COMSATS University Islamabad,
Lahore Campus

Assignment-4

Course Title: Introduction to Data Science
Course Code: CSC461

Name: Ayesha Tariq
Roll Number: SP20-BCS-020-B
Group: IV
Semester: 6th
Program Name: Bachelor of Computer Science
Submitted To: Dr. Muhammad Sharjeel

Q1: Provide responses to the following questions about the dataset.

1. **How many instances does the dataset contain?**
   There are 80 instances in the data set. As we can see, after running it shows 80 rows.

```
In [5]: df = pd.read_csv(r'C:\Users\hp\Downloads\gender-prediction.csv')
        print(df)
```

```
    height  weight beard hair_length  shoe_size scarf eye_color  gender
0       71     176   yes       short         44    no     black    male
1       68     165    no        bald         41    no     black    male
2       62     132    no      medium         37   yes      blue  female
3       65     138    no        long         38    no      gray  female
4       70     197   yes      medium         43    no      gray    male
..     ...     ...   ...         ...        ...   ...       ...     ...
75      65      99    no       short         39   yes     green  female
76      61      98    no       short         37    no     brown  female
77      67     119   yes       short         40    no     black    male
78      70     190   yes      medium         43    no      gray    male
79      62     142   yes        long         37    no      blue  female

[80 rows x 8 columns]
```

2. **How many input attributes does the dataset contain?**
   There are 7 input attributes:
   - height
   - weight
   - beard
   - hair_length
   - shoe_size
   - scarf
   - eye_color
3. **How many possible values does the output attribute have?**
   There are two possible values of output attribute "gender", which are:
   - Male
   - Female

   These are encoded as 0 and 1

4. **How many input attributes are categorical?**
   There are four categorical input attributes:
   - beard
   - hair_length
   - scarf
   - eye_color
5. **What is the class ratio (male vs female) in the dataset?**
   There are 46 male and 34 female and the ratio of male vs female is 23:17.

Q2: Apply Random Forest, Support Vector Machines, and Multilayer Perceptron classification algorithms (using Python) on the gender prediction dataset with standard train/test split ratio and answer the following questions.

1. **How many instances are incorrectly classified?**
   In **Random Forest** classifier:

```
model_cm = metrics.confusion_matrix(Y_test, prediction)
print("The confusion matrix is:\n",model_cm)
```

```
The confusion matrix is:
 [[10  0]
 [ 0 17]]
```

There are zero incorrect classified instances.

In **Support Vector machine**:

```
model_cm = metrics.confusion_matrix(Y_test, prediction)
print("The confusion matrix is:\n",model_cm)
```

```
The confusion matrix is:
 [[ 7  3]
 [ 3 14]]
```

There are six incorrect classified instances.

In **Multilayer Perceptron classification:**

```
In [35]: model_cm = metrics.confusion_matrix(Y_test, prediction)
         print("The confusion matrix is:\n",model_cm)
```

```
The confusion matrix is:
 [[ 0 11]
 [ 0 16]]
```

There are eleven incorrect classified instances.

2. **Rerun the experiment using train/test split ratio of 80/20. Do you see any change in the results? Explain.**

Yes, there are changes in the experiment:

Random Forest:

<table>
<tr><th>67/33</th><th>80/20</th></tr>
<tr><td>

Confusion Matrix:

<pre>
[[10 0]
 [0 17]]
</pre>

Classification Report:

<pre>
              precision   recall f1-score   support

           0   1.00   1.00   1.00   10
           1   1.00   1.00   1.00   17

    accuracy   1.00   27
   macro avg   1.00   1.00   1.00   27
weighted avg   1.00   1.00   1.00   27
</pre>

</td><td>

Confusion Matrix:

<pre>
[[6 0]
 [0 10]]
</pre>

Classification Matrix:

<pre>
              precision   recall f1-score   support

           0   1.00   1.00   1.00   6
           1   1.00   1.00   1.00   10

    accuracy   1.00   16
   macro avg   1.00   1.00   1.00   16
weighted avg   1.00   1.00   1.00   16
</pre>

</td></tr>
</table>

SVC:

67/33	80/20

Confusion Matrix:

```
[[ 7  3]
 [ 3 14]]
```

Classification Report:

	precision	recall	f1-score	support
0	0.70	0.70	0.70	10
1	0.82	0.82	0.82	17
accuracy			0.78	27
macro avg	0.76	0.76	0.76	27
weighted avg	0.78	0.78	0.78	27

Confusion Matrix:

```
[[4 2]
 [1 9]]
```

Classification Report:

	precision	recall	f1-score	support
0	0.80	0.67	0.73	6
1	0.82	0.90	0.86	10
accuracy			0.81	16
macro avg	0.81	0.78	0.79	16
weighted avg	0.81	0.81	0.81	16

MLP:

67/33

Confusion Matrix:

```
[[ 8  4]
 [ 0 15]]
```

Classification Report:

	precision	recall	f1-score	support
0	1.00	0.67	0.80	12
1	0.79	1.00	0.88	15
accuracy			0.85	27
macro avg	0.89	0.83	0.84	27
weighted avg	0.88	0.85	0.85	27

80/20

Confusion Matrix:

```
[[ 6  0]
 [ 0 10]]
```

Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	6
1	1.00	1.00	1.00	10
accuracy			1.00	16
macro avg	1.00	1.00	1.00	16
weighted avg	1.00	1.00	1.00	16

3. Name 2 attributes that you believe are the most “powerful” in the prediction task. Explain why?

According to me, beard and scarf are the most powerful instances. Because if Beard= yes, then gender = male, and if scarf=yes, then gender=female. There are no instance in which Beard= yes, then gender = female and if scarf=yes, then gender=male.

4. Try to exclude these 2 attribute(s) from the dataset. Rerun the experiment (using 80/20 train/test split), did you find any change in the results? Explain.

No, there is no change.

Random Forest:

Before

Confusion Matrix:

```
[[ 6  0]
 [ 0 10]]
```

After

Confusion Matrix:

```
[[ 6  0]
 [ 0 10]]
```

Classification Report:

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         6
           1       1.00      1.00      1.00        10

    accuracy                           1.00        16
   macro avg       1.00      1.00      1.00        16
weighted avg       1.00      1.00      1.00        16
```

Classification Report:

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         6
           1       1.00      1.00      1.00        10

    accuracy                           1.00        16
   macro avg       1.00      1.00      1.00        16
weighted avg       1.00      1.00      1.00        16
```

SVC:

Before

Confusion Matrix:

```
[[4 2]
 [1 9]]
```

Classification Report:

```
              precision    recall  f1-score   support

           0       0.80      0.67      0.73         6
           1       0.82      0.90      0.86        10

    accuracy                           0.81        16
   macro avg       0.81      0.78      0.79        16
weighted avg       0.81      0.81      0.81        16
```

After

Confusion Matrix:

```
[[4 2]
 [1 9]]
```

Classification Report:

```
              precision    recall  f1-score   support

           0       0.80      0.67      0.73         6
           1       0.82      0.90      0.86        10

    accuracy                           0.81        16
   macro avg       0.81      0.78      0.79        16
weighted avg       0.81      0.81      0.81        16
```

MLP:

Before

Confusion Matrix:

```
[[ 6  0]
 [ 0 10]]
```

Classification Report

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         6
           1       1.00      1.00      1.00        10

    accuracy                           1.00        16
   macro avg       1.00      1.00      1.00        16
weighted avg       1.00      1.00      1.00        16
```

After

Confusion Matrix:

```
[[ 6  0]
 [ 0 10]]
```

Classification Report:

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         6
           1       1.00      1.00      1.00        10

    accuracy                           1.00        16
   macro avg       1.00      1.00      1.00        16
weighted avg       1.00      1.00      1.00        16
```

Q3: Apply Decision Tree Classifier classification algorithm (using Python) on the gender prediction dataset with Monte Carlo cross-validation and Leave P-Out cross-validation. Report F1 score for both cross-validation strategies.

Note: You are free to choose any parameter values for both cross-validation strategies, however, you have to provide these values in your submission document.

(On Notebook)

Q4: Add 5 sample instances into the dataset (you can ask your friends/relatives/sibling for the data). Rerun the ML experiment (using Python) by training the model using Gaussian Naïve Bayes

classification algorithm and all the instances from the gender prediction dataset. Evaluate the trained model using the newly added test instances. Report accuracy, precision, and recall scores.

Note: You have to add the test instances in your assignment submission document

In [144]:
```
model_cm = metrics.confusion_matrix(Y_test, prediction)
print("The confusion matrix is:\n",model_cm)
```

```
The confusion matrix is:
 [[10  1]
 [ 0 18]]
```

In [145]:
```
model_cl_rep = metrics.classification_report(Y_test, prediction)
print(model_cl_rep)
```

```
              precision    recall  f1-score   support

           0       1.00      0.91      0.95        11
           1       0.95      1.00      0.97        18

    accuracy                           0.97        29
   macro avg       0.97      0.95      0.96        29
weighted avg       0.97      0.97      0.97        29
```

In [147]:
```
model_acc = accuracy_score(Y_test, prediction)*100
print("Accuracy score of the model is:",model_acc)
```

```
Accuracy score of the model is: 96.55172413793103
```