]Comparing machine learning algorithms[

/Machine learning/

Name: ahmed Mohamed salem Eslam Mohamed Arafa

DR: Islam EL Kabani
Eng: Rokaya Eltehewy

Description of the dataset and features:

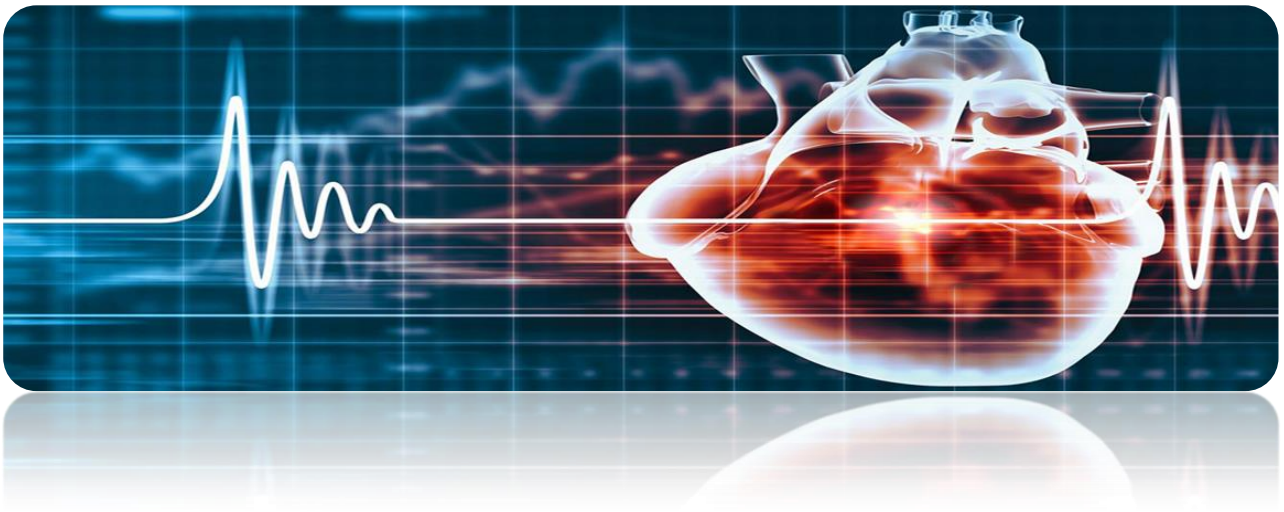


Table of contents:

Introduction:

Cardiovascular diseases (CVDs) are the number 1 cause of death globally, taking an estimated 17.9 million lives each year, which accounts for 31% of all deaths worldwide. Four out of 5CVD deaths are due to heart attacks and strokes, and one-third of these deaths occur prematurely in people under 70 years of age. Heart failure is a common event caused by CVDs and this dataset contains 11 features that can be used to predict a possible heart disease.

People with cardiovascular disease or who are at high cardiovascular risk (due to the presence of one or more risk factors such as hypertension, diabetes, hyperlipidemia or already established disease) need early detection and management wherein a machine learning model can be of great help.

Source:

This dataset was created by combining different datasets already available independently but not combined before. In this dataset, 5 heart datasets are combined over 11 common features which make it the largest heart disease dataset available so far for research purposes. The five datasets used for its curation are:

- Cleveland: 303 observations
- Hungarian: 294 observations
- Switzerland: 123 observations
- Long Beach VA: 200 observations
- Stalog (Heart) Data Set: 270 observations

Total: 1190 observations
Duplicated: 272 observations

Features :

name	Description
age	age of the patient [years]
sex	0: female. 1: male
Chest pain type	chest pain type [TA: Typical Angina, ATA: Atypical Angina, NAP: Non-Angina Pain, ASY: Asymptomatic]
trestbps	resting blood pressure
chol	serum cholesterol
fbs	• fasting blood sugar > 120 mg/dl) • 1 = true 0 = false
restecg	resting electrocardiogram results [Normal: Normal, ST: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV), LVH: showing probable or definite left ventricular hypertrophy by Estes' criteria]
thalach	maximum heart rate achieved
exang	: exercise induced angina • 1 = yes • 0 = no
oldpeak	ST depression induced by exercise relative to rest
slope	the slope of the peak exercise ST segment Value 0: upsloping Value 1: flat Value 2: downsloping
Ca	**number of major vessels (0-3) colored by flourosopy**
Thal	• 1 = fixed defect • 2 = normal

	• 3 = reversable defect Thalassemia
target	• 0 = no disease, • 1 = disease
Smoking	0=no smoking 1= smoking
BMI	The math of the body calculated by (high over weight)

Brief explanation for the machine learning:

1. Logistic Regression

Logistic regression is a type of regression model that is used to predict the probability of a particular event occurring. The advantage of this model is that it can be used to predict binary outcomes, such as whether or not a customer will buy a product. The disadvantage is that it is more complex than linear regression and can be difficult to interpret.

2. Decision Trees

A decision tree is a type of classification model that uses a hierarchy of nodes to classify data. The algorithm starts by identifying a root node, which is the category that all new data will be classified into. From there, the algorithm splits the data into two categories, and determines which category each new piece of data should belong to. This process is repeated until all the data has been classified.

3. Support Vector Machines

Support vector machines (SVMs) are a type of classification model that is similar to decision trees, but has the advantage of being less fragile. Like decision trees, SVMs use a hierarchy of nodes to classify data, but they are able to do this with a much higher accuracy.

4. Naive Bayes

A naive Bayes classifier is an algorithm that uses Bayes' theorem to classify objects. Naive Bayes classifiers assume strong, or naive, independence between attributes of data points. Popular uses of naive Bayes classifiers include spam filters, text analysis and medical

diagnosis. These classifiers are widely used for machine learning because they are simple to implement.

Naive Bayes is also known as simple Bayes or independence Bayes.

5. Neural network

A neural network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates. In this sense, neural networks refer to systems of neurons, either organic or artificial in nature.

Neural networks can adapt to changing input; so the network generates the best possible result without needing to redesign the output criteria. The concept of neural networks, which has its roots in artificial intelligence, is swiftly gaining popularity in the development of trading systems.

6. KNN

K-nearest neighbors (KNN) is a type of supervised learning algorithm used for both regression and classification. KNN tries to predict the correct class for the test data by calculating the distance between the test data and all the training points. Then select the K number of points which is closet to the test data. The KNN algorithm calculates the probability of the test data belonging to the classes of 'K' training data and class holds the highest probability will be selected. In the case of regression, the value is the mean of the 'K' selected training point

7. Random forest:

Random forest is a ***Supervised Machine Learning Algorithm*** that is ***used widely in Classification and Regression problems***. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.

One of the most important features of the Random Forest Algorithm is that it can handle the data set containing ***continuous variables*** as in the case of regression and ***categorical variables*** as in the case of classification. It performs better results for classification problems.

Description of the experiments:

Experiment1: we create logistic regression model to predict the value of the target feature andsplit the data into train and test and train the model using X_train and Y_train and after training we just pass X_test to the model to predict the output of the test's value and compare it with actual values

for checking the accuracy of the model. Finally, we check the values of f1_score, precision, recall, mean_square_error, mean_absoluate_error.

Experiment2: we create naive bays model to predict the value of the target feature and split the data into train and test and train the model using X_train and Y_train and after training we just pass X_test to the model to predict the output of the test's value and compare it with actual values for checking the accuracy of the model. Finally, we check the values of f1_score, precision, recall, mean_square_error, mean_absoluate_error

Experiment3: we create support vector machine model to predict the value of the target feature and split the data into train and test and train the model using X_train and Y_train and after training we just pass X_test to the model to predict the output of the test's value and compare it with actual values for checking the accuracy of the model. Finally, we check the values of f1_score, precision, recall, mean_square_error, mean_absoluate_error

Experiment4: we create KNN model to predict the value of the target feature and split the data into train and test and train the model using X_train and Y_train and after training we just pass X_test to the model to predict the output of the test's value and compare it with actual values for checking the accuracy of the model. Finally, we check the values of f1_score, precision, recall, mean_square_error, mean_absoluate_error.

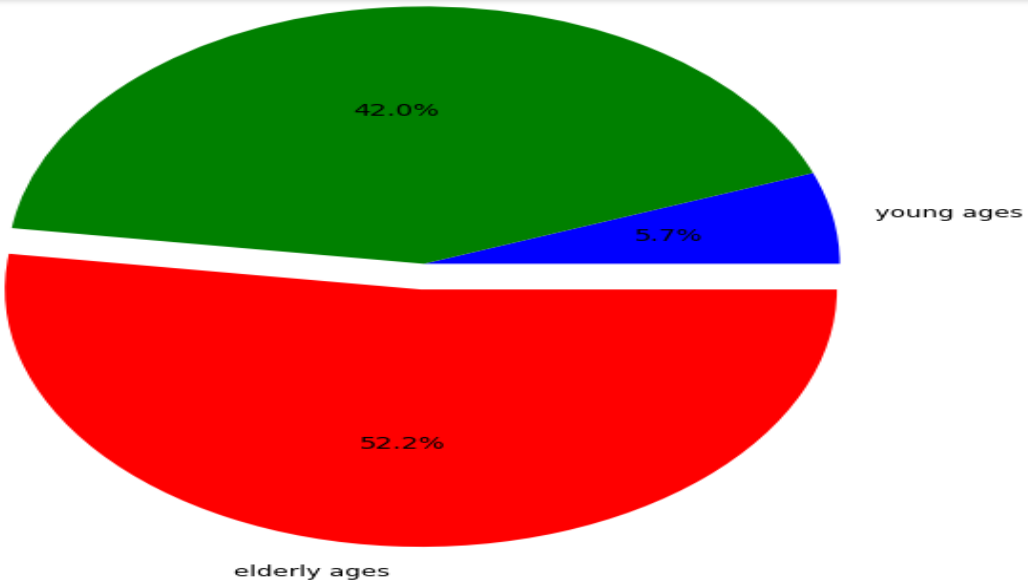
Experiment5: we create Decision tree model to predict the value of the target feature and split the data into train and test and train the model using X_train and Y_train and after training we just pass X_test to the model to predict the output of the test's value and compare it with actual values for checking the accuracy of the model. Finally, we check the values of f1_score, precision, recall, mean_square_error, mean_absoluate_error.

Experiment6: we create Random forest model to predict the value of the target feature and split the data into train and test and train the model using X_train and Y_train and after training we just pass X_test to the model to predict the output of the test's value and compare it with actual values for checking the accuracy of the model. Finally, we check the values of f1_score, precision, recall, mean_square_error, mean_absoluate_error.

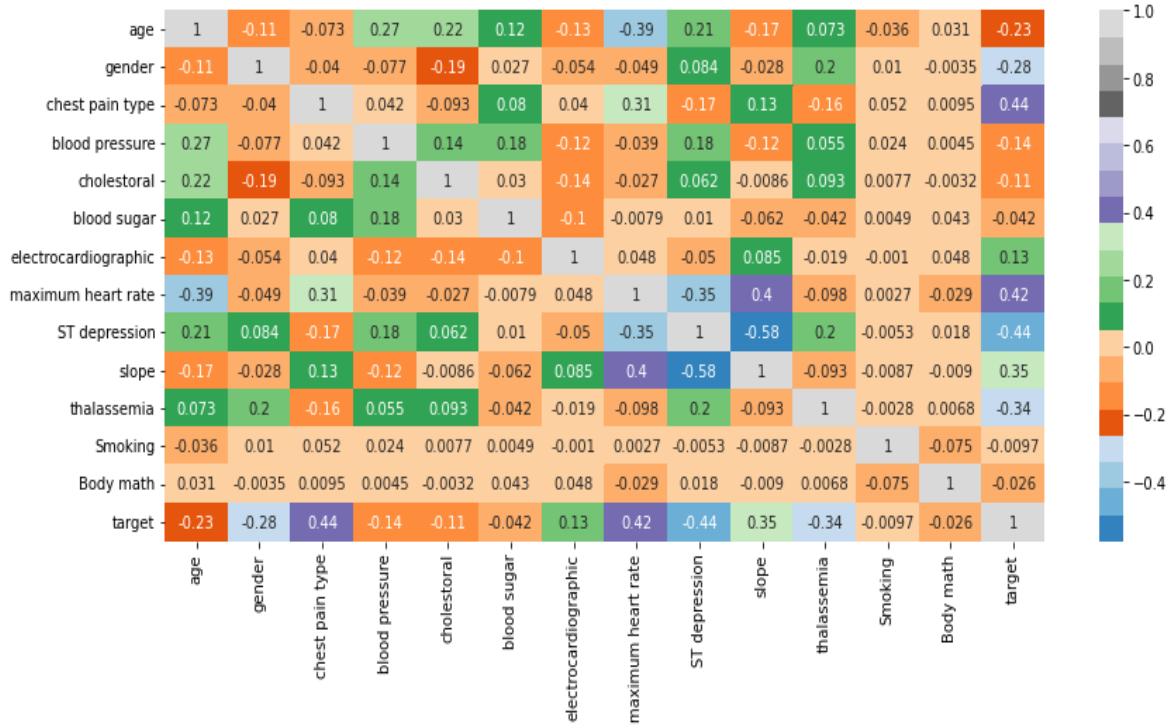
Experiment7: we create Neural network model to predict the value of the target feature and split the data into train and test and train the model using X_train and Y_train and after training we just pass X_test to the model to predict the output of the test's value and compare it with actual values for

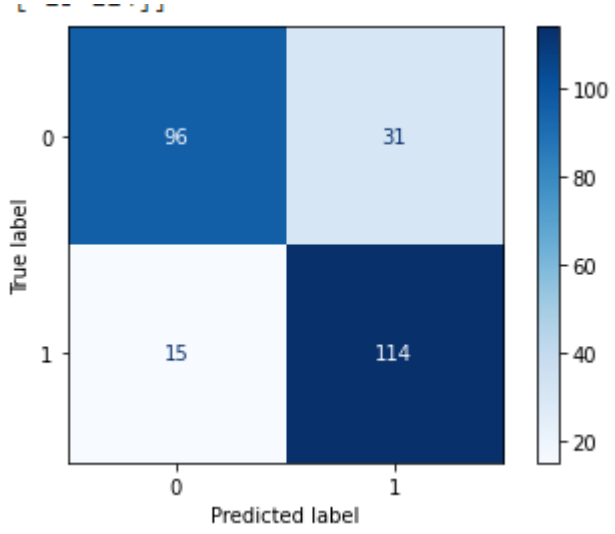
checking the accuracy of the model. Finally, we check the values of f1_score, precision, recall, mean_square_error, mean_absoluate_error.

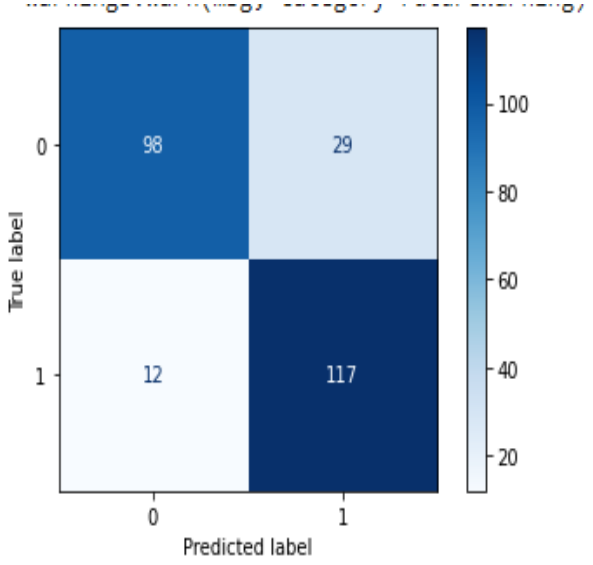
Data visualization:

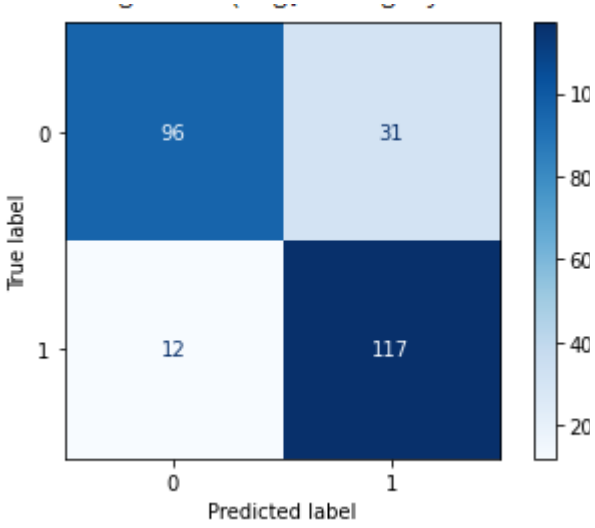


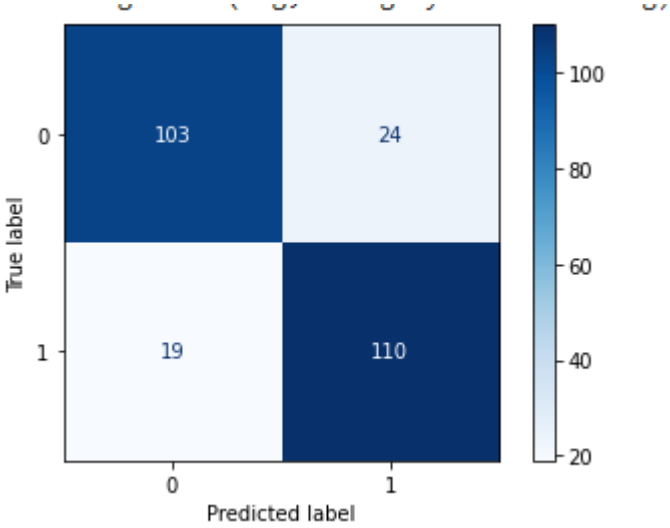

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f6f271d0410>
```

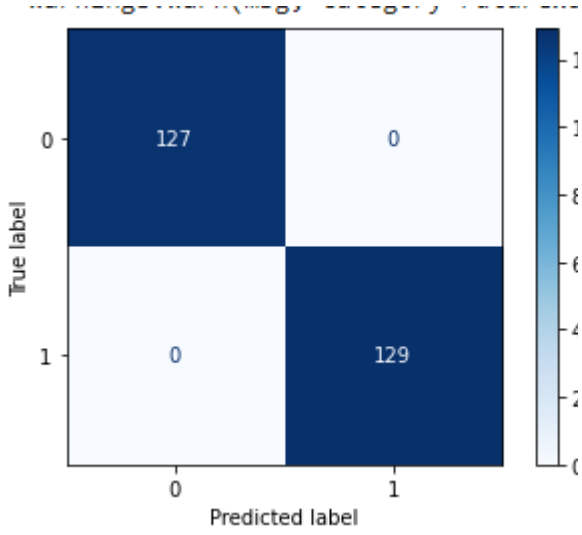


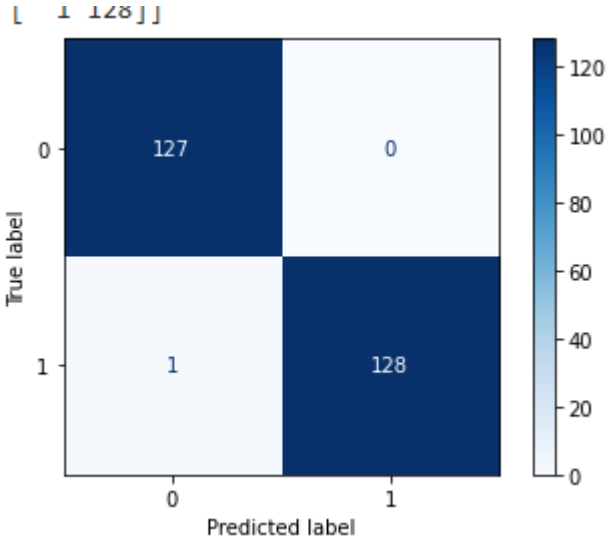

96
31
15
114
0
1
True label
Predicted label
100
80
60
40
20


98
29
12
117
0
1
True label
Predicted label
100
80
60
40
20

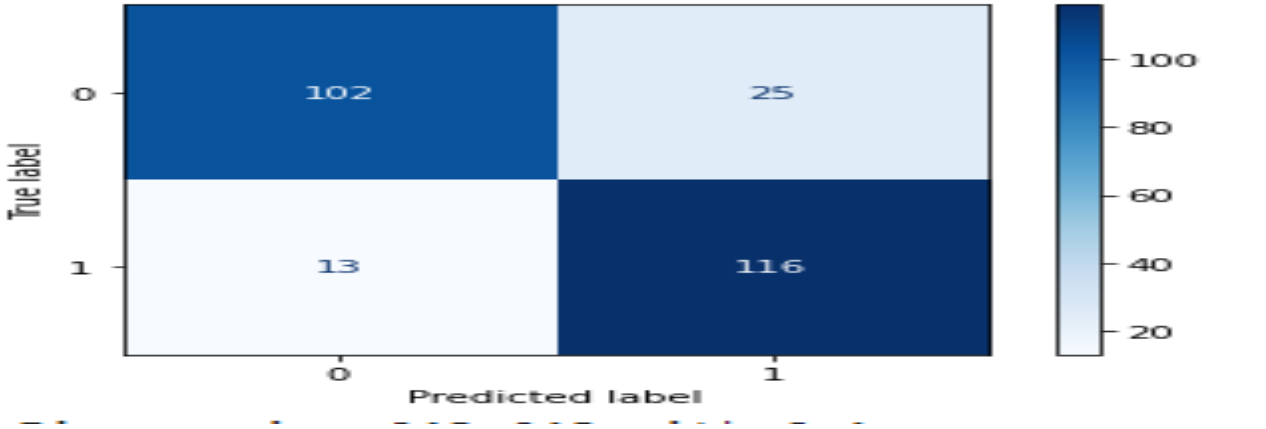

96
31
12
117
0
1
True label
Predicted label

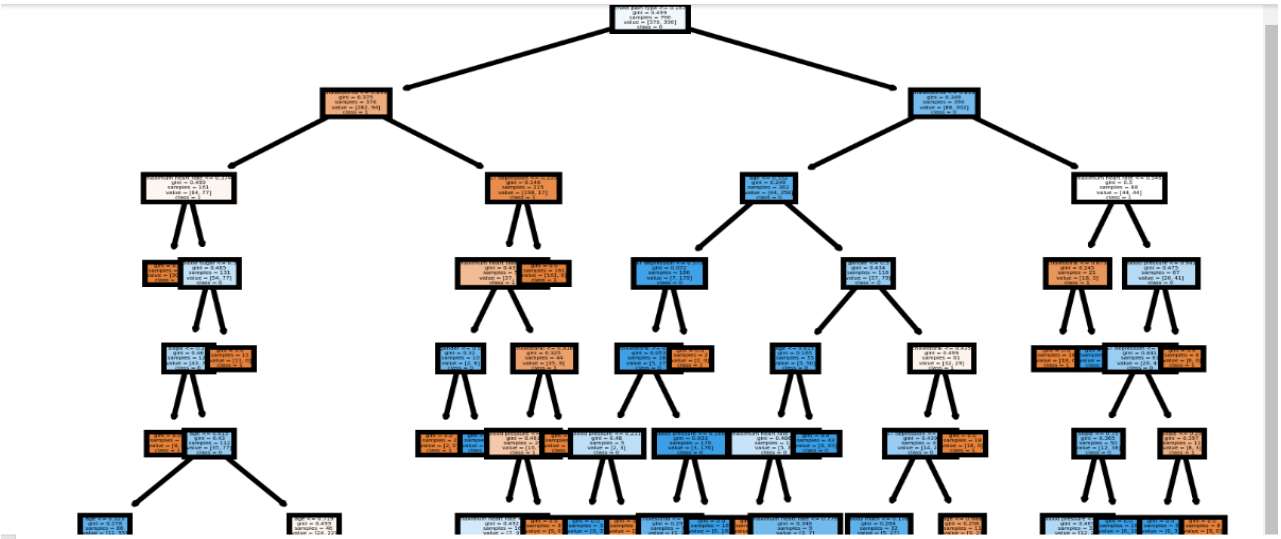

103
24
19
110
0
1
True label
Predicted label
100
80
60
40
20


127
0
0
129
0
1
True label
Predicted label


127
0
1
128
0
1
True label
Predicted label
120
100
80
60
40
20
0

```
warnings.warn(msg, category=FutureWarnir
```





Show the result:

	train_score	test_score	mean_squared_error	mean_absolute_error	f1_score	precision_score	recall_score	accuracy
logistic_regression	0.825065	0.839844	0.160156	0.160156	0.850909	0.801370	0.906977	0.839844
Naive Bayes	0.823760	0.832031	0.160156	0.160156	0.850909	0.801370	0.906977	0.839844
support vector machine	0.835509	0.832031	0.167969	0.167969	0.844765	0.790541	0.906977	0.832031
KNN	0.851175	0.828125	0.171875	0.171875	0.834586	0.810219	0.860465	0.828125
Decision tree.	1.000000	0.980469	0.019531	0.019531	0.980695	0.980695	0.984496	0.980469
Random forest	1.000000	0.996094	0.003906	0.003906	0.996109	0.996109	0.992248	0.996094
Neural network	0.892950	0.851562	0.148438	0.148438	0.859259	0.859259	0.899225	0.851562

The Random forest and Decision tree algorithm is overfiting because the training score is 1.00000 and test score is 99 and 98 and the accuracy is 99 and 98 respectively.

how you can improve the testing results.:

1. Add more data
2. Ensemble methods.

3. Feature Selection.
4. Using cross validation correctly

5. Testing multiple models

THANK YOU