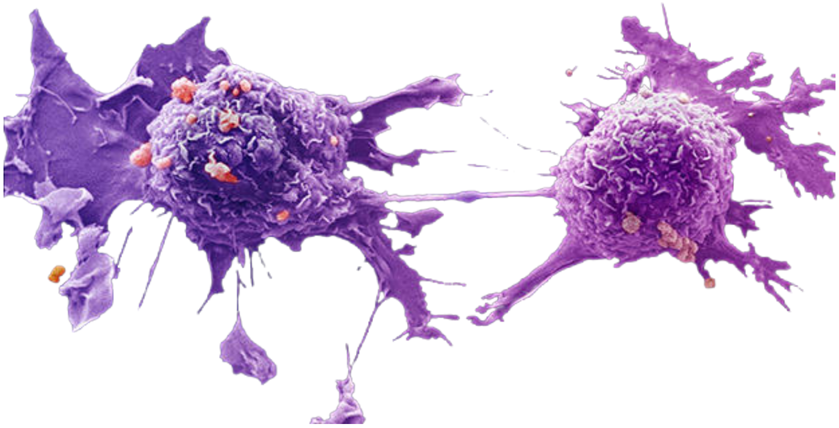
Tumor Cancer Prediction



Team members:

Salma Ayman Mohamed El-Sayed Alassal
ID : 20201700345

Mariam Ahmed Ismail Mahmoud
ID : 20201700800

Mohamed Nasr Abd-Alazem Ahmed Ali
ID : 20201700749

Abdelrahman Ashraf Fathy Hassan
ID : 20201700430

Mohamed Salah Ahmed Mansour Abdelrahman
ID : 20201700696

Ahmed Ayman Ameen Mohamed El-Badawy
ID : 20201700027

Dr. Dina El-Sayad
Dr. Alaa Tarek

May-2022

Table of Contents

Introduction

A tumor is an abnormal lump or growth of cells. When the cells in the tumor are normal, it is benign. Then something just went wrong, and they overgrew and produced a lump. When the cells are abnormal and can grow uncontrollably, then they are cancerous cells, and the tumor is malignant.

The early diagnosis of the tumor can improve the prognosis and chance of treatment significantly, as it can promote timely clinical treatment to patients. Further accurate classification of benign tumors can prevent patients undergoing unnecessary treatments.

Objective

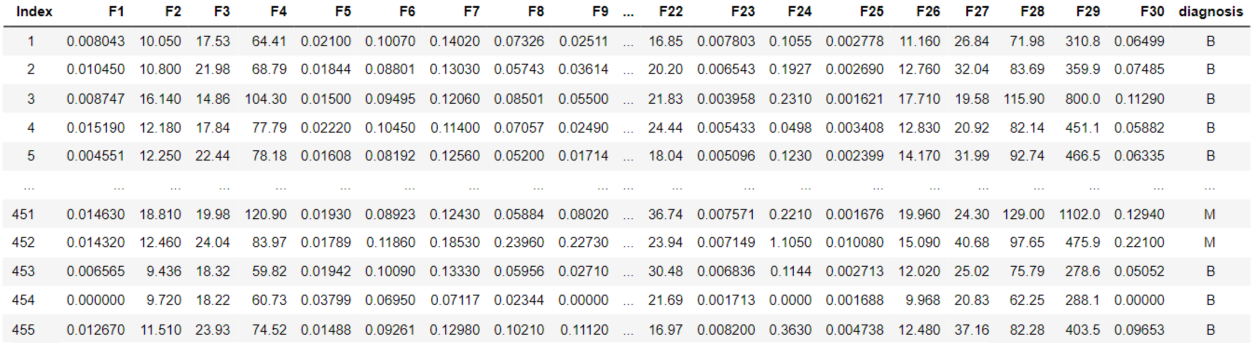
The main objective of this project is to develop a machine learning model to predict the patient diagnosis based on the given features using **SVM, Logistic Regression, Naive Bayes, KNN** and **ID3** algorithms.

Dataset

The dataset contains 455 row each row consists of **30** independent features(F1 -> F30) and 1 dependent feature (The diagnosis).

The diagnosis column contains 2 values : **M** = malignant , **B** = benign.

Dataset Snapshot



Index	F1	F2	F3	F4	F5	F6	F7	F8	F9	...	F22	F23	F24	F25	F26	F27	F28	F29	F30	diagnosis
1	0.008043	10.050	17.53	64.41	0.02100	0.10070	0.14020	0.07326	0.02511	...	16.85	0.007803	0.1055	0.002778	11.160	26.84	71.98	310.8	0.06499	B
2	0.010450	10.800	21.98	68.79	0.01844	0.08801	0.13030	0.05743	0.03614	...	20.20	0.006543	0.1927	0.002690	12.760	32.04	83.69	359.9	0.07485	B
3	0.008747	16.140	14.86	104.30	0.01500	0.09495	0.12060	0.08501	0.05500	...	21.83	0.003958	0.2310	0.001621	17.710	19.58	115.90	800.0	0.11290	B
4	0.015190	12.180	17.84	77.79	0.02220	0.10450	0.11400	0.07057	0.02490	...	24.44	0.005433	0.0498	0.003408	12.830	20.92	82.14	451.1	0.05882	B
5	0.004551	12.250	22.44	78.18	0.01608	0.08192	0.12560	0.05200	0.01714	...	18.04	0.005096	0.1230	0.002399	14.170	31.99	92.74	466.5	0.06335	B
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
451	0.014630	18.810	19.98	120.90	0.01930	0.08923	0.12430	0.05884	0.08020	...	36.74	0.007571	0.2210	0.001676	19.960	24.30	129.00	1102.0	0.12940	M
452	0.014320	12.460	24.04	83.97	0.01789	0.11860	0.18530	0.23960	0.22730	...	23.94	0.007149	1.1050	0.010080	15.090	40.68	97.65	475.9	0.22100	M
453	0.006565	9.436	18.32	59.82	0.01942	0.10090	0.13330	0.05956	0.02710	...	30.48	0.006836	0.1144	0.002713	12.020	25.02	75.79	278.6	0.05052	B
454	0.000000	9.720	18.22	60.73	0.03799	0.06950	0.07117	0.02344	0.00000	...	21.69	0.001713	0.0000	0.001688	9.968	20.83	62.25	288.1	0.00000	B
455	0.012670	11.510	23.93	74.52	0.01488	0.09261	0.12980	0.10210	0.11120	...	16.97	0.008200	0.3630	0.004738	12.480	37.16	82.28	403.5	0.09653	B

Data Preparation

Data Reduction

It is a process that reduces the volume of original data and represents it in a much smaller volume.

Index column is an irrelevant data that is not actually needed, and doesn't fit under the context of the problem (patient diagnosis prediction) we're trying to solve so this column will be dropped using the drop() method.

Handling Missing Values

Missing or null values means there is no value present at all. In order to validate that the dataset doesn't contain null values we will use isnull().sum() method which will show that data is free of nulls.

Mapping

The Diagnosis column contains text data but our machine could not understand the text data so we need to convert it into numerical data. For this conversion, we use map() method on the diagnosis column to convert B and M to 0 and 1 respectively.

Train and test

We need to split our dataset into **dependent** and **independent** variables where independent variables (F1 -> F30) contain the columns which are used to predict the result but they are not dependent on each other, the dependent variable (diagnosis) contains the column for which the result needs to be predicted (benign or malignant) .

Sklearns's train_test_split() is used here for splitting of data. We will put **75%** (341 row) of the data in the **training set** and **25%** (114 row) in the **test set.**

Random State

If there is no random state provided, the system will use a random state which is generated internally. So whenever we will run the code we may see different results, and in order to avoid that we will put the random state equal to **1**.

Feature Scaling

Feature Scaling is a technique to standardize the independent features present in the data in a **fixed range**.

Since the range of values of data may vary widely, in some machine learning algorithms, objective functions will not work properly without normalization.

Using **normalization technique** in which values are shifted and rescaled so that they end up ranging between 0 and 1 our data will present in a fixed range.

Algorithm	Accuracy	
—	**Normalization**	**No Normalization**
SVM	97%	94%
Logistic Regression	97%	94%
ID3	96%	96%
Naive Bayes	95%	96%
KNN	96%	93%

Feature Selection

The objective of feature selection is finding out the best subset of attributes which better explains the relationship of independent variables with target variable so to achieve that we will use **correlation coefficient** technique.

Correlation Coefficient

Correlation is a measure of the linear relationship of 2 or more variables. Through correlation, we can predict one variable from the other. The logic behind using correlation for feature selection is that the good variables are highly correlated with the target.

Correlation statistics

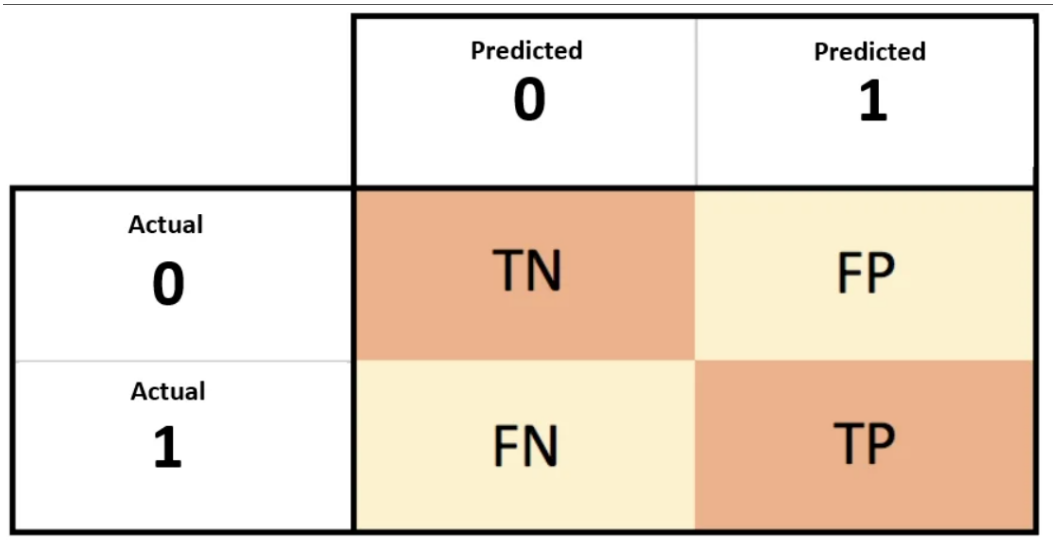
Feature	Correlation	Feature	Correlation
F1	r = 0.38 (Moderate)	F16	r = 0.72 (Strong)
F2	r = 0.72 (Strong)	F17	r = 0.56 (Moderate)
F3	r = 0.41 (Moderate)	F18	r = 0.42 (Moderate)
F4	r = 0.74 (Strong)	F19	r = 0.6 (Moderate)
F5	r = - 0.01 (Weak)	F20	r = - 0.02 (Weak)
F6	r = 0.36 (Moderate)	F21	r = 0.55 (Moderate)
F7	r = 0.42 (Moderate)	F22	r = 0.54 (Moderate)
F8	r = 0.61 (Strong)	F23	r = - 0.09 (Weak)
F9	r = 0.69 (Strong)	F24	r = 0.66 (Strong)
F10	r = 0.2 (Weak)	F25	r = 0.06 (Weak)
F11	r = 0.78 (Strong)	F26	r = 0.77 (Strong)
F12	r = 0.28 (Weak)	F27	r = 0.45 (Moderate)
F13	r = 0.32 (Moderate)	F28	r = 0.78 (Strong)
F14	r = 0.33 (Moderate)	F29	r = 0.7 (Strong)
F15	r = - 0.03 (Weak)	F30	r = 0.79 (Strong)

It can be seen that F5, F10, F12, F15, F20, F23 and F25 have weak relations so in order to improve the **accuracy** and achieve **lower computational cost** they will be dropped.

Evaluation

Confusion Matrix :

It represents counts from predicted and actual values (is used to define the performance of a classification algorithm).



	Predicted 0	Predicted 1
Actual 0	TN	FP
Actual 1	FN	TP

The Mean Squared Error :

The Mean Squared Error , Which tells you how close a regression line is to a set of points. It does this by taking the distances from the points to the regression line (these distances are the “errors”) and squaring them. The squaring is necessary to remove any negative signs. **The lower the MSE, the better the forecast.**

Accuracy :

Accuracy is the number of correctly predicted data points out of all the data points (can be computed by comparing actual and predicted values ,How well the model explains the data it was trained with).

Recall :

Recall or the **true positive rate** is the number of positive samples that are correctly classified as ‘positive’. If all of them are identified correctly, then recall will be 1. If all of them were classified incorrectly, then recall will be 0. With some positive samples classified as negative, recall with be in between 0 and 1

Precision :

Precision is one indicator of a machine learning model's performance – **the quality of a positive prediction made by the model.** It refers to the number of true positives divided by the total number of positive predictions (true positive + false positive).

Classification report :

Classification report which is used to measure the quality of predictions from the classification algorithm.

F1 Score :

F1-score is one of the most important evaluation metrics in machine learning . It sums up the predictive performance of a model by combining two otherwise competing metrics — precision and recall . F1 scores can range from **0 to 1**, with 1 representing a model that perfectly classifies each observation into the correct class and 0 representing a model that is unable to classify any observation into the correct class .

Machine Learning Algorithms

The machine learning algorithms applied are :

- Logistic Regression
- SVM
- ID3
- Naive Bayes
- KNN

Using random state = **1**

Algorithm	Accuracy	Mean Squared Error
SVM	97%	3%
Logistic Regression	97%	3%
ID3	96%	4%
Naive Bayes	95%	5%
KNN	96%	4%

Confusion Matrix

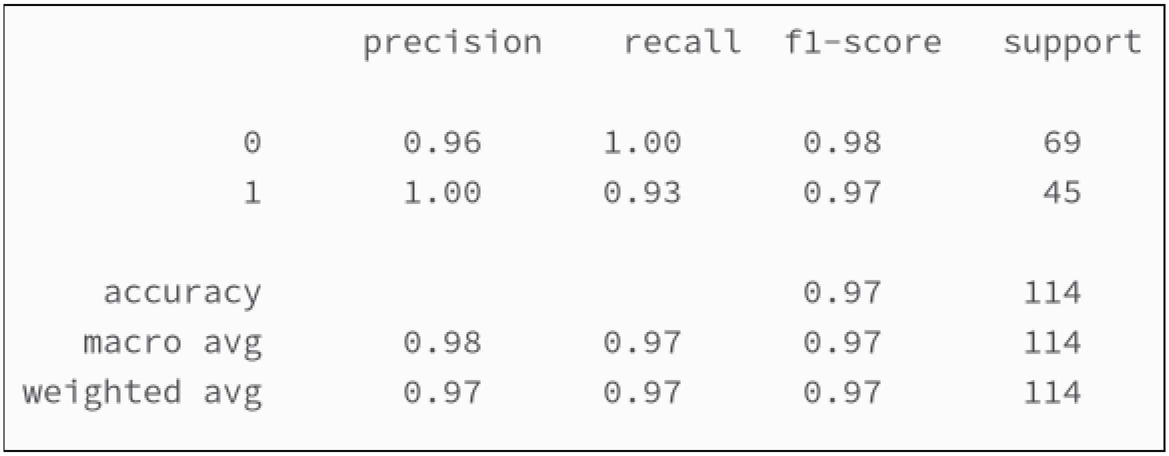
<table>
<tr><td rowspan="2">SVM</td><td>69</td><td>0</td><td rowspan="3">Naive Bayes</td><td rowspan="2">67</td><td rowspan="2">2</td></tr>
<tr><td>3</td><td>42</td></tr>
<tr><td rowspan="2">Logistic Regression</td><td>69</td><td>0</td><td>4</td><td>41</td></tr>
<tr><td>3</td><td>42</td><td rowspan="3">KNN</td><td>67</td><td>2</td></tr>
<tr><td rowspan="2">ID3</td><td>68</td><td>4</td><td rowspan="2">3</td><td rowspan="2">42</td></tr>
<tr><td>4</td><td>41</td></tr>
</table>

Logistic Regression

Logistic regression is a type of regression analysis we use when the response variable is binary.

Logistic regression was used to analyze the relationship between independent variables (F1 -> F30) and the dependent variable (diagnosis) to know the patient diagnosis based on the given features.

ACCURACY
96%
98%
97%
95%

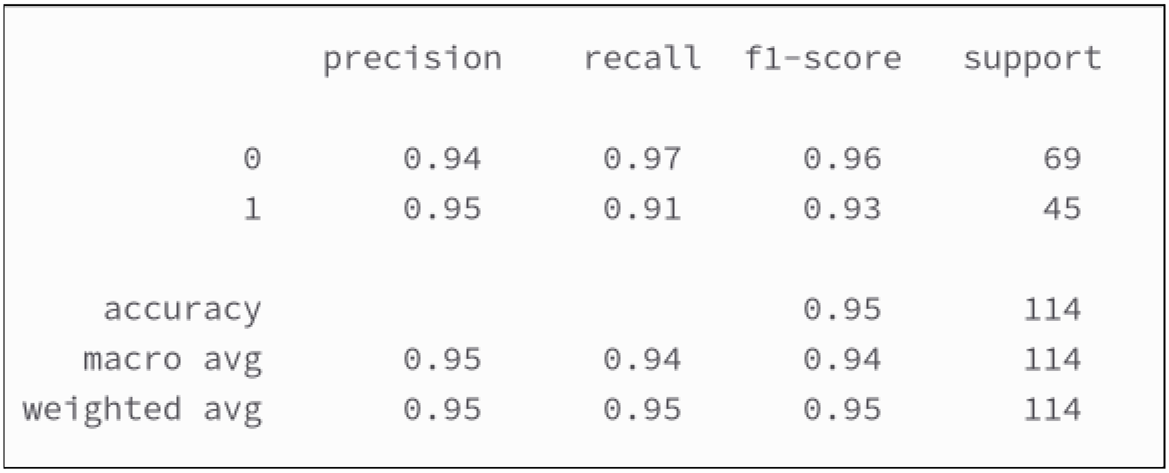


```
              precision    recall  f1-score   support

           0       0.96      1.00      0.98        69
           1       1.00      0.93      0.97        45

    accuracy                           0.97       114
   macro avg       0.98      0.97      0.97       114
weighted avg       0.97      0.97      0.97       114
```

Naive Bayes

Naive Bayes is a probabilistic machine learning algorithm based on the **Bayes Theorem**, used in a wide variety of classification tasks, with an assumption of independence of the features.

ACCURACY
97%
94%
92%
98%



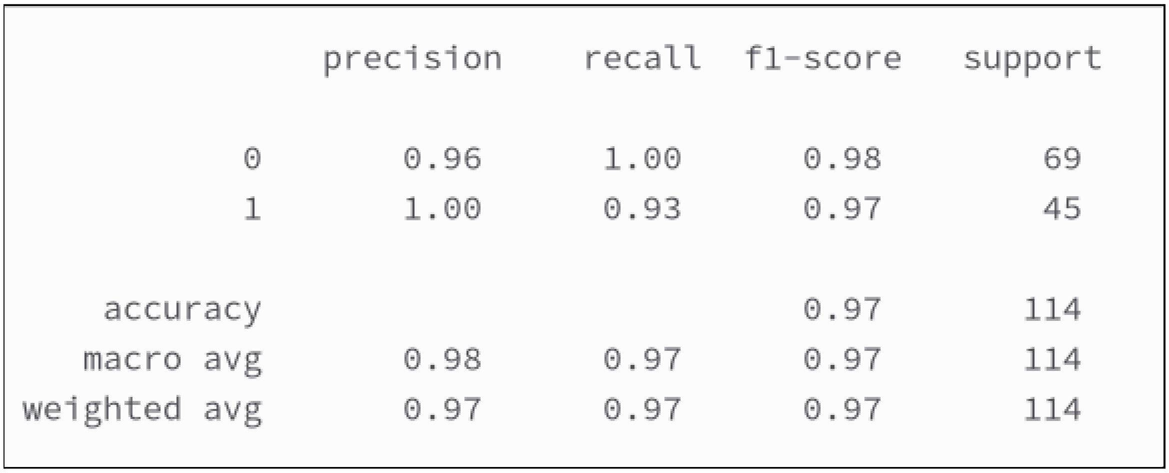
```
              precision    recall  f1-score   support

           0       0.94      0.97      0.96        69
           1       0.95      0.91      0.93        45

    accuracy                           0.95       114
   macro avg       0.95      0.94      0.94       114
weighted avg       0.95      0.95      0.95       114
```

Support Vector Machine (SVM)

SVM works by mapping data to a high-dimensional feature space so that data points can be categorized, even when the data are not otherwise linearly separable. A separator between the categories is found, then the data are transformed in such a way that the separator could be drawn as a hyperplane.

This separator is placed so as to separate the two classes with the maximum possible margin, and as the data is not linearly distributed, Kernel function does the job by adding a higher dimension to the features to make it possible to split the samples.

ACCURACY
97%
96%
94%
99%



```
              precision    recall  f1-score   support

           0       0.96      1.00      0.98        69
           1       1.00      0.93      0.97        45

    accuracy                           0.97       114
   macro avg       0.98      0.97      0.97       114
weighted avg       0.97      0.97      0.97       114
```

Decision Tree (ID3)

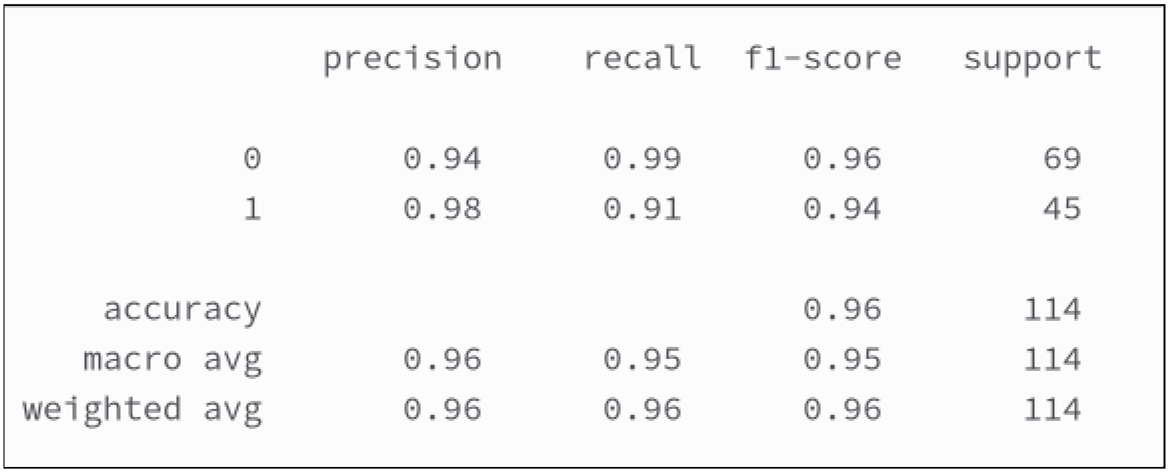
A Decision Tree is a map of the possible outcomes of a series of related choices. It is easy to interpret and visualize.

It also requires fewer data preprocessing from the user, for example, there is no need to normalize columns.

It can be used for feature engineering such as predicting missing values, suitable for variable selection.

The decision tree has no assumptions about distribution because of the non-parametric nature of the algorithm.

ACCURACY
94%
95%
97%
96%

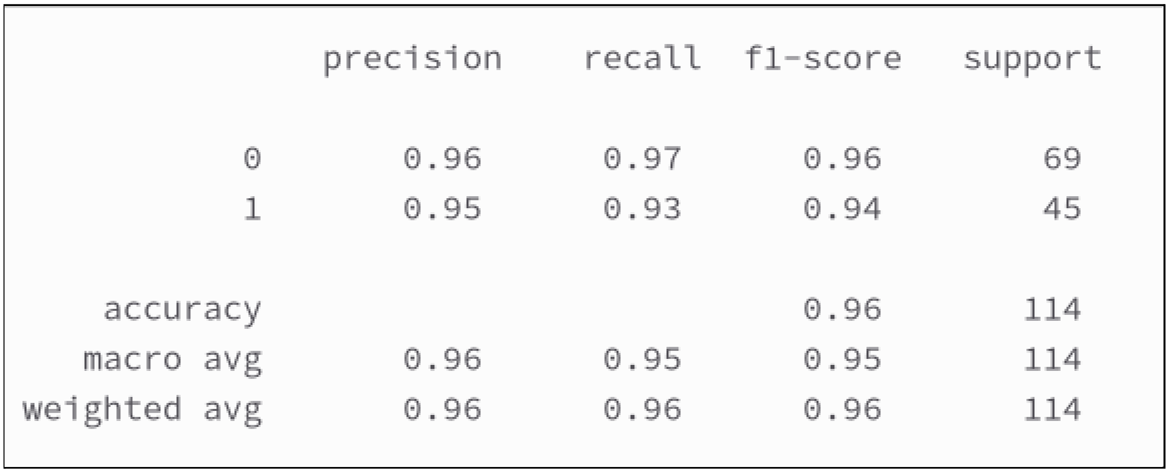


```
              precision    recall  f1-score   support

           0       0.94      0.99      0.96        69
           1       0.98      0.91      0.94        45

    accuracy                           0.96       114
   macro avg       0.96      0.95      0.95       114
weighted avg       0.96      0.96      0.96       114
```

K-Nearest Neighbors Model (KNN)

KNN Is a supervised algorithm that can be used to solve both classification and regression problems, and the principal of KNN is the value or class of a specific data point that is determined by the data points around this value. .

ACCURACY
96%
98%
97%
99%



```
              precision    recall  f1-score   support

           0       0.96      0.97      0.96        69
           1       0.95      0.93      0.94        45

    accuracy                           0.96       114
   macro avg       0.96      0.95      0.95       114
weighted avg       0.96      0.96      0.96       114
```

Algorithm Tuning

The objective of parameter tuning is to find the **optimum value** for each parameter to improve the accuracy of the model so to achieve that we will use **Grid Search technique**.

Using this technique, we simply build a model for some possible combination of some of the hyperparameter values provided, evaluating each model, and selecting the architecture which produces the best results.

Conclusion

- It can be seen that SVM and Logistic regression perform better than Naive Bayes, ID3 and KNN.
- KNN performance depends on the value of K as it gives poor results for lower values of K and best results as the value of K increases.
- Logistic regression model does not work if we have NaN values in the dataset.