

CHAPTER 1

ABOUT TECH FORTUNE TECHNOLOGIES

1.1 Vision

With an unyielding focus on integrity and backed by strong founders and management team, Sourcing wants to make a mark in the field of IT services by applying innovation to simplify complex business processes and add value to clients' business.

1.2 Mission

Being slow and steady, our mission is to gain the confidence of our clients and by dint of our integrity, innovation and dynamism, deliver their requirements on time with full quality thus bridging the gap between demand and delivery.

1.3 About company:

Tech Fortune was incorporated with a goal.” To provide high quality and optimal Technological Solutions to business requirements of our clients”. Every business is a different and has a unique business model and so are the technological requirements. They understand this and hence the solutions provided to these requirements are different as well. They focus on clients requirements and provide them with tailor made technological solutions. They also understand that Reach of their Product to its targeted market or the automation of the existing process into efficient and simple process are the key features that our clients desire from Technological Solution they are looking for and these are the features that we focus on while designing the solutions for their clients.

Tech Fortune is a Technology Organization providing solutions for all web design and development, MYSQL, PYTHON Programming, C#, HTML, CSS, ASP .NET and LINQ. Meeting the ever increasing automation requirements Tech Fortune specialize in ERP, Connectivity, SEO Services, Conference Management, effective web promotion and tailor made software products, designing solutions best suiting client's requirements.

Tech Fortune, strive to be the front runner in creativity and innovation in software development through their well researched expertise and establish it as an out of the

box software development company in Bangalore, India. As a software development company they translate this software development expertise into value for their customers through their professional solutions.

They understand that the best desired output can be achieved only by understanding the client's demand better. Tech Fortune work with their clients and help them to define their exact solution requirement. Sometimes even they wonder that they have completely redefined their solution or new application requirement during the brainstorming session, and here they position themselves as an IT solutions consulting group comprising of high caliber consultants.

1.4 Objectives:

Tech Fortune brings together and integrates technology and expertise to create the ICT-enabled society of tomorrow.

We collaborate closely with partners and customers around the world, orchestrating each project to ensure all its parts are fine-tuned to local needs.

Every day, our innovative solutions for society contribute to greater safety, security, efficiency and equality, and enable people to live brighter lives.

Tech Fortune is committed to creating social value with our partners.

Working together, we can create safer, more secure, more efficient and more equal societies, so that people around the world can lead better lives.

We are confident that our global leadership in information and network technologies will be an invaluable asset for achieving this meaningful goal.

1.5 Quality Policy:

Tech Fortune Technologies is a process driven organization and adopts Global Standards for providing software solutions and services to its customers.

The processes and practices at NTI are derived from the International Standards, e.g., CMMi, PCMM, ITIL.

Thus, the Quality Management System at NTI is a comprehensive repository of Industry Best Practices ranging from Project Management, Software Development Methodologies, Service Management, Quantitative Project Management, etc.

Services:

- Enterprise Software Development
- Enterprise Mobility Development
- Digital Transformation Services
- Data Analytics and Visualization
- Business Technology Consulting

CHAPTER 2

ABOUT THE DEPARTMENT

Tech-fortune plays an essential role as an institute, the level of education, development of student's skills are based on their trainers. If you do not have a good mentor then you may learn many things from others and that is why we at Tech- fortune gives you the facility of skilled employees so that you do not feel unsecured about the academics. Personality development and academic status are some of those things which lie on mentor's hands. If you are trained well then you can do well in your future and knowing its importance of Tech-fortune always tries to give you the best.

They have a great team of skilled mentors who are always ready to direct their trainees in the best possible way they can and to ensure the skills of mentors we held many skill development programs as well so that each and every mentor can develop their own skills with the demands of the companies so that they can prepare a complete packaged trainee.

Various Solutions Provided by Tech Fortune are:

- **Payment Gateway:** Tech Fortune Payment Solutions India Pvt. Ltd. provides electronic payment platform solutions and services in India. It offers a variety of payment terminals, including dial-up, GPRS/Mobile, PC-POS, and M-POS for banks/merchants. It offers Fortune Payments platform, a payment network certified third-party payments processing infrastructure that enable banks to acquire transactions securely using a variety of devices. Fortune Gateway, an Internet payment gateway solution that provides payment SDK, acquiring front-end, PCI certification of the platform, integration with payment scheme, and support for a variety of multi-factor authentication schemes, including 3D-secure based (VbV/MSC), SMS, OTP.
- **Web Application:** Tech Fortune deliver neat, easy-to-use and business friendly website designing, a web site that drives customers into it. In the domain of website design India, they have served almost every segment of the trade. Their website designing experts ensures each Tech Fortune deliver neat,

easy-to-use and business friendly website designing, a web site that drives customers into it. In the domain of website design India, they have served almost every segment of the trade. Their website designing experts ensures each quality parameter such as easy-navigation, overall consistency and content quality within quality parameter such as easy-navigation, overall consistency and content quality within the time frame and budget. They always ensure that their clients are delighted with their web designing services.

Chapter 3

TASKS PERFORMED

3.1 PROJECT SUMMARY

Time series analysis comprises methods for analyzing time series data in order to extract meaningful statistics and other characteristics of the data. Time series forecasting is the use of a model to predict future values based on previously observed values. Time series are widely used for non-stationary data, like economic, weather, stock price, and retail sales in this post.

Root Mean Square Error:

The root-mean-square error (RMSE) is a frequently used measure of the differences between values (sample or population values) predicted by a model or an estimator and the values observed. The RMSD represents the square root of the second sample moment of the differences between predicted values and observed values or the quadratic mean of these differences.

RMSE in Data Science: Subtleties of Using RMSE

In data science, RMSE has a double purpose:

- To serve as a heuristic for training models
- To evaluate trained models for usefulness / accuracy

3.2 CONCEPTS AND TOOLS USED

Titanic Survival Exploration is an application developed using Machine Learning Models. Machine Learning (ML) is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to perform the task.

ML is comprised of three types of algorithms:

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning

3.2.1 ROOT MEAN SQUARE ERROR

Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). Residuals are a measure of how far from the regression line data points are; RMSE is a measure of how spread out these residuals are. In other words, it tells you how concentrated the data is around the line of best fit. Root mean square error is commonly used in climatology, forecasting, and regression analysis to verify experimental results.

3.2.2 DATA VISUALIZATION

Data visualization is the discipline of trying to understand data by placing it in a visual context so that patterns, trends and correlations that might not otherwise be detected can be exposed.

Python offers multiple great graphing libraries that come packed with lots of different features. A few popular plotting libraries:

- Matplotlib: low level, provides lots of freedom
- Pandas Visualization: easy to use interface, built on Matplotlib
- Seaborn: high-level interface, great default styles
- ggplot: based on R's ggplot2, uses Grammar of Graphics
- Plotly: can create interactive plots

3.2.3 MATPLOTLIB

Matplotlib is the most popular python plotting library. It is a low-level library with a Matlab like interface which offers lots of freedom at the cost of having to write more code.

Matplotlib is specifically good for creating basic graphs like line charts, bar charts, histograms and many more.

3.2.4 PANDAS VISUALIZATION

Pandas is an open source high-performance, easy-to-use library providing data structures, such as dataframes, and data analysis tools like the visualization tools we will use in this article.

Pandas Visualization makes it really easy to create plots out of a pandas dataframe and series.

3.3 SNAPSHOTS

```
import warnings
import itertools
import numpy as np
import matplotlib.pyplot as plt
warnings.filterwarnings("ignore")
plt.style.use('fivethirtyeight')
import pandas as pd
import statsmodels.api as sm
import matplotlib
matplotlib.rcParams['axes.labelsize'] = 14
matplotlib.rcParams['xtick.labelsize'] = 12
matplotlib.rcParams['ytick.labelsize'] = 12
matplotlib.rcParams['text.color'] = 'k'

[ ] from google.colab import files
    uploaded = files.upload()

Choose Files | No file chosen | Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.
Saving Superstore.xls to Superstore.xls

[ ] import io
    df = pd.read_excel(io.BytesIO(uploaded['Superstore.xls']))
    furniture = df.loc[df['Category'] == 'Furniture']

[ ] furniture['Order Date'].min(), furniture['Order Date'].max()

(Timestamp('2014-01-06 00:00:00'), Timestamp('2017-12-30 00:00:00'))
```



```
[ ] cols = ['Row ID', 'Order ID', 'Ship Date', 'Ship Mode', 'Customer ID', 'Customer Name', 'Segment', 'Country', 'City', 'State', 'Postal Code', 'Region', 'Product ID', 'Category', 'Sub-Category']
furniture.drop(cols, axis=1, inplace=True)
furniture = furniture.sort_values('Order Date')
furniture.isnull().sum()
```

```
Order Date    0
Sales         0
dtype: int64
```

Indexing with Time Series Data

```
[ ] furniture = furniture.set_index('Order Date')
furniture.index
```

```
DatetimeIndex(['2014-01-06', '2014-01-07', '2014-01-10', '2014-01-11',
               '2014-01-13', '2014-01-13', '2014-01-14', '2014-01-16',
               '2014-01-19', '2014-01-20',
               ...,
               '2017-12-28', '2017-12-28', '2017-12-28', '2017-12-29',
               '2017-12-29', '2017-12-29', '2017-12-29', '2017-12-29',
               '2017-12-29', '2017-12-30'],
              dtype='datetime64[ns]', name='Order Date', length=2121, freq=None)
```

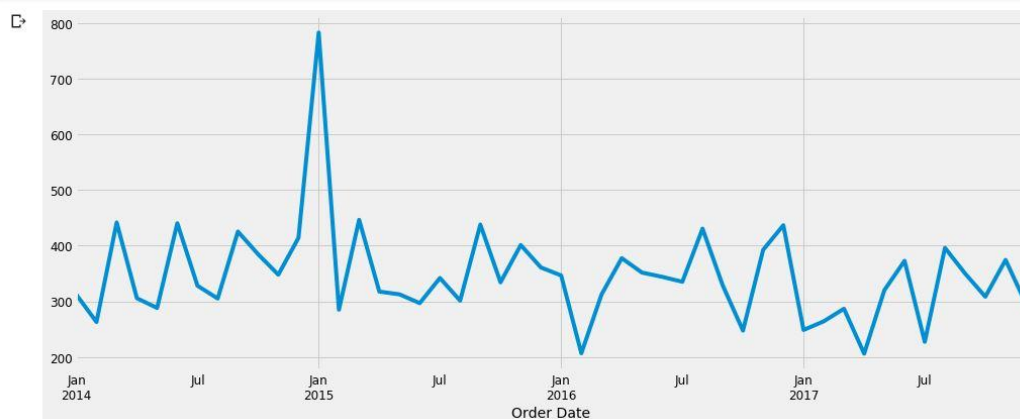
```
[ ] y = furniture['Sales'].resample('MS').mean()
```

```
[ ] y['2017'::]
```

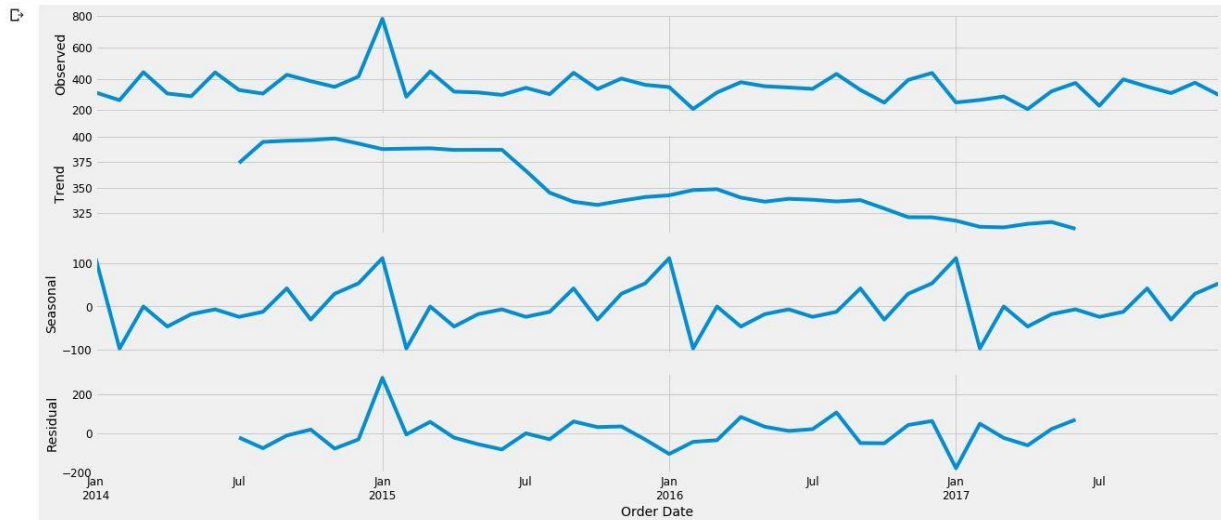
```
Order Date
2017-01-01    248.501333
2017-02-01    264.089900
2017-03-01    286.669600
2017-04-01    206.044502
2017-05-01    319.953928
2017-06-01    372.717386
2017-07-01    227.173500
2017-08-01    395.945487
2017-09-01    349.737422
2017-10-01    308.226313
2017-11-01    374.310253
2017-12-01    296.296857
Freq: MS, Name: Sales, dtype: float64
```

Visualizing Furniture Sales Time Series Data

```
[ ] y.plot(figsize=(15, 6))
plt.show()
```



```
[ ] from pylab import rcParams
rcParams['figure.figsize'] = 18, 8
decomposition = sm.tsa.seasonal_decompose(y, model='additive')
fig = decomposition.plot()
plt.show()
```



Time series forecasting with ARIMA

```
[ ] p = d = q = range(0, 2)
pdq = list(itertools.product(p, d, q))
seasonal_pdq = [(x[0], x[1], x[2], 12) for x in list(itertools.product(p, d, q))]
print('Examples of parameter combinations for Seasonal ARIMA...')
print('SARIMAX: {} x {}'.format(pdq[1], seasonal_pdq[1]))
print('SARIMAX: {} x {}'.format(pdq[1], seasonal_pdq[2]))
print('SARIMAX: {} x {}'.format(pdq[2], seasonal_pdq[3]))
print('SARIMAX: {} x {}'.format(pdq[2], seasonal_pdq[4]))
```

Examples of parameter combinations for Seasonal ARIMA...

SARIMAX: (0, 0, 1) x (0, 0, 1, 12)

SARIMAX: (0, 0, 1) x (0, 1, 0, 12)

SARIMAX: (0, 1, 0) x (0, 1, 1, 12)

SARIMAX: (0, 1, 0) x (1, 0, 0, 12)

```
[ ] for param in pdq:
    for param_seasonal in seasonal_pdq:
        try:
            mod = sm.tsa.statespace.SARIMAX(y,
                                             order=param,
                                             seasonal_order=param_seasonal,
                                             enforce_stationarity=False,
                                             enforce_invertibility=False)

            results = mod.fit()
            print('ARIMA{0}x{1}12 - AIC:{2}'.format(param, param_seasonal, results.aic))
        except:
            continue
```

```
➤ ARIMA(0, 0, 0)x(0, 0, 0, 12)12 - AIC:688.3112416991725
ARIMA(0, 0, 0)x(0, 0, 1, 12)12 - AIC:1447.3867340897314
ARIMA(0, 0, 0)x(0, 1, 0, 12)12 - AIC:426.20464898591723
ARIMA(0, 0, 0)x(1, 0, 0, 12)12 - AIC:452.8238602501692
/usr/local/lib/python3.6/dist-packages/statsmodels/base/model.py:512: ConvergenceWarning:
  "Check mle_retvals", ConvergenceWarning)
/usr/local/lib/python3.6/dist-packages/statsmodels/base/model.py:512: ConvergenceWarning:
  "Check mle_retvals", ConvergenceWarning)
ARIMA(0, 0, 0)x(1, 0, 1, 12)12 - AIC:1317.6576376825706
ARIMA(0, 0, 0)x(1, 1, 0, 12)12 - AIC:295.446456472235
ARIMA(0, 0, 1)x(0, 0, 0, 12)12 - AIC:641.124068066613
ARIMA(0, 0, 1)x(0, 0, 1, 12)12 - AIC:3179.1590875170373
ARIMA(0, 0, 1)x(0, 1, 0, 12)12 - AIC:416.5496453364607
/usr/local/lib/python3.6/dist-packages/statsmodels/base/model.py:512: ConvergenceWarning:
  "Check mle_retvals", ConvergenceWarning)
/usr/local/lib/python3.6/dist-packages/statsmodels/base/model.py:512: ConvergenceWarning:
  "Check mle_retvals", ConvergenceWarning)
```

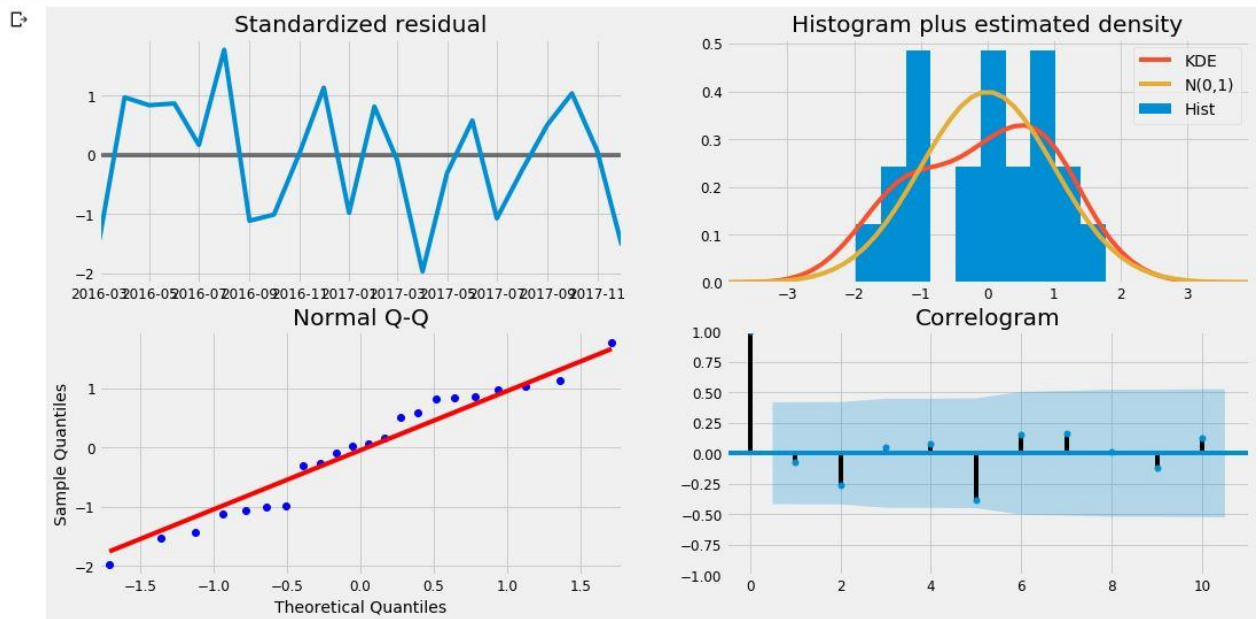
Fitting the ARIMA model

```
[ ] mod = sm.tsa.statespace.SARIMAX(y,
                                     order=(1, 1, 1),
                                     seasonal_order=(1, 1, 0, 12),
                                     enforce_stationarity=False,
                                     enforce_invertibility=False)

results = mod.fit()
print(results.summary().tables[1])
```

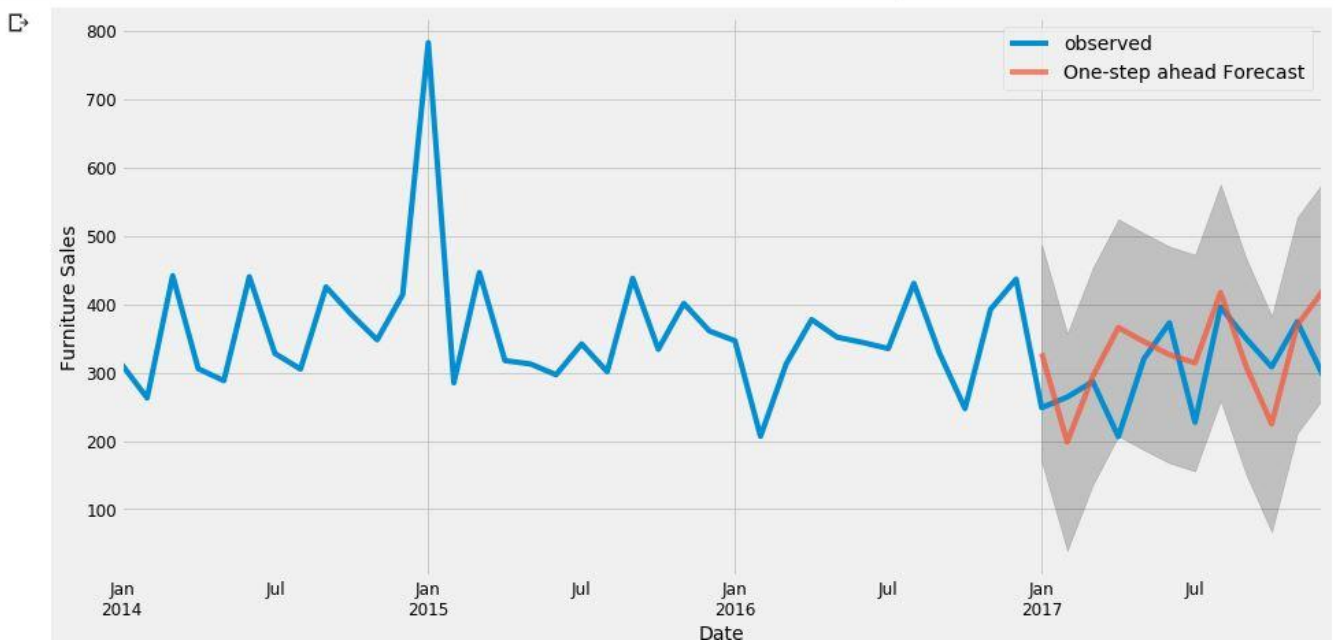
```
➤ =====
              coef      std err          z      P>|z|      [0.025      0.975]
-----
ar.L1          -0.0771         0.259      -0.298      0.766      -0.584         0.430
ma.L1          -1.0000         0.434      -2.305      0.021      -1.850        -0.150
ar.S.L12        -0.0007         0.029      -0.025      0.980      -0.057         0.056
sigma2        6323.9741      6.86e+05      9.22e+07      0.000      6323.974      6323.974
=====
```

```
[ ] results.plot_diagnostics(figsize=(16, 8))
plt.show()
```



Validating forecasts

```
[ ] pred = results.get_prediction(start=pd.to_datetime('2017-01-01'), dynamic=False)
pred_ci = pred.conf_int()
ax = y['2014:'].plot(label='observed')
pred.predicted_mean.plot(ax=ax, label='One-step ahead Forecast', alpha=.7, figsize=(14, 7))
ax.fill_between(pred_ci.index,
               pred_ci.iloc[:, 0],
               pred_ci.iloc[:, 1], color='k', alpha=.2)
ax.set_xlabel('Date')
ax.set_ylabel('Furniture Sales')
plt.legend()
plt.show()
```




```

y_forecasted = pred.predicted_mean
y_truth = y['2017-01-01':]
mse = ((y_forecasted - y_truth) ** 2).mean()
print('The Mean Squared Error of our forecasts is {}'.format(round(mse, 2)))

```

The Mean Squared Error of our forecasts is 5911.26

```

[ ] print('The Root Mean Squared Error of our forecasts is {}'.format(round(np.sqrt(mse), 2)))

```

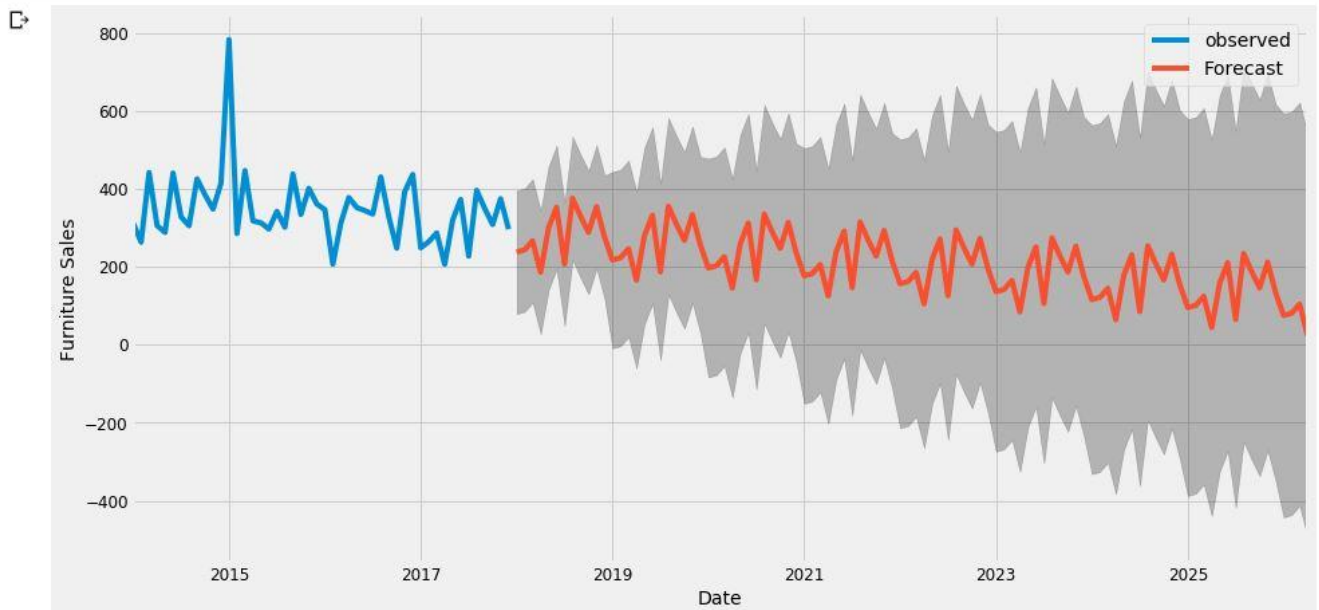
The Root Mean Squared Error of our forecasts is 76.88

Producing and visualizing forecasts

```

[ ] pred_uc = results.get_forecast(steps=100)
    pred_ci = pred_uc.conf_int()
    ax = y.plot(label='observed', figsize=(14, 7))
    pred_uc.predicted_mean.plot(ax=ax, label='Forecast')
    ax.fill_between(pred_ci.index,
                    pred_ci.iloc[:, 0],
                    pred_ci.iloc[:, 1], color='k', alpha=.25)
    ax.set_xlabel('Date')
    ax.set_ylabel('Furniture Sales')
    plt.legend()
    plt.show()

```



Time Series of Furniture vs. Office Supplies

```
[ ] furniture = df.loc[df['Category'] == 'Furniture']
office = df.loc[df['Category'] == 'Office Supplies']
furniture.shape, office.shape
```

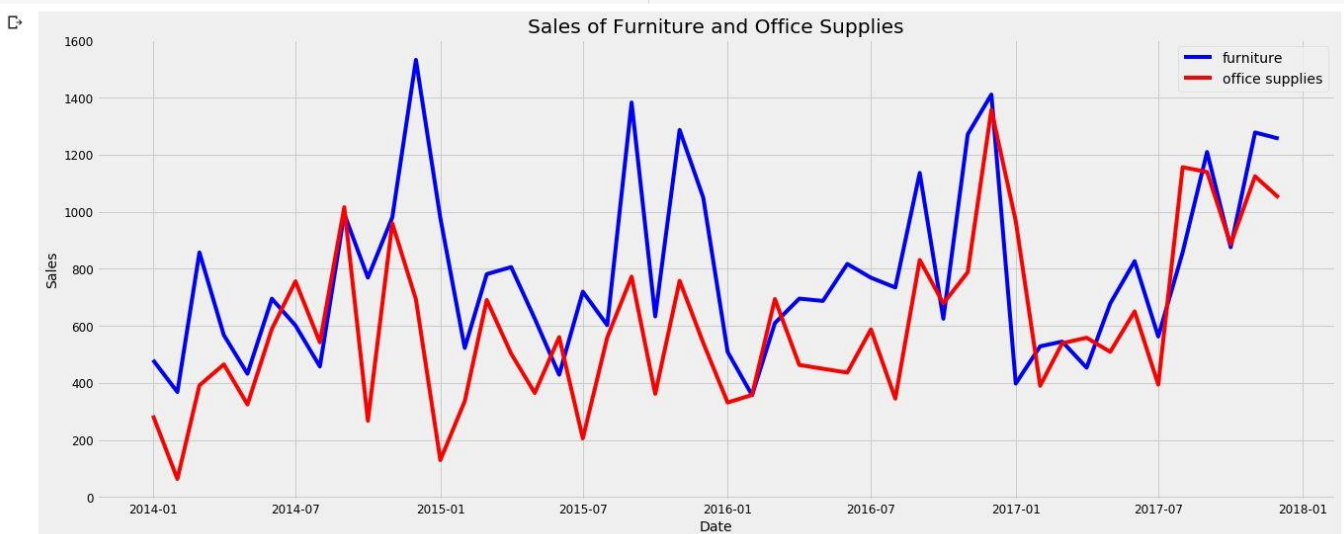
```
((2121, 21), (6026, 21))
```

Data Exploration

```
cols = ['Row ID', 'Order ID', 'Ship Date', 'Ship Mode', 'Customer ID', 'Customer Name', 'Segment', 'Country', 'City', 'State', 'Postal Code', 'Region', 'Product ID', 'Category', 'Sub-Category']
furniture.drop(cols, axis=1, inplace=True)
office.drop(cols, axis=1, inplace=True)
furniture = furniture.sort_values('Order Date')
office = office.sort_values('Order Date')
furniture = furniture.groupby('Order Date')['Sales'].sum().reset_index()
office = office.groupby('Order Date')['Sales'].sum().reset_index()
furniture = furniture.set_index('Order Date')
office = office.set_index('Order Date')
y_furniture = furniture['Sales'].resample('MS').mean()
y_office = office['Sales'].resample('MS').mean()
furniture = pd.DataFrame({'Order Date': y_furniture.index, 'Sales': y_furniture.values})
office = pd.DataFrame({'Order Date': y_office.index, 'Sales': y_office.values})
store = furniture.merge(office, how='inner', on='Order Date')
store.rename(columns={'Sales_x': 'furniture_sales', 'Sales_y': 'office_sales'}, inplace=True)
store.head()
```

	Order Date	furniture_sales	office_sales
0	2014-01-01	480.194231	285.357647
1	2014-02-01	367.931600	63.042588
2	2014-03-01	857.291529	391.176318
3	2014-04-01	567.488357	464.794750
4	2014-05-01	432.049188	324.346545

```
[ ] plt.figure(figsize=(20, 8))
plt.plot(store['Order Date'], store['furniture_sales'], 'b-', label = 'furniture')
plt.plot(store['Order Date'], store['office_sales'], 'r-', label = 'office supplies')
plt.xlabel('Date'); plt.ylabel('Sales'); plt.title('Sales of Furniture and Office Supplies')
plt.legend();
```



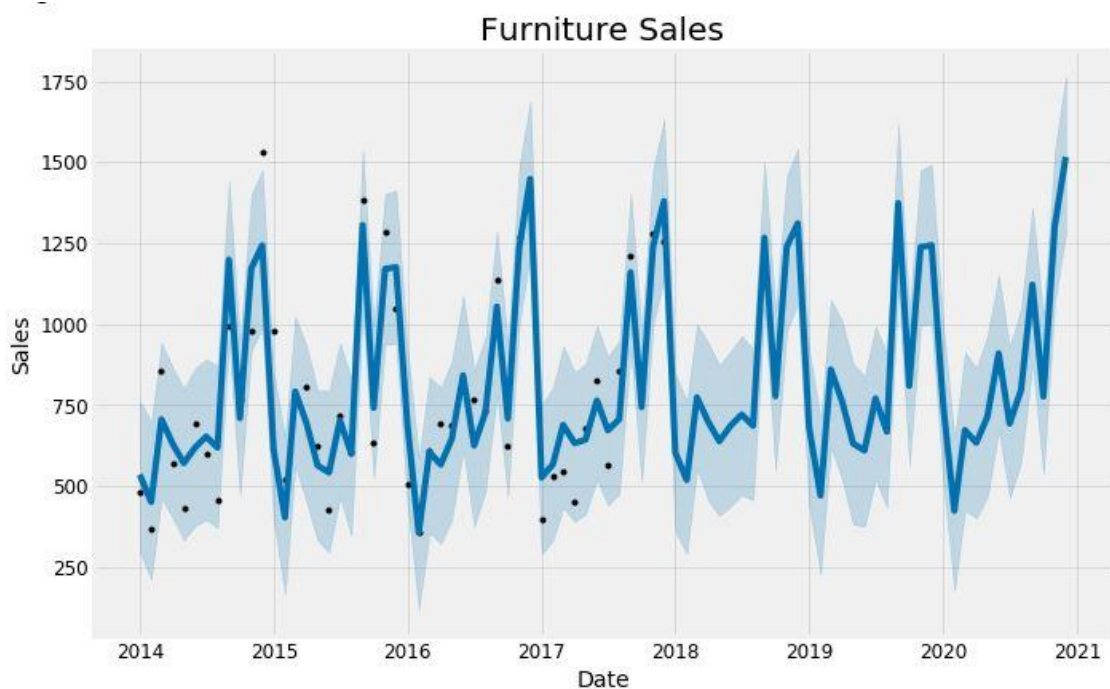
```
[ ] first_date = store.ix[np.min(list(np.where(store['office_sales'] > store['furniture_sales'])[0])), 'Order Date']
    print("Office supplies first time produced higher sales than furniture is {}".format(first_date.date()))
```

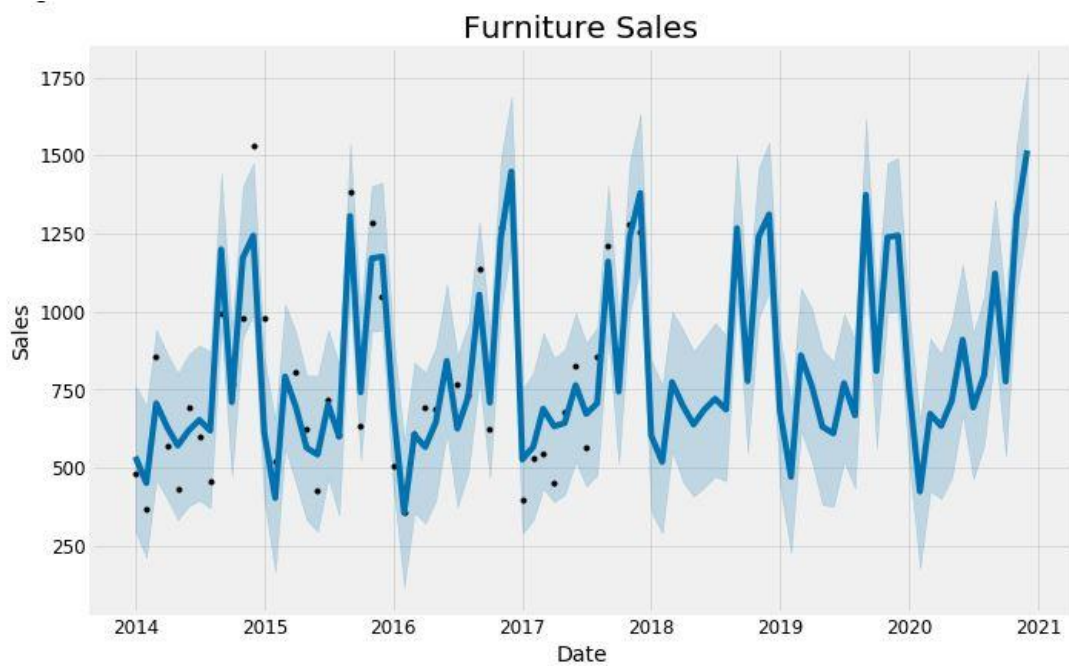
Office supplies first time produced higher sales than furniture is 2014-07-01.

Time Series Modeling with Prophet

```
[ ] from fbprophet import Prophet
    furniture = furniture.rename(columns={'Order Date': 'ds', 'Sales': 'y'})
    furniture_model = Prophet(interval_width=0.95)
    furniture_model.fit(furniture)
    office = office.rename(columns={'Order Date': 'ds', 'Sales': 'y'})
    office_model = Prophet(interval_width=0.95)
    office_model.fit(office)
    furniture_forecast = furniture_model.make_future_dataframe(periods=36, freq='MS')
    furniture_forecast = furniture_model.predict(furniture_forecast)
    office_forecast = office_model.make_future_dataframe(periods=36, freq='MS')
    office_forecast = office_model.predict(office_forecast)
    plt.figure(figsize=(18, 6))
    furniture_model.plot(furniture_forecast, xlabel = 'Date', ylabel = 'Sales')
    plt.title('Furniture Sales');
```

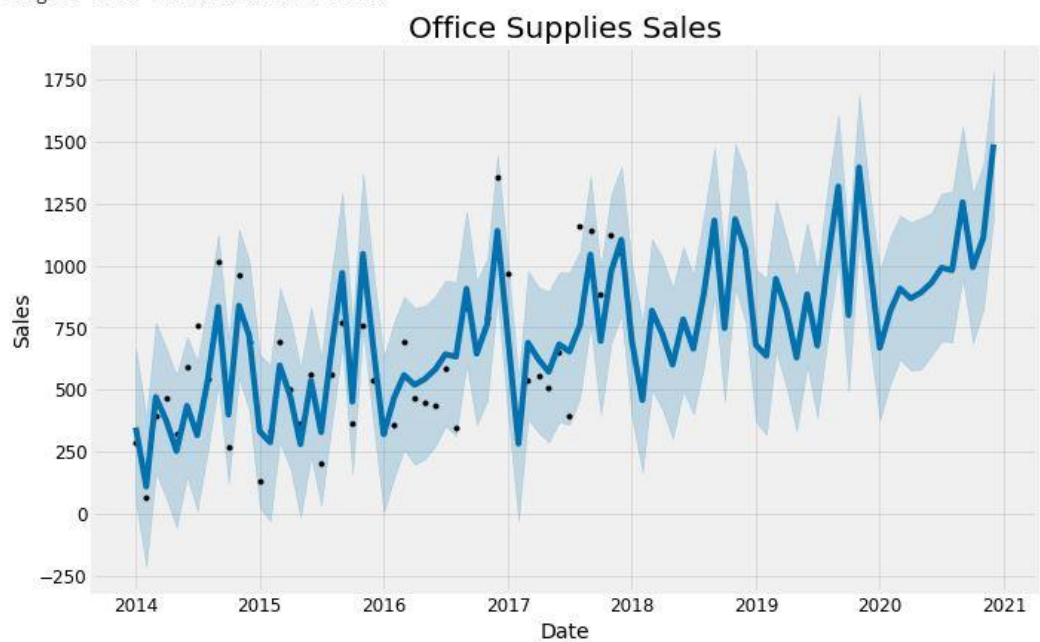
INFO:fbprophet:Disabling weekly seasonality. Run prophet with weekly_seasonality=True to override this.
 INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to override this.
 INFO:fbprophet:Disabling weekly seasonality. Run prophet with weekly_seasonality=True to override this.
 INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to override this.
 <Figure size 1296x432 with 0 Axes>





```
[ ] plt.figure(figsize=(18, 6))
office_model.plot(office_forecast, xlabel = 'Date', ylabel = 'Sales')
plt.title('Office Supplies Sales');
```

<Figure size 1296x432 with 0 Axes>



Compare Forecasts

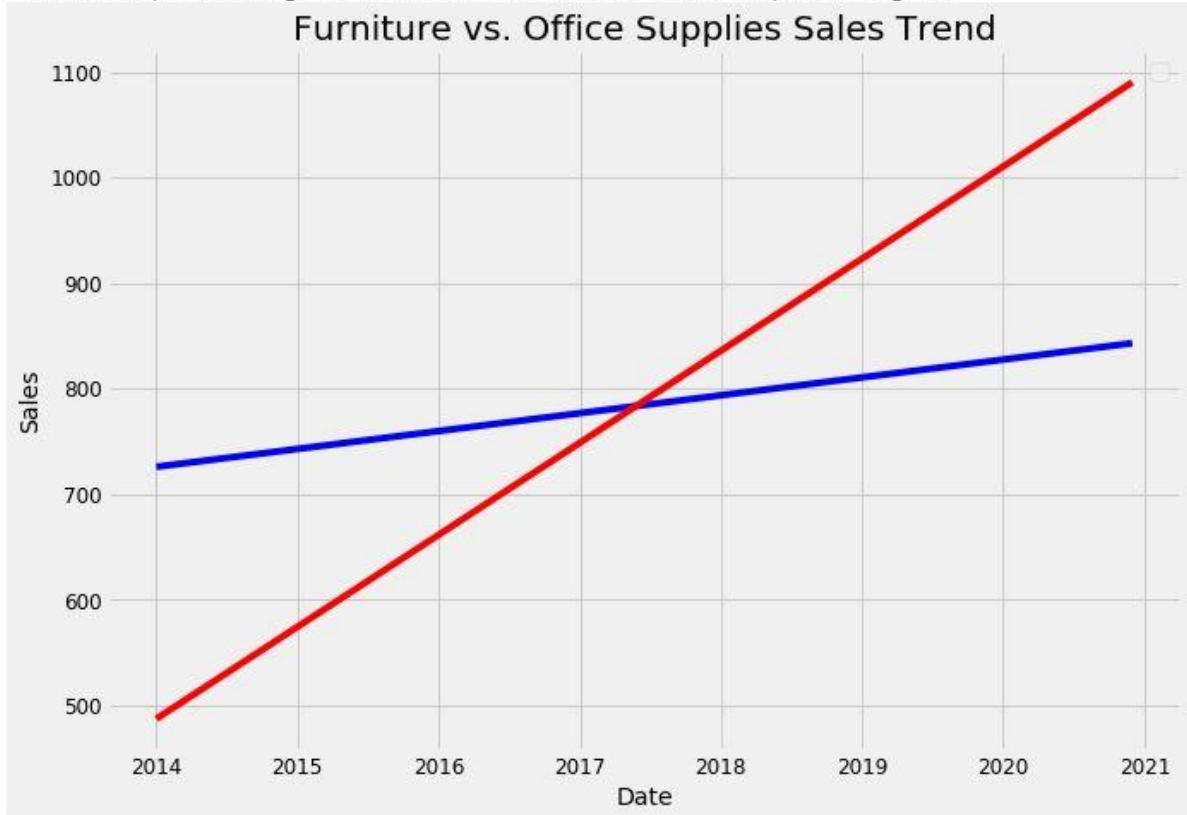
```
[ ] furniture_names = ['furniture_{}'.format(column) for column in furniture_forecast.columns]
office_names = ['office_{}'.format(column) for column in office_forecast.columns]
merge_furniture_forecast = furniture_forecast.copy()
merge_office_forecast = office_forecast.copy()
merge_furniture_forecast.columns = furniture_names
merge_office_forecast.columns = office_names
forecast = pd.merge(merge_furniture_forecast, merge_office_forecast, how = 'inner', left_on = 'furniture_ds', right_on = 'office_ds')
forecast = forecast.rename(columns={'furniture_ds': 'Date'}).drop('office_ds', axis=1)
forecast.head()
```

	Date	furniture_trend	furniture_yhat_lower	furniture_yhat_upper	furniture_trend_lower	furniture_trend_upper	furniture_additive_terms	furniture_additive_terms_lower	furniture_additive_terms_upper
0	2014-01-01	726.057713	295.119472	762.253540	726.057713	726.057713	-190.685662	-190.685662	-190.685662
1	2014-02-01	727.494023	212.847567	698.537161	727.494023	727.494023	-276.377703	-276.377703	-276.377703
2	2014-03-01	728.791335	462.362807	941.383328	728.791335	728.791335	-22.389755	-22.389755	-22.389755
3	2014-04-01	730.227645	400.268119	865.674796	730.227645	730.227645	-100.141158	-100.141158	-100.141158
4	2014-05-01	731.617622	334.061054	801.916278	731.617622	731.617622	-160.815662	-160.815662	-160.815662

Trend and Forecast Visualization

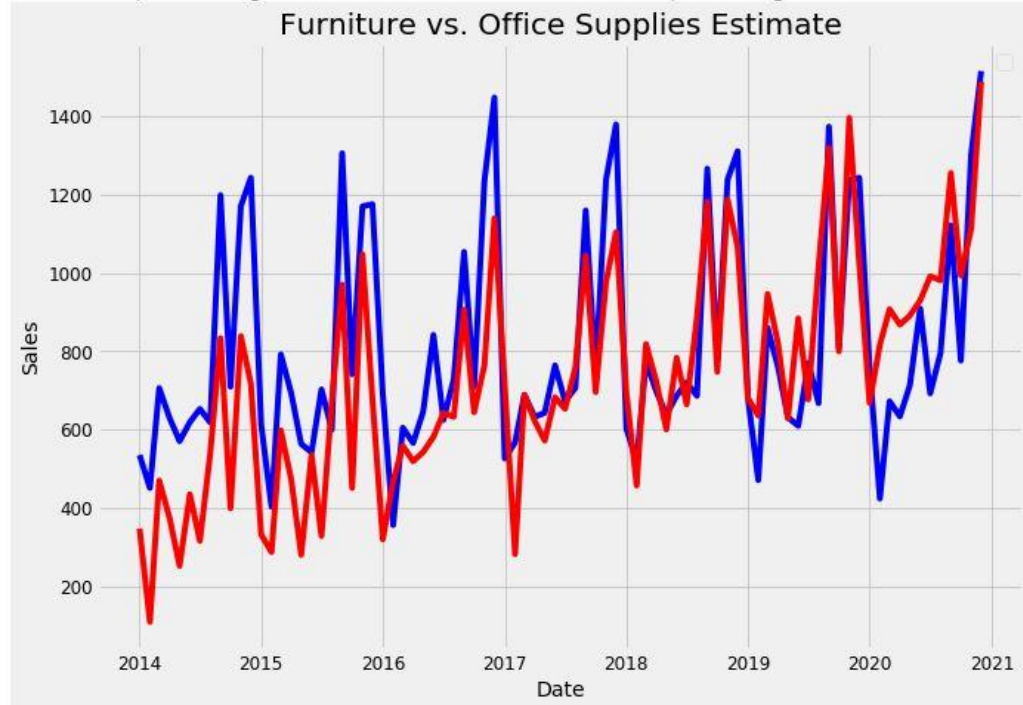
```
[ ] plt.figure(figsize=(10, 7))
plt.plot(forecast['Date'], forecast['furniture_trend'], 'b-')
plt.plot(forecast['Date'], forecast['office_trend'], 'r-')
plt.legend(); plt.xlabel('Date'); plt.ylabel('Sales')
plt.title('Furniture vs. Office Supplies Sales Trend');
```

WARNING:matplotlib.legend.No handles with labels found to put in legend.



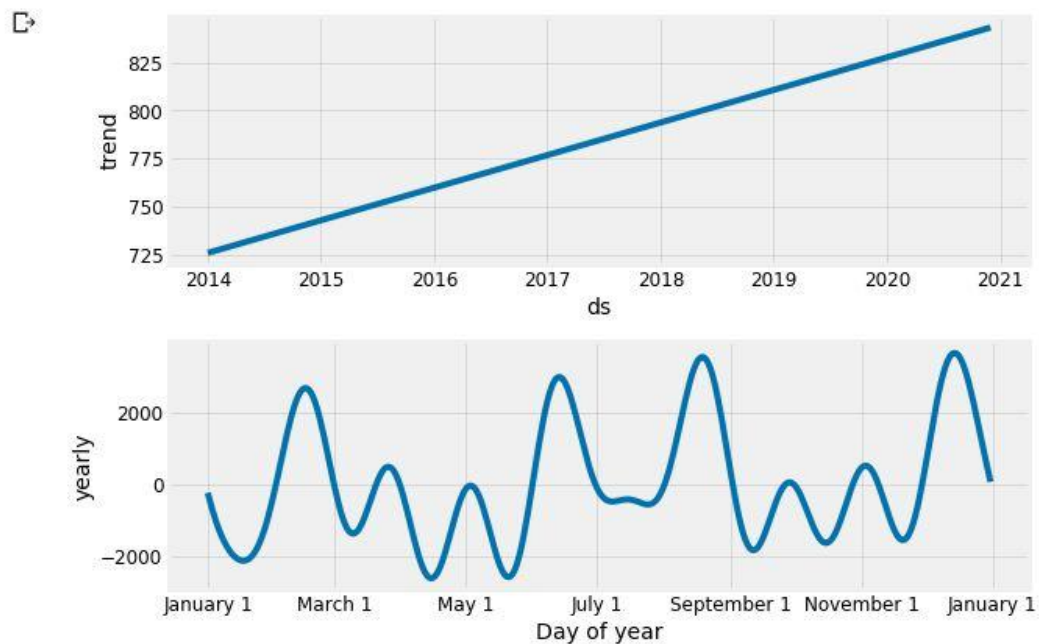
```
[ ] plt.figure(figsize=(10, 7))
plt.plot(forecast['Date'], forecast['furniture_yhat'], 'b-')
plt.plot(forecast['Date'], forecast['office_yhat'], 'r-')
plt.legend(); plt.xlabel('Date'); plt.ylabel('Sales')
plt.title('Furniture vs. Office Supplies Estimate');
```

WARNING:matplotlib.legend:No handles with labels found to put in legend.

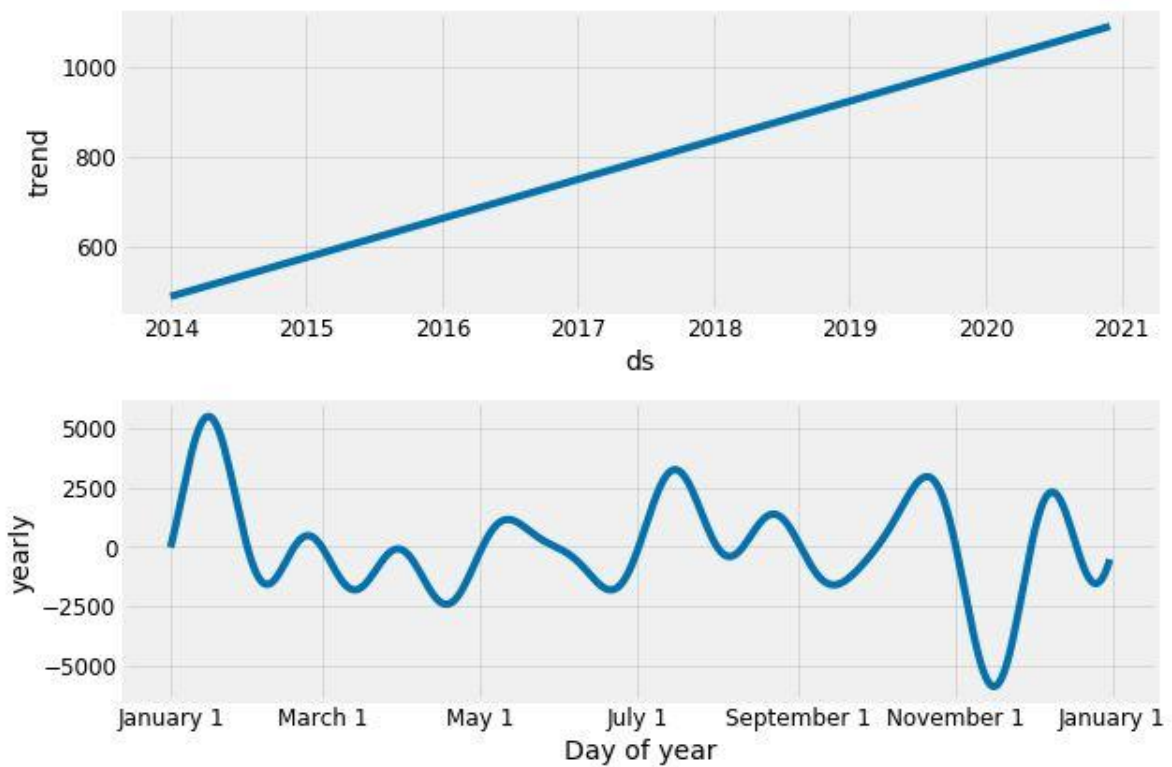


Trends and Patterns

```
[ ] furniture_model.plot_components(furniture_forecast);
```



```
[ ] office_model.plot_components(office_forecast);
```



```
[ ]
```

Chapter 4

REFLECTION NOTES

Working at Tech Fortune Technologies enabled me to understand some important aspects of being successful as an Application Developer and Data Analyst. The most important thing is that good communication is extremely important for developing the best version of the required product. It is also important for an individual to have the ability of multi-tasking during the working hours. This will make one flexible in the workplace. It is very important to maintain punctuality in order to be successful. Another key to success entails that an individual should have a professional attitude in order to fit into the competitive market. At the same time, one should have the ability to network with people in the chosen professional field. The amount of experience gained during the internship period in Tech Fortune Technologies was immense.

Reflecting back, I was a person without any experience in Application Development. But, this project provided me with an opportunity to learn about all the new concepts in the field of Application Development. Concepts related to Machine Learning that seemed difficult before working on this project seems very easy to accomplish now. The entire office environment was new, and this project helped me develop team spirit to work as a team instead of as an individual. This project helped me understand the importance working under a deadline. I was able to learn how to handle stress at times when the desired objective seemed distant. I learnt how to interact with my fellow colleagues at higher positions in the corporate environment.

All in all, this internship made me grow into the corporate world and helped me blend into the industry. It helped me boost my confidence to speak efficiently and fluently without being afraid of any person and express my own views without any trouble.

Chapter 5

CONCLUSION

Tech Fortune Technologies is a leader in the integration of IT and services technologies and brings more than 5 years of expertise in technological innovation to provide solutions for empowering people, businesses and society. Headquartered in India, Tech Fortune started operations in India in the 2013, accelerating its growth through the expansion of business to global markets. Tech Fortune expanded its business to provide services in Data Analysis, Software Development, Digital Transformation services etc. With its Centre of Excellence for analytics platform solutions, big data, mobile and software development, Tech Fortune offers innovative new services and solutions for India. Tech Fortune operates across Karnataka with offices in Bangalore (head office), Huballi, Belagavi and Vijayapura.

The main task that was given during this tenure of 6 weeks was to gain a first-hand experience of the Machine Learning and its Models. Along with the technical skills the internship program also provided insights on how a particular IT company works and how the process of project planning and management takes place in Tech Fortune Technologies. Working under the Data Analytics department also helped get to know a lot of information about how the department works and what technologies they use. This internship has successfully helped in bridging the gap between the curriculum and the industry.

EXECUTIVE SUMMARY

Tech Fortune Technologies is a leader in the integration of IT and services technologies and brings more than 5 years of expertise in technological innovation to provide solutions for empowering people, businesses and society. Headquartered in India, Tech Fortune started operations in India in the 2013, accelerating its growth through the expansion of business to global markets. Tech Fortune expanded its business to provide services in Data Analysis, Software Development, Digital Transformation services etc. With its Centre of Excellence for analytics platform solutions, big data, mobile and software development, Tech Fortune offers innovative new services and solutions for India. Tech Fortune operates across Karnataka with offices in Bangalore (head office), Huballi, Belagavi and Vijayapura.

The main task that was given during this tenure of 6 weeks was to gain a first-hand experience of the Machine Learning and its Models. Along with the technical skills the internship program also provided insights on how a particular IT company works and how the process of project planning and management takes place in Tech Fortune Technologies. Working under the Data Analytics department also helped get to know a lot of information about how the department works and what technologies they use. This internship has successfully helped in bridging the gap between the curriculum and the industry.