

# Ordinal\_Encoding\_Technique

March 18, 2023

```
[1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings("ignore")
```

```
[2]: df = pd.read_csv('online_profit.csv')
```

```
[3]: df.head()
```

```
[3]:
```

	Marketing Spend	Administration	Transport	Area	Profit
0	114523.61	136897.80	471784.10	Dhaka	192261.83
1	NaN	151377.59	443898.53	Ctg	191792.06
2	153441.51	101145.55	407934.54	NaN	191050.39
3	144372.41	118671.85	383199.62	Dhaka	182901.99
4	142107.34	91391.77	366168.42	Rangpur	166187.94

```
[4]: df.isnull().sum()
```

```
[4]:
```

Marketing Spend	2
Administration	0
Transport	0
Area	3
Profit	1

dtype: int64

```
[5]: mean = df['Marketing Spend'].mean()
```

```
[6]: mean
```

```
[6]: 70691.35312500001
```

```
[7]: df['Marketing Spend'] = df['Marketing Spend'].fillna(mean)
```

```
[8]: df.head()
```

```
[8]: Marketing Spend Administration Transport Area Profit
0 114523.610000 136897.80 471784.10 Dhaka 192261.83
1 70691.353125 151377.59 443898.53 Ctg 191792.06
2 153441.510000 101145.55 407934.54 NaN 191050.39
3 144372.410000 118671.85 383199.62 Dhaka 182901.99
4 142107.340000 91391.77 366168.42 Rangpur 166187.94
```

```
[9]: df['Area'] = df['Area'].fillna(method='ffill')
```

```
[10]: median = df['Profit'].median()
```

```
[11]: median
```

```
[11]: 107404.34
```

```
[12]: df['Profit'] = df['Profit'].fillna(median)
```

```
[13]: df.head()
```

```
[13]: Marketing Spend Administration Transport Area Profit
0 114523.610000 136897.80 471784.10 Dhaka 192261.83
1 70691.353125 151377.59 443898.53 Ctg 191792.06
2 153441.510000 101145.55 407934.54 Ctg 191050.39
3 144372.410000 118671.85 383199.62 Dhaka 182901.99
4 142107.340000 91391.77 366168.42 Rangpur 166187.94
```

```
[14]: from sklearn.preprocessing import OrdinalEncoder
```

```
[15]: df.Area.unique()
```

```
[15]: array(['Dhaka', 'Ctg', 'Rangpur'], dtype=object)
```

```
[16]: city_list = ['Dhaka', 'Ctg', 'Rangpur']
```

```
[17]: ordinal = OrdinalEncoder(categories=[city_list])
```

```
[18]: encoded_values = ordinal.fit_transform(df[["Area"]])
```

```
[19]: new_area = pd.DataFrame(encoded_values, columns=['Area'])
```

```
[20]: df.Area = new_area
```

```
[21]: df.head()
```

```
[21]: Marketing Spend Administration Transport Area Profit
0 114523.610000 136897.80 471784.10 0.0 192261.83
1 70691.353125 151377.59 443898.53 1.0 191792.06
2 153441.510000 101145.55 407934.54 1.0 191050.39
```

```

3    144372.410000    118671.85  383199.62    0.0  182901.99
4    142107.340000    91391.77  366168.42    2.0  166187.94

```

```
[22]: new_df = df.drop("Area",axis=1)
```

```
[23]: new_df.head()
```

```

[23]:   Marketing Spend  Administration  Transport    Profit
0    114523.610000    136897.80  471784.10  192261.83
1     70691.353125    151377.59  443898.53  191792.06
2    153441.510000    101145.55  407934.54  191050.39
3    144372.410000    118671.85  383199.62  182901.99
4    142107.340000     91391.77  366168.42  166187.94

```

```
[24]: df = pd.concat([new_df,df.Area],axis=1)
```

```
[25]: df.head()
```

```

[25]:   Marketing Spend  Administration  Transport    Profit  Area
0    114523.610000    136897.80  471784.10  192261.83    0.0
1     70691.353125    151377.59  443898.53  191792.06    1.0
2    153441.510000    101145.55  407934.54  191050.39    1.0
3    144372.410000    118671.85  383199.62  182901.99    0.0
4    142107.340000     91391.77  366168.42  166187.94    2.0

```

```
[26]: x = df.drop(['Profit'], axis=1)
```

```
[27]: y = df['Profit']
```

```
[28]: x.head()
```

```

[28]:   Marketing Spend  Administration  Transport  Area
0    114523.610000    136897.80  471784.10    0.0
1     70691.353125    151377.59  443898.53    1.0
2    153441.510000    101145.55  407934.54    1.0
3    144372.410000    118671.85  383199.62    0.0
4    142107.340000     91391.77  366168.42    2.0

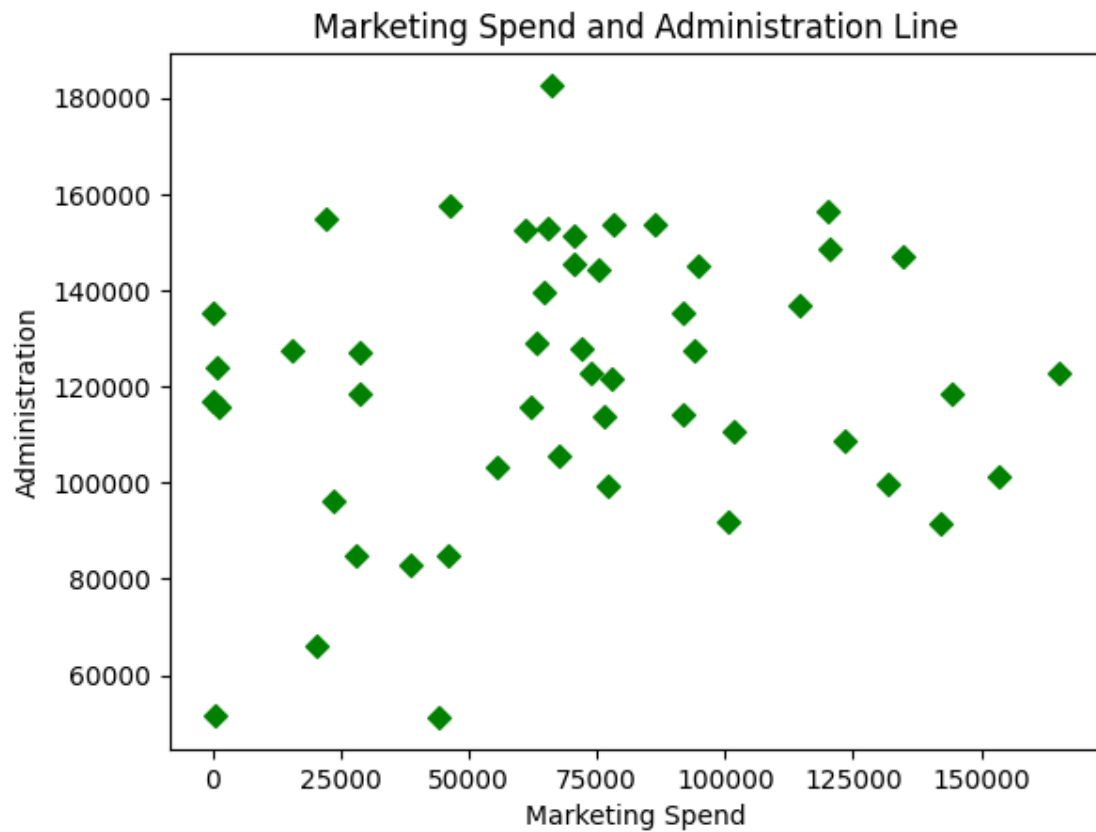
```

```

[29]: plt.title("Marketing Spend and Administration Line")
      plt.xlabel("Marketing Spend")
      plt.ylabel("Administration")
      plt.scatter(df['Marketing Spend'],df['Administration'],marker="D",color="Green")

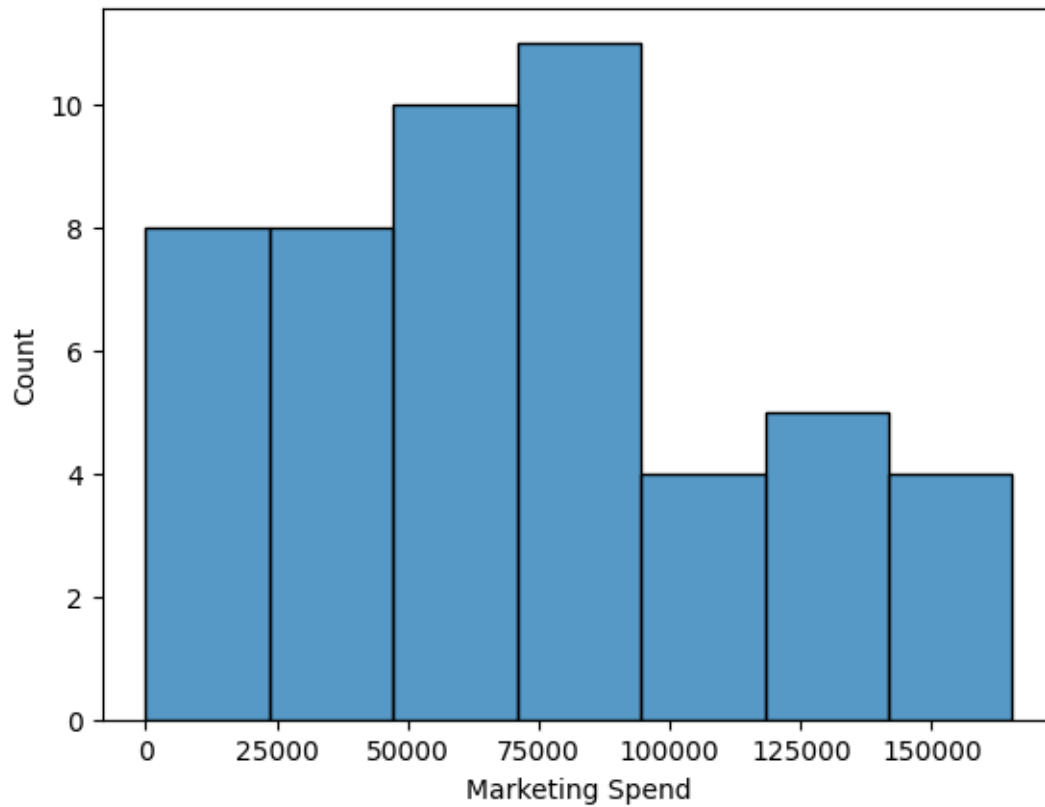
```

```
[29]: <matplotlib.collections.PathCollection at 0x291d71a5360>
```



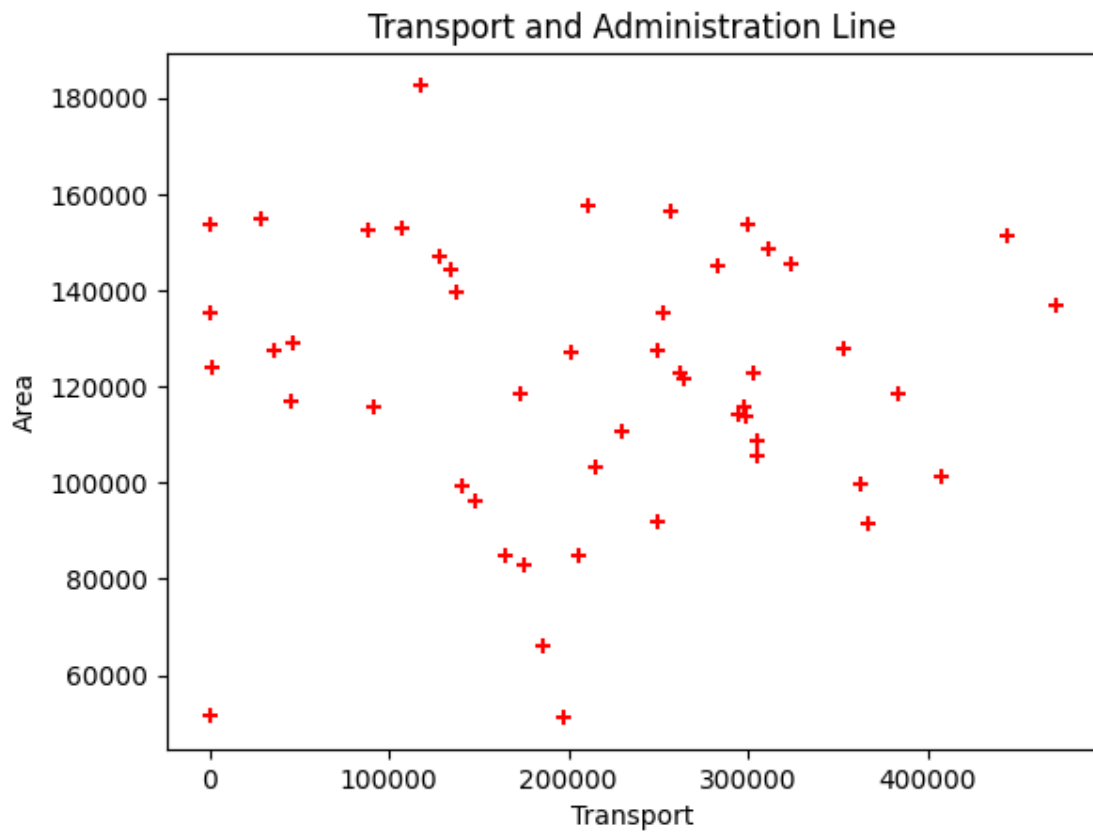
```
[30]: sns.histplot(df['Marketing Spend'])
```

```
[30]: <AxesSubplot: xlabel='Marketing Spend', ylabel='Count'>
```



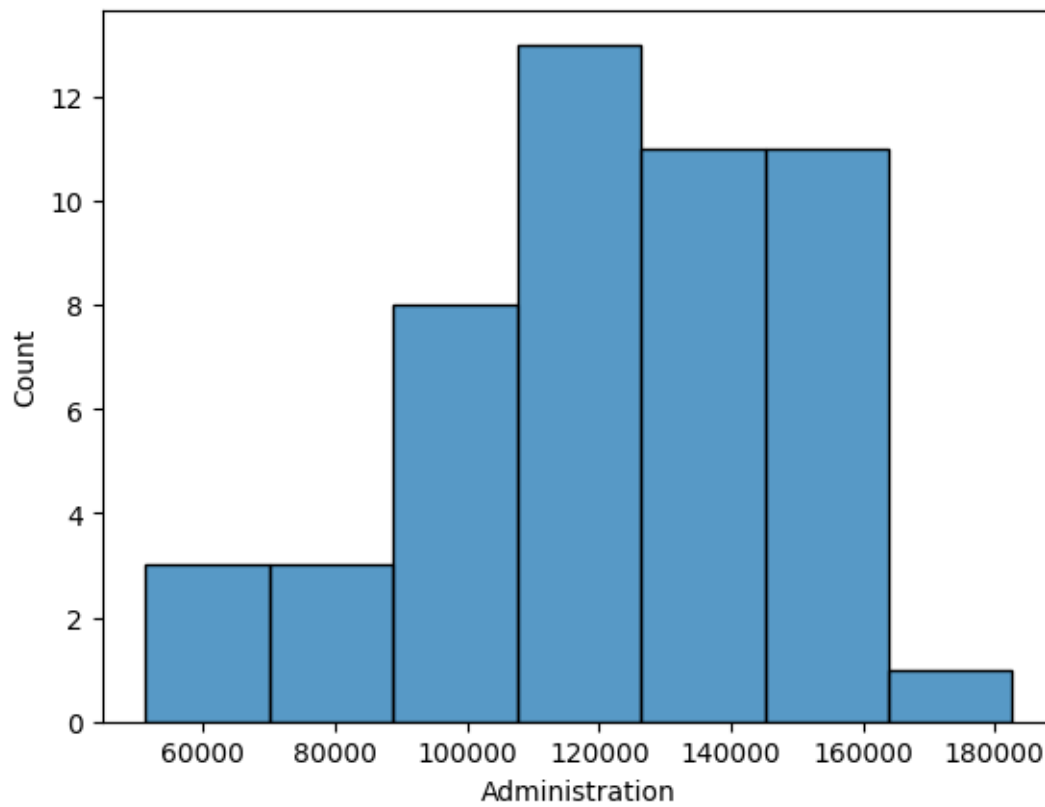
```
[31]: plt.title("Transport and Administration Line")  
      plt.xlabel("Transport")  
      plt.ylabel("Area")  
      plt.scatter(df['Transport'],df['Administration'],marker="+",color="Red")
```

```
[31]: <matplotlib.collections.PathCollection at 0x291d7390c10>
```



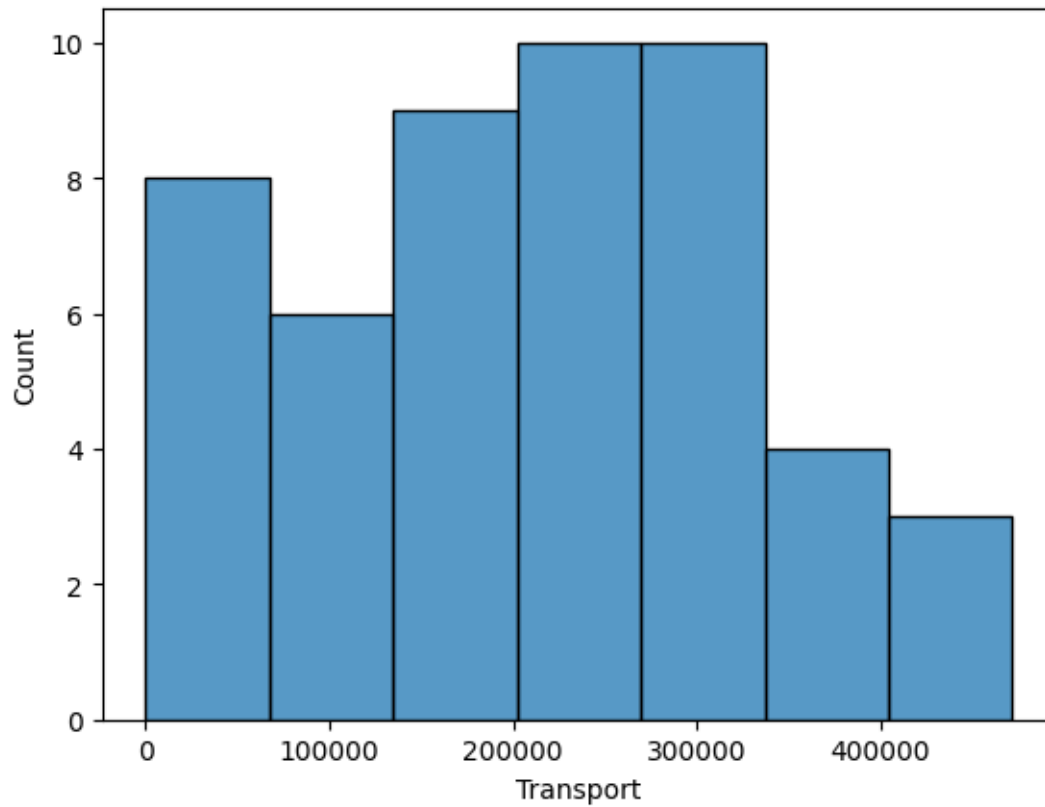
```
[32]: sns.histplot(df['Administration'])
```

```
[32]: <AxesSubplot: xlabel='Administration', ylabel='Count'>
```



```
[33]: sns.histplot(df['Transport'])
```

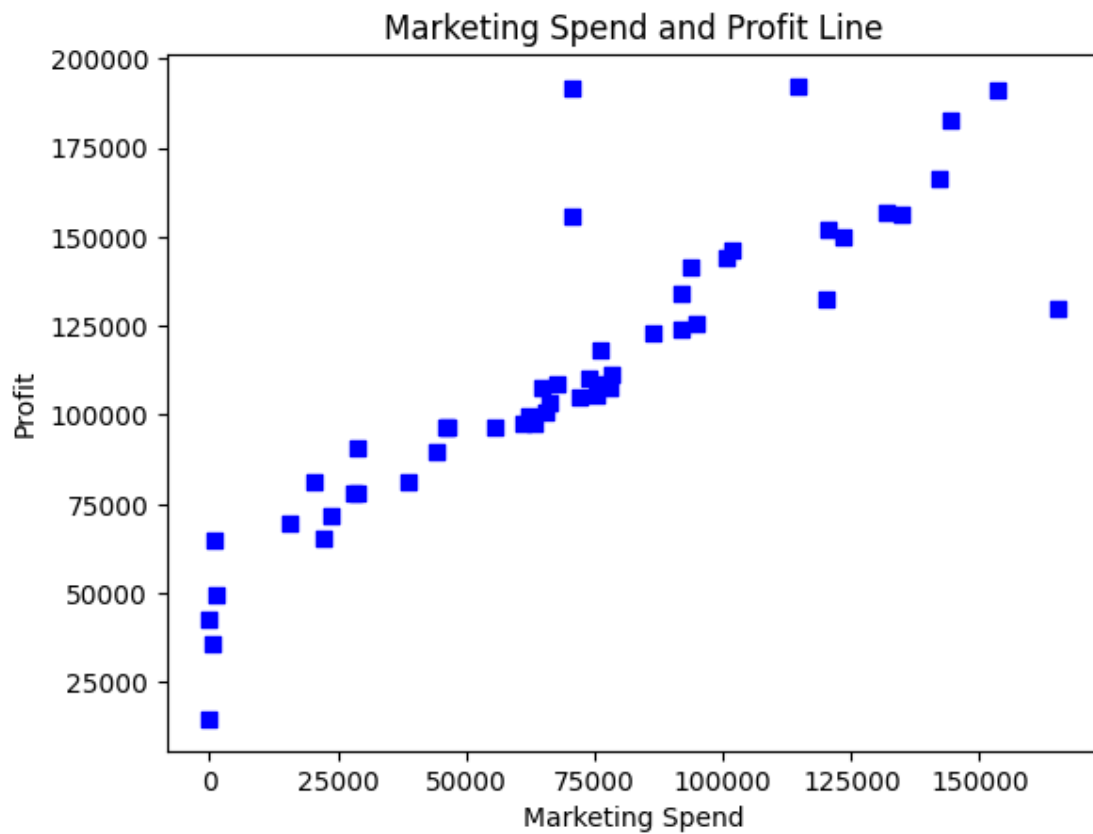
```
[33]: <AxesSubplot: xlabel='Transport', ylabel='Count'>
```



```
[34]: plt.title("Marketing Spend and Profit Line")
plt.xlabel("Marketing Spend")
plt.ylabel("Profit")
plt.scatter(df['Marketing Spend'],df['Profit'],marker="s",color="Blue")
```

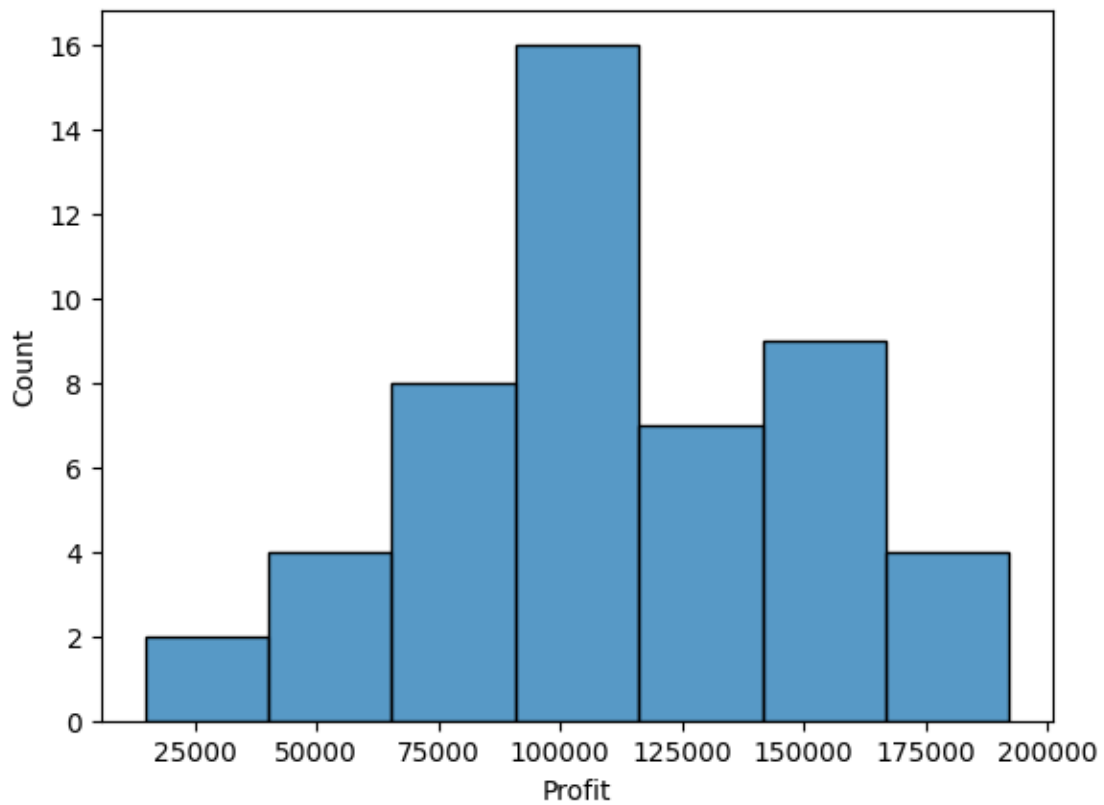
```
[34]: <matplotlib.collections.PathCollection at 0x291d95536d0>
```





```
[35]: sns.histplot(df['Profit'])
```

```
[35]: <AxesSubplot: xlabel='Profit', ylabel='Count'>
```



```
[36]: from sklearn.model_selection import train_test_split
```

```
[37]: xtrain, xtest, ytrain, ytest = train_test_split(x,y,train_size=.  
↪70,random_state=42)
```

```
[38]: xtrain.shape
```

```
[38]: (35, 4)
```

```
[39]: xtest.shape
```

```
[39]: (15, 4)
```

```
[40]: ytrain.shape
```

```
[40]: (35,)
```

```
[41]: ytest.shape
```

```
[41]: (15,)
```

```
[42]: from sklearn.linear_model import LinearRegression
```

```
[43]: reg = LinearRegression()
```

```
[44]: reg.fit(xtrain,ytrain)
```

```
[44]: LinearRegression()
```

```
[45]: ytest
```

```
[45]: 13      134307.35
      39      81005.76
      30      99937.59
      45      64926.08
      17     125370.37
      48      35673.41
      26     105733.54
      25     107404.34
      32      97427.84
      19     122776.86
      12     141585.52
      4      166187.94
      37      89949.14
      8      152211.77
      3      182901.99
      Name: Profit, dtype: float64
```

```
[46]: reg.score(xtest.values,ytest)
```

```
[46]: 0.8726448123190611
```

```
[47]: reg.coef_
```

```
[47]: array([ 5.60631094e-01,  1.67194619e-01,  1.49138930e-01, -4.95687455e+02])
```

```
[48]: reg.intercept_
```

```
[48]: 18213.515106813546
```

```
[49]: reg.predict([[142107.34,91391.77,366168.42,1]])
```

```
[49]: array([167277.79952355])
```