Walmart

Redefining Retail

Team 5 (9:50am Class)
Accidental Analysts

- Case Scenario -

Can Walmart Continue to thrive on cost leadership?

2012
Amazon Acquired Kiva Systems

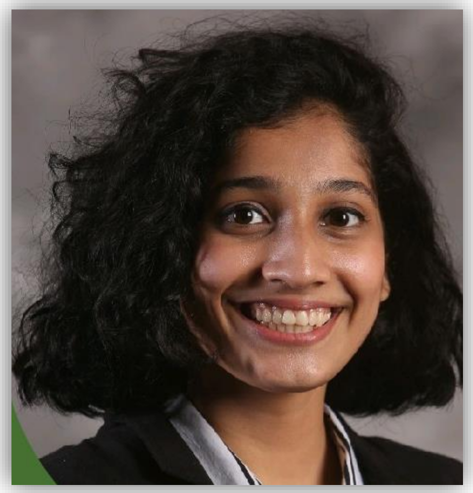
2013
Amazon invests in Drone Tech

2013
Walmart : $3bn Loss estimated due to stock operations issues

2014
Walmart facing threat of staff-strikes

- Team Members -





Accidental Analysts

Walmart - 2013

19.3%

General expense increase in last 3 years

$85

Stock price : Least growth in last 3 years

$467bn

Least growth in last 3 years

- Objective -

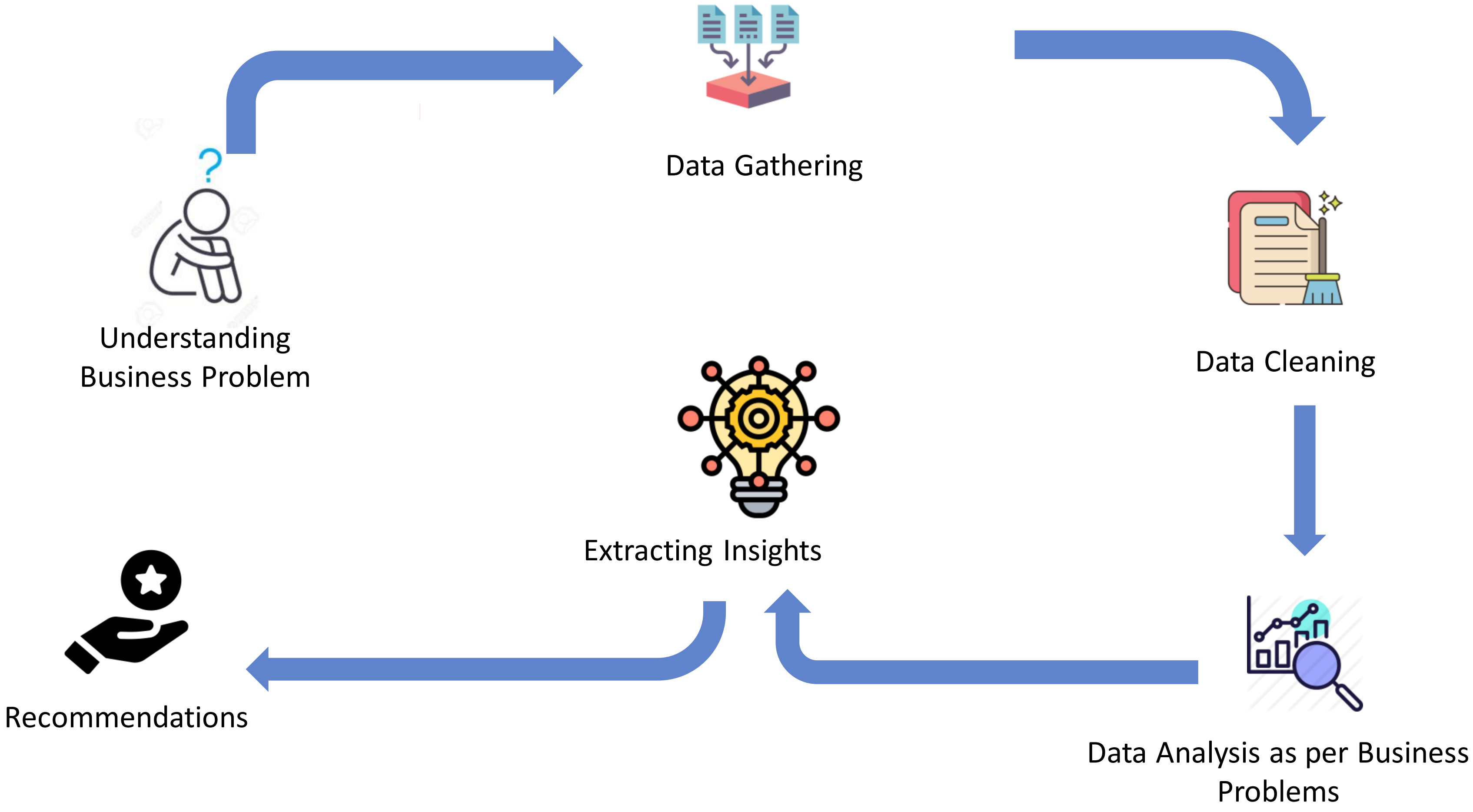
Mission

Prevent Walmart from loosing Cost leadership in Retail segment

Problem Statement

Analyze Walmart data from 45 stores and give recommendation on pricing and growth strategy

Process Flow



DATASET INTRO

Rows : 6436
Columns: 9

Sales Data

- Store ID
- Date – Week
- Temperature – Avg temp in area
- Fuel Price – cost of fuel in area
- Markdown 1-5 – Anonymized data related to promotional markdowns
- CPI – Consumer Price Index
- Unemployment Rate
- ISholiday – Whether a week is a holiday week

Store Opening data

- Store ID
- Opening Date
- Address
- ZipCode
- State
- City

Store Data

- Store ID
- Store Type
- Size

Features

- Markdown 1
- Markdown 2
- Markdown 3
- Markdown 4
- Markdown 5
- Store
- Date

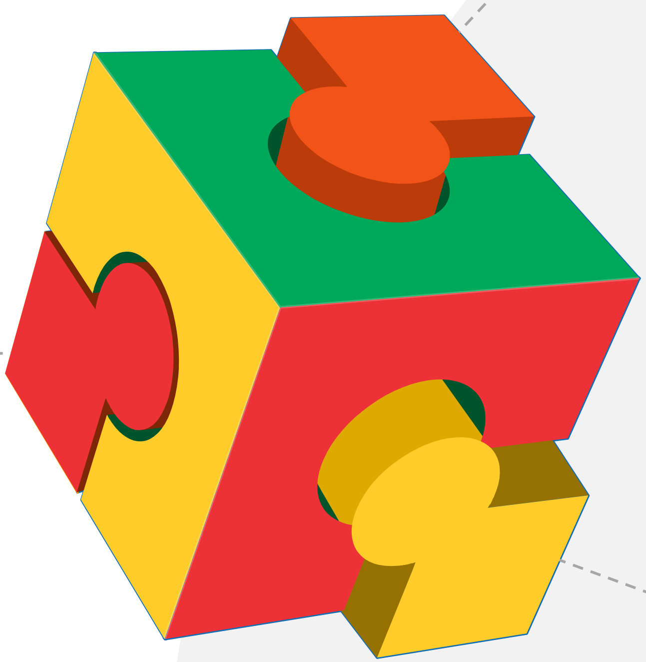
Data Cleaning

Handle Missing Data

All columns – if rows are null – replace with 0

Check for critical rows – IF there is consistent data

Removing Duplicate columns



Merge Tables

All three data tables were Merged into single data frame

Subset data frames created for analysis

Remove unwanted Info

Drop columns from which there is no consistent data

Drop columns where there was there was inaccurate mapping (Structural errors) – Different values for same column

User Stories Tackled

Q1. Store classification Overview

Q2. Impact of Holidays on sales?

Q3. Impact of Markdown on Sales?

Q4. Impact of other external factors like temp, fuel price unemployment rate & CPI on sales?

Background - Store types

TYPE A	TYPE B	TYPE C
Walmart Supercenter	Discount Stores	Neighborhood Markets
Average size - 187,000 square feet	Average size - 107,000 square feet	Average size - 42,000 square feet

Walmart Supercenter

Offer 142,000 different items.
Employ 350 or more associates on average

Offerings

Combining full grocery lines and general merchandise ,specialty shops such as vision centers, hair salons

Discount Stores

Offer around 120,000 items
Employ an average of 225 associates

Offerings

Value-priced general merchandise.

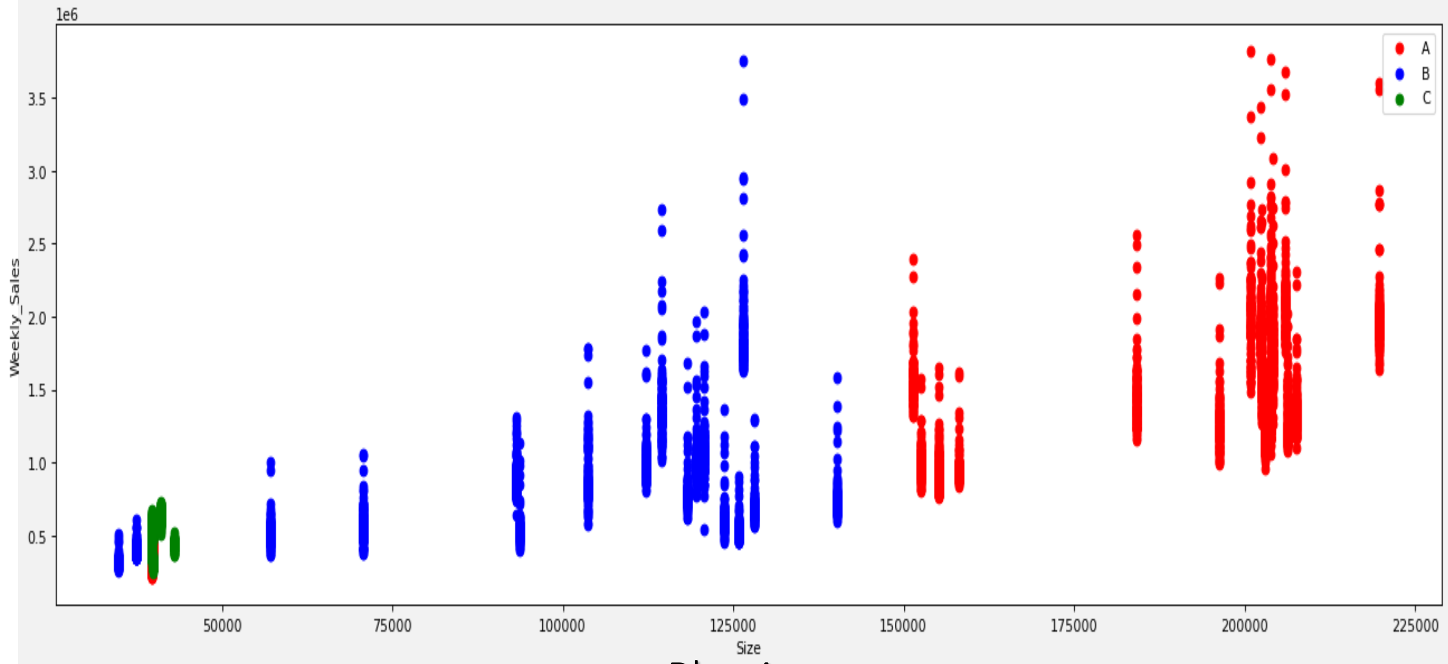
Neighborhood Markets

employ 95 associates on average and offer about 29,000 items

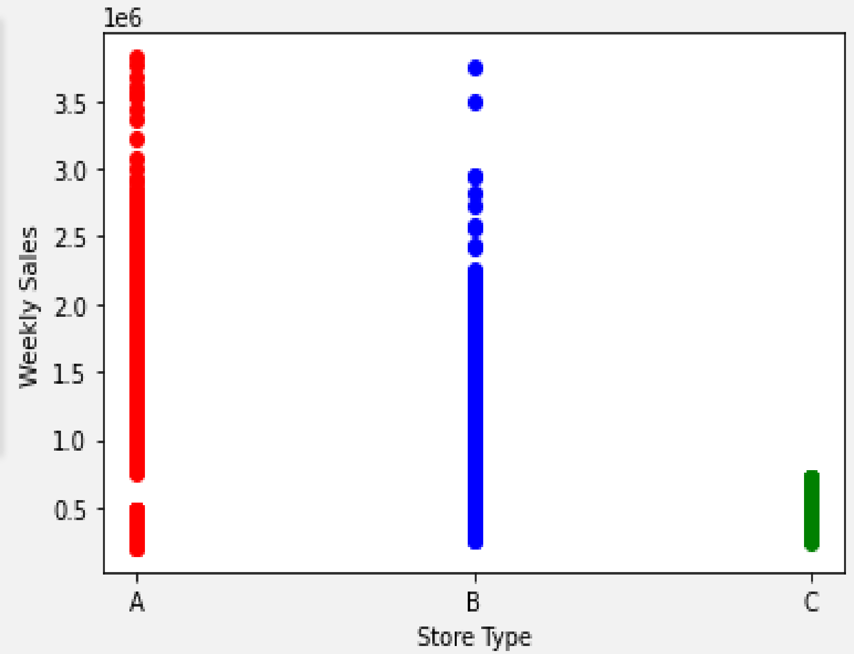
Offerings

Groceries, pharmaceuticals and general merchandise

Store Size/Type Sales Overview



Plot A

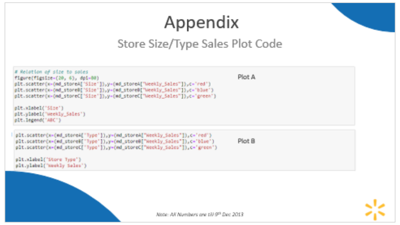


Plot B

Type A stores will have higher sales and higher transactions – due to size

Type B stores will have moderate sales and moderate transactions

Type C stores will have Lower sales volume and Lower transactions



Note: All Numbers are till 9th Jan 2013

Sales Analysis

Sales Trends

Quarterly Performance

State wise Performance

Effect of Holidays

Store type trends

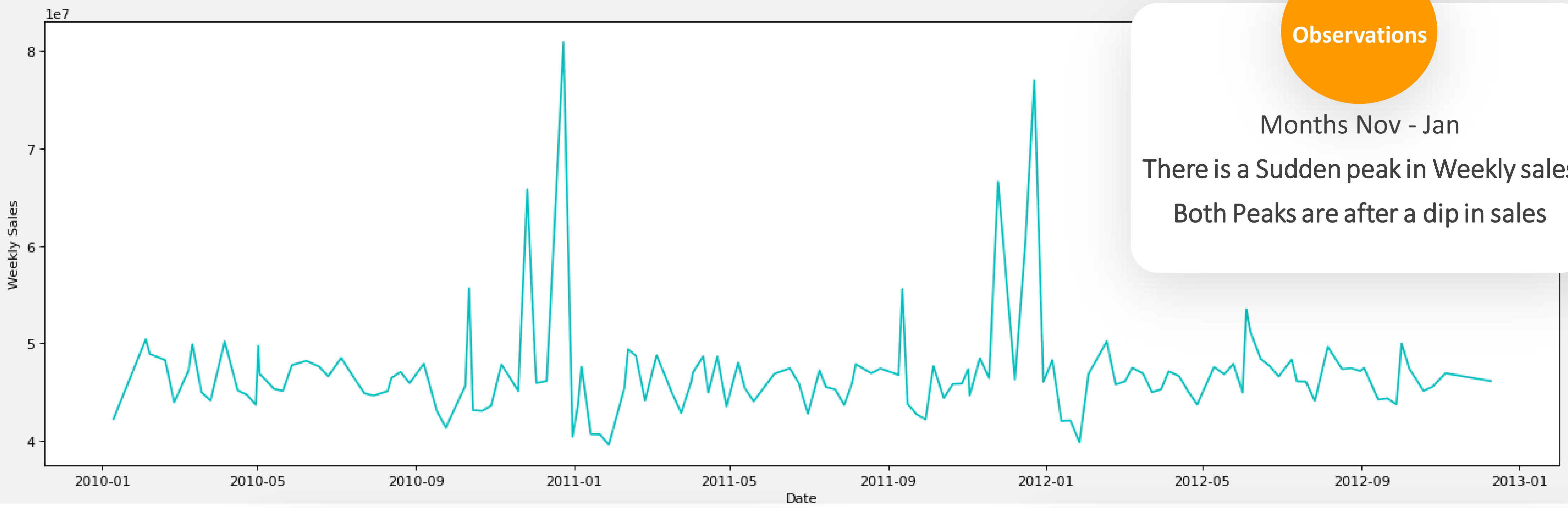
Holiday type trends

4 X

Effect of Markdowns

State wise trends

Sales Overview



Observations

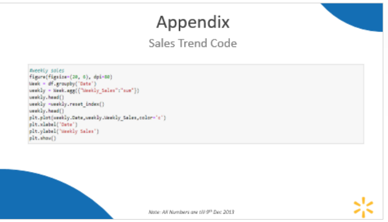
Months Nov - Jan

There is a Sudden peak in Weekly sales

Both Peaks are after a dip in sales

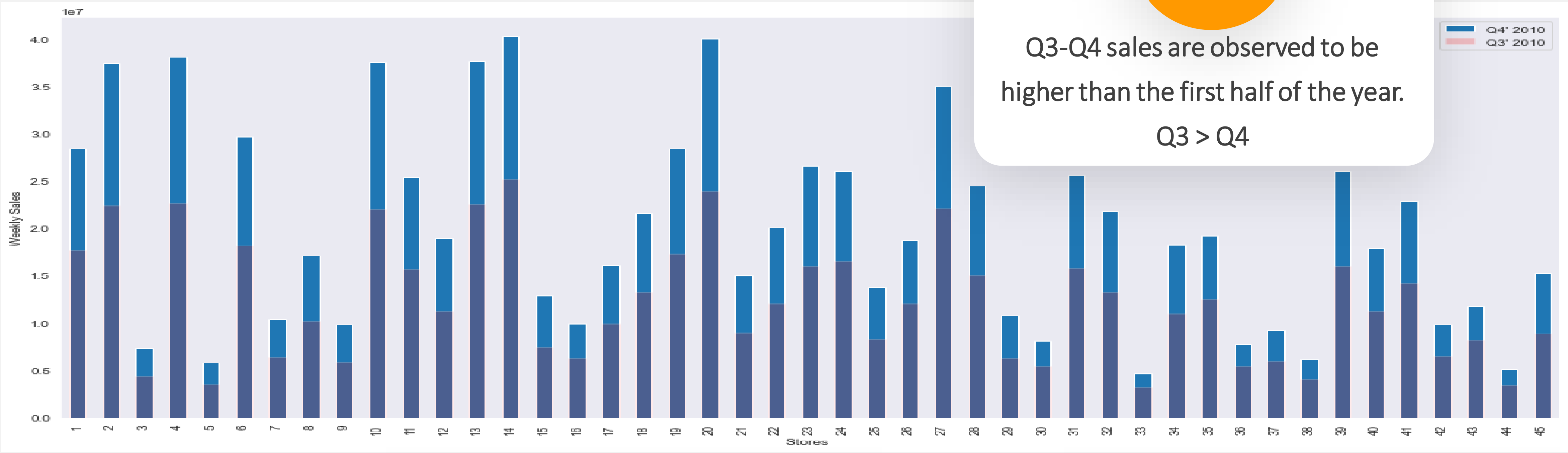
November to January has two holiday seasons – Thanksgiving and Christmas – This could be possible reason for the spike

Sudden Dip before and after the week – Could indicate savings behavior & people usually are spending time with their families



Note: All Numbers are till 9th Jan 2013

Sales Q3 on Q4



Observations

Q3-Q4 sales are observed to be higher than the first half of the year.

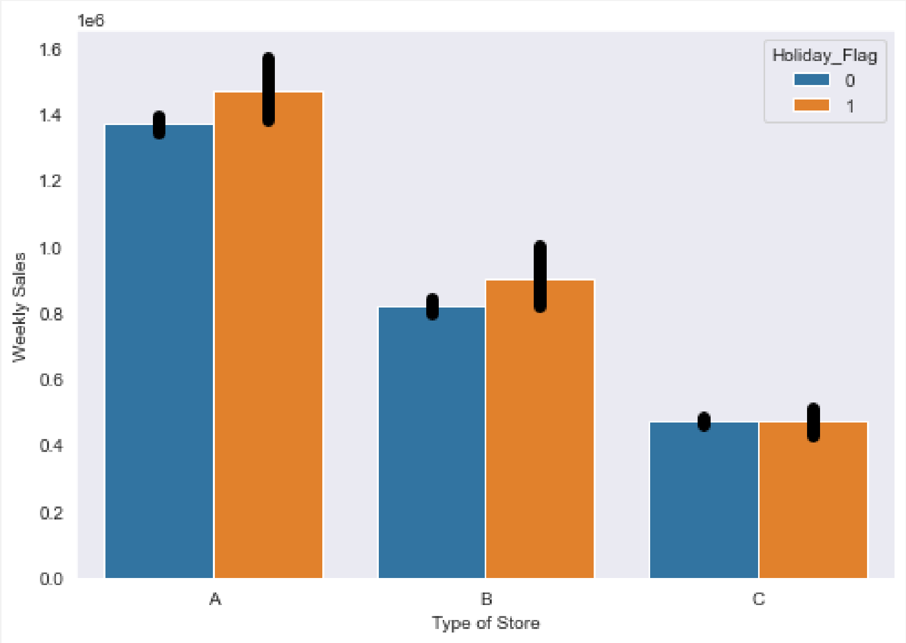
Q3 > Q4

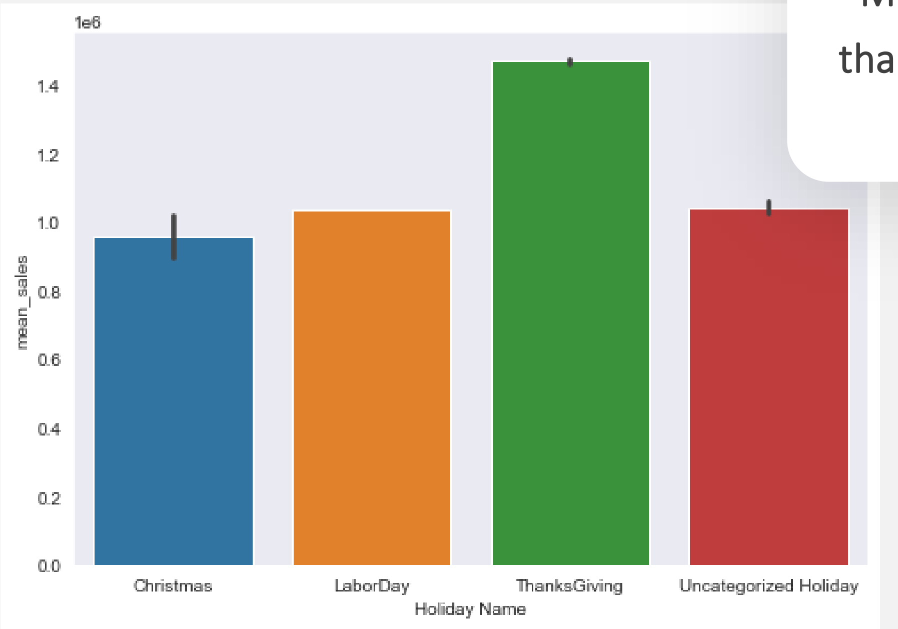
Second half of the year is clubbed with festive seasons and hence has more sales but Q3>Q4 is counterintuitive finding, and we will need more data to understand this.

Walmart should create Gift bundles in second half of the year and create more occasions to buy in first half.
An occasion like 'Last day of Feb' or 'Spring-Fest Sales' on the lines of Black Friday will create the required engagement.

Note: All Numbers are till 9th Jan 2013

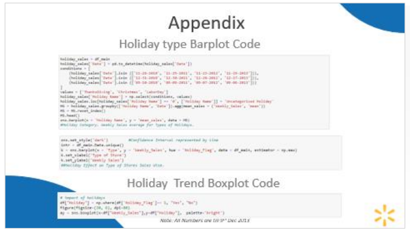
Holidays





Observations

Mean weekly sales is higher during thanksgiving weeks. But type C stores have no impact of Holidays

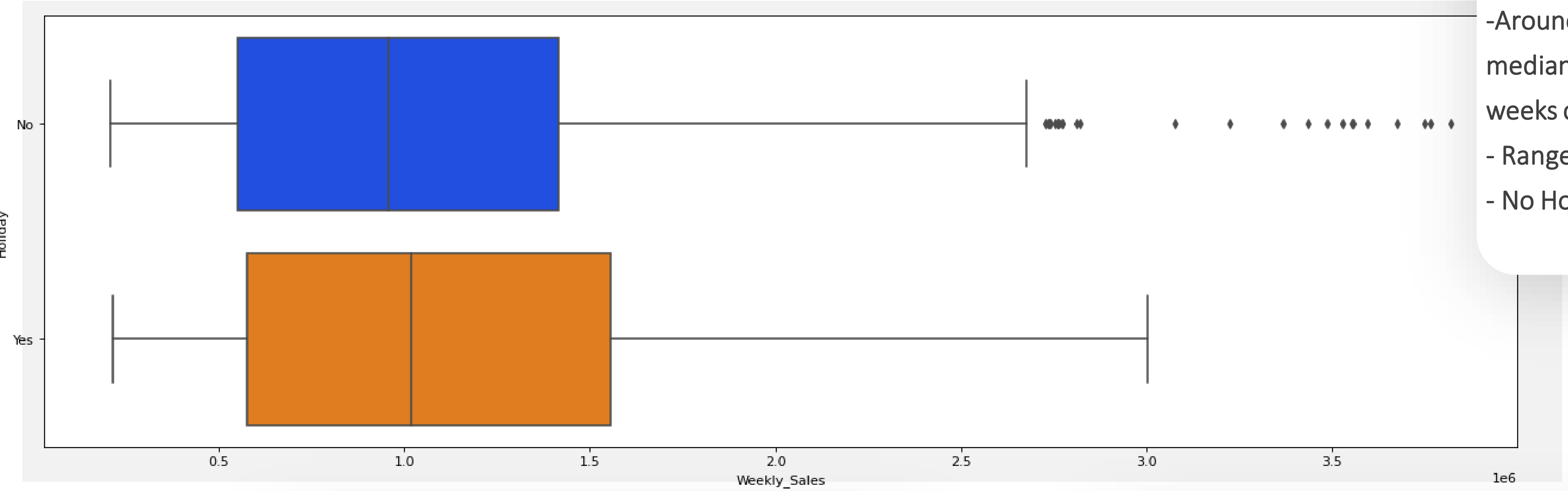


Since Holidays ask for more diverse purchases and require planned visits, neighborhood stores do not see any change in the purchase behaviors.

Higher sales during Thanksgiving can be attributed to – shorter shopping window and Black Friday sales that follows – leading to higher sales volume

Note: All Numbers are till 9^{th} Dec 2012

Holidays



Observations

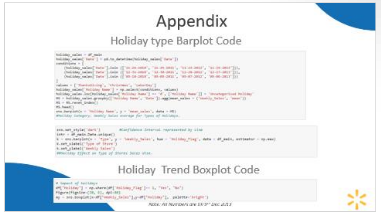
-Around $100K difference in the median(50%) weekly sales during holiday weeks compared normal weeks

- Range of sales is larger in holiday weeks

- No Holiday weeks have more outliers

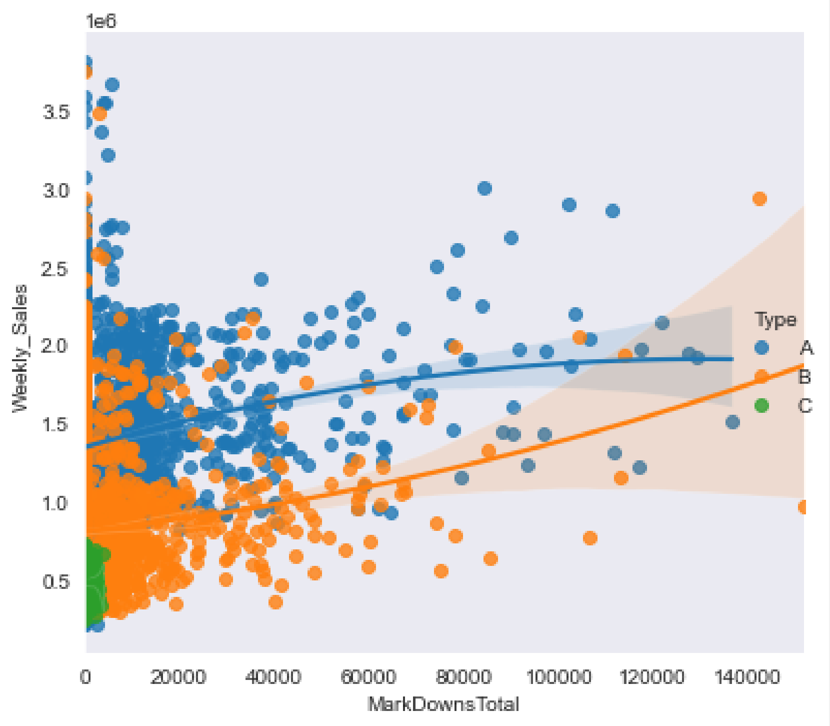
No Holiday weeks – erratic outliers could be due to customers who have lesser frequency and hence bigger market basket size, we need to tap this market.

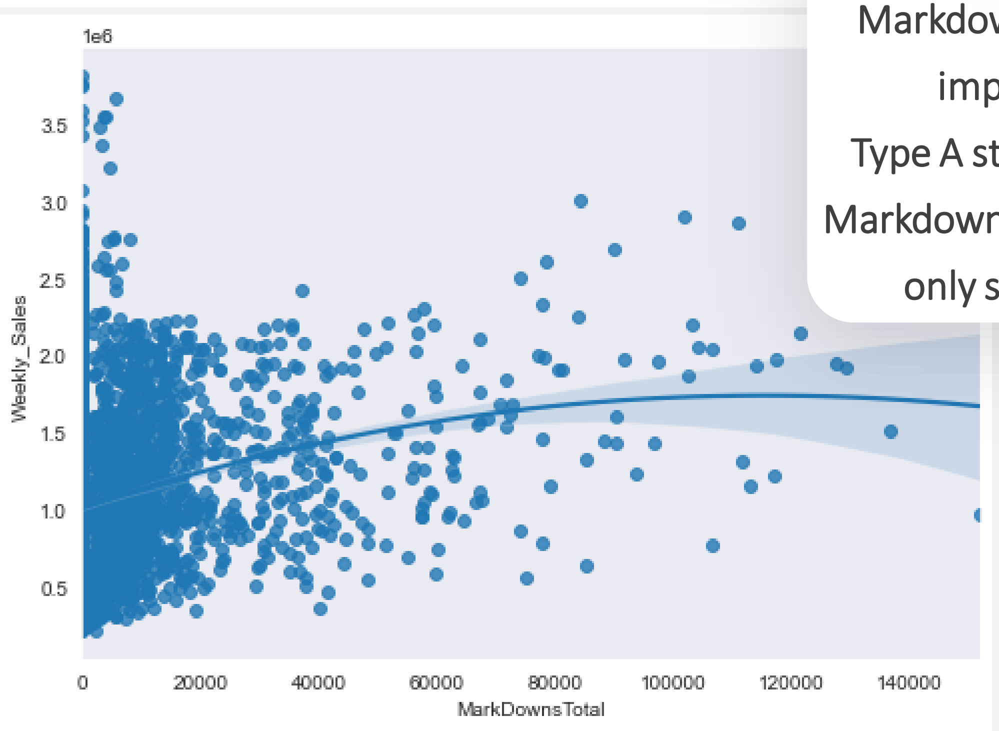
$100 K difference in median can be attributed to added purchase of gifts - increasing the sales volume.



Note: All Numbers are till 9th Dec 2012

Markdown





Observations

Markdown's trend show a positive impact for Type B stores.

Type A store remains relatively flat.

Markdown effect on total weekly sales only sees a moderate impact

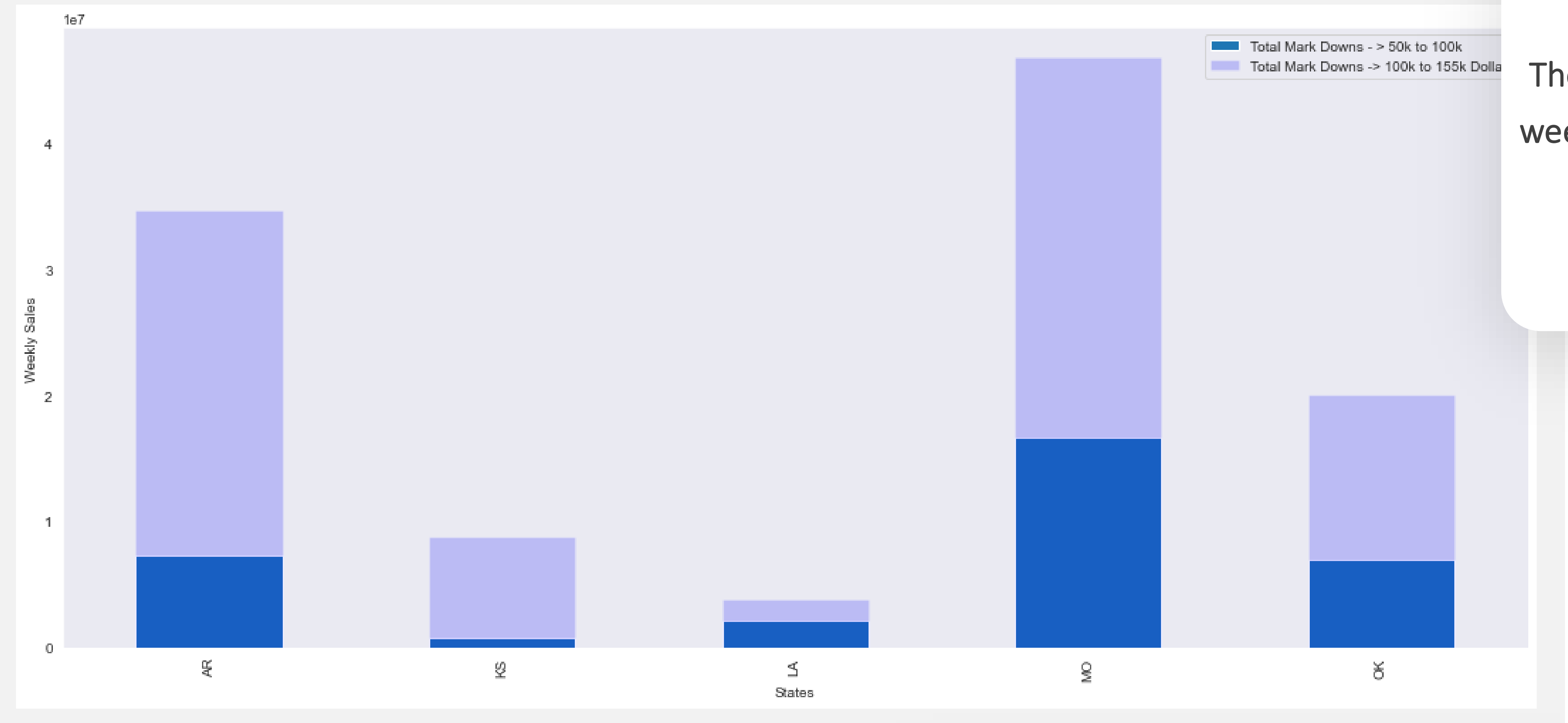
Markdowns are the total amount reduced on the price for a set of slow-moving products –so as to enable their sales

Trend show a positive impact of Markdown on Type B stores. This can be because Type A stores are mainly dependent on bulk planned purchase and type C on quick convenience purchase. Both behaviors are difficult to change

Markdowns show a diminishing returns plot. Hence Mark down should be targeted only on type B stores in Moderate Magnitude to get the best return on investment

Note: All Numbers are till 9th Jan 2013

Markdown



Observations

There is more than 100% increase in weekly sales when markdown value is in range of 100k to 155k

Stores in different states behave differently on markdown- Hence Walmart should have a separate markdown strategy for each store

Note: All Numbers are till 9th Jan 2013

Pricing- Recommendations

01 Better Planned Markdowns For Type B stores

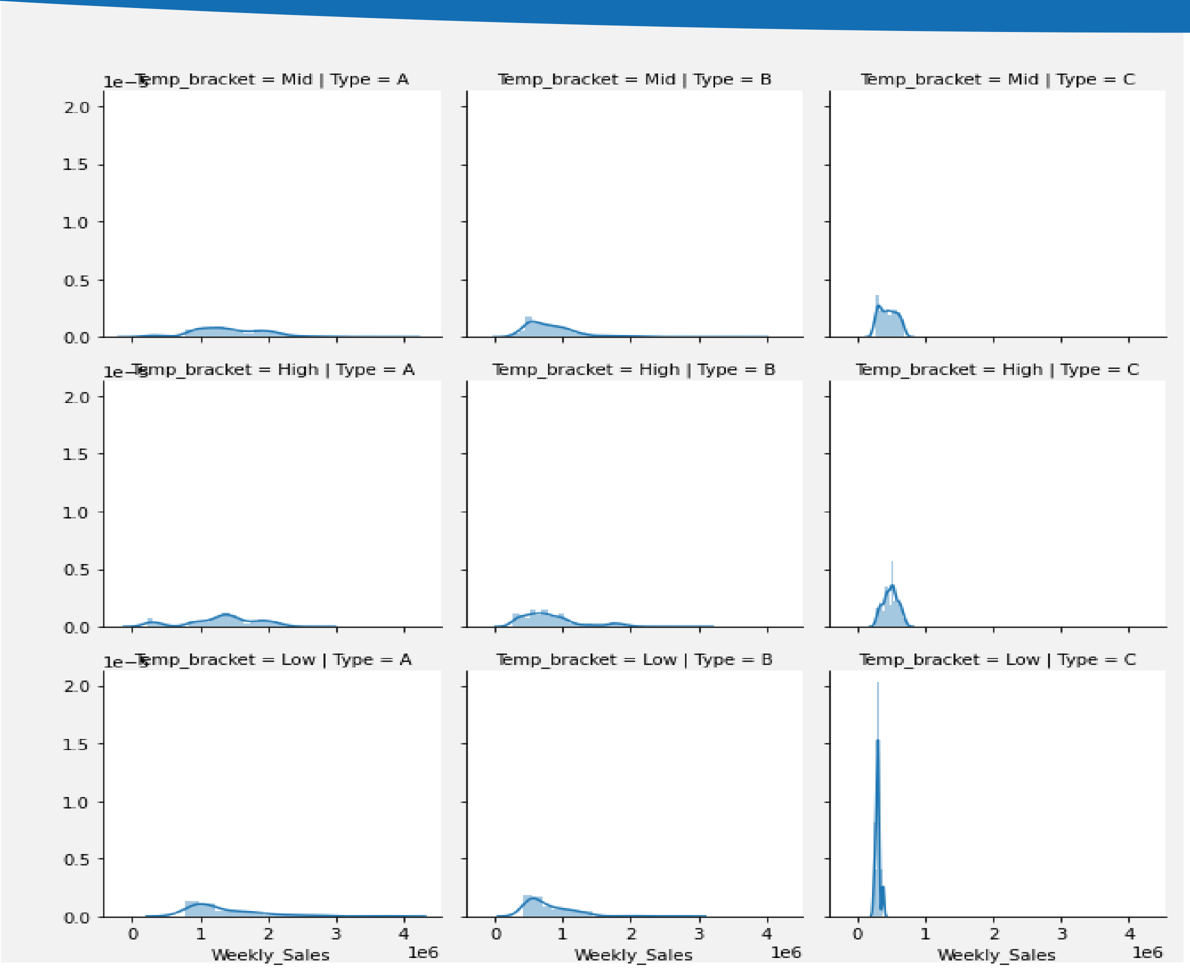
02 Prepare for advance purchasing cycle during Holiday season

03 Offer Gift Bundles in Second Half of the year. Create Sales Occasions in first half of the year

04 During no holiday week, promote bulk buying by providing **free delivery service on orders above certain limit**.
During holiday season, **promote bulk buying by providing store points.**

External Factors Analysis

Temperature

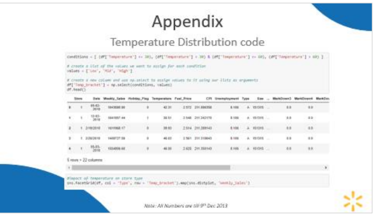


Observations

Type C stores peak when temperature is low

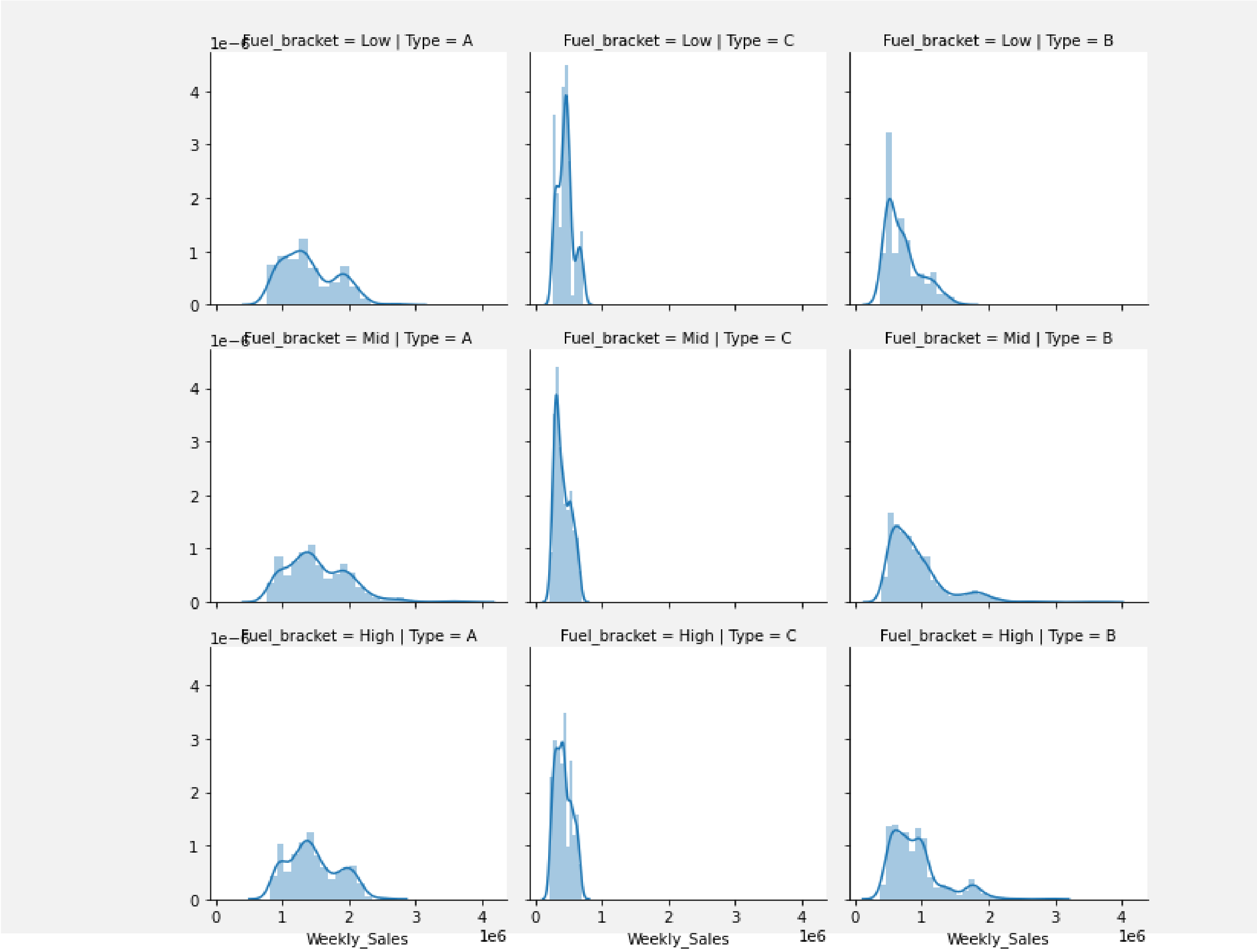
Type A and B don't have much difference

Type C peak can be explained by the convenience factor for consumers during winters.



Note: All Numbers are till 9th Dec 2013

Fuel Price

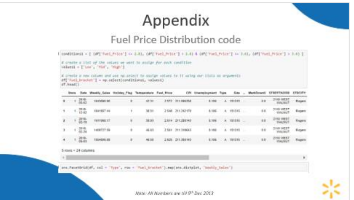


Observations

Type C stores peak when fuel price is low

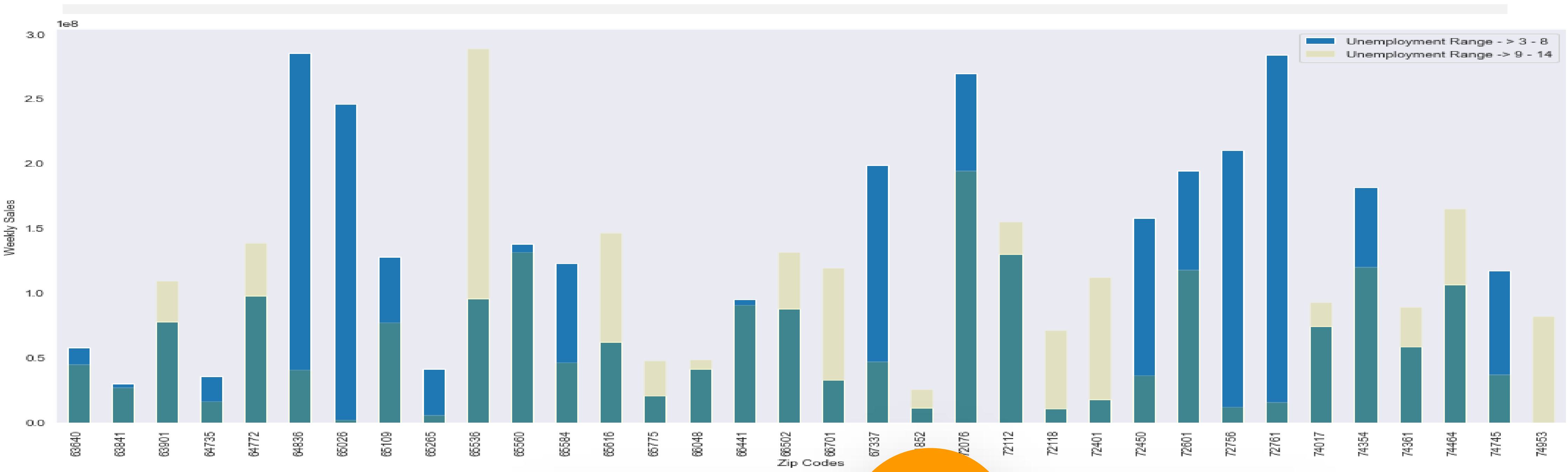
Type A and B don't have much difference

More number of sales in Type C when fuel price is low can indicate that the type C stores might be spread apart –in distance, hence uneconomical to travel if fuel price is high



Note: All Numbers are till 9th Dec 2012

Unemployment



Observations

•There are few zip codes such as 63901 where the Unemployment range has no effect on the Weekly Sales.
•This graph shows how unemployment ranges are affecting weekly sales in various Zip codes.

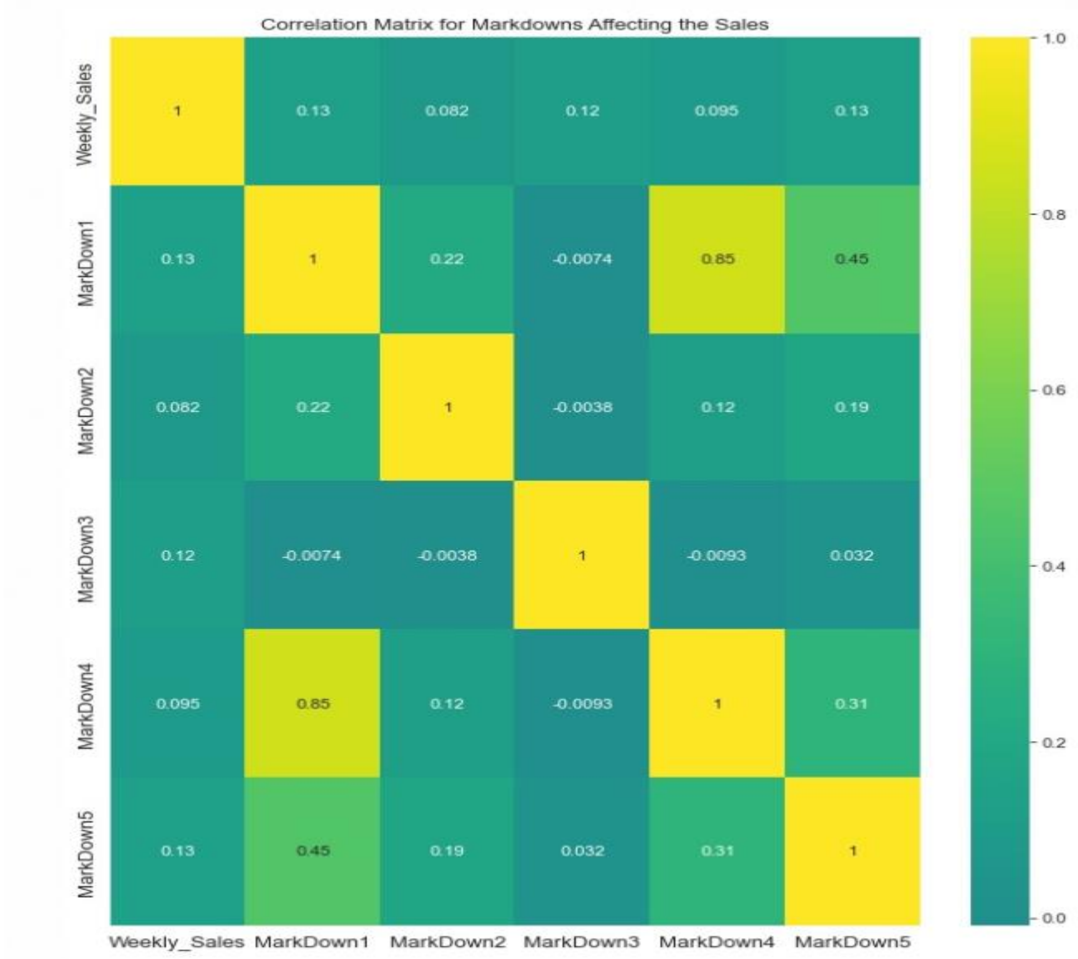
Note: All Numbers are till 9th Dec 2012

Correlation Matrix – Other Factors

Observations

As per the correlation Matrix, we can say that size of store is significantly & positively correlated with Weekly sales. On the other hand, Unemployment shares significant but negative correlation.

Correlation Matrix – Markdown



Observations

As per the correlation Matrix, we can say that Markdown 1 , 3 and 5 are is significantly & positively correlated with Weekly Sales

Note: All Numbers are till 9th Dec 2012

Modelling

OLS Regression Model

```
                            OLS Regression Results
==============================================================================
Dep. Variable:           Weekly_Sales   R-squared:                       0.693
Model:                            OLS   Adj. R-squared:                  0.692
Method:                 Least Squares   F-statistic:                     1810.
Date:                Sat, 02 Oct 2021   Prob (F-statistic):               0.00
Time:                        21:05:00   Log-Likelihood:                -90557.
No. Observations:                6435   AIC:                         1.811e+05
Df Residuals:                    6426   BIC:                         1.812e+05
Df Model:                           8
Covariance Type:            nonrobust
```

LGBM Regression Model

```
LGBMClassifier()
The R squared value of the model is 1.0
```

Random Forest Regression Model Results

```
Mean Absolute Error: 81592.97601195649
Mean Squared Error: 26623448840.89384
Root Mean Squared Error: 163166.93550132588
Variance score: 0.9213834359666249
R^2 score :  0.9213834359666249
```

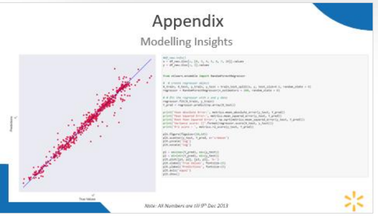
We have run three models from scikitlearn, lightbgm and statsmodels.api

1. Ordinary Least Squares Regression
2. Random Forest
3. Light Gradient Boosted machine from lightgbm library.

The purpose of running three models is to justify which regression works best for our parameters.

The key factor we are looking in these regression models is the R-square values and how it has increased as we have used different models.

Our Analysis has been based on the Test and the Predicted Data for Weekly Sales based on various factors



Note: All Numbers are till 9th Dec 2012

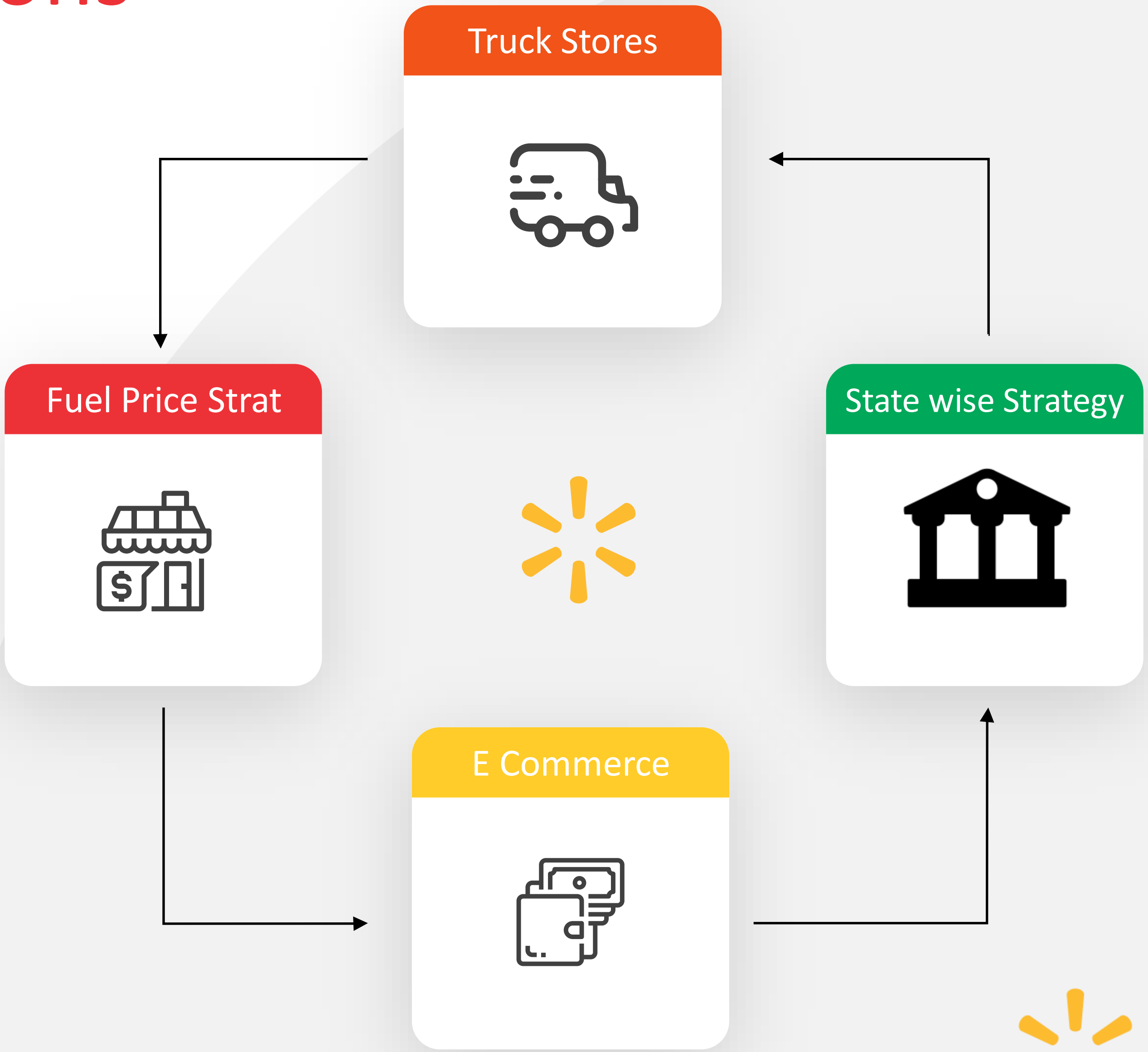
Strategy - Recommendations

01 Open smaller truck store for these neighborhoods during winter season.

02 In Areas and States with higher fuel price, Walmart should open more smaller neighborhood stores

03 Move towards ecommerce

04 Separate Marketing Strategies for states with different CPI & Unemployment



Appendix

Data Consistency check

```
for index, rows in df1.iterrows():
    if(rows['Fuel_Price'] == '' or 0):
        print(True)
    else:
        exit
print('Fuel Price Column has consistent Data')
```

```
Fuel Price Column has consistent Data
```

```
for index, rows in df1.iterrows():
    if(rows['Holiday_Flag'] != 0 or rows['Holiday_Flag'] != 1):
        exit
    else:
        print(True)
print('Holiday Flag Column has consistent Data')
```

```
Holiday Flag Column has consistent Data
```

```
def check_range(df1, x, y):
    for index, rows in df1.iterrows():
        if(x <= rows['Store'] <= y):
            exit
        else:
            print(True)
    print('Store Column has consistent Data')

x = 1
y = 45
check_range(df1, x, y)
```

```
Store Column has consistent Data
```

Null values check

```
df_ma['MarkDown1'] = df_ma['MarkDown1'].replace(np.nan, 0)
```

```
df_ma.head()
```

	Unnamed: 0	Store	Date	Weekly_Sales	Holiday_Flag	Temperature	Fuel_Price	CPI	Unemployment	Type	Size	MarkDown1	MarkDown2	Mar
0	0	1	2010-05-02	1643690.90	0	42.31	2.572	211.096358	8.106	A	151315	0.0	NaN	
1	1	1	2010-12-02	1641957.44	1	38.51	2.548	211.242170	8.106	A	151315	0.0	NaN	
2	2	1	2010-02-19	1611968.17	0	39.93	2.514	211.289143	8.106	A	151315	0.0	NaN	
3	3	1	2010-02-26	1409727.59	0	46.63	2.561	211.319643	8.106	A	151315	0.0	NaN	
4	4	1	2010-05-03	1554806.68	0	46.50	2.625	211.350143	8.106	A	151315	0.0	NaN	

Appendix

Data frames Merging

```
pd.merge(df1, df2, left_on = 'Store', right_on = 'Store', how = 'left')
```

	Store	Date	Weekly_Sales	Holiday_Flag	Temperature	Fuel_Price	CPI	Unemployment	Type	Size
0	1	2010-05-02	1643690.90	0	42.31	2.572	211.096358	8.106	A	151315
1	1	2010-12-02	1641957.44	1	38.51	2.548	211.242170	8.106	A	151315
2	1	2010-02-19	1611968.17	0	39.93	2.514	211.289143	8.106	A	151315
3	1	2010-02-26	1409727.59	0	46.63	2.561	211.319643	8.106	A	151315
4	1	2010-05-03	1554806.68	0	46.50	2.625	211.350143	8.106	A	151315
...	...	...	...	...	...	...	...	...	...	...
6430	45	2012-09-28	713173.95	0	64.88	3.997	192.013558	8.684	B	118221
6431	45	2012-05-10	733455.07	0	64.89	3.985	192.170412	8.667	B	118221
6432	45	2012-12-10	734464.36	0	54.47	4.000	192.327265	8.667	B	118221
6433	45	2012-10-19	718125.53	0	56.47	3.969	192.330854	8.667	B	118221
6434	45	2012-10-26	760281.43	0	58.85	3.882	192.308899	8.667	B	118221

6435 rows × 10 columns

```
df_main = pd.merge(df1, df2, left_on = 'Store', right_on = 'Store', how = 'left')
```

```
df_main.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 6435 entries, 0 to 6434
Data columns (total 10 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Store         6435 non-null   int64
 1   Date          6435 non-null   datetime64[ns]
 2   Weekly_Sales  6435 non-null   float64
 3   Holiday_Flag  6435 non-null   int64
 4   Temperature   6435 non-null   float64
 5   Fuel_Price    6435 non-null   float64
 6   CPI           6435 non-null   float64
 7   Unemployment  6435 non-null   float64
 8   Type          6435 non-null   object
 9   Size          6435 non-null   int64
dtypes: datetime64[ns](1), float64(5), int64(3), object(1)
memory usage: 553.0+ KB
```

```
Store_df2 = Store_df.drop(['date_super','conversion','type_store'],axis=1)
Store_df2.head()
```

	storenum	OPENDATE	st	county	STREETADDR	STRCITY	STRSTATE	ZIPCODE
0	1	7/1/1962	5	7	2110 WEST WALNUT	Rogers	AR	72756.0
1	2	8/1/1964	5	9	1417 HWY 62/65 N	Harrison	AR	72601.0
2	3	4/12/1988	13	11	30983 HWY 441 SOUTH	Commerce	GA	30529.0
3	4	8/1/1965	5	7	2901 HWY 412 EAST	Siloam Springs	AR	72761.0
4	5	5/1/1972	5	45	1155 HWY 65 NORTH	Conway	AR	72032.0

```
cols_to_use = Store_df2.columns.difference(df.columns)
pd.merge(df, Store_df2[cols_to_use], left_on = 'Store', right_on = 'storenum', how = 'left')
```

Note: All Numbers are till 9th Dec 2013

Appendix

Store Size/Type Sales Plot Code

```
# Relation of size to sales
figure(figsize=(20, 6), dpi=80)
plt.scatter(x=(md_storeA['Size']),y=(md_storeA["Weekly_Sales"]),c='red')
plt.scatter(x=(md_storeB['Size']),y=(md_storeB["Weekly_Sales"]),c='blue')
plt.scatter(x=(md_storeC['Size']),y=(md_storeC["Weekly_Sales"]),c='green')

plt.xlabel('Size')
plt.ylabel('Weekly_Sales')
plt.legend('ABC')
```

Plot A

```
plt.scatter(x=(md_storeA['Type']),y=(md_storeA["Weekly_Sales"]),c='red')
plt.scatter(x=(md_storeB['Type']),y=(md_storeB["Weekly_Sales"]),c='blue')
plt.scatter(x=(md_storeC['Type']),y=(md_storeC["Weekly_Sales"]),c='green')

plt.xlabel('Store Type')
plt.ylabel('Weekly Sales')
```

Plot B

Note: All Numbers are till 9th Dec 2013

Appendix

Sales Trend Code

```
#weekly sales
figure(figsize=(20, 6), dpi=80)
Week = df.groupby('Date')
weekly = Week.agg({"Weekly_Sales":"sum"})
weekly.head()
weekly =weekly.reset_index()
weekly.head()
plt.plot(weekly.Date,weekly.Weekly_Sales,color='c')
plt.xlabel('Date')
plt.ylabel('Weekly Sales')
plt.show()
```

Note: All Numbers are till 9th Dec 2013

Appendix

Sales Q3 on Q4 barplot Code

```
q3_sales = df_new[(df_new['Date'] >= '2010-07-01') & (df_new['Date'] <= '2010-09-30')].groupby('Store')['Weekly_Sales'].sum()
q4_sales= df_new[(df_new['Date'] >= '2010-09-01') & (df_new['Date'] <= '2010-12-31')].groupby('Store')['Weekly_Sales'].sum()

# Plotting the difference between sales for Third and Fourth
plt.figure(figsize=(15,7))
q3_sales.plot(ax=q4_sales.plot(kind ='bar'),kind='bar',color='r',alpha=0.2,legend=True)
plt.legend(["Q4' 2010", "Q3' 2010"])
plt.ylabel('Weekly Sales')
plt.xlabel('Stores ')
```

Note: All Numbers are till 9th Dec 2013

Appendix

Holiday type Barplot Code

```
holiday_sales = df_main
holiday_sales['Date'] = pd.to_datetime(holiday_sales['Date'])
conditions = [
    (holiday_sales['Date'].isin (['11-26-2010', '11-25-2011', '11-23-2012', '11-29-2013'])),
    (holiday_sales['Date'].isin (['12-31-2010', '12-30-2011', '12-28-2012', '12-27-2013'])),
    (holiday_sales['Date'].isin (['09-10-2010', '09-09-2011', '09-07-2012', '09-06-2013']))
]
values = ['ThanksGiving', 'Christmas', 'LaborDay']
holiday_sales['Holiday Name'] = np.select(conditions, values)
holiday_sales.loc[holiday_sales['Holiday Name'] == '0', ['Holiday Name']] = 'Uncategorized Holiday'
HS = holiday_sales.groupby(['Holiday Name', 'Date']).agg(mean_sales = ('Weekly_Sales', 'mean'))
HS = HS.reset_index()
HS.head()
sns.barplot(x = 'Holiday Name', y = 'mean_sales', data = HS)
#Holiday Category. Weekly Sales average for Types of Holidays.
```

```
sns.set_style('dark')          #Confidence Interval represented by Line
intr = df_main.Date.unique()
k = sns.barplot(x = 'Type', y = 'Weekly_Sales', hue = 'Holiday_Flag', data = df_main, estimator = np.max)
k.set_xlabel('Type of Store')
k.set_ylabel('Weekly Sales')
##Holiday Effect on Type of Stores Sales Wise.
```

Holiday Trend Boxplot Code

```
# impact of holidays
df["Holiday"] = np.where(df['Holiday_Flag']== 1, "Yes", "No")
figure(figsize=(20, 6), dpi=80)
ay = sns.boxplot(x=df["Weekly_Sales"],y=df["Holiday"],  palette='bright')
```

Note: All Numbers are till 9th Dec 2013

Appendix

Markdown by Store type trend Regression plot code

```
df_new['MarkDownsTotal'] = df_new.iloc[:, 10:14].sum(axis=1)
plt.figure()
sns.lmplot(y = 'Weekly_Sales', x = 'MarkDownsTotal', data = df_new, hue = 'Type', order = 2, ci = 95)
plt.show()
#Regression plot for Markdown Total values
```

Total Markdown trend Regression plot code

```
df_new['MarkDownsTotal'] = df_new.iloc[:, 10:14].sum(axis=1)
plt.figure()
sns.regplot(y = 'Weekly_Sales', x = 'MarkDownsTotal', data = df_new, order = 2, ci = 95)
plt.show()
```

Total Markdown trend Bar plot Code

```
m1_sales = df_new[(df_new['MarkDownsTotal'] >= 0 ) & (df_new['MarkDownsTotal'] <= 50000)].groupby('STRSTATE')['Weekly_Sales']
m2_sales = df_new[(df_new['MarkDownsTotal'] > 50000 ) & (df_new['MarkDownsTotal'] <= 100000)].groupby('STRSTATE')['Weekly_Sal
m3_sales = df_new[(df_new['MarkDownsTotal'] > 100000 ) & (df_new['MarkDownsTotal'] <= 155000)].groupby('STRSTATE')['Weekly_Sa

plt.figure(figsize=(15,7))
m2_sales.plot(ax = m3_sales.plot(kind='bar'), kind = 'bar', color='b',alpha=0.2,legend=True)
plt.legend(["Total Mark Downs - > 50k to 100k", "Total Mark Downs -> 100k to 155k Dollars"])
plt.ylabel('Weekly Sales')
plt.xlabel('States')
plt.show()
```

Note: All Numbers are till 9th Dec 2013

Appendix

Temperature Distribution code

```
conditions = [ (df['Temperature'] <= 30), (df['Temperature'] > 30) & (df['Temperature'] <= 60), (df['Temperature'] > 60) ]

# create a list of the values we want to assign for each condition
values = ['Low', 'Mid', 'High']

# create a new column and use np.select to assign values to it using our lists as arguments
df['Temp_bracket'] = np.select(conditions, values)
df.head()
```

	Store	Date	Weekly_Sales	Holiday_Flag	Temperature	Fuel_Price	CPI	Unemployment	Type	Size	...	MarkDown3	MarkDown4	MarkDov
0	1	05-02-2010	1643690.90	0	42.31	2.572	211.096358	8.106	A	151315	...	0.0	0.0	
1	1	12-02-2010	1641957.44	1	38.51	2.548	211.242170	8.106	A	151315	...	0.0	0.0	
2	1	2/19/2010	1611968.17	0	39.93	2.514	211.289143	8.106	A	151315	...	0.0	0.0	
3	1	2/26/2010	1409727.59	0	46.63	2.561	211.319643	8.106	A	151315	...	0.0	0.0	
4	1	05-03-2010	1554806.68	0	46.50	2.625	211.350143	8.106	A	151315	...	0.0	0.0	

5 rows × 22 columns

```
#impact of temperature on store type
sns.FacetGrid(df, col = 'Type', row = 'Temp_bracket').map(sns.distplot, 'Weekly_Sales')
```

Note: All Numbers are till 9th Dec 2013

Appendix

Fuel Price Distribution code

```
conditions1 = [ (df['Fuel_Price'] <= 2.8), (df['Fuel_Price'] > 2.8) & (df['Fuel_Price'] <= 3.6), (df['Fuel_Price'] > 3.6) ]

# create a list of the values we want to assign for each condition
values1 = ['Low', 'Mid', 'High']

# create a new column and use np.select to assign values to it using our lists as arguments
df['Fuel_bracket'] = np.select(conditions1, values1)
df.head()
```

	Store	Date	Weekly_Sales	Holiday_Flag	Temperature	Fuel_Price	CPI	Unemployment	Type	Size	...	MarkDown5	STREETADDR	STRCITY
0	1	2010-05-02	1643690.90	0	42.31	2.572	211.096358	8.106	A	151315	...	0.0	2110 WEST WALNUT	Rogers
1	1	2010-12-02	1641957.44	1	38.51	2.548	211.242170	8.106	A	151315	...	0.0	2110 WEST WALNUT	Rogers
2	1	2010-02-19	1611968.17	0	39.93	2.514	211.289143	8.106	A	151315	...	0.0	2110 WEST WALNUT	Rogers
3	1	2010-02-26	1409727.59	0	46.63	2.561	211.319643	8.106	A	151315	...	0.0	2110 WEST WALNUT	Rogers
4	1	2010-05-03	1554806.68	0	46.50	2.625	211.350143	8.106	A	151315	...	0.0	2110 WEST WALNUT	Rogers

5 rows × 24 columns

```
sns.FacetGrid(df, col = 'Type', row = 'Fuel_bracket').map(sns.distplot, 'Weekly_Sales')
```

Note: All Numbers are till 9th Dec 2013

Appendix

Unemployment Bar plot code

```
u1_sales = df_new[(df_new['Unemployment'] >= 3) & (df_new['Unemployment'] <= 8)].groupby('ZIPCODE')['Weekly_Sales'].sum()
u2_sales= df_new[(df_new['Unemployment'] > 8) & (df_new['Unemployment'] <= 14)].groupby('ZIPCODE')['Weekly_Sales'].sum()

# Plotting the difference between sales for Third and Fourth
plt.figure(figsize=(15,7))
u2_sales.plot(ax=u1_sales.plot(kind ='bar'),kind='bar',color='y',alpha=0.2,legend=True)
plt.legend(["Unemployment Range - > 3 - 8", "Unemployment Range -> 9 - 14"])
plt.ylabel('Weekly Sales')
plt.xlabel('Zip Codes')
plt.show()
```

Note: All Numbers are till 9th Dec 2013

Appendix

Correlation Matrix code

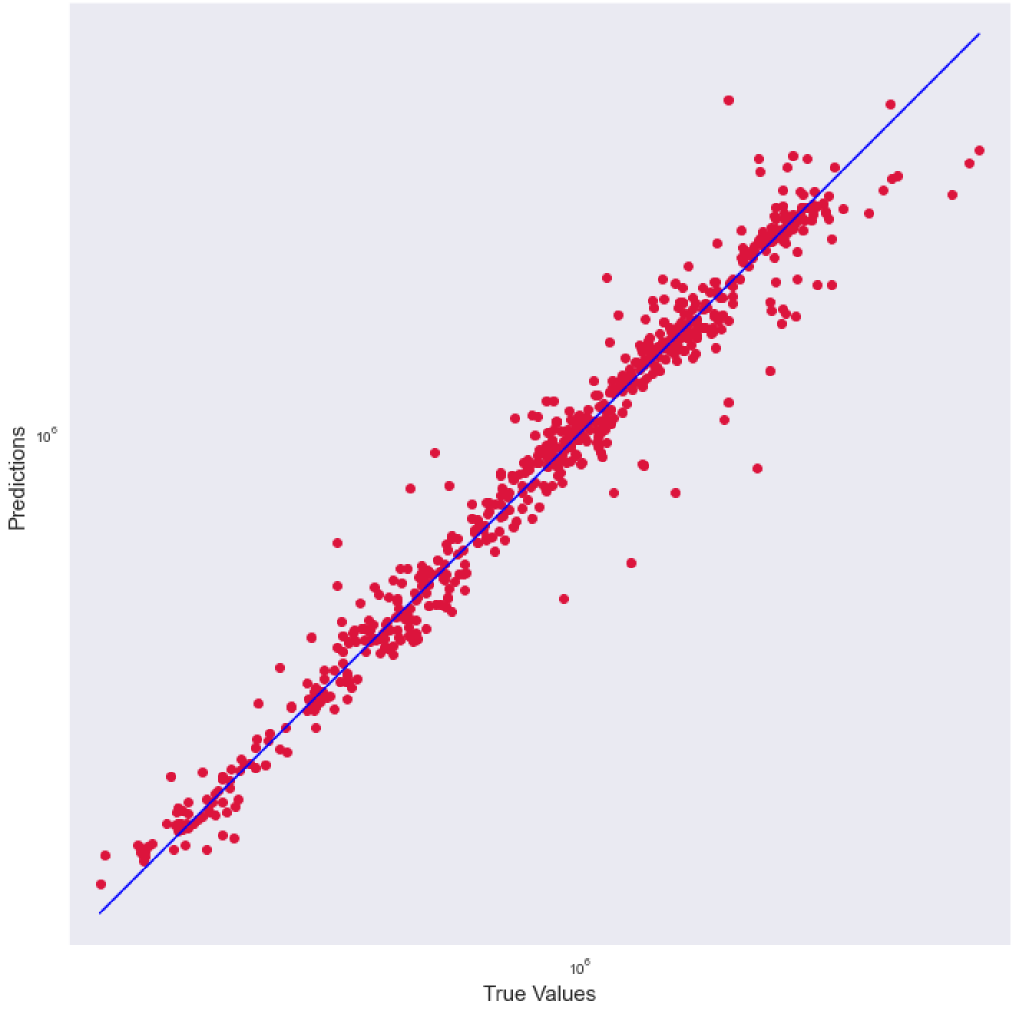
```
select_columns = df_main[['Weekly_Sales', 'Temperature', 'CPI', 'Unemployment', 'Size', 'Fuel_Price']]
new_df = select_columns.copy()
plt.figure(figsize=(10,12), dpi= 80)

sns.heatmap(new_df.corr(), xticklabels=new_df.corr().columns, yticklabels=new_df.corr().columns, cmap='YlGnBu',
            center=0, annot=True, )
plt.title('Correlation Matrix for Parameters Affecting the Sales', fontsize=12)
plt.xticks(fontsize=12)
plt.yticks(fontsize=12)
plt.show()
```

```
#df_main.info()
allMarkDowns = df_main[['Weekly_Sales', 'Store', 'MarkDown1', 'MarkDown2', 'MarkDown3', 'MarkDown4', 'MarkDown5']]
df_markDW = allMarkDowns.copy()
plt.figure(figsize=(10,12), dpi= 80)
sns.heatmap(df_markDW.corr(), xticklabels=df_markDW.corr().columns, yticklabels=df_markDW.corr().columns, cmap='viridis',
            center=0, annot=True)
plt.title('Correlation Matrix for Markdowns Affecting the Sales', fontsize=12)
plt.xticks(fontsize=12)
plt.yticks(fontsize=12)
plt.show()
```

Note: All Numbers are till 9th Dec 2013

Appendix

Modelling Insights




```
#df_new.info()
x = df_new.iloc[:, [0, 3, 4, 5, 6, 7, 20]].values
y = df_new.iloc[:, 2].values


from sklearn.ensemble import RandomForestRegressor

#  # create regressor object
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.1, random_state = 0)
regressor = RandomForestRegressor(n_estimators = 100, random_state = 0)

# # fit the regressor with x and y data
regressor.fit(X_train, y_train)
Y_pred = regressor.predict(np.array(X_test))

print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, Y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(y_test, Y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test, Y_pred)))
print('Variance score: {}'.format(regressor.score(X_test, y_test)))
print('R^2 score : ', metrics.r2_score(y_test, Y_pred))

plt.figure(figsize=(10,10))
plt.scatter(y_test, Y_pred, c='crimson')
plt.yscale('log')
plt.xscale('log')

p1 = max(max(Y_pred), max(y_test))
p2 = min(min(Y_pred), min(y_test))
plt.plot([p1, p2], [p1, p2], 'b-')
plt.xlabel('True Values', fontsize=15)
plt.ylabel('Predictions', fontsize=15)
plt.axis('equal')
plt.show()
```

Note: All Numbers are till 9th Dec 2013