AI Assignment-4

Name:- Vanisha Singh
Roll No:- 2020347

We first uploaded our database to google collab and then read it using panda's library.

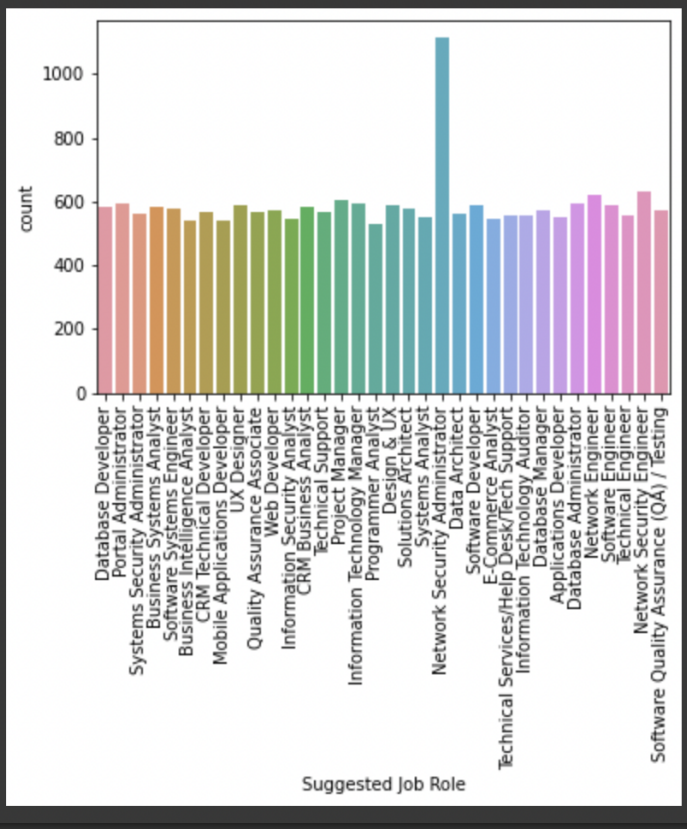
```
df.columns
```

```
Index(['Acedamic percentage in Operating Systems', 'percentage in Algorithms',
       'Percentage in Programming Concepts',
       'Percentage in Software Engineering', 'Percentage in Computer Networks',
       'Percentage in Electronics Subjects',
       'Percentage in Computer Architecture', 'Percentage in Mathematics',
       'Percentage in Communication skills', 'Hours working per day',
       'Logical quotient rating', 'hackathons', 'coding skills rating',
       'public speaking points', 'can work long time before system?',
       'self-learning capability?', 'Extra-courses did', 'certifications',
       'workshops', 'talenttests taken?', 'olympiads',
       'reading and writing skills', 'memory capability score',
       'Interested subjects', 'interested career area ', 'Job/Higher Studies?',
       'Type of company want to settle in?',
       'Taken inputs from seniors or elders', 'interested in games',
       'Interested Type of Books', 'Salary Range Expected',
       'In a Realtionship?', 'Gentle or Tuff behaviour?',
       'Management or Technical', 'Salary/work', 'hard/smart worker',
       'worked in teams ever?', 'Introvert', 'Suggested Job Role'],
      dtype='object')
```

```
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20000 entries, 0 to 19999
Data columns (total 39 columns):
 #   Column                                  Non-Null Count  Dtype 
---  ------                                  --------------  ----- 
 0   Acedamic percentage in Operating Systems  20000 non-null  int64 
 1   percentage in Algorithms                20000 non-null  int64 
 2   Percentage in Programming Concepts      20000 non-null  int64 
 3   Percentage in Software Engineering      20000 non-null  int64 
 4   Percentage in Computer Networks         20000 non-null  int64 
 5   Percentage in Electronics Subjects      20000 non-null  int64 
 6   Percentage in Computer Architecture     20000 non-null  int64 
 7   Percentage in Mathematics               20000 non-null  int64 
 8   Percentage in Communication skills      20000 non-null  int64 
 9   Hours working per day                   20000 non-null  int64 
 10  Logical quotient rating                 20000 non-null  int64 
 11  hackathons                              20000 non-null  int64 
 12  coding skills rating                    20000 non-null  int64 
 13  public speaking points                  20000 non-null  int64 
 14  can work long time before system?       20000 non-null  object
 15  self-learning capability?               20000 non-null  object
 16  Extra-courses did                       20000 non-null  object
 17  certifications                          20000 non-null  object
 18  workshops                               20000 non-null  object
 19  talenttests taken?                      20000 non-null  object
 20  olympiads                               20000 non-null  object
 21  reading and writing skills              20000 non-null  object
 22  memory capability score                 20000 non-null  object
 23  Interested subjects                     20000 non-null  object
 24  interested career area                  20000 non-null  object
 25  Job/Higher Studies?                     20000 non-null  object
 26  Type of company want to settle in?      20000 non-null  object
 27  Taken inputs from seniors or elders     20000 non-null  object
 28  interested in games                     20000 non-null  object
 29  Interested Type of Books                20000 non-null  object
 30  Salary Range Expected                   20000 non-null  object
 31  In a Realtionship?                      20000 non-null  object
 32  Gentle or Tuff behaviour?               20000 non-null  object
 33  Management or Technical                 20000 non-null  object
 34  Salary/work                             20000 non-null  object
 35  hard/smart worker                       20000 non-null  object
 36  worked in teams ever?                   20000 non-null  object
 37  Introvert                               20000 non-null  object
 38  Suggested Job Role                      20000 non-null  object
dtypes: int64(14), object(25)
memory usage: 6.0+ MB
None
```

```
chart = sns.countplot(x=y)
chart.set_xticklabels(chart.get_xticklabels(),rotation=90)
plt.show()
```



Then we do one hot encoding of the entire feature data and then train the model for at 80-20 train-test split.

```
[39] X_new = OneHotEncoder().fit_transform(X)

[40] Y_new = Y.copy(deep=True)

[41] X11train, X11test, Y11train, Y11test = train_test_split(X_new,Y_new,test_size=0.2)

m11 = MLPClassifier(max_iter = 300,random_state=1).fit(X11train,Y11train)

a11 = accuracy_score(m11.predict(X11test),Y11test)
print("Accuracy (at testing size = 0.2)")
print(a11)
```

```
Accuracy (at testing size = 0.2)
0.02825
```

```
print("Train confusion matrix")
print(cm11train)
print("Test confusion matrix")
print(cm11test)
print("Train classwise accuracies")
print(cm11train.diagonal()/cm11train.sum(axis=1))
print("Test classwise accuracies")
print(cm11test.diagonal()/cm11test.sum(axis=1))
```

```
Train confusion matrix
[[413   3   2 ...   1   0   0]
 [  2 365   2 ...   0   1   2]
 [  1   1 420 ...   1   1   1]
 ...
 [  0   0   1 ... 443   0   0]
 [  0   0   1 ...   1 440   0]
 [  0   0   1 ...   1   0 442]]
Test confusion matrix
[[2 6 1 ... 6 4 4]
 [1 3 1 ... 1 0 3]
 [4 4 5 ... 2 3 2]
 ...
 [4 4 1 ... 2 2 3]
 [3 8 5 ... 0 4 4]
 [4 8 1 ... 2 6 1]]
```

Then we did standardization and normalization of the given data

```
X13 = preprocessing.normalize(X_new)
Y13 = Y.copy(deep=True)
X13train, X13test, Y13train, Y13test = train_test_split(X13,Y13,test_size=0.2)
m13 = MLPClassifier(max_iter = 300,random_state=50).fit(X13train,Y13train)
a13 = accuracy_score(m13.predict(X13test),Y13test)
print("Accuracy (after normalisation at testing size = 0.2)")
print(a13)
m13.predict(X13test)
cm13train = confusion_matrix(m13.predict(X13train),Y13train)
cm13test = confusion_matrix(m13.predict(X13test),Y13test)
print("Train confusion matrix")
print(cm13train)
print("Test confusion matrix")
print(cm13test)
print("Train classwise accuracies")
print(cm13train.diagonal()/cm13train.sum(axis=1))
print("Test classwise accuracies")
print(cm13test.diagonal()/cm13test.sum(axis=1))
```

Then we did the same step for 60-40,90-10,70-30 train-test splits and found the accuracy, confusion matrix, and class-wise accuracies.

Then we combined the suggested job roles and found the accuracies

```
[64] y1 = Y.copy(deep=True)
```

```
[65] y1 = y1.replace(['Project Manager','Information Technology Manager','Database Manager'],'Manager')
     y1 = y1.replace(['Business Systems Analyst','Business Intelligence Analyst','E-Commerce Analyst','Information Security Analyst','Systems Analyst','CRM Busin
     y1 = y1.replace(['Mobile Applications Developer','Web Developer','Software Developer','Applications Developer','CRM Technical Developer','Database Developer
     y1 = y1.replace(['UX Designer','Design & UX'] , 'UX/Design')
     y1 = y1.replace(['Database Administrator','Portal Administrator','Network Security Administrator','Systems Security Administrator'] , 'Administrator')
     y1 = y1.replace(['Software Quality Assurance (QA) / Testing','Quality Assurance Associate','Solutions Architect','Data Architect','Information Technology Au
     y1 = y1.replace(['Technical Engineer','Technical Services/Help Desk/Tech Support','Technical Support'],'Technical')
     y1 = y1.replace(['Network Security Engineer','Network Engineer','Software Systems Engineer','Software Engineer'],'Engineer')
```

```
[69] X5train, X5test, Y5train, Y5test = train_test_split(X_new,y1,test_size=0.1)
     m5 = MLPClassifier(random_state=1).fit(X5train,Y5train)
     a5 = accuracy_score(m5.predict(X5test),Y5test)
     print("Accuracy (at testing size = 0.1)")
     print(a5)
```

```
Accuracy (at testing size = 0.1)
0.14
```

We changed the database according to assignment 1 and then found out the accuracies.

```
record = df.rename(columns = {'Acedamic percentage in Operating Systems':'Acedamic percentage in Machine Learning','percentage in Algorithms'
:'percentage in Signal Processing','Percentage in Programming Concepts':'Percentage in Image Processing','Percentage in Electronics Subjects'
:'Percentage in Big data mining in healthcare','Percentage in Communication skills':'Percentage in Convex Optimization'})
```

```
record.columns
```

```
Index(['Acedamic percentage in Machine Learning',
       'percentage in Signal Processing', 'Percentage in Image Processing',
       'Percentage in Software Engineering', 'Percentage in Computer Networks',
       'Percentage in Big data mining in healthcare',
       'Percentage in Computer Architecture', 'Percentage in Mathematics',
       'Percentage in Convex Optimization', 'Hours working per day',
       'Logical quotient rating', 'hackathons', 'coding skills rating',
       'public speaking points', 'can work long time before system?',
       'self-learning capability?', 'Extra-courses did', 'certifications',
       'workshops', 'talenttests taken?', 'olympiads',
       'reading and writing skills', 'memory capability score',
       'Interested subjects', 'interested career area ', 'Job/Higher Studies?',
       'Type of company want to settle in?',
       'Taken inputs from seniors or elders', 'interested in games',
       'Interested Type of Books', 'Salary Range Expected',
       'In a Realtionship?', 'Gentle or Tuff behaviour?',
       'Management or Technical', 'Salary/work', 'hard/smart worker',
       'worked in teams ever?', 'Introvert', 'Suggested Job Role'],
      dtype='object')
```

```
[74] X6 = record.iloc[:,:-1]
     X6_new = OneHotEncoder().fit_transform(X6)
     Y6_new = Y.copy(deep=True)
     X6train, X6test, Y6train, Y6test = train_test_split(X6_new,y1,test_size=0.1)
     m6 = MLPClassifier(random_state=1).fit(X6train,Y6train)
     a6 = accuracy_score(m6.predict(X6test),Y6test)
     print("Accuracy (at testing size = 0.1)")
     print(a6)
```

```
Accuracy (at testing size = 0.1)
0.1415
```