Edward Odhiambo

Github: https://github.com/EddOdhiambo (https://github.com/EddOdhiambo) Linkedin: https://www.linkedin.com/in/edward-odhiambo-656225276/ (https://www.linkedin.com/in/edward-odhiambo-656225276/)

Exploratory Data Analysis in Python

In [24]:
```
import pandas as pd
import numpy as np
from sklearn import preprocessing
from sklearn.preprocessing import Normalizer
from sklearn.preprocessing import Binarizer
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt
```

In [4]:
```
#importing data into the notebook
Data = pd.read_csv('C:/Edward/hotel_revenue.csv')
```

In [6]:
```
Data
```

Out[6]:

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arrival_date_week_number	arrival_date_day_of_month
0	Resort Hotel	1	85	2018	July	27	1
1	Resort Hotel	1	75	2018	July	27	1
2	Resort Hotel	1	23	2018	July	27	1
3	Resort Hotel	1	60	2018	July	27	1
4	Resort Hotel	1	96	2018	July	27	1
...	...	...	...	...	...	...	...
21991	City Hotel	1	24	2018	December	53	27
21992	City Hotel	1	1	2018	December	53	27
21993	City Hotel	1	66	2018	December	53	28
21994	City Hotel	1	54	2018	December	53	30
21995	City Hotel	1	54	2018	December	53	30

21996 rows × 32 columns

In [7]:
```
#reading first five entries of the dataset
Data.head(5)
```

Out[7]:

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arrival_date_week_number	arrival_date_day_of_month	sta
0	Resort Hotel	1	85	2018	July	27	1	
1	Resort Hotel	1	75	2018	July	27	1	
2	Resort Hotel	1	23	2018	July	27	1	
3	Resort Hotel	1	60	2018	July	27	1	
4	Resort Hotel	1	96	2018	July	27	1	

5 rows × 32 columns

In [8]:
```
Data.tail(5)
```

Out[8]:

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arrival_date_week_number	arrival_date_day_of_month
21991	City Hotel	1	24	2018	December	53	27
21992	City Hotel	1	1	2018	December	53	27
21993	City Hotel	1	66	2018	December	53	28
21994	City Hotel	1	54	2018	December	53	30
21995	City Hotel	1	54	2018	December	53	30

5 rows × 32 columns

MEASURES OF CENTRAL TENDENCY

We will select univariate data, ie stays_in_weekend_nights

In [9]:
```
#reading leadtime data
lead_time = Data['lead_time']
print(lead_time)
```

```
0        85
1        75
2        23
3        60
4        96
        ..
21991    24
21992     1
21993    66
21994    54
21995    54
Name: lead_time, Length: 21996, dtype: int64
```

In [10]:
```
#reading stay in weekend nights data
stays_in_weekend_nights = Data['stays_in_weekend_nights']
print(stays_in_weekend_nights)
```

```
0        0
1        0
2        0
3        2
4        2
       ..
21991    2
21992    2
21993    0
21994    1
21995    1
Name: stays_in_weekend_nights, Length: 21996, dtype: int64
```

In [11]:
```
stays_in_week_nights = Data["stays_in_week_nights"]
print(stays_in_week_nights)
```

```
0        3
1        3
2        4
3        5
4        8
        ..
21991    1
21992    4
21993    5
21994    4
21995    4
Name: stays_in_week_nights, Length: 21996, dtype: int64
```

1.MEAN

In [12]:
```
#Lead time mean
Average_lead_time = lead_time.mean()
print("Average_lead_time:",Average_lead_time)
Average_stays_in_weekend_nights = stays_in_weekend_nights.mean()
print("Average_stays_in_weekend_nights:",Average_stays_in_weekend_nights)
Average_stays_in_week_nights = stays_in_week_nights.mean()
print("Average_stays_in_week_nights:",Average_stays_in_week_nights)
```

```
Average_lead_time: 97.24599927259501
Average_stays_in_weekend_nights: 0.9297144935442808
Average_stays_in_week_nights: 2.46126568466994
```

2.MODE

In [13]:
```
#Lead time mode
Mode_lead_time = lead_time.mode().iloc[0]
print("Mode_lead_time:", Mode_lead_time)
Mode_stays_in_weekend_nights = stays_in_weekend_nights.mode().iloc[0]
print("Mode_stays_in_weekend_nights:", Mode_stays_in_weekend_nights)
Mode_stays_in_week_nights = stays_in_week_nights.mode().iloc[0]
print("Mode_stays_in_week_nights:", Mode_stays_in_week_nights)
```

```
Mode_lead_time: 0
Mode_stays_in_weekend_nights: 0
Mode_stays_in_week_nights: 2
```

MEASURES OF DISPERSION

1.RANGE

In [14]:
```
#Lead time range
minimum_lead_time = lead_time.min()
print('minimum_lead_time:', minimum_lead_time)
maximum_lead_time = lead_time.max()
print("maximum_lead_time:", maximum_lead_time)
maximum_stays_in_weekend_nights = stays_in_weekend_nights.max()
print("stays_in_weekend_nights:", stays_in_weekend_nights)
maximum_stays_in_week_nights = stays_in_week_nights.max()
print("stays_in_week_nights:",stays_in_week_nights)
```

```
minimum_lead_time: 0
maximum_lead_time: 737
stays_in_weekend_nights: 0        0
1        0
2        0
3        2
4        2
        ..
21991    2
21992    2
21993    0
21994    1
21995    1
Name: stays_in_weekend_nights, Length: 21996, dtype: int64
stays_in_week_nights: 0        3
1        3
2        4
3        5
4        8
        ..
21991    1
21992    4
21993    5
21994    4
21995    4
Name: stays_in_week_nights, Length: 21996, dtype: int64
```

QUARTILES

In [15]:
```
#first Quartile
q1 = np.quantile(stays_in_week_nights, 0.25)
stays_in_week_nights_Lower_quartile = stays_in_week_nights.quantile(0.25)

#third quartile
q3 = np.quantile(stays_in_week_nights, 0.75)
stays_in_week_nights_LQ = stays_in_week_nights.quantile(0.75)

#Inter quartile region
iqr = q3-q1

#upper and lower bound
upper_bound = q3+(1.5*iqr)
lower_bound = q1-(1.5*iqr)
print(q1, q3 ,iqr, upper_bound, lower_bound,stays_in_week_nights_Lower_quartile)
```

```
1.0 3.0 2.0 6.0 -2.0 1.0
```

INTER QUARTILE RANGE(IQR)

In [16]:
```
# Import stats from scipy library
from scipy import stats
# IQR
IQR = stats.iqr(stays_in_week_nights, interpolation = 'midpoint')
print(IQR)
```

```
2.0
```

In [17]:
```
# IMPORTING STATISTCS MODULE
import statistics
# Variance calculation for AGE
AGE_VAR = statistics.variance(stays_in_week_nights)
print("VARIANCE OF AGE\n",AGE_VAR)
```

```
VARIANCE OF AGE
 3.5811774659394766
```

In [18]:
```
stays_in_week_nights_std= stays_in_week_nights.std()
print("standard deviation of stays_in_week_nights\n",stays_in_week_nights_std)
stays_in_weekend_nights_std= stays_in_weekend_nights.std()
print(" standard deviation of stays_in_weekend_nights\n",stays_in_weekend_nights_std)
```

```
standard deviation of stays_in_week_nights
 1.892399922304908
 standard deviation of stays_in_weekend_nights
 1.0055842852835148
```

In [19]:
```
### BIVARATE DATA
BIVARIATE = Data.iloc[:, [7,8]]
BIVARIATE.head()
```

Out[19]:

	stays_in_weekend_nights	stays_in_week_nights
0	0	3
1	0	3
2	0	4
3	2	5
4	2	8

In [20]:
```
from numpy import cov

#select data Feature or Attribute
stays_in_week_nights = Data['stays_in_week_nights']
# select SEX Feature /attribute
lead_time = Data ['lead_time']

COV = cov(stays_in_weekend_nights,stays_in_week_nights)
print("Covariance matrix of bivariate data \n",COV)
def covariance(x, y):
    # Finding the mean of the series x and y
    mean_x = sum(x)/float(len(x))
    mean_y = sum(y)/float(len(y))
    # Subtracting mean from the individual elements
    sub_x = [i - mean_x for i in x]
    sub_y = [i - mean_y for i in y]
    numerator = sum([sub_x[i]*sub_y[i] for i in range(len(sub_x))])
    denominator = len(x)-1
    cov = numerator/denominator
    return cov

cov_func = covariance(stays_in_weekend_nights,stays_in_week_nights)
print("Covariance from the custom function between stays_in_weekend_nights and stays_in_week_nights:", co
```

```
Covariance matrix of bivariate data
 [[1.01119975 0.96158749]
 [0.96158749 3.58117747]]
Covariance from the custom function between stays_in_weekend_nights and stays_in_week_nights: 0.9615874
857242421
```

Correlation

(Perfect — values near to ±1)

(High degree — values between ±0.5 and ±1)

(Moderate degree — values between ±0.3 and ±0.49)

(Low degree — values below ±0.29)

(No correlation — values close to 0)

In [21]:
```
BIVARIATE[["stays_in_weekend_nights","stays_in_week_nights"]].corr(method='pearson')
```

Out[21]:

	stays_in_weekend_nights	stays_in_week_nights
stays_in_weekend_nights	1.000000	0.505309
stays_in_week_nights	0.505309	1.000000

In [22]:
```
Data[['lead_time','arrival_date_week_number']].corr(method='spearman')
```

Out[22]:

	lead_time	arrival_date_week_number
lead_time	1.000000	-0.219298
arrival_date_week_number	-0.219298	1.000000

In [23]:
```
Data[['arrival_date_day_of_month','stays_in_week_nights']].corr(method='spearman')
```

Out[23]:

	arrival_date_day_of_month	stays_in_week_nights
arrival_date_day_of_month	1.000000	-0.008466
stays_in_week_nights	-0.008466	1.000000

In [ ]:

In [ ]: