Assignment – 4

2303A52330

B.Rithwik

Batch – 35

Code –

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
QUESTION 1
```

```
[4] import pandas as pd
    import numpy as np
    import matplotlib.pyplot as plt
    import seaborn as sns
    from sklearn.model_selection import train_test_split
    from sklearn.linear_model import LogisticRegression
    from sklearn.metrics import accuracy_score
    data=pd.read_csv('/content/train.csv')
    columns=['battery_power', 'clock_speed', 'mobile_wt', 'talk_time', 'price_range']
    data=data.loc[:,columns]
    data.head()
    print(data)
```

```
      battery_power  clock_speed  mobile_wt  talk_time  price_range
0               842          2.2        188         19            1
1              1021          0.5        136          7            2
2               563          0.5        145          9            2
3               615          2.5        131         11            2
4              1821          1.2        141         15            1
...             ...          ...        ...        ...          ...
1995            794          0.5        106         19            0
1996           1965          2.6        187         16            2
1997           1911          0.9        108          5            3
1998           1512          0.9        145         19            0
1999            510          2.0        168          2            3

[2000 rows x 5 columns]
```

```
print(data)
data.tail(-1)
xa=np.max(data)
print(xa)
xb=np.min(data)
print(xb)
for i in data:
  data[i]=(data[i]-xb)/(xa-xb)
print(data)
```

```
      battery_power  clock_speed  mobile_wt  talk_time  price_range
0               842          2.2        188         19            1
1              1021          0.5        136          7            2
2               563          0.5        145          9            2
3               615          2.5        131         11            2
4              1821          1.2        141         15            1
...             ...          ...        ...        ...          ...
1995            794          0.5        106         19            0
1996           1965          2.6        187         16            2
1997           1911          0.9        108          5            3
1998           1512          0.9        145         19            0
1999            510          2.0        168          2            3

[2000 rows x 5 columns]
1998.0
0.0
      battery_power  clock_speed  mobile_wt  talk_time  price_range
0          0.421421     0.001101   0.094094   0.009510     0.000501
1          0.511011     0.000250   0.068068   0.003504     0.001001
2          0.281782     0.000250   0.072573   0.004505     0.001001
3          0.307808     0.001251   0.065566   0.005506     0.001001
4          0.911411     0.000601   0.070571   0.007508     0.000501
...             ...          ...        ...        ...          ...
1995       0.397397     0.000250   0.053053   0.009510     0.000000
1996       0.983483     0.001301   0.093594   0.008008     0.001001
1997       0.956456     0.000450   0.054054   0.002503     0.001502
1998       0.756757     0.000450   0.072573   0.009510     0.000000
1999       0.255255     0.001001   0.084084   0.001001     0.001502

[2000 rows x 5 columns]
```

```
features=['battery_power', 'clock_speed', 'mobile_wt', 'talk_time']
x=data.loc[:, features]
y=data.loc[:, 'price_range']
print(x)
print(y)
```

```
      battery_power  clock_speed  mobile_wt  talk_time
0          0.421421     0.001101   0.094094   0.009510
1          0.511011     0.000250   0.068068   0.003504
2          0.281782     0.000250   0.072573   0.004505
3          0.307808     0.001251   0.065566   0.005506
4          0.911411     0.000601   0.070571   0.007508
...             ...          ...        ...        ...
1995       0.397397     0.000250   0.053053   0.009510
1996       0.983483     0.001301   0.093594   0.008008
1997       0.956456     0.000450   0.054054   0.002503
1998       0.756757     0.000450   0.072573   0.009510
1999       0.255255     0.001001   0.084084   0.001001

[2000 rows x 4 columns]
0       0.000501
1       0.001001
2       0.001001
3       0.001001
4       0.000501
          ...
1995    0.000000
1996    0.001001
1997    0.001502
1998    0.000000
1999    0.001502
Name: price_range, Length: 2000, dtype: float64
```

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=0)
print(x_train)
print(y_train)
print(x_test)
print(y_test)
```

```
      battery_power  clock_speed  mobile_wt  talk_time
582        0.616617     0.001451   0.084585   0.008008
159        0.920921     0.000250   0.071071   0.005005
1827       0.846847     0.001051   0.053053   0.003504
318        0.254254     0.000400   0.047047   0.003504
708        0.488989     0.001401   0.082583   0.005005
...             ...          ...        ...        ...
835        0.612613     0.000801   0.078579   0.006507
1216       0.579580     0.000350   0.061562   0.007508
1653       0.595596     0.001001   0.046547   0.004505
559        0.596096     0.001201   0.084585   0.004004
684        0.353353     0.000250   0.054054   0.002002

[1600 rows x 4 columns]
582     0.000000
159     0.000501
1827    0.001502
318     0.000000
708     0.001502
          ...
835     0.001502
1216    0.000501
1653    0.001502
559     0.000000
684     0.000501
Name: price_range, Length: 1600, dtype: float64
      battery_power  clock_speed  mobile_wt  talk_time
405        0.727728     0.000250   0.041542   0.002503
1190       0.546547     0.000250   0.083584   0.005506
1132       0.762763     0.000901   0.087087   0.006507
731        0.904404     0.001051   0.062563   0.006507
1754       0.543544     0.000851   0.055556   0.008509
...             ...          ...        ...        ...
638        0.477978     0.000250   0.071572   0.004505
360        0.404905     0.000400   0.045546   0.001502
1810       0.297297     0.001301   0.098098   0.008509
1743       0.837337     0.000801   0.091091   0.010010
563        0.645646     0.000250   0.071572   0.007508

[400 rows x 4 columns]
405     0.001502
1190    0.000000
1132    0.001001
731     0.001001
1754    0.001001
```

```
638     0.000501
360     0.000501
1810    0.000000
1743    0.001001
563     0.000501
Name: price_range, Length: 400, dtype: float64
```

JESTION 2

```
size_x=x.size
size_y=y.size
print(size_x)
print(size_y)
print(data.size)
print('The shape of the data is', data.shape)
print('The X shape of the data is', x.shape)
print('The type of the data is ', type(x))
bool_series = pd.isnull(data["battery_power"])
missing_values_count = bool_series.sum()
print("Count of missing values in the 'Team' column:", missing_values_count)
print(bool_series)
```

```
8000
2000
10000
The shape of the data is (2000, 5)
The X shape of the data is (2000, 4)
The type of the data is  <class 'pandas.core.frame.DataFrame'>
Count of missing values in the 'Team' column: 0
0       False
1       False
2       False
3       False
4       False
        ...
1995    False
1996    False
1997    False
1998    False
1999    False
Name: battery_power, Length: 2000, dtype: bool
```