Sales Data Forecasting

August 8, 2023

1 Sales Data Forecasting

Sale data forecasting based on 6 months of sales data using SQL and the Prophet python package.

- Access Data Repository HERE

[54]:
```
# import packages

import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import sklearn as sklearn
```

[7]:
```
# import data

df=pd.read_csv(r"/Users/scipio/Downloads/Faux_Sales_Data.csv")
```

[18]:
```
#Making Month an index

df.set_index('Month')
```

[18]:

Month	Sales	MoM_Change	Avg_MoM_Change
1	1000	0	250.0
2	3000	2000	250.0
3	500	-2500	250.0
4	4000	3500	250.0
5	700	-3300	250.0
6	2500	1800	250.0

[19]:
```
# df properties

df.shape
```

[19]:
```
(6, 4)
```

[25]:
```
# statistical metrics of sales column
df['Sales'].describe()
```

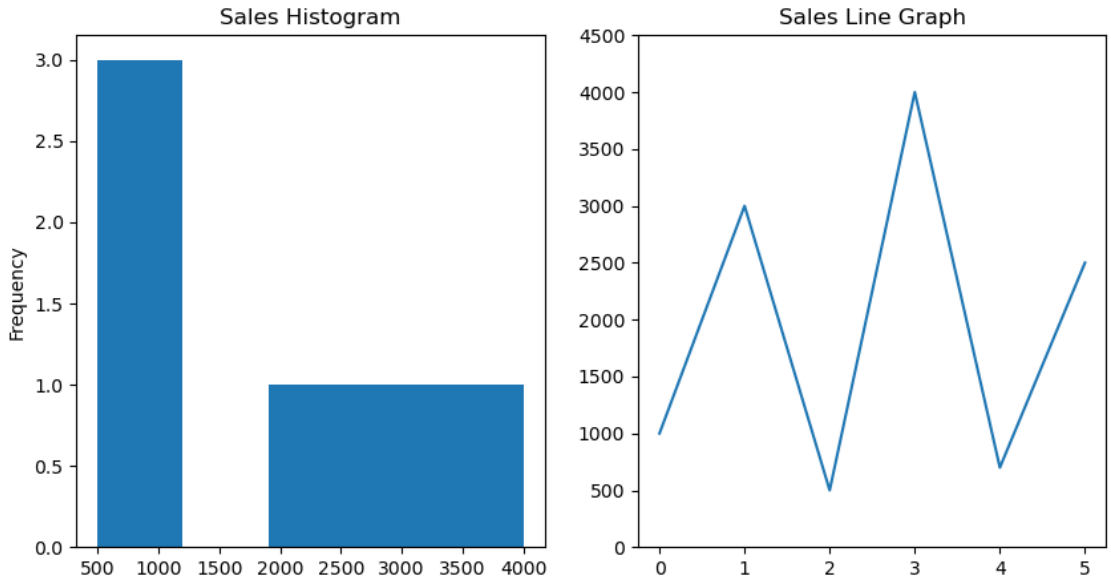
[25]:
```
count       6.000000
mean     1950.000000
std      1426.534262
min       500.000000
25%       775.000000
50%      1750.000000
75%      2875.000000
max      4000.000000
Name: Sales, dtype: float64
```

[37]:
```
#Sales data visualizations

fig,axs = plt.subplots(nrows = 1, ncols =2, figsize = (10,5))

df['Sales'].plot(kind = 'hist', bins = 5, ax = axs[0], title = 'Sales␣
↪Histogram')
df['Sales'].plot(kind = 'line', figsize = (10,5), ylim = (0,4500), ax = axs␣
↪[1], title = 'Sales Line Graph')
```

[37]:
```
<Axes: title={'center': 'Sales Line Graph'}>
```



1.0.1 Sales Data Distribution

There is an uneven distribution in the sales data. So, the median, not the mean, 1750 is indicative of the typical month of sales. Additionally, there is a positive trend in the dataset, with a +250 increase month sales on average.

[142]:
```
from prophet import Prophet
```

[143]:
```
df1 = pd.read_csv(r"/Users/scipio/Downloads/Prophet_Sales_Data.csv")

df1['ds'] = df1['Date_']

df1['y'] = df1['Sales']

drop_columns = ['Sales','MoM_Change','Avg_MoM_Change','Date_']

df1 = df1.drop(columns = drop_columns)

df1
```

[143]:

	ds	y
0	2023-01-01	1000
1	2023-02-01	3000
2	2023-03-01	500
3	2023-04-01	4000
4	2023-05-01	700
5	2023-06-01	2500

[144]:
```
model = Prophet()
model.fit(df1)
```

```
20:26:32 - cmdstanpy - INFO - Chain [1] start processing
20:26:32 - cmdstanpy - INFO - Chain [1] done processing
```

[144]:
```
<prophet.forecaster.Prophet at 0x7fefd0932d10>
```

[145]:
```
# creating 6 months intervals into the future

future = model.make_future_dataframe(periods = 6, freq = 'M')

future.tail()
```

[145]:

	ds
7	2023-07-31
8	2023-08-31
9	2023-09-30
10	2023-10-31
11	2023-11-30

[147]:
```
forecast = model.predict(future)

round(forecast[['ds','yhat','yhat_lower','yhat_upper']],2)
```

[147]:

	ds	yhat	yhat_lower	yhat_upper
0	2023-01-01	1642.36	70.83	3230.09
1	2023-02-01	1768.83	135.00	3366.40

2	2023-03-01	1883.06	306.98	3364.91
3	2023-04-01	2009.53	417.69	3546.83
4	2023-05-01	2131.92	567.70	3813.37
5	2023-06-01	2258.39	645.40	3807.11
6	2023-06-30	2376.70	860.71	3982.89
7	2023-07-31	2503.17	826.32	4112.36
8	2023-08-31	2629.63	1171.35	4175.32
9	2023-09-30	2752.02	1107.82	4160.46
10	2023-10-31	2878.49	1432.22	4447.93
11	2023-11-30	3000.88	1437.19	4650.56

[150]:
```
fig1 = model.plot(forecast, xlabel = 'Date', ylabel = 'Sale Amount')
```