“”” **Rankin of universities**

Name - Shubham Thorat “””

“”” DATASHEET (1st 20 rows)

World_Rank	University_Name	Country	Teaching_Rating	Inter_Outlook_Rating	Research_Rating	Citations_Rating	Industry_Income_Rating	Total_Score	Num_Students	Student/Staff_Ratio	%_Inter_Students	%_Female_Students
1	Harvard University	United States of America	99.7	72.4	98.7	98.8	34.5	96.1	20152	8.9	25	
2	California Institute of Technology	United States of America	97.7	54.6	98	99.9	83.7	96	2243	6.9	27	33
3	Massachusetts Institute of Technology	United States of America	97.8	82.3	91.4	99.9	87.5	95.6	11074	9	33	37
4	Stanford University	United States of America	98.3	29.5	98.1	99.2	64.3	94.3	15596	7.8	22	42
5	Princeton University	United States of America	90.9	70.3	95.4	99.9		94.2	7929	8.4	27	45
6	University of Cambridge	United Kingdom	90.5	77.7	94.1	94	57	91.2	18812	11.8	34	46
6	University of Oxford	United Kingdom	88.2	77.2	93.9	95.1	73.5	91.2	19919	11.6	34	46
8	University of California, Berkeley	United States of America	84.2	39.6	99.3	97.8		91.1	36186	16.4	15	50
9	Imperial College London	United Kingdom	89.2	90	94.5	88.3	92.9	90.6	15060	11.7	51	37
10	Yale University	United States of America	92.1	59.2	89.7	91.5		89.5	11751	4.4	20	50
11	University of California, Los Angeles	United States of America	83	48.1	92.9	93.2		87.7	38206	10.3	15	52
12	University of Chicago	United States of America	79.1	62.8	87.9	96.9		86.9	14221	6.9	21	42
13	Johns Hopkins University	United States of America	80.9	58.5	89.2	92.3		100	86.4	15128	3.6	23	50
14	Cornell University	United States of America	82.2	62.4	88.8	88.1	34.7	83.9	21424	10.2	19	48

15 ETH Zurich â Swiss Federal Institute of Technology Zurich Switzerland 77.5 93.7 87.8 83.1 83.4 18178 14.7 37 31

15 University of Michigan United States of America 83.9 53.3 89.1 84.1 59.6 83.4 41786 9 16 48

17 University of Toronto Canada 75.8 87.9 82.2 82 66198 19.5 15

18 Columbia University United States of America 73.8 90.9 73.8 92.6 81 25055 5.9 28

“””

#Code

importing the libraries

import pandas as pd

import matplotlib.pyplot as plt

import numpy as np

importing the data set

dataset=pd.read_csv('university_ranking.csv')

X = dataset.iloc[:,1:13].values

Y = dataset.iloc[:,0].values

/*

X=

```
array([['Harvard University', 'United States of America', 99.7, 72.4, 98.7,
        98.8, 34.5, 96.1, 20152.0, 8.9, 25.0, nan],
       ['California Institute of Technology', 'United States of America',
        97.7, 54.6, 98.0, 99.9, 83.7, 96.0, 2243.0, 6.9, 27.0, 33.0],
       ['Massachusetts Institute of Technology',
        'United States of America', 97.8, 82.3, 91.4, 99.9, 87.5, 95.6,
```

11074.0, 9.0, 33.0, 37.0],

['Stanford University', 'United States of America', 98.3, 29.5,

98.1, 99.2, 64.3, 94.3, 15596.0, 7.8, 22.0, 42.0],

['Princeton University', 'United States of America', 90.9, 70.3,

95.4, 99.9, nan, 94.2, 7929.0, 8.4, 27.0, 45.0],

['University of Cambridge', 'United Kingdom', 90.5, 77.7, 94.1,

94.0, 57.0, 91.2, 18812.0, 11.8, 34.0, 46.0],

['University of Oxford', 'United Kingdom', 88.2, 77.2, 93.9, 95.1,

73.5, 91.2, 19919.0, 11.6, 34.0, 46.0],

['University of California, Berkeley', 'United States of America',

84.2, 39.6, 99.3, 97.8, nan, 91.1, 36186.0, 16.4, 15.0, 50.0],

['Imperial College London', 'United Kingdom', 89.2, 90.0, 94.5,

88.3, 92.9, 90.6, 15060.0, 11.7, 51.0, 37.0],

['Yale University', 'United States of America', 92.1, 59.2, 89.7,

91.5, nan, 89.5, 11751.0, 4.4, 20.0, 50.0]]

Y=

array([1, 2, 3, 4, 5, 6, 6, 8, 9, 10]

*/

taking care of missing data

from sklearn.preprocessing import Imputer

imputer = Imputer(missing_values="NaN", strategy="mean",axis=0)

imputer = imputer.fit(X[:,2:13], y=None)

X[:,2:14] = imputer.transform(X[:,2:14])

/*

array([['Harvard University', 'United States of America', 99.7, 72.4, 98.7,

98.8, 34.5, 96.1, 20152.0, 8.9, 25.0, **49.29545454545455**],

['California Institute of Technology', 'United States of America',
97.7, 54.6, 98.0, 99.9, 83.7, 96.0, 2243.0, 6.9, 27.0, 33.0],
['Massachusetts Institute of Technology',
'United States of America', 97.8, 82.3, 91.4, 99.9, 87.5, 95.6,
11074.0, 9.0, 33.0, 37.0],
['Stanford University', 'United States of America', 98.3, 29.5,
98.1, 99.2, 64.3, 94.3, 15596.0, 7.8, 22.0, 42.0],
['Princeton University', 'United States of America', 90.9, 70.3,
95.4, 99.9, **51.479136690647486**, 94.2, 7929.0, 8.4, 27.0, 45.0],
['University of Cambridge', 'United Kingdom', 90.5, 77.7, 94.1,
94.0, 57.0, 91.2, 18812.0, 11.8, 34.0, 46.0],
['University of Oxford', 'United Kingdom', 88.2, 77.2, 93.9, 95.1,
73.5, 91.2, 19919.0, 11.6, 34.0, 46.0],
['University of California, Berkeley', 'United States of America',
84.2, 39.6, 99.3, 97.8, **51.479136690647486**, 91.1, 36186.0, 16.4,
15.0, 50.0],
['Imperial College London', 'United Kingdom', 89.2, 90.0, 94.5,
88.3, 92.9, 90.6, 15060.0, 11.7, 51.0, 37.0],
['Yale University', 'United States of America', 92.1, 59.2, 89.7,
91.5, **51.479136690647486**, 89.5, 11751.0, 4.4, 20.0, 50.0]],

***/**

handling categorical (encoding) data

from sklearn.preprocessing import LabelEncoder

label_encoder_X = LabelEncoder()

label_encoder_Y = LabelEncoder()

X[:,0] = label_encoder_X.fit_transform(X[:,0])

X[:,1] = label_encoder_X.fit_transform(X[:,1])

/*

X=

[[31, 25, 99.7, 72.4, 98.7, 98.8, 34.5, 96.1, 20152.0, 8.9, 25.0,

49.29545454545455],

[10, 25, 97.7, 54.6, 98.0, 99.9, 83.7, 96.0, 2243.0, 6.9, 27.0, 33.0],

[54, 25, 97.8, 82.3, 91.4, 99.9, 87.5, 95.6, 11074.0, 9.0, 33.0,

37.0],

[86, 25, 98.3, 29.5, 98.1, 99.2, 64.3, 94.3, 15596.0, 7.8, 22.0,

42.0],

[77, 25, 90.9, 70.3, 95.4, 99.9, 51.479136690647486, 94.2, 7929.0,

8.4, 27.0, 45.0],

[121, 24, 90.5, 77.7, 94.1, 94.0, 57.0, 91.2, 18812.0, 11.8, 34.0,

46.0],

[159, 24, 88.2, 77.2, 93.9, 95.1, 73.5, 91.2, 19919.0, 11.6, 34.0,

46.0],

[113, 25, 84.2, 39.6, 99.3, 97.8, 51.479136690647486, 91.1, 36186.0,

16.4, 15.0, 50.0],

[37, 24, 89.2, 90.0, 94.5, 88.3, 92.9, 90.6, 15060.0, 11.7, 51.0,

37.0],

[190, 25, 92.1, 59.2, 89.7, 91.5, 51.479136690647486, 89.5, 11751.0,

4.4, 20.0, 50.0]]

***/**

dividing dataset into test and training dataset

from sklearn.model_selection import train_test_split

X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size = .3, random_state = 0)

/*

X_train =

```
array([[74, 8, 51.9, 30.7, 37.2, 71.5, 26.4, 52.2, 27862.0, 8.7, 18.0, 48.0],
       [92, 0, 51.8, 74.2, 53.4, 69.0, 57.1, 59.1, 34718.0, 32.7, 27.0,
        53.0],
       [11, 25, 70.3, 39.1, 79.3, 95.7, 53.7, 79.3, 11885.0, 13.1, 35.0,
        39.0],
       [127, 25, 38.4, 16.8, 51.9, 61.3, 100.0, 50.4, 19262.0, 15.9, 10.0,
        56.0],
       [181, 21, 56.6, 87.9, 47.0, 65.0, 43.8, 57.7, 26583.0, 6.5, 19.0,
        57.0]]
Y_train = array([140,  81,  20, 159,  90]
X-test =
array([[121, 24, 90.5, 77.7, 94.1, 94.0, 57.0, 91.2, 18812.0, 11.8, 34.0,
        46.0],
       [42, 2, 57.7, 29.6, 62.9, 45.2, 97.7, 54.8, 42503.0, 41.9, 18.0,
        54.0],
       [40, 25, 80.9, 58.5, 89.2, 92.3, 100.0, 86.4, 15128.0, 3.6, 23.0,
        50.0],
       [17, 12, 55.5, 47.4, 67.7, 29.0, 99.4, 51.3, 15920.0, 19.4, 25.0,
        26.0]]
Y_test =
array([  6, 119,  13, 151]
*/
```

feature scaling

from sklearn.preprocessing import StandardScaler

sc_X = StandardScaler()

X_train = sc_X.fit_transform(X_train)

X_test = sc_X.fit_transform(X_test)

/*

X_train =

```
array([[ -4.04763141e-01,  -1.15764070e+00,  -1.52168502e-01,
         -1.09063038e+00,  -1.08154740e+00,  -6.53767509e-02,
         -1.28357406e+00,  -6.78415565e-01,   2.20222478e-01,
         -7.65960802e-01,  -1.22863071e-01,  -1.34094097e-01],
        [ -8.77922575e-02,  -2.08644545e+00,  -1.58912001e-01,
          8.10593387e-01,  -7.16875795e-02,  -2.18811983e-01,
          3.23575635e-01,  -1.03386812e-01,   6.51816994e-01,
          1.89704024e+00,   7.43236590e-01,   3.92976952e-01],
        [ -1.51416123e+00,   8.16069389e-01,   1.08863516e+00,
         -7.23497512e-01,   1.54284138e+00,   1.41987629e+00,
          1.45585115e-01,   1.58003070e+00,  -7.85551383e-01,
         -2.77743944e-01,   1.51310296e+00,  -1.08282198e+00],
        [  5.28540016e-01,   8.16069389e-01,  -1.06254076e+00,
         -1.69814786e+00,  -1.65193118e-01,  -6.91392497e-01,
          2.56939719e+00,  -8.28423066e-01,  -3.21159208e-01,
          3.29395118e-02,  -8.92729437e-01,   7.09219580e-01],
        [  1.47945267e+00,   3.51667015e-01,   1.64775912e-01,
          1.40936961e+00,  -4.70644544e-01,  -4.64308354e-01,
         -3.72681399e-01,  -2.20059313e-01,   1.39707690e-01,
         -1.01006923e+00,  -2.66297752e-02,   8.14633790e-01],
        [ -1.54938022e+00,   8.16069389e-01,   3.73824356e-01,
          2.11817167e-01,   1.52725713e-01,   3.15142624e-01,
          2.93225810e-02,   3.79970690e-01,  -9.89010175e-01,
```

-6.10619074e-01, -2.66297752e-02, 2.46522032e-03],
[-1.58459921e+00, 8.16069389e-01, -9.47901288e-01,
-1.05129471e+00, -1.30596069e+00, 3.33554852e-01,
2.93225810e-02, -8.36756816e-01, -7.01763124e-01,
1.99377077e-01, -2.66297752e-02, 4.98391161e-01],
[9.86386847e-01, 8.16069389e-01, 2.32210894e-01,
-1.42716884e+00, 9.07003723e-01, 2.35356303e-01,
2.93225810e-02, 4.38306941e-01, 1.41396910e+00,
2.65952103e-01, -6.04029549e-01, 2.87562742e-01],
[1.35618621e+00, -1.73814367e+00, -1.43343316e+00,
-6.13471669e-02, -3.89606410e-01, 4.01066354e-01,
-1.23122391e+00, -5.78410564e-01, -4.26980443e-01,
6.54306423e-01, -7.96496141e-01, 7.09219580e-01],
[1.00399634e+00, -1.73814367e+00, 1.31058421e-01,
-6.13471669e-02, -3.21035682e-01, -1.26830897e+00,
2.93225810e-02, -6.61748065e-01, 8.75042513e-01,
5.21156370e-01, 5.50769999e-01, 8.14633790e-01]])

X_test =

array([[0.32873875, 0.79480595, 2.46698117, 1.09473929, 2.19183891,
1.30951793, 0.21425824, 2.41037865, -0.44016799, -0.4420552 ,
1.61149983, -0.34818888],
[-1.0369244 , -1.8747713 , 0.0735777 , -0.87903318, 0.30629944,
-1.35687185, 2.27454172, -0.50530508, 1.56065337, 2.25751567,

```
       -0.05950789,  0.48349503],
      [-1.07149816,  0.91615037,  1.76647284,  0.30687169,  1.89571252,
       1.2166314 ,  2.39097051,  2.02589289, -0.75129989, -1.17748646,
       0.46268202,  0.06765308],
      [-1.46909629, -0.66132709, -0.08695546, -0.14861426,  0.59638244,
       -2.24202583,  2.36059778, -0.78565928, -0.6844116 ,  0.23956403,
       0.67155799, -2.42739866]])

*/
```