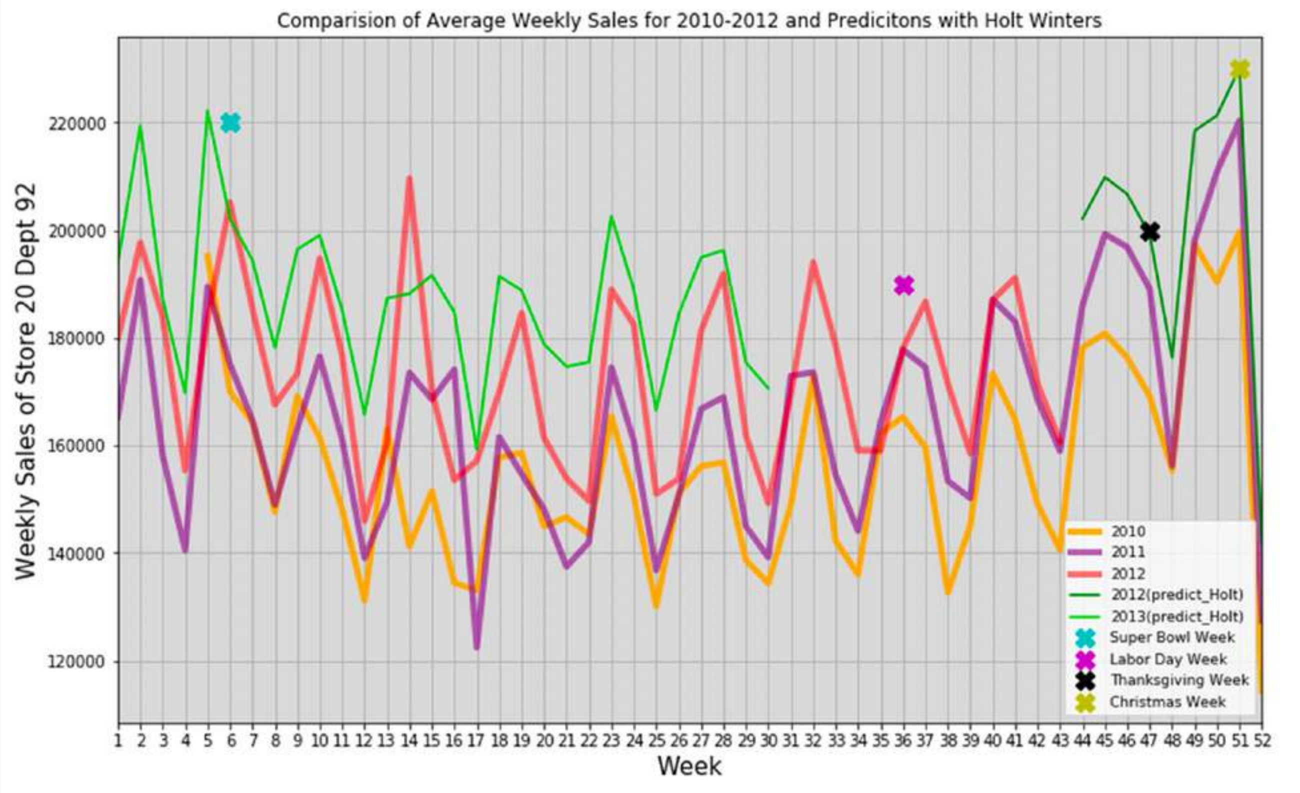
Sales Prediction

- *Regression algorithms comparison for train and test set (top left)*
- *Time series algorithms comparison for train and test set (top right)*









- *HW Method for S20_D92 (down left)*
- *HW prediction for S20_D92 and previous year with holidays (down right)*

Note: Store 20 had the highest average weekly sales from 2010-2012, with department 92 (Grocery) having the highest sales of $160K, making it the chosen department.

Conclusion & Recommendation

- **Time series models** outperform regression models, especially Holt-Winters (HW), due to their ability to predict past values.
- **Regression models** may not capture complex patterns in time series data due to reliance on known attributes.
- Limited **correlation** between attributes and weekly sales may have hindered regression model performance.
- **Improvement areas** include increasing hyperparameter tuning parameters, applying Holt-Winters to all stores and departments, and addressing technical issues with WMAE.

Challenges of the Project

Data Quality

Ensuring data quality and completeness of Walmart dataset from Kaggle.

Complex Relationship

Modeling complex relationships between factors influencing sales.

Overfitting

Preventing overfitting of models to training data.

Deployment

Overcoming challenges in deploying the model into production.

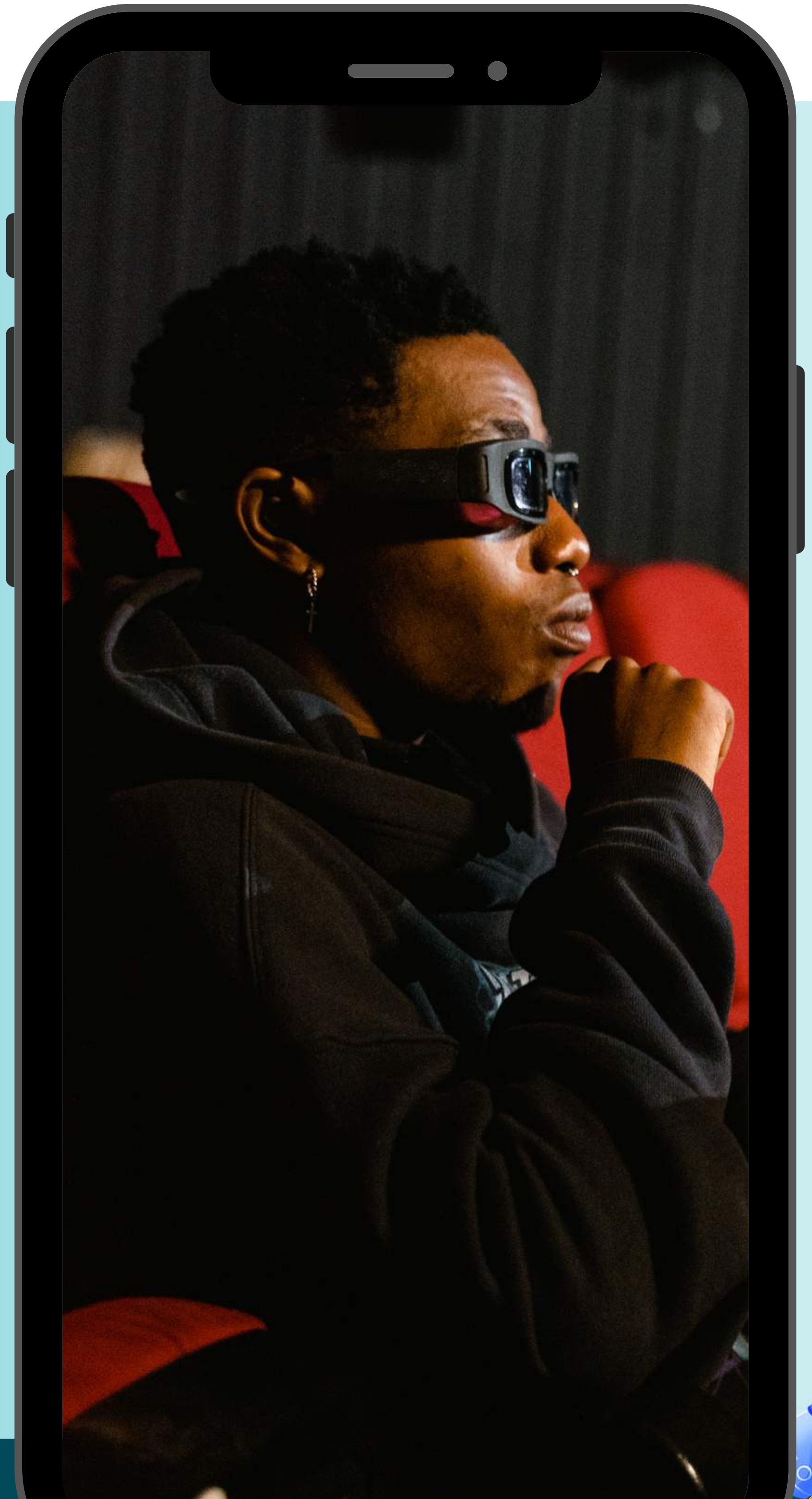
Model Selection

Selecting appropriate predictive algorithms considering interpretability, scalability, computational complexity.

Resource Constraints

Managing resource constraints throughout the project lifecycle.

Future Scopes of Walmart Sales Prediction



I **Expansion to other retail sectors**
Explore applications in fashion, electronics, automotive.

II **Real-time sales forecasting:**
Develop capabilities for immediate response to market changes and consumer behavior.

III **Integration of external data sources**
Enhance sales predictions accuracy and granularity.

IV **Personalized marketing strategies**
Tailor marketing strategies to individual customer preferences.

V **Inventory optimization**
Leverage predictive analytics for accurate demand forecasting and inventory optimization.

VI **Omnichannel retailing**
Support both online and offline sales channels.

VII **IoT and sensor data integration**
Capture real-time insights into customer behavior and inventory movement.

Used Tools and Technologies

GitHub

Repository Hosting Manager

GitLab

Version control

Google Collaborator

Canva

Presentation

Microsoft Teams

References

- *Bonnes, K. (2014). Predictive analytics for supply chains: A systematic literature review. In 21st twente student conference on IT. Netherlands.*
- *Catal, C., Kaan, E. C. E., Arslan, B., & Akbulut, A. (2019). Benchmarking of Regression Algorithms and Time Series Analysis Techniques for Sales Forecasting. Balkan Journal of Electrical and Computer Engineering, 7(1), 20-26.*
- *Hülsmann, M., Borscheid, D., Friedrich, C. M., & Reith, D. (2012). General sales forecast models for automobile markets and their analysis. Trans. MLDM, 5(2), 65-86.*
- *Jain, A., Menon, M. N., & Chandra, S. (2015). Sales forecasting for retail chains.*
- *Knott, B., Liu, H., & Simpson, A (2015). Predicting Sales for Rossmann Drug Stores.*
- *Lasek, A., Cercone, N., & Saunders, J. (2016). Restaurant sales and customer demand forecasting: Literature survey and categorization of methods. In Smart City 360° (pp. 479-491). Springer, Cham.*
- *Makatjane, K. D., & Moroke, N. D. (2016). Comparative study of holt-winters triples exponential smoothing and seasonal*
- *Arima: Forecasting short term seasonal car sales in South Africa. Risk Governance and Control: Financial Markets and Institutions, 6(1), 71-82.*
- *Massaro, A., Maritati, V., & Galiano, A. (2018). Data Mining model performance of sales predictive algorithms based on RapidMiner workflows. International Journal of Computer Science & Information Technology (IJCSIT), 10(3), 39-56.*
- *Mentzer, J. T., & Moon, M. A. (2004). Sales forecasting management: a demand management approach. Sage Publications.*

Team Everest

Thank You

For Your Attention

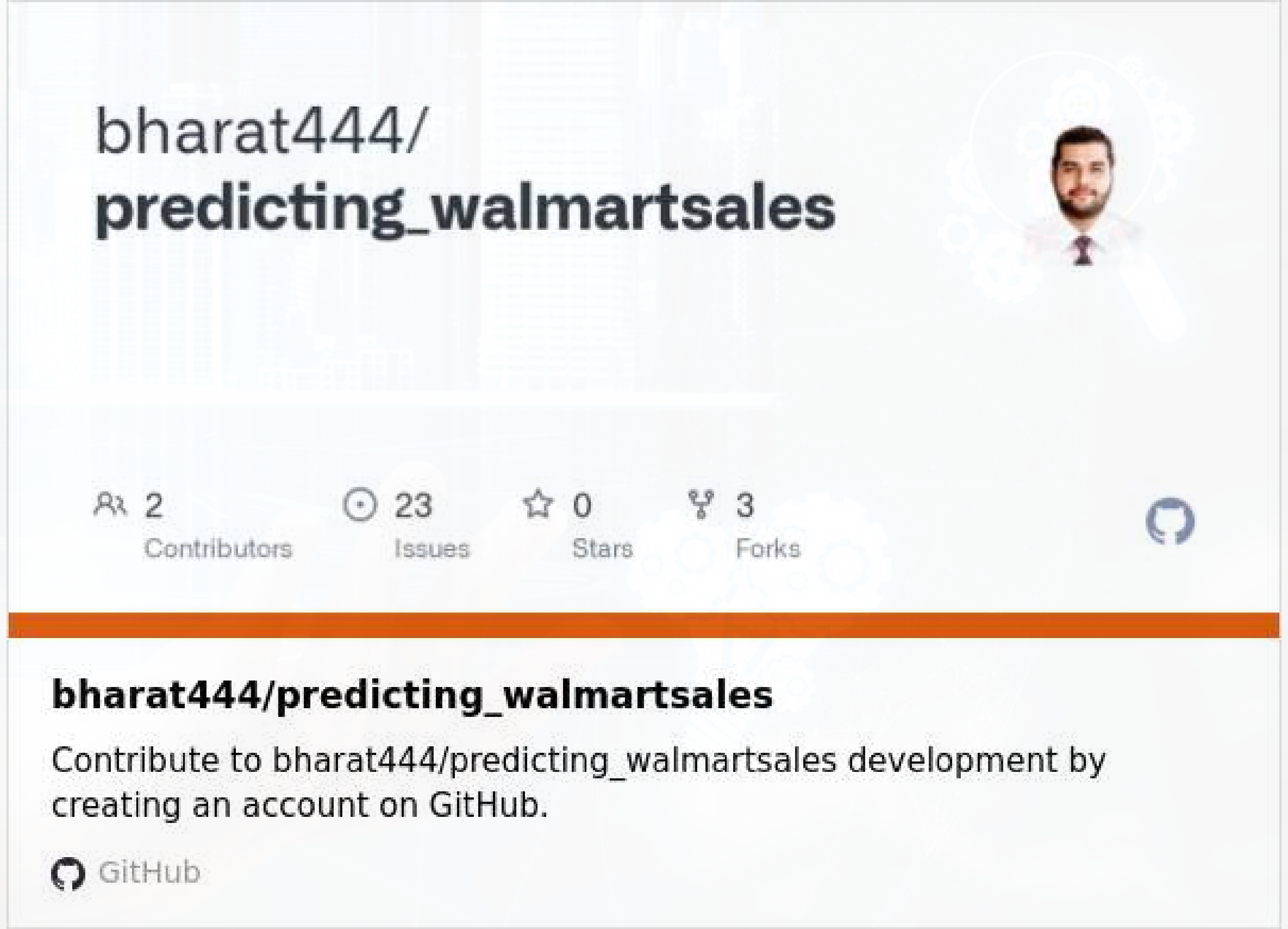
Any Queries?

We are open for discussion.

Access this ppt via: BDM Project Presentation

Appendix

The Link for the Github mentioned here will provide an access to every python files used during this project.



bharat444/
predicting_walmartsales

2 Contributors
23 Issues
0 Stars
3 Forks

bharat444/predicting_walmartsales

Contribute to bharat444/predicting_walmartsales development by creating an account on GitHub.

GitHub