

CLASSIFICATION ASSIGNMENT

1. Stage 1: Machine Learning (Inputs are in Number)
Stage 2: Supervised Learning (Inputs and Outputs are well known)
Stage 3: Binary Classification (Outputs are in Yes or No Format)
2. 399 Rows
25 Columns
3. One Hot Encoding – To convert String into Number
4. Here we are using Logistic Regression, KNN, Navie Bayes, SVM Classification, Decision Tree Classification and Random Forest Classification algorithms to find Best Model.
5. 1) Logistic Regression

```
print("The report:\n",clf_report)
```

The report:

	precision	recall	f1-score	support
0	0.98	1.00	0.99	51
1	1.00	0.99	0.99	82
accuracy			0.99	133
macro avg	0.99	0.99	0.99	133
weighted avg	0.99	0.99	0.99	133

Classification Report of Logistic Regression

Overall Performance of this model is **0.99**

CLASSIFICATION ASSIGNMENT

2) KNN Classifier

```
print("The report:\n",clf_report)
```

The report:

	precision	recall	f1-score	support
0	0.93	1.00	0.96	51
1	1.00	0.95	0.97	82
accuracy			0.97	133
macro avg	0.96	0.98	0.97	133
weighted avg	0.97	0.97	0.97	133

Classification Report of KNN Classifier

Overall Performance of this model is **0.97**

3) Navie Bayes

I. MultinomialNB

Fitting 5 folds for each of 8 candidates, totalling 40 fits
{'alpha': 0.01, 'fit_prior': True}

Confusion Matrix:

```
[[50  1]
 [14 68]]
```

Classification Report:

	precision	recall	f1-score	support
0	0.78	0.98	0.87	51
1	0.99	0.83	0.90	82
accuracy			0.89	133
macro avg	0.88	0.90	0.89	133
weighted avg	0.91	0.89	0.89	133

CLASSIFICATION ASSIGNMENT

Overall Performance of this model is **0.89**

II. BernoulliNB

Fitting 5 folds for each of 24 candidates, totalling 120 fits
{'alpha': 0.01, 'binarize': 0.0, 'fit_prior': True}

Confusion Matrix:

```
[[51  0]
 [ 3 79]]
```

Classification Report:

	precision	recall	f1-score	support
0	0.94	1.00	0.97	51
1	1.00	0.96	0.98	82
accuracy			0.98	133
macro avg	0.97	0.98	0.98	133
weighted avg	0.98	0.98	0.98	133

Overall Performance of this model is **0.98**

III. ComplementNB

Fitting 5 folds for each of 16 candidates, totalling 80 fits
{'alpha': 0.01, 'fit_prior': True, 'norm': False}

Confusion Matrix:

```
[[50  1]
 [14 68]]
```

Classification Report:

	precision	recall	f1-score	support
0	0.78	0.98	0.87	51
1	0.99	0.83	0.90	82
accuracy			0.89	133
macro avg	0.88	0.90	0.89	133
weighted avg	0.91	0.89	0.89	133

Overall Performance of this model is **0.89**

CLASSIFICATION ASSIGNMENT

IV. GaussianNB

Confusion Matrix:

```
[[51  0]
 [ 3 79]]
```

Classification Report:

	precision	recall	f1-score	support
0	0.94	1.00	0.97	51
1	1.00	0.96	0.98	82
accuracy			0.98	133
macro avg	0.97	0.98	0.98	133
weighted avg	0.98	0.98	0.98	133

Overall Performance of this model is **0.98**

4) SVM Classification

```
print("The report:\n",clf_report)
```

The report:

	precision	recall	f1-score	support
0	0.98	1.00	0.99	51
1	1.00	0.99	0.99	82
accuracy			0.99	133
macro avg	0.99	0.99	0.99	133
weighted avg	0.99	0.99	0.99	133

Overall Performance of this model is **0.99**

CLASSIFICATION ASSIGNMENT

5) Decision Tree Classification

```
print("The report:\n",clf_report)
```

The report:

	precision	recall	f1-score	support
0	0.96	1.00	0.98	51
1	1.00	0.98	0.99	82
accuracy			0.98	133
macro avg	0.98	0.99	0.98	133
weighted avg	0.99	0.98	0.99	133

Overall Performance of this model is **0.98**

6) Random Forest Classification

```
print("The report:\n",clf_report)
```

The report:

	precision	recall	f1-score	support
0	0.96	0.98	0.97	51
1	0.99	0.98	0.98	82
accuracy			0.98	133
macro avg	0.97	0.98	0.98	133
weighted avg	0.98	0.98	0.98	133

Overall Performance of this model is **0.98**

6. Two models (Logistic Regression and SVM Classification) are giving 0.99 accuracy. I have chosen **Logistic Regression** for deployment due to its **interpretability, speed, lower complexity, and reliability**.