AI_assign_3

November 22, 2020

[1]:
```python
import pandas as pd
```

[2]:
```python
df = pd.read_excel('data.xlsx')
```

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

[3]:
```
       ID Gender        DOB  10percentage                         10board  \
0  203097      f 1990-02-19          84.3  board ofsecondary education,ap
1  579905      m 1989-10-04          85.4                            cbse
2  810601      f 1992-08-03          85.0                            cbse
3  267447      m 1989-12-05          85.6                            cbse
4  343523      m 1991-02-27          78.0                            cbse

   12graduation  12percentage                             12board  CollegeID  \
0          2007          95.8  board of intermediate education,ap       1141
1          2007          85.0                                cbse       5807
2          2010          68.2                                cbse         64
3          2007          83.6                                cbse       6920
4          2008          76.8                                cbse      11368

   CollegeTier  … MechanicalEngg ElectricalEngg  TelecomEngg  CivilEngg  \
0            2  …             -1             -1           -1         -1
1            2  …             -1             -1           -1         -1
2            2  …             -1             -1           -1         -1
3            1  …             -1             -1           -1         -1
4            2  …             -1             -1           -1         -1

   conscientiousness agreeableness  extraversion  nueroticism  \
0             0.9737        0.8128        0.5269      1.35490
1            -0.7335        0.3789        1.2396     -0.10760
2             0.2718        1.7109        0.1637     -0.86820
3             0.0464        0.3448       -0.3440     -0.40780
4            -0.8810       -0.2793       -1.0697      0.09163

   openess_to_experience  High-Salary
0                -0.4455            1
1                 0.8637            1
```

```
2                 0.6721           1
3                -0.9194           1
4                -0.1295           0

[5 rows x 34 columns]
```

[4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3998 entries, 0 to 3997
Data columns (total 34 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   ID                     3998 non-null   int64
 1   Gender                 3998 non-null   object
 2   DOB                    3998 non-null   datetime64[ns]
 3   10percentage           3998 non-null   float64
 4   10board                3998 non-null   object
 5   12graduation           3998 non-null   int64
 6   12percentage           3998 non-null   float64
 7   12board                3998 non-null   object
 8   CollegeID              3998 non-null   int64
 9   CollegeTier            3998 non-null   int64
 10  Degree                 3998 non-null   object
 11  Specialization         3998 non-null   object
 12  collegeGPA             3998 non-null   float64
 13  CollegeCityID          3998 non-null   int64
 14  CollegeCityTier        3998 non-null   int64
 15  CollegeState           3998 non-null   object
 16  GraduationYear         3998 non-null   int64
 17  English                3998 non-null   int64
 18  Logical                3998 non-null   int64
 19  Quant                  3998 non-null   int64
 20  Domain                 3998 non-null   float64
 21  ComputerProgramming    3998 non-null   int64
 22  ElectronicsAndSemicon  3998 non-null   int64
 23  ComputerScience        3998 non-null   int64
 24  MechanicalEngg         3998 non-null   int64
 25  ElectricalEngg         3998 non-null   int64
 26  TelecomEngg            3998 non-null   int64
 27  CivilEngg              3998 non-null   int64
 28  conscientiousness      3998 non-null   float64
 29  agreeableness          3998 non-null   float64
 30  extraversion           3998 non-null   float64
 31  nueroticism            3998 non-null   float64
 32  openess_to_experience  3998 non-null   float64
 33  High-Salary            3998 non-null   int64
```

```
dtypes: datetime64[ns](1), float64(9), int64(18), object(6)
memory usage: 1.0+ MB
```

[5]:
```python
from datetime import date
today = date.today()
ages = []
for i in range(len(df)):
    born = df.loc[i, "DOB"]
    age = today.year - born.year - ((today.month, today.day) < (born.month,␣
 ↪born.day))
    ages.append(age)
df["age"] = ages
```

[6]:
```python
df.describe()
```

[6]:
```
                 ID  10percentage  12graduation  12percentage     CollegeID  \
count  3.998000e+03   3998.000000   3998.000000   3998.000000   3998.000000
mean   6.637945e+05     77.925443   2008.087544     74.466366   5156.851426
std    3.632182e+05      9.850162      1.653599     10.999933   4802.261482
min    1.124400e+04     43.000000   1995.000000     40.000000      2.000000
25%    3.342842e+05     71.680000   2007.000000     66.000000    494.000000
50%    6.396000e+05     79.150000   2008.000000     74.400000   3879.000000
75%    9.904800e+05     85.670000   2009.000000     82.600000   8818.000000
max    1.298275e+06     97.760000   2013.000000     98.700000  18409.000000

       CollegeTier   collegeGPA  CollegeCityID  CollegeCityTier  \
count  3998.000000  3998.000000    3998.000000      3998.000000
mean      1.925713    71.486171    5156.851426         0.300400
std       0.262270     8.167338    4802.261482         0.458489
min       1.000000     6.450000       2.000000         0.000000
25%       2.000000    66.407500     494.000000         0.000000
50%       2.000000    71.720000    3879.000000         0.000000
75%       2.000000    76.327500    8818.000000         1.000000
max       2.000000    99.930000   18409.000000         1.000000

       GraduationYear  …  ElectricalEngg  TelecomEngg    CivilEngg  \
count     3998.000000  …     3998.000000  3998.000000  3998.000000
mean      2012.105803  …       16.478739    31.851176     2.683842
std         31.857271  …       87.585634   104.852845    36.658505
min          0.000000  …       -1.000000    -1.000000    -1.000000
25%       2012.000000  …       -1.000000    -1.000000    -1.000000
50%       2013.000000  …       -1.000000    -1.000000    -1.000000
75%       2014.000000  …       -1.000000    -1.000000    -1.000000
max       2017.000000  …      676.000000   548.000000   516.000000

       conscientiousness  agreeableness  extraversion  nueroticism  \
count        3998.000000    3998.000000   3998.000000  3998.000000
```

```
mean          -0.037831      0.146496     0.002763    -0.169033
std            1.028666      0.941782     0.951471     1.007580
min           -4.126700     -5.781600    -4.600900    -2.643000
25%           -0.713525     -0.287100    -0.604800    -0.868200
50%            0.046400      0.212400     0.091400    -0.234400
75%            0.702700      0.812800     0.672000     0.526200
max            1.995300      1.904800     2.535400     3.352500

       openess_to_experience  High-Salary          age
count            3998.000000  3998.000000  3998.000000
mean               -0.138110     0.534017    29.474487
std                 1.008075     0.498904     1.773015
min                -7.375700     0.000000    23.000000
25%                -0.669200     0.000000    28.000000
50%                -0.094300     1.000000    29.000000
75%                 0.502400     1.000000    31.000000
max                 1.822400     1.000000    43.000000

[8 rows x 28 columns]
```

[7]:

```
X = df.loc[:,['10percentage', '12graduation',
       '12percentage', 'CollegeTier', 'collegeGPA', 'CollegeCityTier',
       'GraduationYear', 'English', 'Logical', 'Quant',
       'Domain', 'ComputerProgramming', 'ElectronicsAndSemicon',
       'ComputerScience', 'MechanicalEngg', 'ElectricalEngg', 'TelecomEngg',
       'CivilEngg', 'conscientiousness', 'agreeableness', 'extraversion',
       'nueroticism', 'openess_to_experience', 'age'] ]
y = df.loc[:,[ 'High-Salary']]
```

[8]:

```
X1 = X.corr()
print(X1)
```

```
                     10percentage  12graduation  12percentage  CollegeTier  \
10percentage             1.000000      0.269957      0.643378    -0.126042
12graduation             0.269957      1.000000      0.259166     0.027691
12percentage             0.643378      0.259166      1.000000    -0.100771
CollegeTier             -0.126042      0.027691     -0.100771     1.000000
collegeGPA               0.312538      0.086001      0.346137    -0.086781
CollegeCityTier          0.116707     -0.003016      0.130462    -0.101494
GraduationYear          -0.013799      0.014457     -0.012933    -0.005557
English                  0.350780      0.147925      0.212888    -0.183843
Logical                  0.316014      0.105887      0.243571    -0.182811
Quant                    0.317640      0.001379      0.312413    -0.251103
Domain                   0.078563     -0.034163      0.074099    -0.061436
ComputerProgramming      0.053600     -0.047995      0.080818    -0.073644
ElectronicsAndSemicon    0.085179     -0.005891      0.117112    -0.031573
ComputerScience         -0.018933      0.293439     -0.043534     0.001053
```

MechanicalEngg	0.050364	0.035459	0.037635	-0.021548
ElectricalEngg	0.074419	0.123751	0.064001	0.002594
TelecomEngg	0.049378	0.023470	0.044201	0.000007
CivilEngg	0.030002	-0.004727	0.005910	-0.033722
conscientiousness	0.067657	0.103329	0.058299	0.055174
agreeableness	0.136645	0.041182	0.103998	-0.038055
extraversion	-0.004679	0.061956	-0.007486	0.009970
nueroticism	-0.132496	-0.074369	-0.094369	0.023778
openess_to_experience	0.036692	-0.015069	0.006332	-0.019179
age	-0.240898	-0.874084	-0.262868	-0.036627

	collegeGPA	CollegeCityTier	GraduationYear	English	\
10percentage	0.312538	0.116707	-0.013799	0.350780	
12graduation	0.086001	-0.003016	0.014457	0.147925	
12percentage	0.346137	0.130462	-0.012933	0.212888	
CollegeTier	-0.086781	-0.101494	-0.005557	-0.183843	
collegeGPA	1.000000	0.017471	0.008706	0.106478	
CollegeCityTier	0.017471	1.000000	0.008152	0.050462	
GraduationYear	0.008706	0.008152	1.000000	-0.024089	
English	0.106478	0.050462	-0.024089	1.000000	
Logical	0.196610	0.020353	-0.024018	0.444357	
Quant	0.217380	0.007896	-0.021781	0.375784	
Domain	0.107252	0.009250	-0.009741	0.089721	
ComputerProgramming	0.136596	0.064272	0.026688	0.125005	
ElectronicsAndSemicon	0.029855	0.041083	0.006179	0.018591	
ComputerScience	0.007601	-0.010643	0.024089	0.059500	
MechanicalEngg	-0.031765	-0.052395	-0.066844	-0.002477	
ElectricalEngg	0.052258	0.010311	0.008525	0.032438	
TelecomEngg	-0.005226	0.049876	0.004226	-0.005822	
CivilEngg	-0.018950	-0.033392	0.001696	-0.007724	
conscientiousness	0.069582	0.014763	-0.013235	0.034943	
agreeableness	0.068282	0.005565	-0.002877	0.194990	
extraversion	-0.032684	-0.008203	0.008397	0.018755	
nueroticism	-0.074859	0.004442	-0.000417	-0.155528	
openess_to_experience	0.028071	-0.016790	0.016855	0.067979	
age	-0.112177	0.029897	-0.017637	-0.105697	

	Logical	Quant	…	MechanicalEngg	\
10percentage	0.316014	0.317640	…	0.050364	
12graduation	0.105887	0.001379	…	0.035459	
12percentage	0.243571	0.312413	…	0.037635	
CollegeTier	-0.182811	-0.251103	…	-0.021548	
collegeGPA	0.196610	0.217380	…	-0.031765	
CollegeCityTier	0.020353	0.007896	…	-0.052395	
GraduationYear	-0.024018	-0.021781	…	-0.066844	
English	0.444357	0.375784	…	-0.002477	
Logical	1.000000	0.500152	…	-0.009861	
Quant	0.500152	1.000000	…	0.019933	

Domain	0.169453	0.207108	…	0.048472
ComputerProgramming	0.183905	0.146035	…	-0.284891
ElectronicsAndSemicon	-0.009994	0.104221	…	-0.109434
ComputerScience	0.044481	-0.043379	…	-0.124355
MechanicalEngg	-0.009861	0.019933	…	1.000000
ElectricalEngg	0.012003	0.020975	…	-0.040522
TelecomEngg	-0.012947	0.021387	…	-0.070947
CivilEngg	-0.011286	0.000528	…	0.076201
conscientiousness	0.025876	-0.005639	…	-0.010858
agreeableness	0.167207	0.103443	…	-0.028586
extraversion	-0.006949	-0.028616	…	-0.017748
nueroticism	-0.178781	-0.131895	…	0.036148
openess_to_experience	0.048420	0.020377	…	-0.027988
age	-0.098083	-0.026773	…	-0.029248

	ElectricalEngg	TelecomEngg	CivilEngg	\
10percentage	0.074419	0.049378	0.030002	
12graduation	0.123751	0.023470	-0.004727	
12percentage	0.064001	0.044201	0.005910	
CollegeTier	0.002594	0.000007	-0.033722	
collegeGPA	0.052258	-0.005226	-0.018950	
CollegeCityTier	0.010311	0.049876	-0.033392	
GraduationYear	0.008525	0.004226	0.001696	
English	0.032438	-0.005822	-0.007724	
Logical	0.012003	-0.012947	-0.011286	
Quant	0.020975	0.021387	0.000528	
Domain	0.042875	0.024442	0.017569	
ComputerProgramming	-0.138224	-0.248269	-0.088249	
ElectronicsAndSemicon	0.036968	0.387140	0.002863	
ComputerScience	-0.083798	-0.148095	-0.052613	
MechanicalEngg	-0.040522	-0.070947	0.076201	
ElectricalEngg	1.000000	-0.051469	-0.020059	
TelecomEngg	-0.051469	1.000000	-0.031492	
CivilEngg	-0.020059	-0.031492	1.000000	
conscientiousness	0.029806	-0.004946	-0.017526	
agreeableness	-0.015454	-0.014627	-0.034254	
extraversion	0.004467	-0.039050	-0.031822	
nueroticism	-0.030870	0.020638	0.010555	
openess_to_experience	-0.012585	-0.000141	-0.031201	
age	-0.111800	-0.010363	0.012167	

	conscientiousness	agreeableness	extraversion	\
10percentage	0.067657	0.136645	-0.004679	
12graduation	0.103329	0.041182	0.061956	
12percentage	0.058299	0.103998	-0.007486	
CollegeTier	0.055174	-0.038055	0.009970	
collegeGPA	0.069582	0.068282	-0.032684	
CollegeCityTier	0.014763	0.005565	-0.008203	

GraduationYear	-0.013235	-0.002877	0.008397
English	0.034943	0.194990	0.018755
Logical	0.025876	0.167207	-0.006949
Quant	-0.005639	0.103443	-0.028616
Domain	-0.039478	0.051944	-0.024647
ComputerProgramming	0.012862	0.076934	0.043504
ElectronicsAndSemicon	-0.026483	-0.024286	-0.044458
ComputerScience	0.090155	0.039866	0.102153
MechanicalEngg	-0.010858	-0.028586	-0.017748
ElectricalEngg	0.029806	-0.015454	0.004467
TelecomEngg	-0.004946	-0.014627	-0.039050
CivilEngg	-0.017526	-0.034254	-0.031822
conscientiousness	1.000000	0.481820	0.355537
agreeableness	0.481820	1.000000	0.454369
extraversion	0.355537	0.454369	1.000000
nueroticism	-0.330312	-0.207480	-0.096491
openess_to_experience	0.395649	0.591541	0.435074
age	-0.105159	-0.025693	-0.054090

	nueroticism	openess_to_experience	age
10percentage	-0.132496	0.036692	-0.240898
12graduation	-0.074369	-0.015069	-0.874084
12percentage	-0.094369	0.006332	-0.262868
CollegeTier	0.023778	-0.019179	-0.036627
collegeGPA	-0.074859	0.028071	-0.112177
CollegeCityTier	0.004442	-0.016790	0.029897
GraduationYear	-0.000417	0.016855	-0.017637
English	-0.155528	0.067979	-0.105697
Logical	-0.178781	0.048420	-0.098083
Quant	-0.131895	0.020377	-0.026773
Domain	-0.017928	0.010412	0.039150
ComputerProgramming	-0.084344	0.043133	0.045663
ElectronicsAndSemicon	0.021026	-0.013460	0.021214
ComputerScience	-0.112652	0.058039	-0.286385
MechanicalEngg	0.036148	-0.027988	-0.029248
ElectricalEngg	-0.030870	-0.012585	-0.111800
TelecomEngg	0.020638	-0.000141	-0.010363
CivilEngg	0.010555	-0.031201	0.012167
conscientiousness	-0.330312	0.395649	-0.105159
agreeableness	-0.207480	0.591541	-0.025693
extraversion	-0.096491	0.435074	-0.054090
nueroticism	1.000000	-0.065795	0.080310
openess_to_experience	-0.065795	1.000000	0.006626
age	0.080310	0.006626	1.000000

[24 rows x 24 columns]

[9]:
```
Columns = X1.columns
for i in range(len(X1)):
    for j in range(len(X1)):
        if(X1.iloc[i, j] >= 0.9 and i!=j):
            print(Columns[i],Columns[j])
```

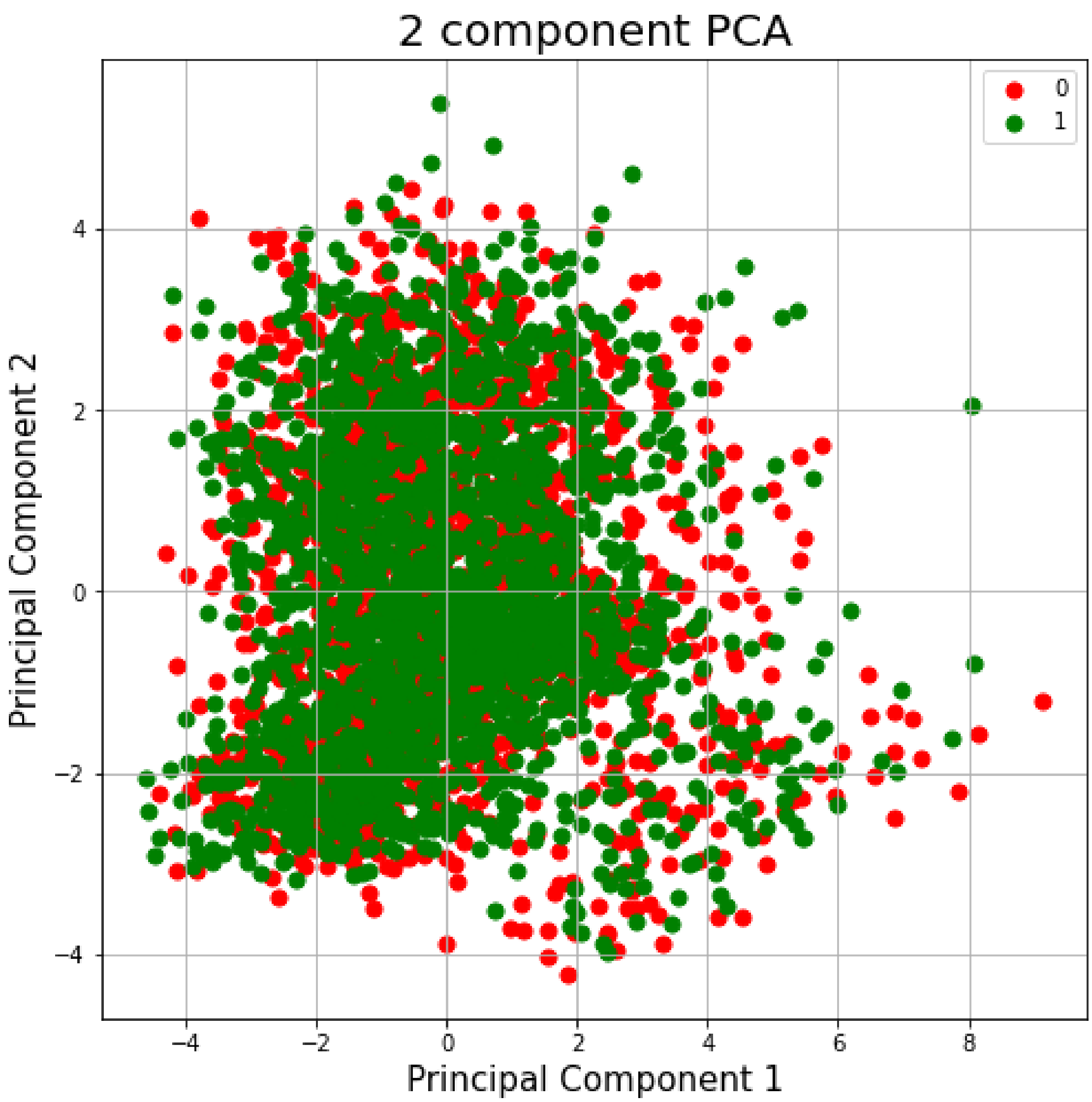
[10]:
```
X_2 = pd.DataFrame(df, columns=['Degree', 'Specialization'])
dum_df = pd.get_dummies(X_2, columns=['Degree', 'Specialization'],␣
↪prefix=["Type_of_",'Type_of_'], drop_first = True )
X = X.join(dum_df)
```

[11]:
```
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt
```

[12]:
```
import numpy as np
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.1,␣
↪random_state=42)

s = StandardScaler()
s.fit(X_train)
X_train = s.transform(X_train)
X_test = s.transform(X_test)

pca = PCA(n_components=2)
principalComponents = pca.fit_transform(X_train)
principalDf = pd.DataFrame(data = principalComponents, columns = ['principal␣
↪component 1', 'principal component 2'])
finalDf = pd.concat([principalDf, y], axis = 1)

fig = plt.figure(figsize = (8,8))
ax = fig.add_subplot(1,1,1)
ax.set_xlabel('Principal Component 1', fontsize = 15)
ax.set_ylabel('Principal Component 2', fontsize = 15)
ax.set_title('2 component PCA', fontsize = 20)
targets = [0, 1]
colors = ['r', 'g']
for target, color in zip(targets,colors):
    indicesToKeep = finalDf['High-Salary'] == target
    ax.scatter(finalDf.loc[indicesToKeep, 'principal component 1']
               , finalDf.loc[indicesToKeep, 'principal component 2']
               , c = color
               , s = 50)
ax.legend(targets)
ax.grid()
```

```python
print(X_train.shape, y_train.shape)
```

```
(3598, 72) (3598, 1)
```



[13]:
```python
logreg = LogisticRegression(max_iter = 1000)
clf = logreg.fit(X_train, y_train.to_numpy().reshape(((y_train.shape[0], ))))
```

[14]:
```python
y_res = clf.predict(X_test)
```

[15]:
```python
from sklearn.metrics import classification_report, accuracy_score,␣
↪confusion_matrix
print(classification_report(y_test, y_res))
```

```
              precision    recall  f1-score   support

           0       0.72      0.63      0.68       189
           1       0.71      0.78      0.74       211

    accuracy                           0.71       400
   macro avg       0.71      0.71      0.71       400
weighted avg       0.71      0.71      0.71       400
```

[16]:
```python
print("Accuracy is %f" % (100*accuracy_score(y_test, y_res)))
```

```
Accuracy is 71.250000
```

[17]:
```python
print("Confusion Matrix: ")
cm = confusion_matrix(y_test, y_res)
print(cm)
```

```
Confusion Matrix:
[[120  69]
 [ 46 165]]
```

[19]:
```python
cs = cm.diagonal()/cm.sum(axis=1)
print("Class-wise accuracy class 0:%f class 1:%f" %(cs[0] ,cs[1]))
```

```
Class-wise accuracy class 0:0.634921 class 1:0.781991
```

[ ]:
```python

```