August 2022

TEXAS
The University of Texas at Austin

Time Series Forecasting – Sales

Meet the Group

Aashi Aashi

Saurabh Arora

Sharon Liu

Anthony Moreno

TEXAS
The University of Texas at Austin

AGENDA FOR THE DAY

Topics to be discussed

- Problem Statement
- Dataset Summary and Description
- Exploratory Data Analysis
- Hypothesis Testing
- Time Series and Pre-Modelling Tests
- Forecasting

TEXAS
The University of Texas at Austin

PROBLEM STATEMENT

PROBLEM STATEMENT | Given the transaction record for a store, use time series forecasting to predict the total sales for a specified date.

Objective is to build a model that accurately forecasts the total sales for thousands of items sold at Favorita stores franchise. We will implement various Time series Forecasting techniques with an approachable training dataset of dates, store, and item information, promotions, and unit sales to forecast the total sales while considering various Macro and Micro Economic Factors.

Data Source:

Kaggle - https://www.kaggle.com/competitions/store-sales-time-series-forecasting

TEXAS
The University of Texas at Austin

DATASET SUMMARY & DESCRIPTION

DATA SUMMARY | There are 5 files- one for Holiday, Transactions, Daily sales, Oil Price and Store information

Daily Sales Data

The Daily sales data, comprising time series of features store_nbr, family, and onpromotion as well as the target sales.

`store_nbr`: identifies the store at which the products are sold.

`family`: identifies the type of product sold.

`sales` gives the total sales for a product family at a particular store at a given date. Fractional values are possible since products can be sold in fractional units (1.5 kg of cheese, for instance, as opposed to 1 bag of chips).

`onpromotion` gives the total number of items in a product family that were being promoted at a store at a given date.

Store data

Store metadata, including city, state, type, and cluster.

cluster is a grouping of similar stores.

Oil data

Daily oil price. Includes values during both the train and test data timeframes. (Ecuador is an oil-dependent country and it's economical health is highly vulnerable to shocks in oil prices.)

Holiday data

Holidays and Events, with metadata

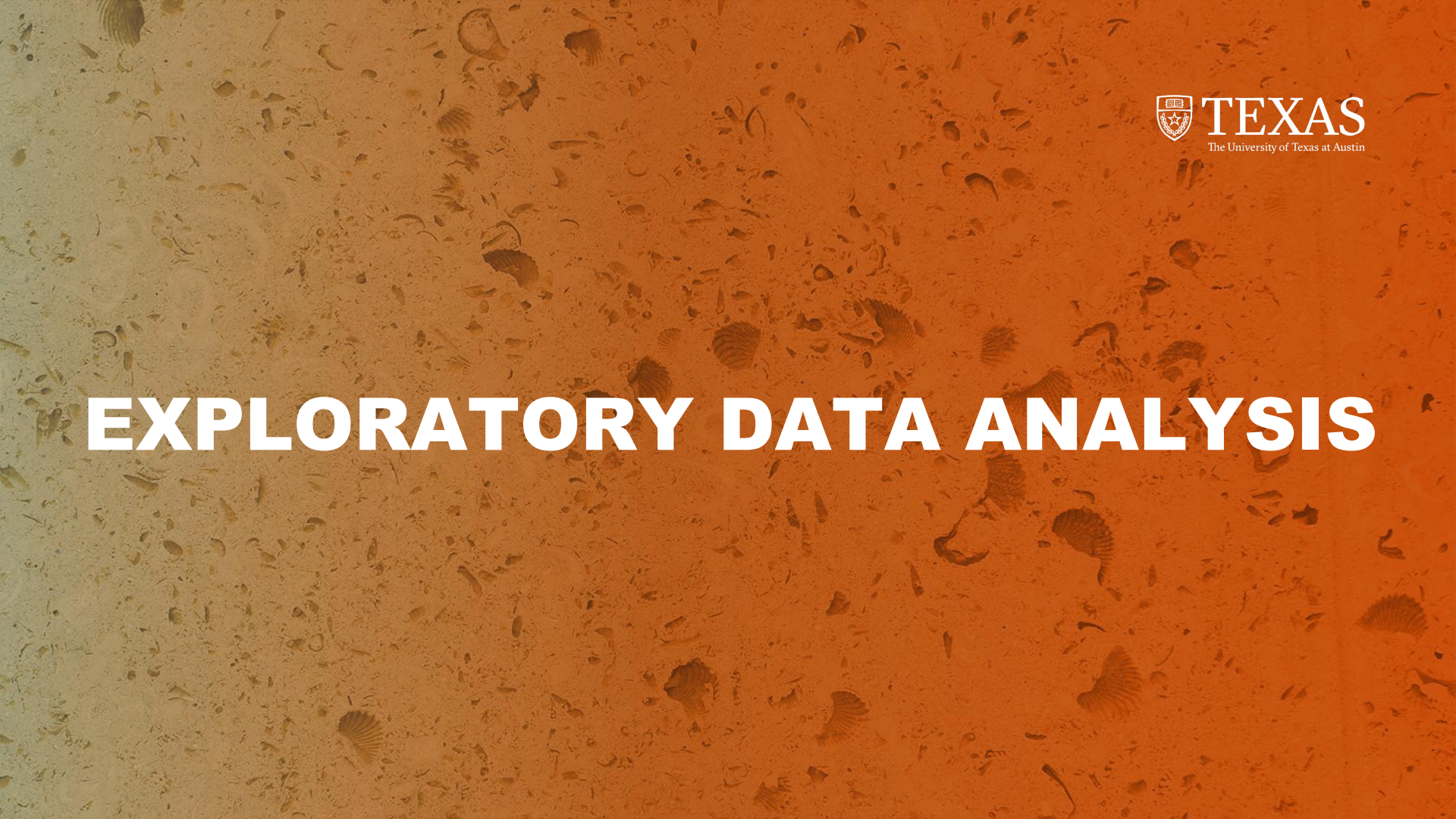
Transaction data

Stores total transactions across each store at a daily level

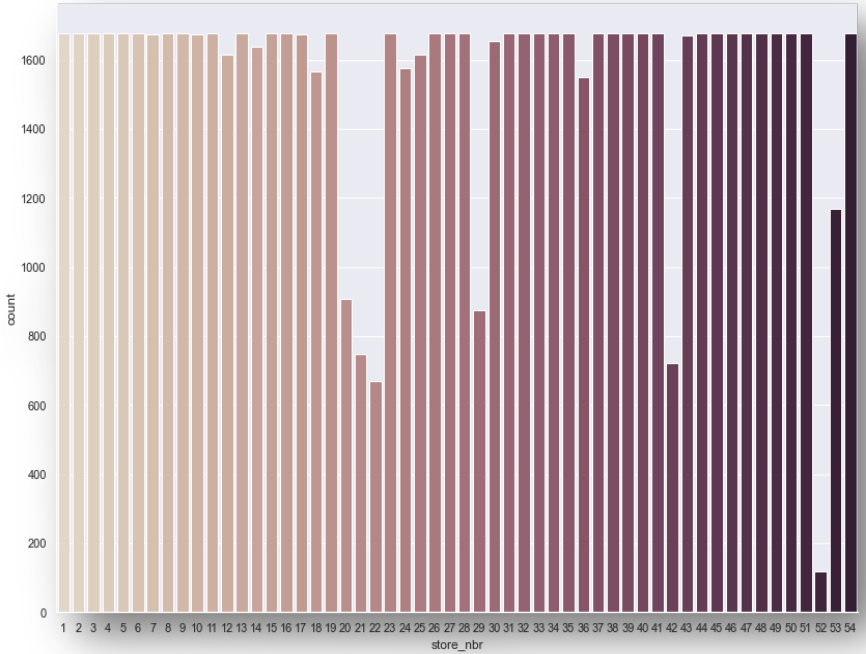
Additional Notes

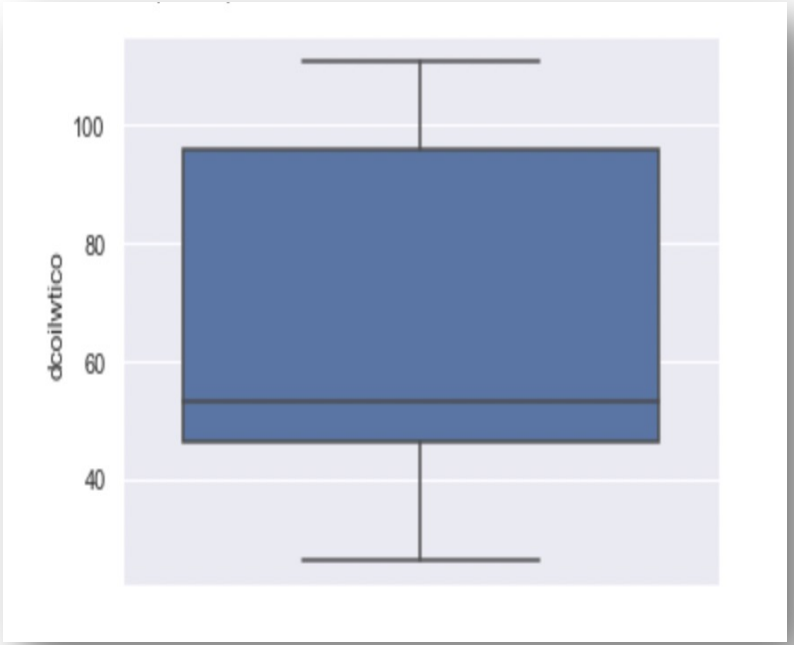
Wages in the public sector are paid every two weeks on the 15 th and on the last day of the month. Supermarket sales could be affected by this.

A magnitude 7.8 earthquake struck Ecuador on April 16, 2016. People rallied in relief efforts donating water and other first need products which greatly affected supermarket sales for several weeks after the earthquake.

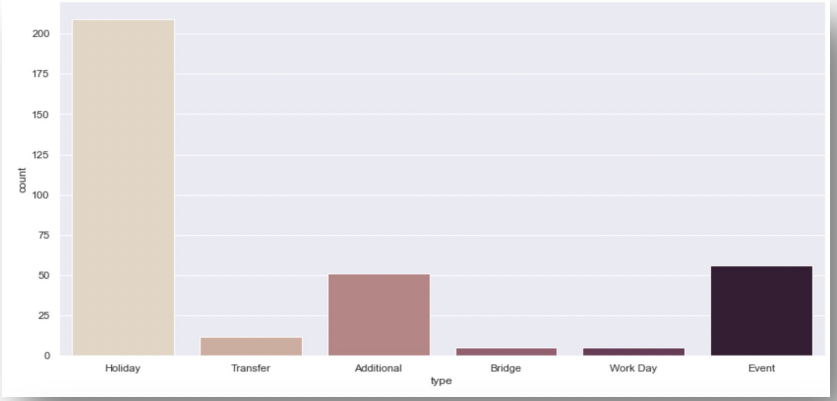

TEXAS
The University of Texas at Austin
EXPLORATORY DATA ANALYSIS

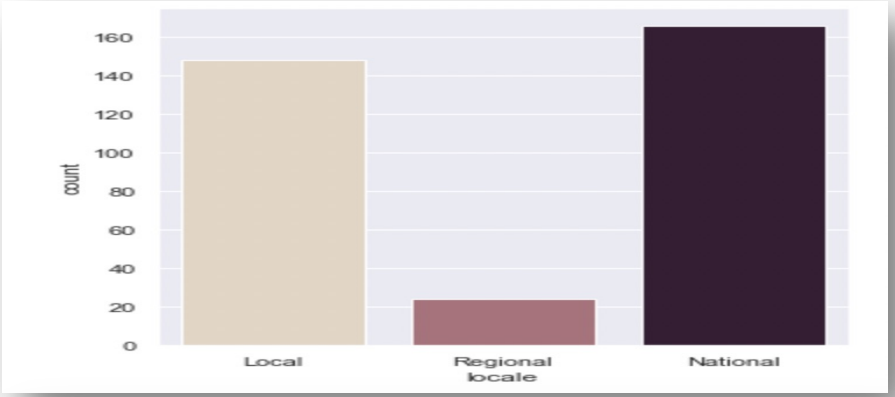
EDA | Uni-Variate Analysis | Transaction Data - Distribution of Stores across number of Transactions | OIL Data -

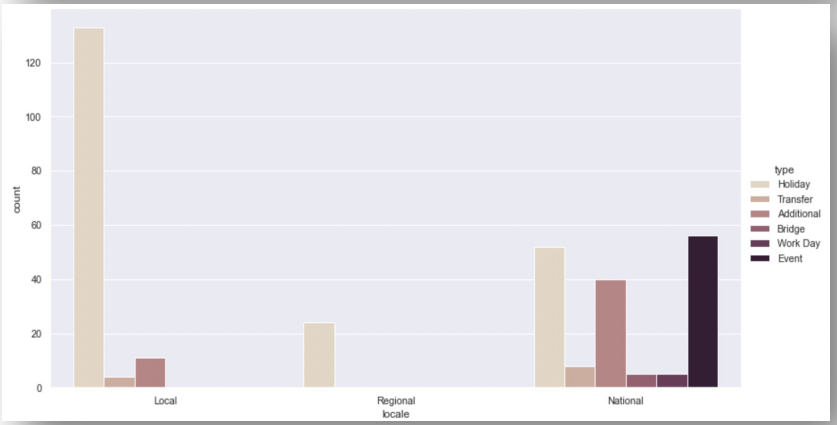




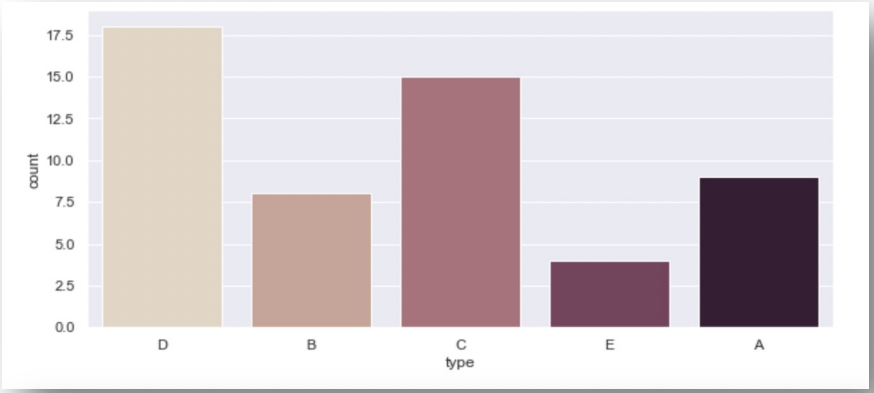
EDA | Uni-Variate Analysis | Holidays Data Frame – Distribution of Holiday Types

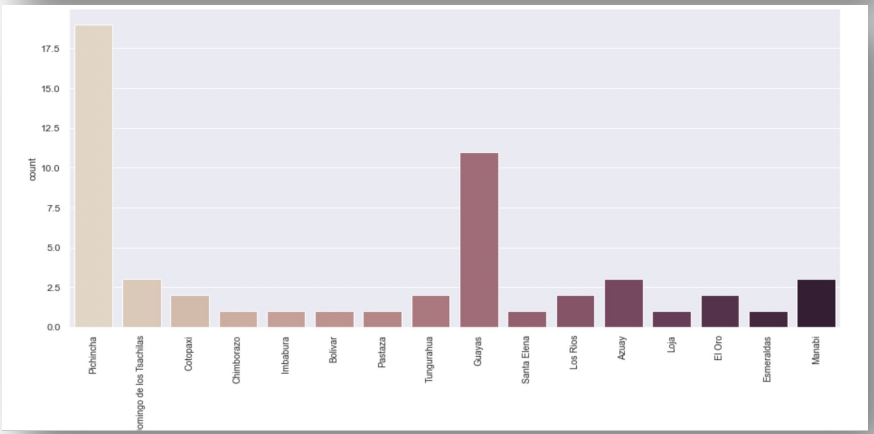


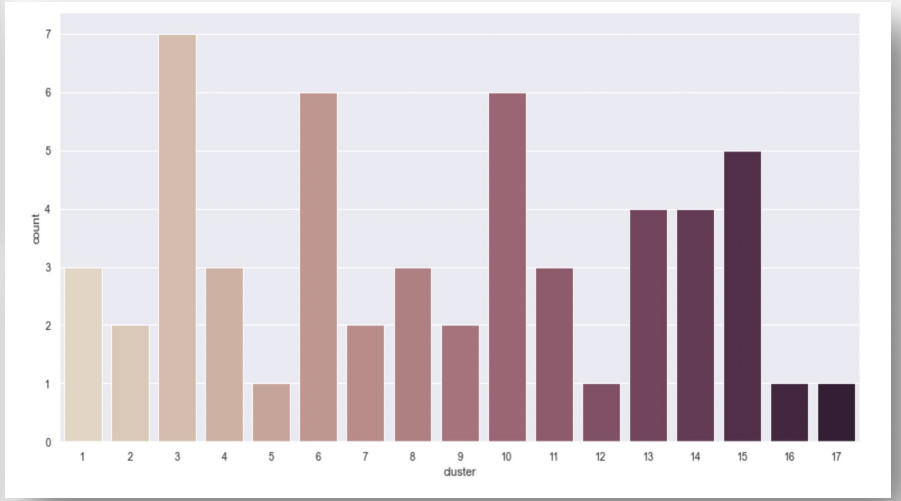




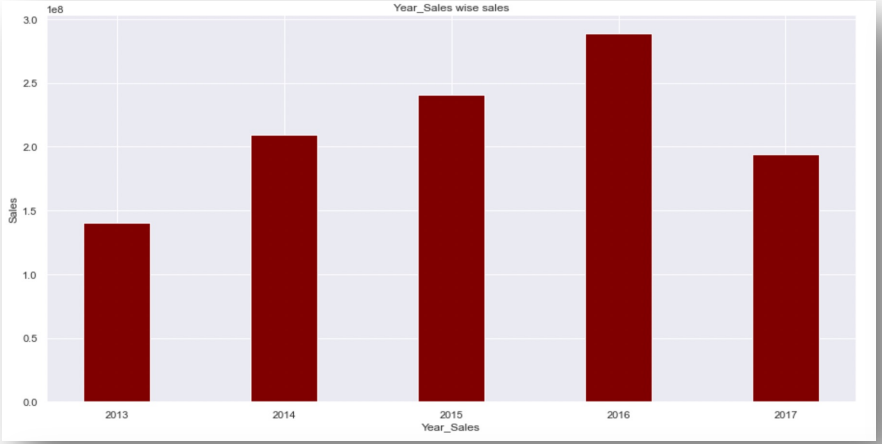
EDA | Uni-Variate Analysis | Checking the distribution of Store Type, Cluster and City in the Store Data Set

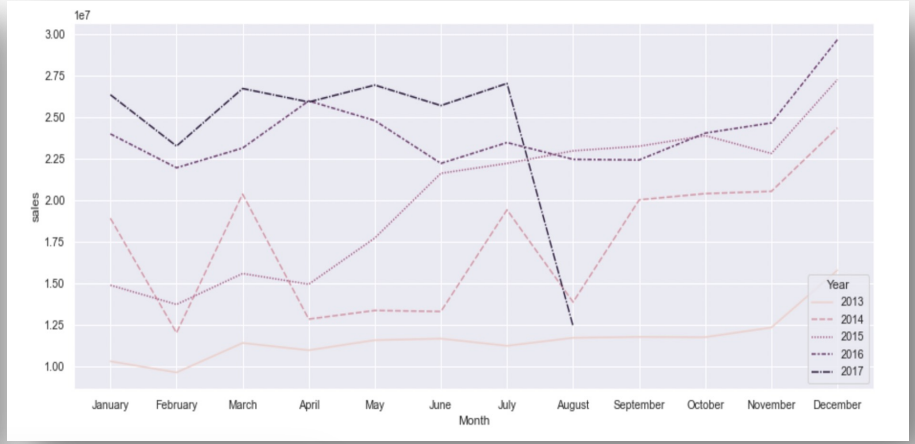


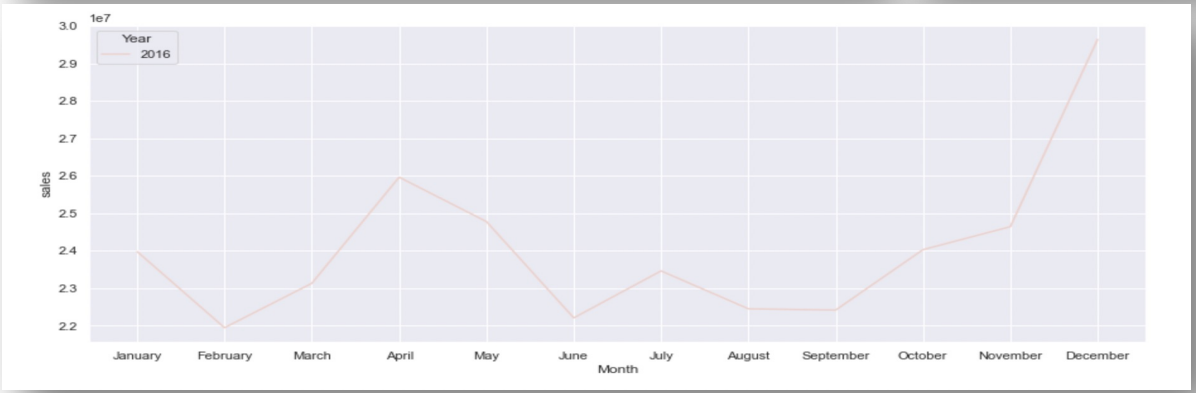




EDA | Plotting Sales across Years to determine if any seasonality or trend is present in the data







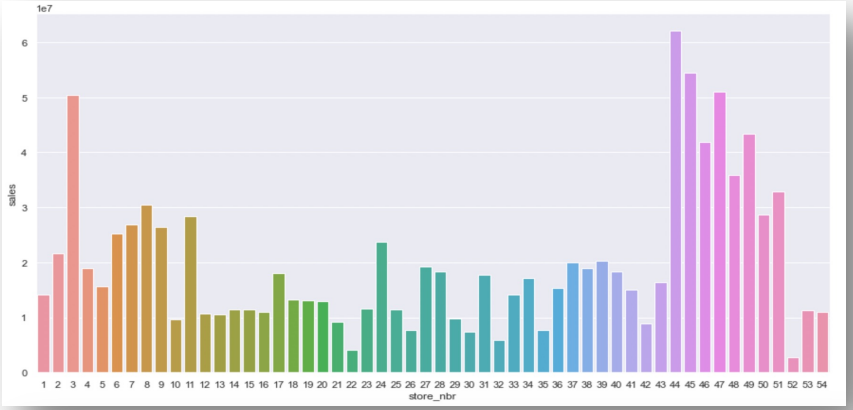
Observations:
a) Out of all five years, 2016 has the highest sales.
b) Further, we looked at monthly sales across all the 5 years to check if all the year has the same trend across the month
c) We found that there is uncommon spike in sales in the month of April for the year of 2016. As it has been mentioned that an earthquake happened, this could be a reason for the spike of sales

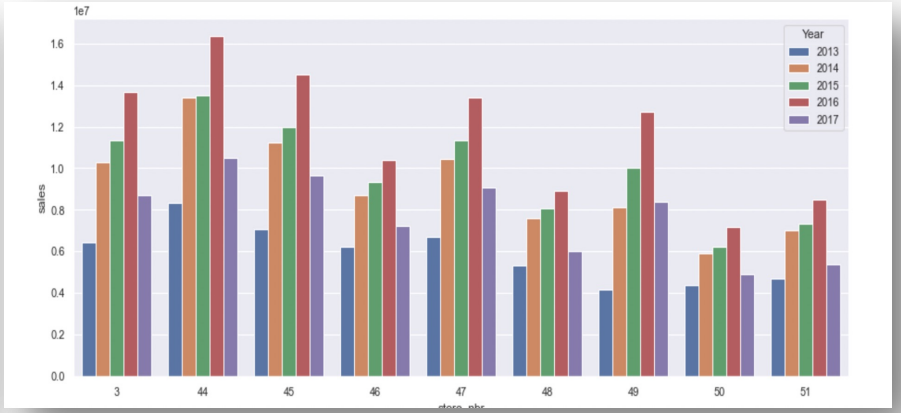
TEXAS
The University of Texas at Austin

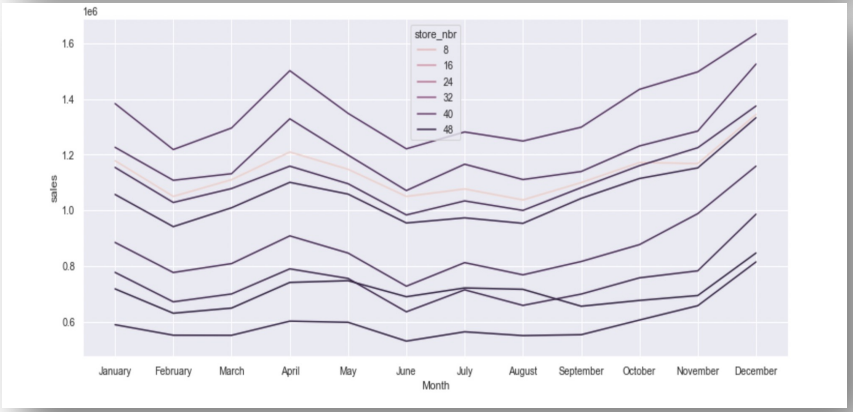
HYPOTHESIS TESTING

HYPOTHESIS TESTING | Hypothesis #1| Store has an effect on Sales of the company

H0: Store has no effect on sales | **H1**: Store has an effect on sales








```
# Ordinary Least Squares (OLS) model
model = ols('sales ~ store_nbr', data=train_df4).fit()
anova_table = sm.stats.anova_lm(model, typ=2)
anova_table
```

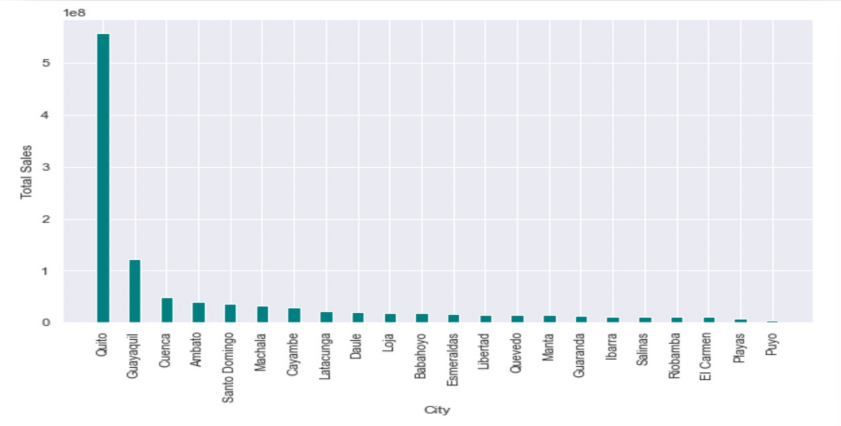
	sum_sq	df	F	PR(>F)
store_nbr	1.685856e+11	53.0	2746.286055	0.0
Residual	3.475688e+12	3000834.0	NaN	NaN

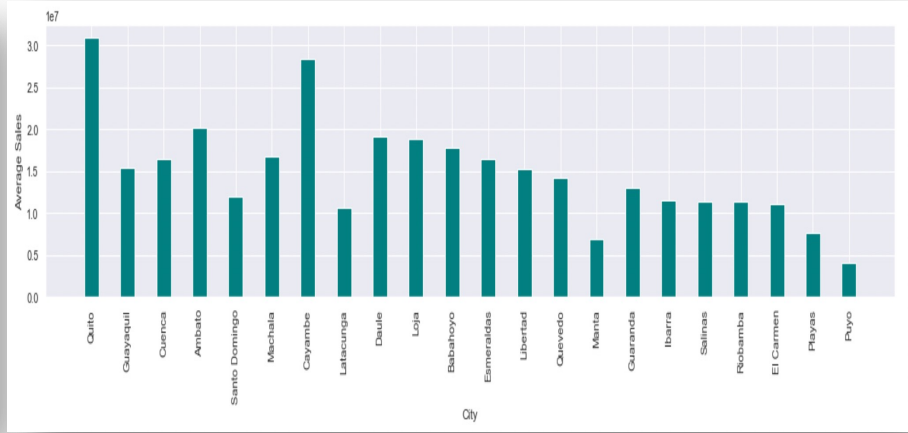
As P<0.05, we can reject the Null Hypothesis; i.e. Store Number has an effect on sales.

All the stores have pretty much similar sales except Store-3,44,45,46,47,48,49,50,51. We looked deeper into them
These stores have a spike in sales in the year of 2016 approximately by 25% in most of the stores; one of the main reason could be the earthquake in April 2016.

HYPOTHESIS TESTING | Hypothesis #2 | Location has an effect on Sales of the company

H0: Location has no effect on sales | **H1**: Location has an effect on sales






```
# Ordinary Least Squares (OLS) model
model = ols('sales ~ state', data=train_df4).fit()
anova_table = sm.stats.anova_lm(model, typ=2)
anova_table
```

	sum_sq	df	F	PR(>F)
state	7.012482e+10	15.0	3925.141397	0.0
Residual	3.574149e+12	3000872.0	NaN	NaN

```
# Ordinary Least Squares (OLS) model
model = ols('sales ~ city', data=train_df4).fit()
anova_table = sm.stats.anova_lm(model, typ=2)
anova_table
```

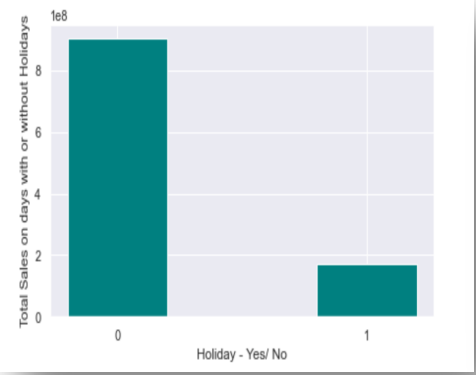
	sum_sq	df	F	PR(>F)
city	7.185915e+10	21.0	2874.401888	0.0
Residual	3.572415e+12	3000866.0	NaN	NaN

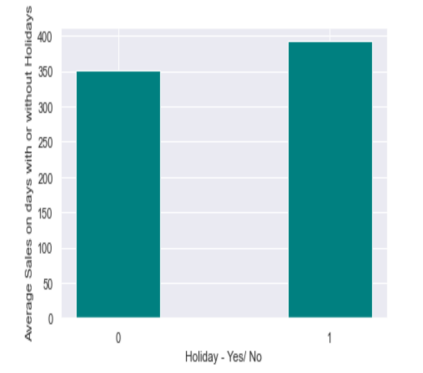
As seen from the graph, Total sales and average sales per store are highest in City Quito, followed by Guayaquil

As P<0.05, we can reject the Null Hypothesis; i.e. City / State has an effect on sales

HYPOTHESIS TESTING | Hypothesis #3 | Holiday has an effect on Sales of the company

H0: Holiday has no effect on sales | **H1**: Holiday has an effect on sales





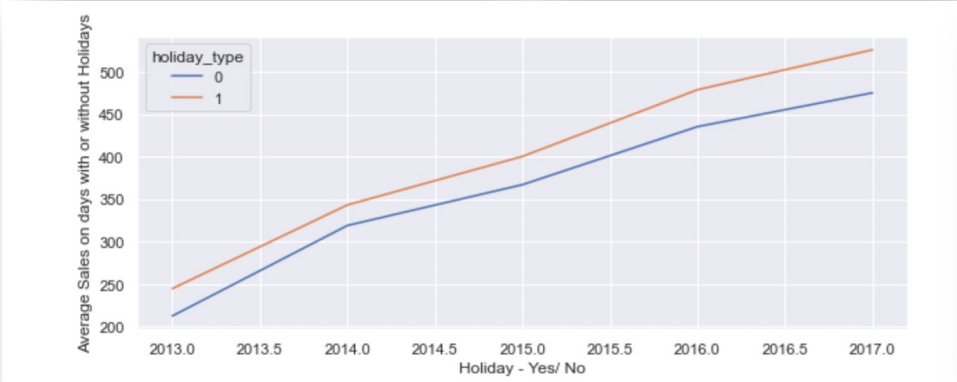

```
# Ordinary Least Squares (OLS) model
model = ols('sales ~ holiday_type', data=train_df4).fit()
anova_table = sm.stats.anova_lm(model, typ=2)
anova_table
```

	sum_sq	df	F	PR(>F)
holiday_type	6.132023e+08	1.0	505.027878	7.821313e-112
Residual	3.643661e+12	3000886.0	NaN	NaN

As P<0.05, we can reject the Null Hypothesis; i.e. Holiday has an effect on sales

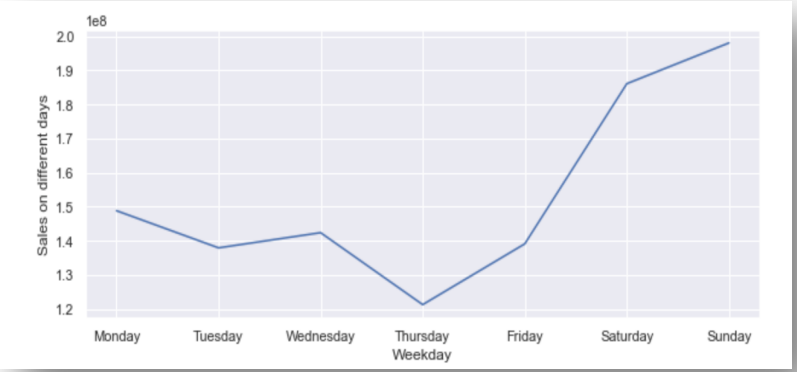
Observations:
Even though total sales on days without holiday is far greater than days with holiday; if we look into average sales; it is just the opposite;

Conclusion:
Sales are much higher on Holidays than without Holidays



HYPOTHESIS TESTING | Hypothesis #4 | Weekday has an effect on Sales of the company

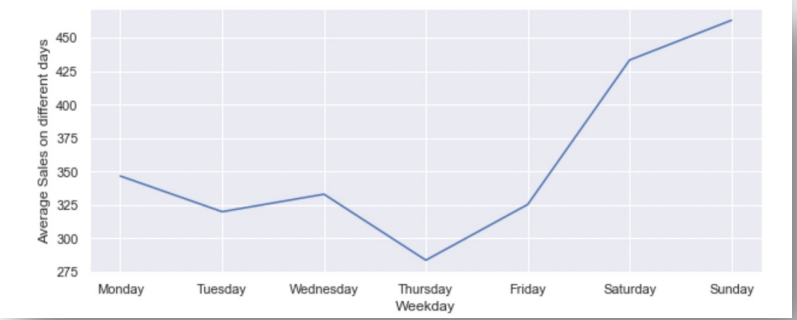
H0: Weekday has no effect on sales | **H1**: Weekday has an effect on sales




```
# Ordinary Least Squares (OLS) model
model = ols('sales ~ weekday', data=train_df4).fit()
anova_table = sm.stats.anova_lm(model, typ=2)
anova_table
```

	sum_sq	df	F	PR(>F)
weekday	1.094443e+10	6.0	1506.558564	0.0
Residual	3.633330e+12	3000881.0	NaN	NaN

As P<0.05, we can reject the Null Hypothesis; i.e. Weekday has an effect on sales



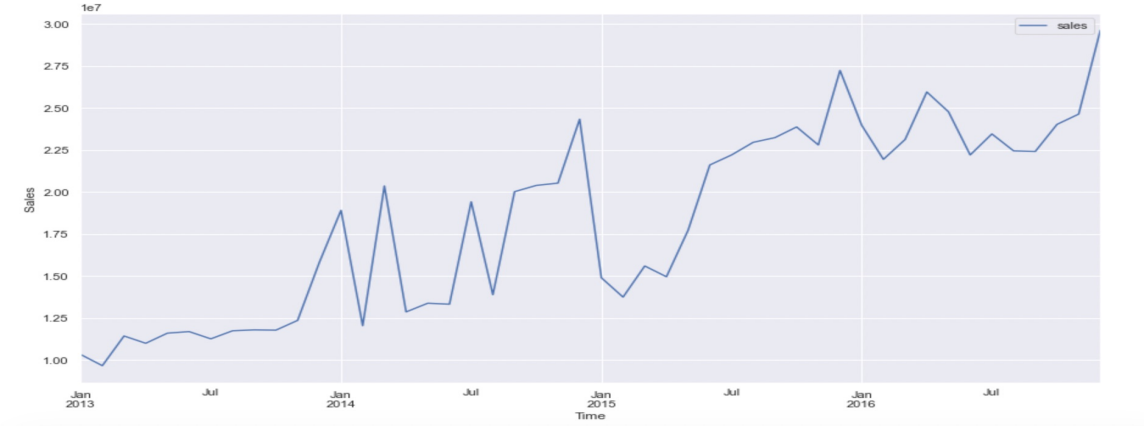
Observations:
Total sales on weekends is far greater than that of weekdays; if we look into average sales; the same trend continues;

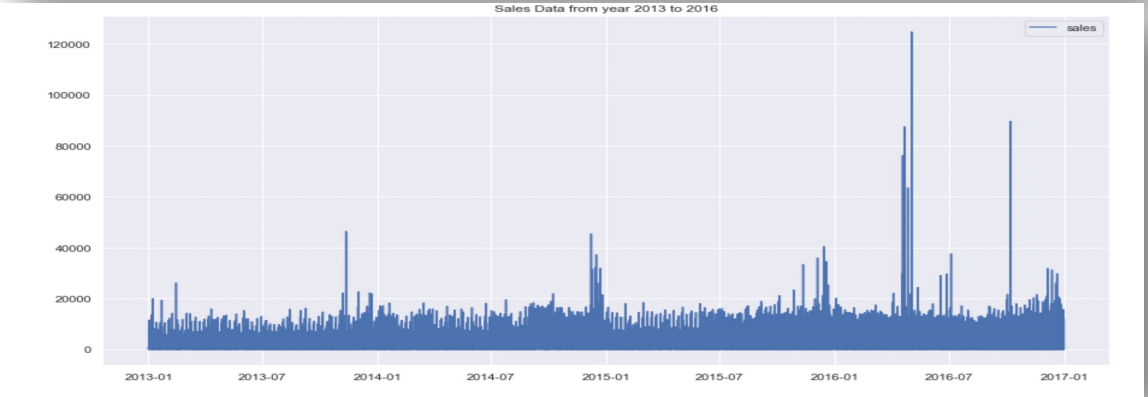
Conclusion:
People tend to shop more during Weekends

TEXAS
The University of Texas at Austin

PRE-MODELING

PRE- MODELING | Running ADF Test to check if the time series is stationary?






```
# ADF Test
# Function to print out results in customised manner
from statsmodels.tsa.stattools import adfuller
dftest = adfuller(overall_dailysales.sales, autolag = 'AIC')
for key, val in dftest[4].items():
    print("\t",key, ": ", val)
```

```
1. ADF :  -2.009672040278186
2. P-Value :  0.2823484153973075
3. Num Of Lags :  22
4. Num Of Observations Used For ADF Regression and Critical Values Calculation : 1434
5. Critical Values :
        1% :  -3.434918371231736
        5% :  -2.8635576234668982
        10% :  -2.5678441693558898
```

Observation: Since p_value>0.5 we can not reject the Null Hypothesis; time series is not stationary

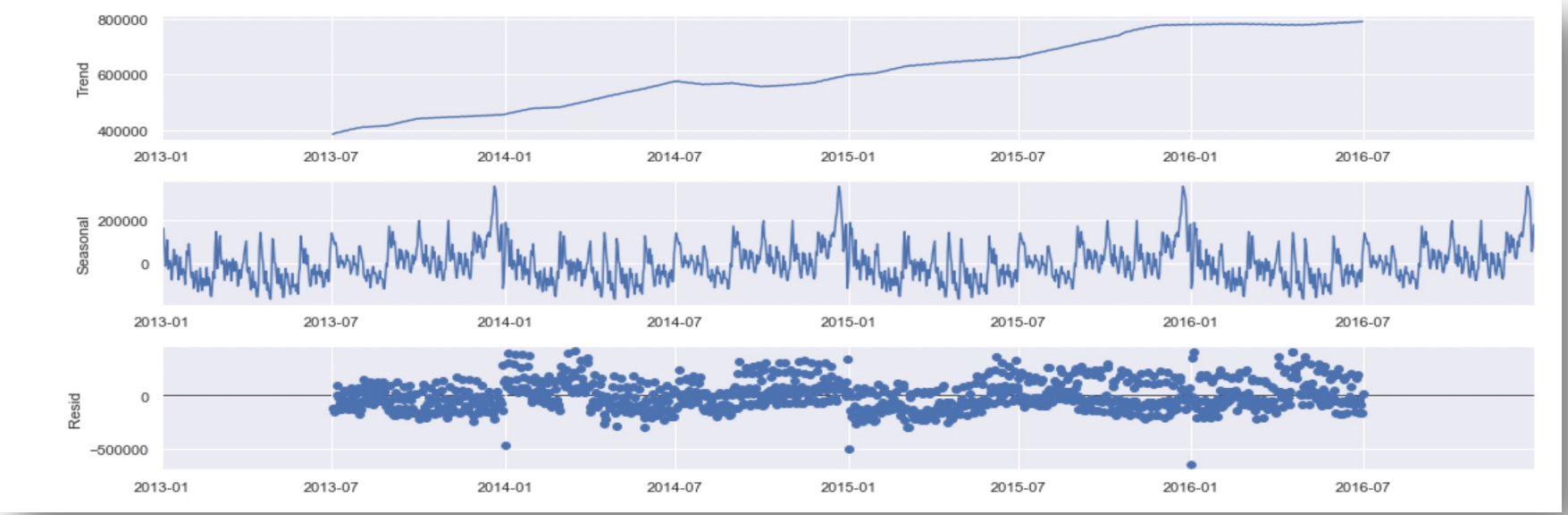
Observation:

Since p_value>0.05, we can conclude that our time series is not stationary.
To run any Time series model; we first need to make the series stationary

Observation:

As seen in the graph, we can clearly see an increasing trend along with spike in sales at a particular time period representing seasonality

PRE- MODELING | Breaking down the time series to see the trend , seasonality and residual in our data



Observation:
Time series is an Additive one not a multiplicative one
Trend is present in the data along with Seasonality

PRE- MODELING | Running ACF and PACF to find the P and Q value for the ARIMA Model along with doing first degree differencing to make our time series stationary

```
overall_dailysales['sales_diff'] = overall_dailysales['sales'] - overall_dailysales['sales'].shift(1)
overall_dailysales['sales_diff'].fillna(0, inplace=True)
overall_dailysales[['sales','sales_diff']].head(5)
```

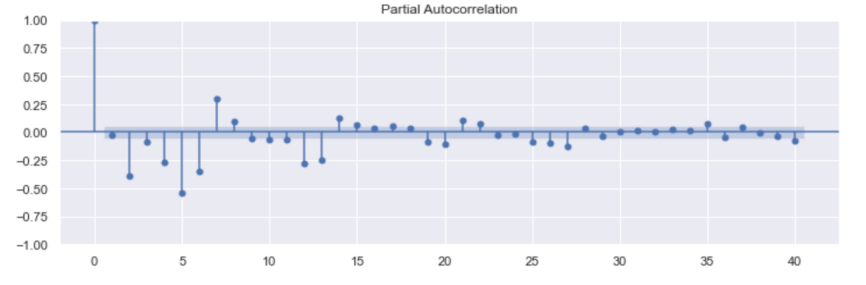
	sales	sales_diff
2013-01-01	2511.618999	0.000000
2013-01-02	496092.417944	493580.798945
2013-01-03	361461.231124	-134631.186820
2013-01-04	354459.677093	-7001.554031
2013-01-05	477350.121229	122890.444136

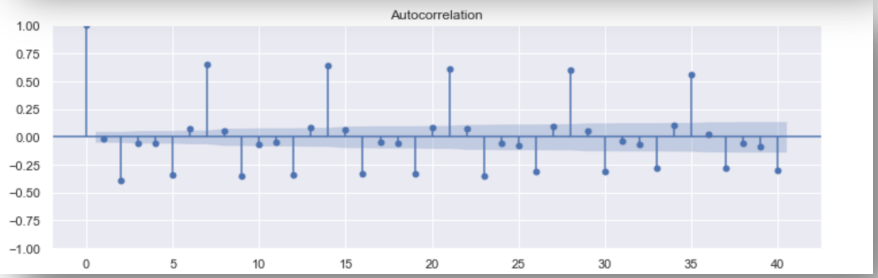
```
# ADF Test
# Function to print out results in customised manner
from statsmodels.tsa.stattools import adfuller
dftest = adfuller(overall_dailysales['sales_diff'], autolag = 'AIC')
for key, val in dftest[4].items():
    print("\t",key, ": ", val)
```

```
1. ADF :  -9.5770669845666
2. P-Value :  2.2088048426585958e-16
3. Num Of Lags :  24
4. Num Of Observations Used For ADF Regression and Critical Values Calculation : 1432
5. Critical Values :
         1% :  -3.4349247631306237
         5% :  -2.8635604442944658
         10% :  -2.5678456715029183
```

As seen, after 1st level of differencing, our data is becoming stationary; hence d=1

Observation:
We did first degree differencing to make our time series stationary for further forecasting. After doing the same, we again did ADF test to make sure our p value is coming less than 0.05; statistically proving that timeseries is stationary now





Observation:
Time series is an Additive one not a multiplicative one
Trend is present in the data along with Seasonality

TEXAS
The University of Texas at Austin

MODELING

MODELING | TOTAL SALES | ARIMA

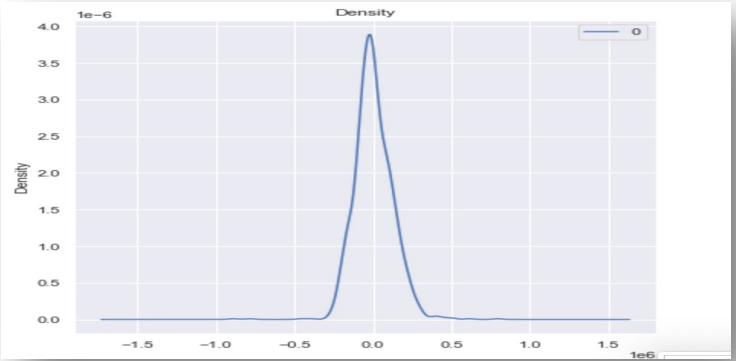
```
from statsmodels.tsa.arima.model import ARIMA
# 1,1,2 ARIMA Model
model = ARIMA(overall_dailysales['sales'],order=(2,1,2))
model_fit = model.fit()
print(model_fit.summary())
```

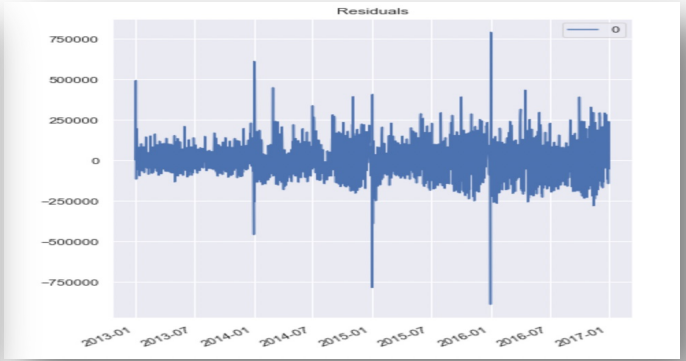
```
                               SARIMAX Results
==============================================================================
Dep. Variable:                   sales   No. Observations:                 1457
Model:                 ARIMA(2, 1, 2)   Log Likelihood              -19124.172
Date:                Sun, 07 Aug 2022   AIC                          38258.345
Time:                        17:29:38   BIC                          38284.762
Sample:                             0   HQIC                         38268.201
                                - 1457
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.3104      0.047      6.587      0.000       0.218       0.403
ar.L2         -0.2283      0.034     -6.628      0.000      -0.296      -0.161
ma.L1         -0.5909      0.048    -12.299      0.000      -0.685      -0.497
ma.L2         -0.2974      0.045     -6.600      0.000      -0.386      -0.209
sigma2      1.255e+10   1.13e-12   1.11e+22      0.000    1.25e+10    1.25e+10
===================================================================================
Ljung-Box (L1) (Q):                   0.09   Jarque-Bera (JB):              1369.46
Prob(Q):                              0.77   Prob(JB):                         0.00
Heteroskedasticity (H):               2.52   Skew:                             0.13
Prob(H) (two-sided):                  0.00   Kurtosis:                         7.74
===================================================================================
```

Observation:
We have used AR (p) =2, the number of lag observations or autoregressive terms in the model;

Integration (d)=1, the difference in the nonseasonal observations;

MA (q)=2, the size of the moving average window.





The residual errors seem fine with near zero mean and uniform variance.

MODELING | TOTAL SALES | ARIMA


```
{'mape': 0.2641178849057131,
 'me': -4426.742652704442,
 'mae': 90965.83439581658,
 'mpe': 0.13602430206612104,
 'rmse': 121791.6291369397,
 'corr': 0.8387254368706629,
 'minmax': 0.13453818302091303}
```

	sales	forecast
2013-01-01	2.511619e+03	0.000000e+00
2013-01-02	4.960924e+05	2.447143e+03
2013-01-03	3.614612e+05	4.767770e+05
2013-01-04	3.544597e+05	1.578492e+05
2013-01-05	4.773501e+05	3.189684e+05
...	...	...
2016-12-27	8.424755e+05	9.576738e+05
2016-12-28	9.515337e+05	9.074954e+05
2016-12-29	8.941082e+05	1.037073e+06
2016-12-30	1.163643e+06	9.227592e+05
2016-12-31	1.109013e+06	1.160603e+06

MAPE value is coming as 0.264 that is we are getting 74% accuracy in the forecasted values

MODELING | TOTAL SALES | AUTO ARIMA

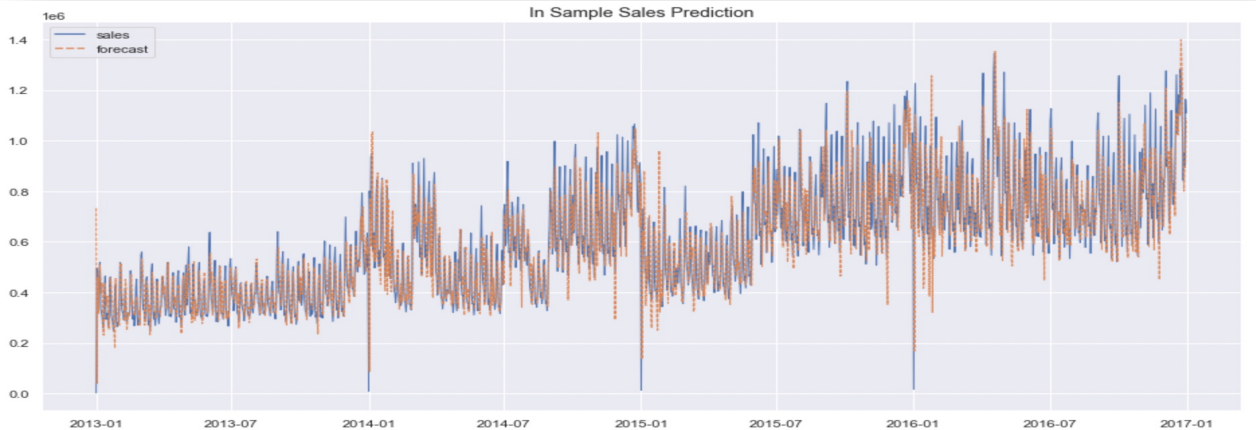
```
model_fit = pm.auto_arima(overall_dailysales['sales'], test='adf',
                          max_p=3, max_d=3, max_q=3,
                          seasonal=True, m=12,
                          max_P=3, max_D=2, max_Q=3,
                          trace=True,
                          error_action='ignore',
                          suppress_warnings=True,
                          stepwise=True)

# summarize the model characteristics
print(model_fit.summary())
```

```
Performing stepwise search to minimize aic
 ARIMA(2,0,2)(1,0,1)[12] intercept   : AIC=38391.738, Time=3.35 sec
 ARIMA(0,0,0)(0,0,0)[12] intercept   : AIC=40028.849, Time=0.02 sec
 ARIMA(1,0,0)(1,0,0)[12] intercept   : AIC=38561.395, Time=0.56 sec
 ARIMA(0,0,1)(0,0,1)[12] intercept   : AIC=39150.765, Time=0.31 sec
 ARIMA(0,0,0)(0,0,0)[12]             : AIC=43110.964, Time=0.01 sec
 ARIMA(2,0,2)(0,0,1)[12] intercept   : AIC=38405.041, Time=0.64 sec
 ARIMA(2,0,2)(1,0,0)[12] intercept   : AIC=38393.145, Time=1.41 sec
 ARIMA(2,0,2)(2,0,1)[12] intercept   : AIC=38430.924, Time=8.40 sec
 ARIMA(2,0,2)(1,0,2)[12] intercept   : AIC=38489.634, Time=9.44 sec
 ARIMA(2,0,2)(0,0,0)[12] intercept   : AIC=38450.225, Time=0.21 sec
 ARIMA(2,0,2)(0,0,2)[12] intercept   : AIC=38402.440, Time=2.21 sec
 ARIMA(2,0,2)(2,0,0)[12] intercept   : AIC=inf, Time=6.30 sec
 ARIMA(2,0,2)(2,0,2)[12] intercept   : AIC=inf, Time=9.51 sec
```

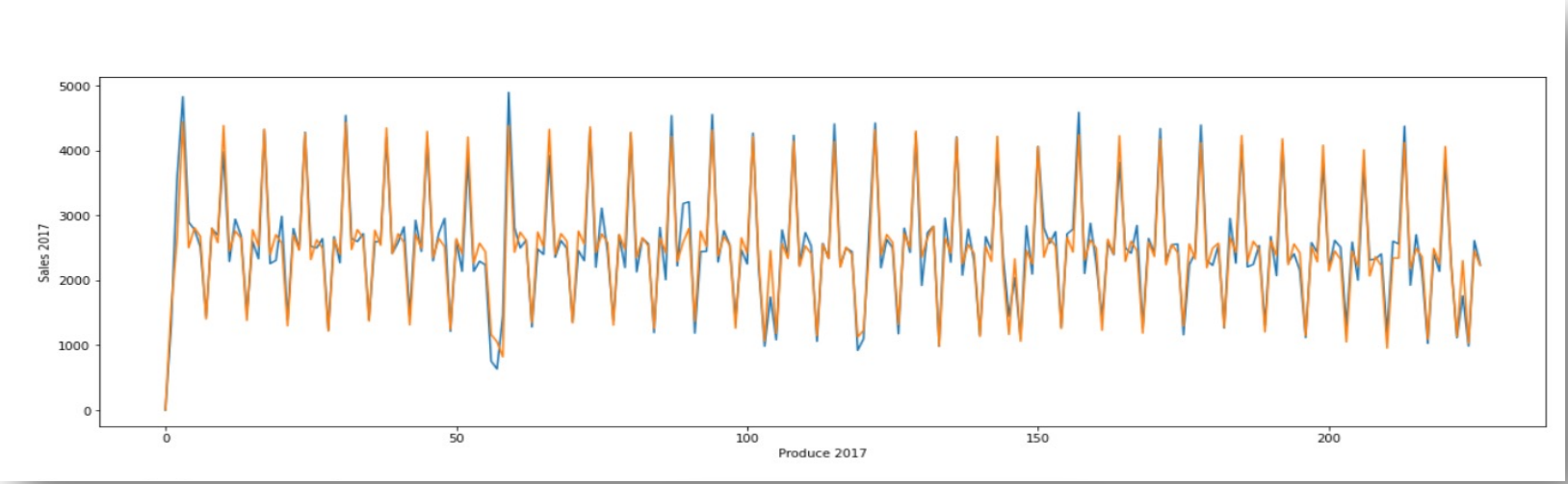
```
Best model:  ARIMA(3,0,2)(2,0,1)[12] intercept
Total fit time: 230.597 seconds
                                SARIMAX Results
==========================================================================================
Dep. Variable:                                 y   No. Observations:                 1457
Model:             SARIMAX(3, 0, 2)x(2, 0, [1], 12)   Log Likelihood           -19045.736
Date:                              Sun, 07 Aug 2022   AIC                        38111.472
Time:                                      17:35:28   BIC                        38164.313
Sample:                                           0   HQIC                       38131.186
                                              - 1457
Covariance Type:                                opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
intercept   6.156e+04   3.33e+04      1.849      0.064   -3698.928    1.27e+05
ar.L1          0.5410      0.012     46.429      0.000       0.518       0.564
ar.L2         -0.5454      0.009    -62.568      0.000      -0.562      -0.528
ar.L3          0.9559      0.013     75.803      0.000       0.931       0.981
ma.L1          0.3319      0.012     26.802      0.000       0.308       0.356
ma.L2          0.8815      0.015     60.309      0.000       0.853       0.910
ar.S.L12      -0.2672      0.056     -4.753      0.000      -0.377      -0.157
ar.S.L24      -0.4669      0.018    -25.881      0.000      -0.502      -0.432
ma.S.L12       0.1931      0.063      3.067      0.002       0.070       0.316
sigma2      1.541e+10      1.925   8.01e+09      0.000    1.54e+10    1.54e+10
==========================================================================
Ljung-Box (L1) (Q):                  18.01   Jarque-Bera (JB):              5966.29
Prob(Q):                              0.00   Prob(JB):                         0.00
Heteroskedasticity (H):               2.48   Skew:                            -0.03
Prob(H) (two-sided):                  0.00   Kurtosis:                        12.91
==========================================================================
```

Observation:
We have run auto arima which automatically takes cares of AR(p), I(d), MA(q) value required for ARIMMA. It also incorporated the seasonality aspect of the data.



	sales	forecast
2013-01-01	2.511619e+03	7.332453e+05
2013-01-02	4.960924e+05	3.978717e+04
2013-01-03	3.614612e+05	4.593572e+05
2013-01-04	3.544597e+05	3.158998e+05
2013-01-05	4.773501e+05	4.644350e+05
...	...	...
2016-12-27	8.424755e+05	9.263562e+05
2016-12-28	9.515337e+05	7.991568e+05
2016-12-29	8.941082e+05	9.051875e+05
2016-12-30	1.163643e+06	9.149700e+05
2016-12-31	1.109013e+06	1.088565e+06

MODELING | XGBoost Regressor




```
Mean Absolute Error = 154.314
Root Mean Squared Log Error = 0.100
```

date	store_nbr	family		
2017-08-16	1	AUTOMOTIVE	3000888	4.258068
		BABY CARE	3000889	0.000000
		BEAUTY	3000890	3.682439
		BEVERAGES	3000891	2285.347832
		BOOKS	3000892	0.418454
...	...	...	...	...
2017-08-31	54	POULTRY	3029263	60.489879
		PREPARED FOODS	3029264	83.797301
		PRODUCE	3029265	497.348026
		SCHOOL AND OFFICE SUPPLIES	3029266	1.670238
		SEAFOOD	3029267	3.481247

28512 rows × 2 columns

TEXAS
The University of Texas at Austin

THANK YOU