Theoretical Foundations of Machine Learning

**MNIST Project**

|  |  |
| --- | --- |
| Team members | ID |
| Eslam Sameh Hamdy | 20180045 |
| Omar Mostafa Abbas | 20180182 |

**Loading the Data**

We use “loadlocal\_mnist ()” function from “mlxtend” library to load our local dataset then we save each set (train, test) in two arrays (features, labels).

**Apply HOG features**

After loading the data we apply histogram of oriented gradients (HOG) to the images, (which give us just the important features we need to classify an image) using “hog ()” function from “skimage” package and save the hogged images in “numpy” array to use them in the different classification models.

**Classification Models**

* **KNN (K Nearest Neighbors)**
* **SVM (Support Vector Machine)**
* **ANN (Artificial Neural Network)**

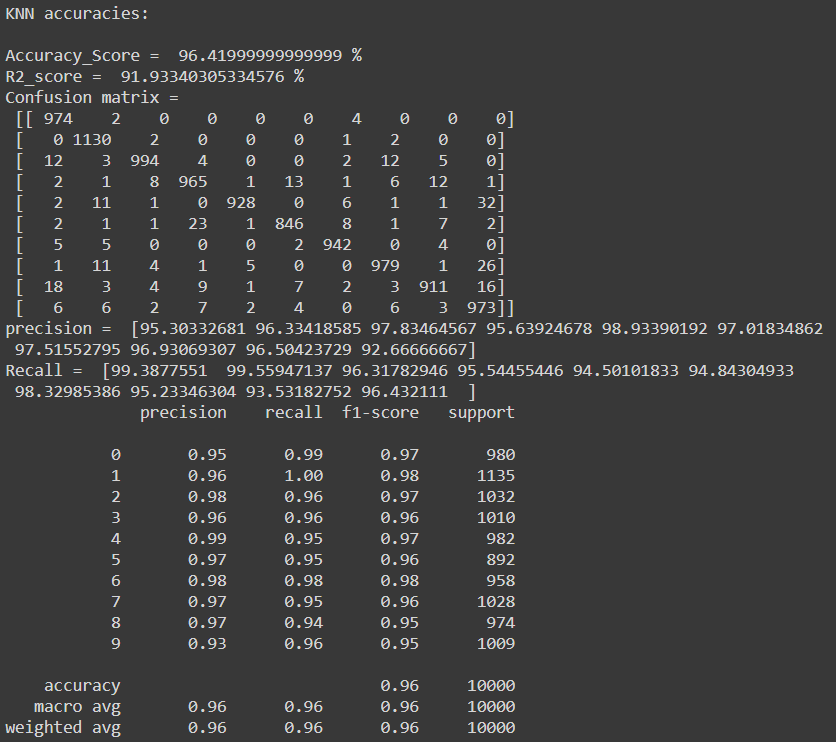
**KNN Model**

**Definition**

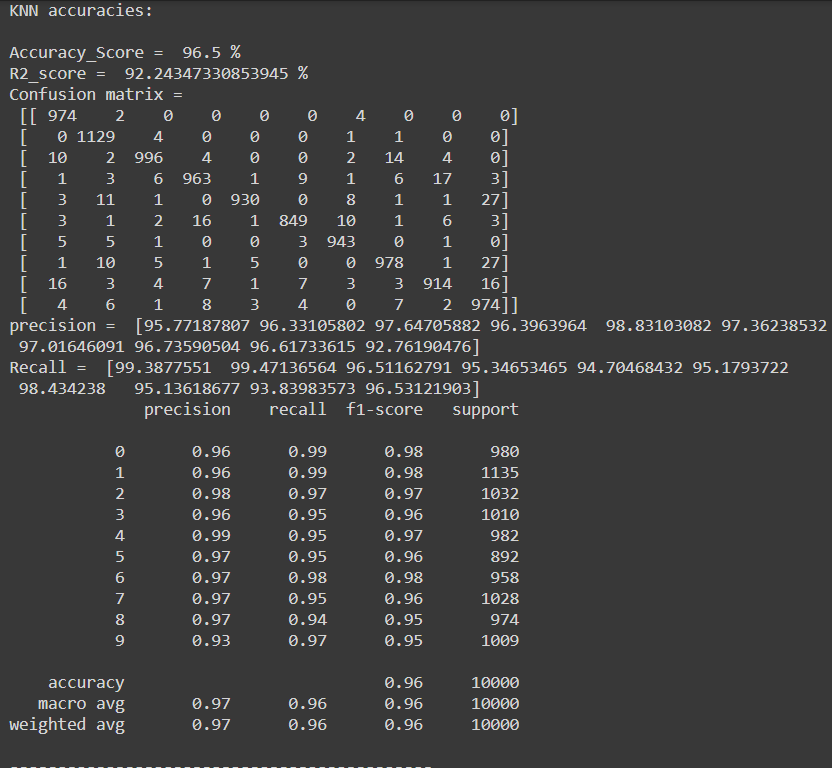
A k-nearest-neighbor algorithm, often abbreviated k-nn, is an approach to data classification that estimates how likely a data point is to be a member of one group or the other depending on what group the data points nearest to it are in.

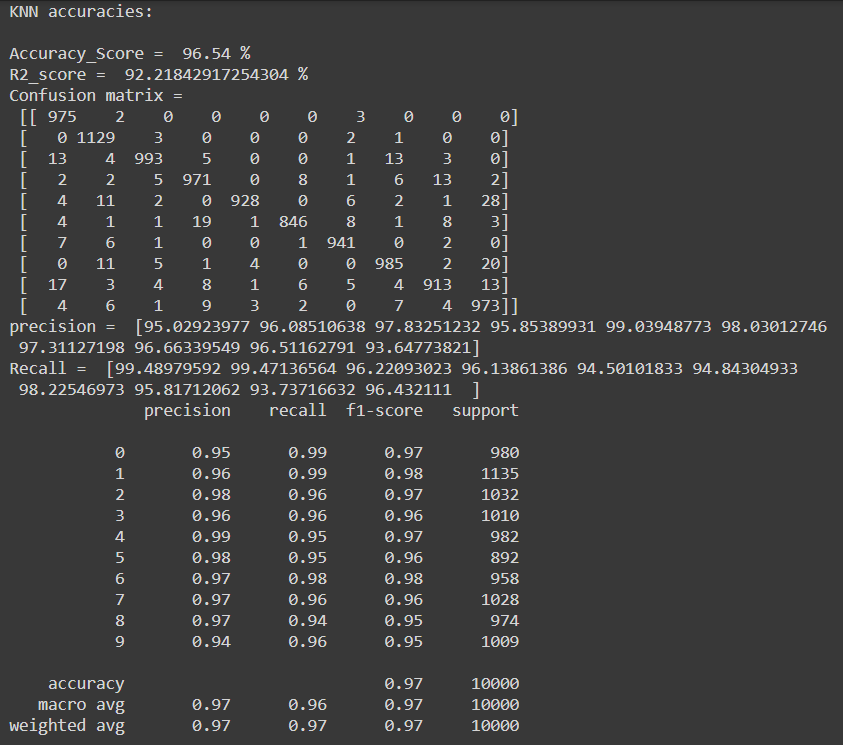
**Experiment the model with different parameters**

**K=3**

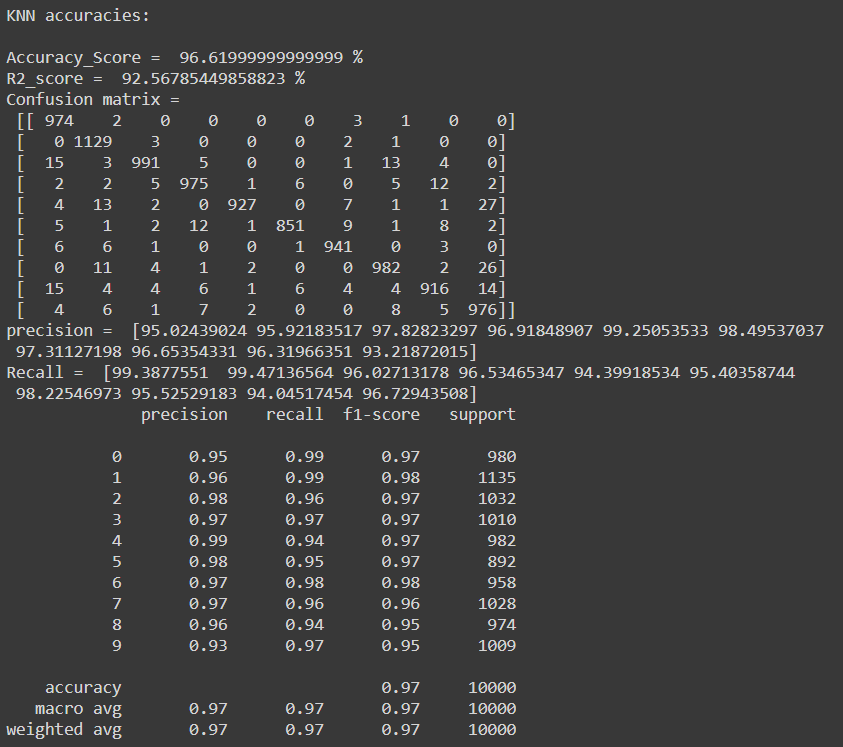
****

**K=5**

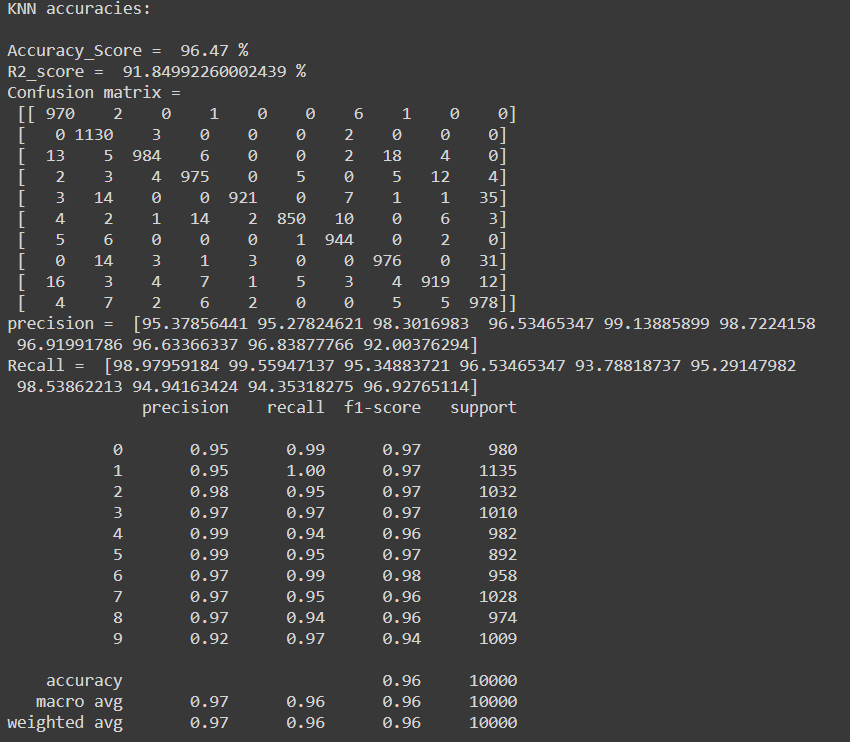
****

**K=8 **

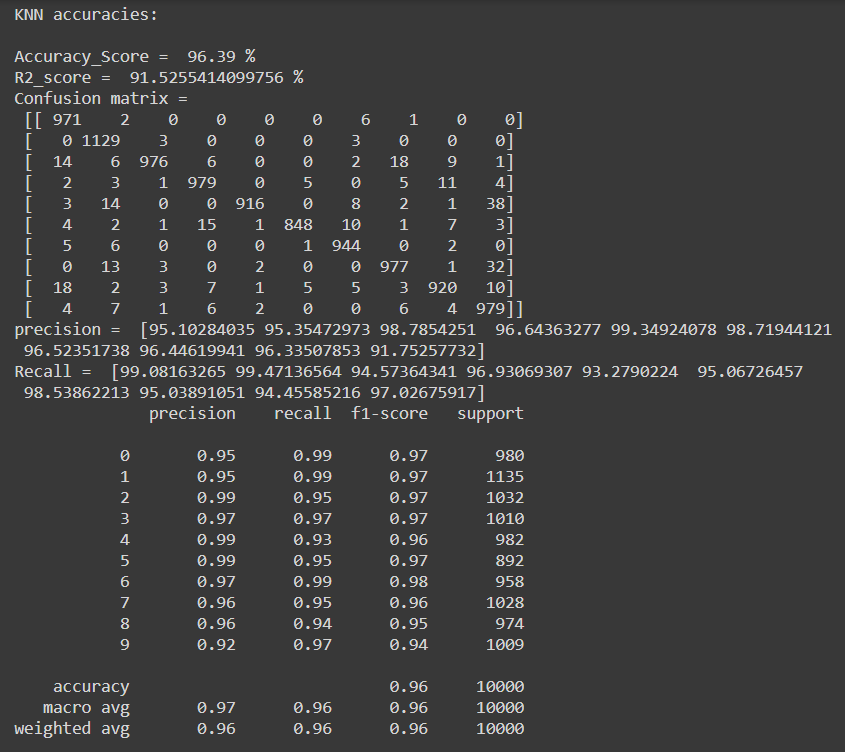
**K=10**

****

**K=15**

****

**K=20**

****

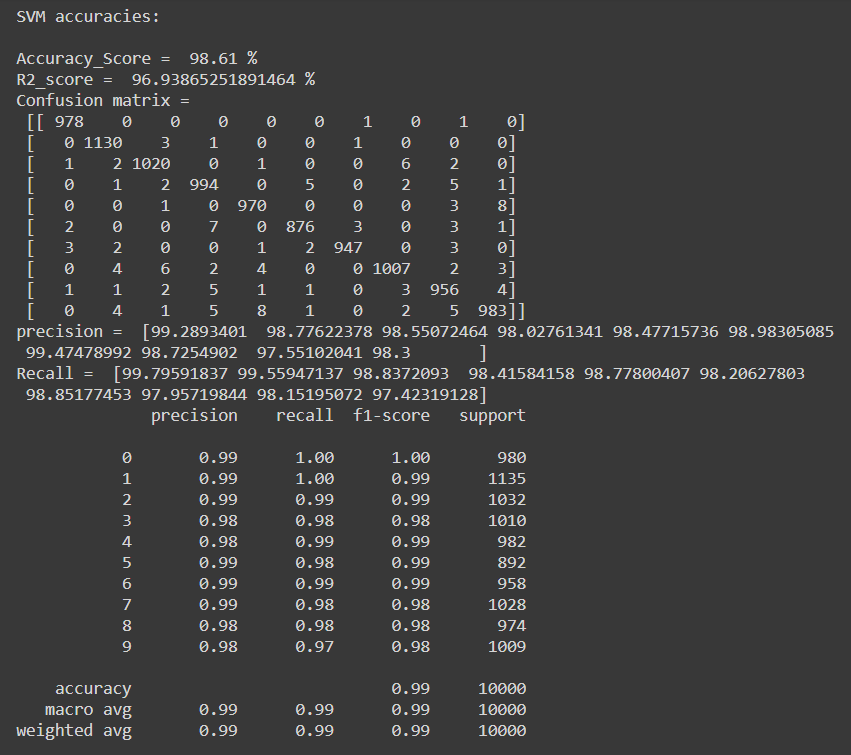
**SVM Model**

**Definition**

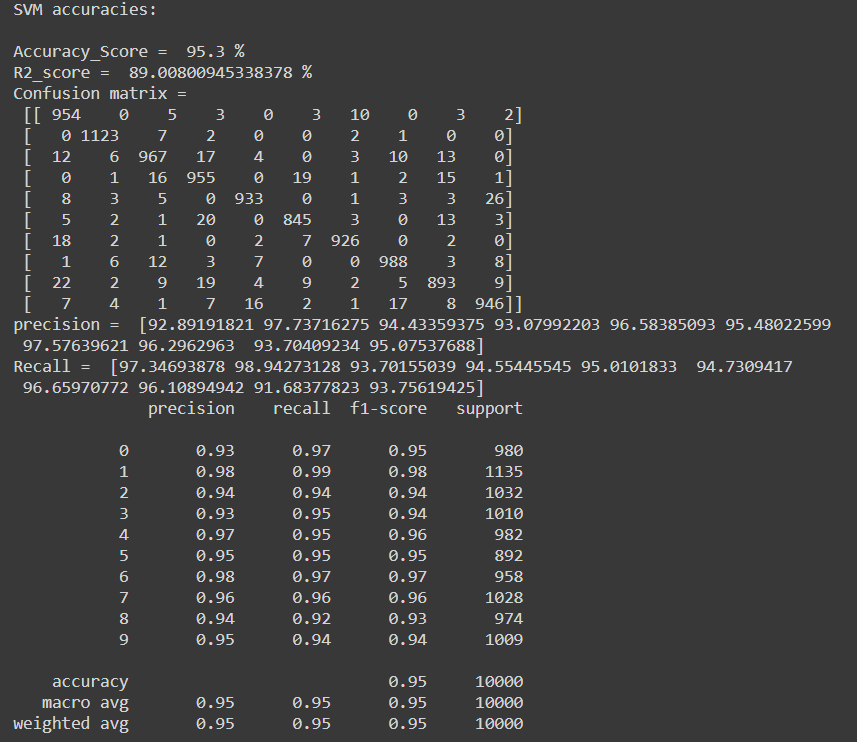
Support Vector Machine is machine learning algorithm that analyzes data for classification and regression analysis. SVM is a supervised learning method that looks at data and sorts it into one of two categories. An SVM outputs a map of the sorted data with the margins between the two as far apart as possible. SVMs are used in text categorization, image classification, handwriting recognition and in the sciences.

**Experiment the model with different parameters**

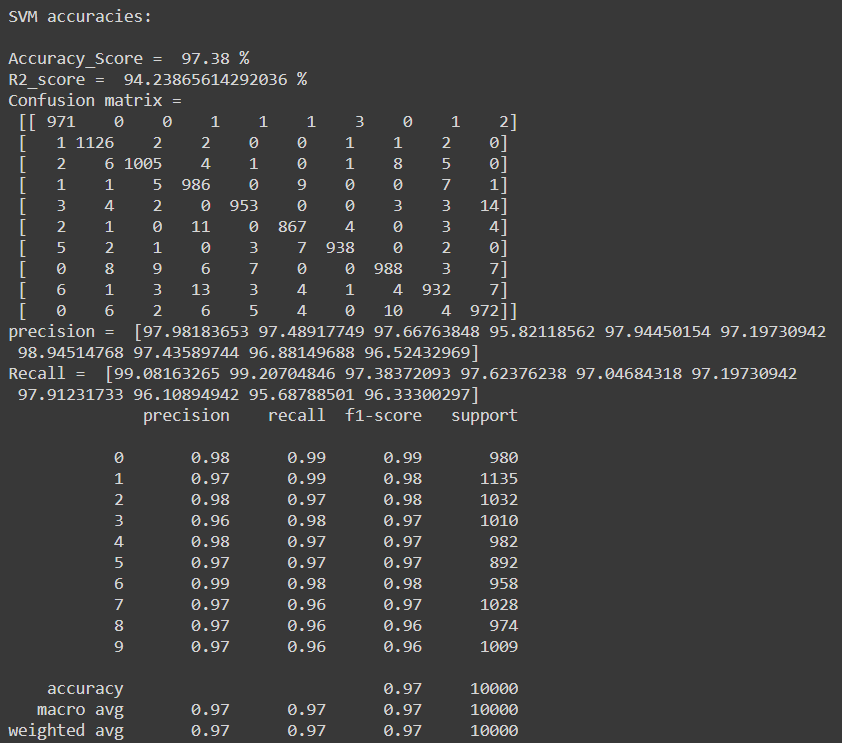
**Gaussian Kernel (Default)**

****

**Sigmoid Kernel**

****

**Linear Kernel**

****

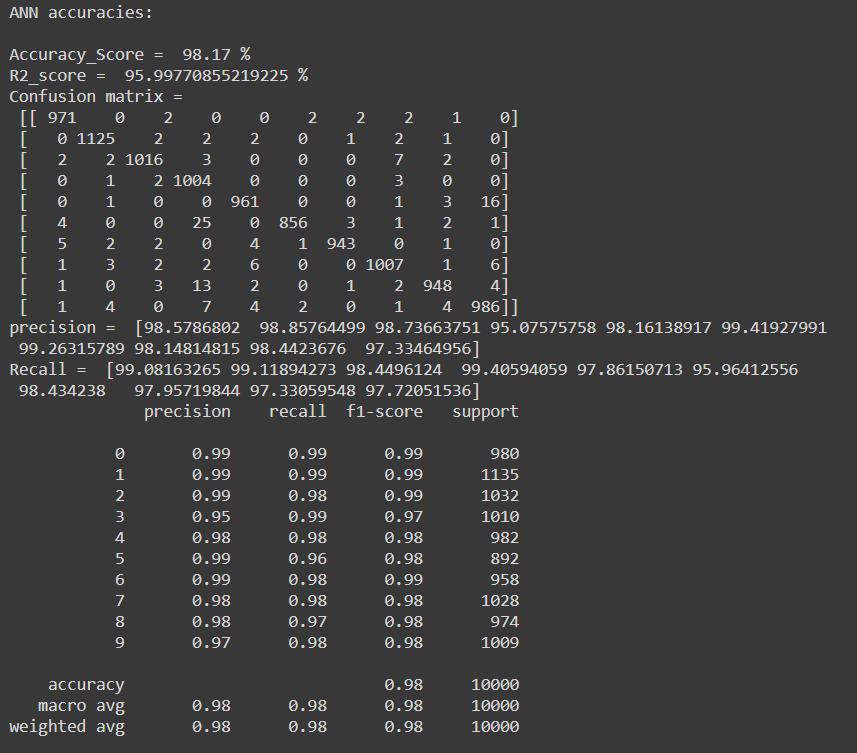
**ANN Model**

**Definition**

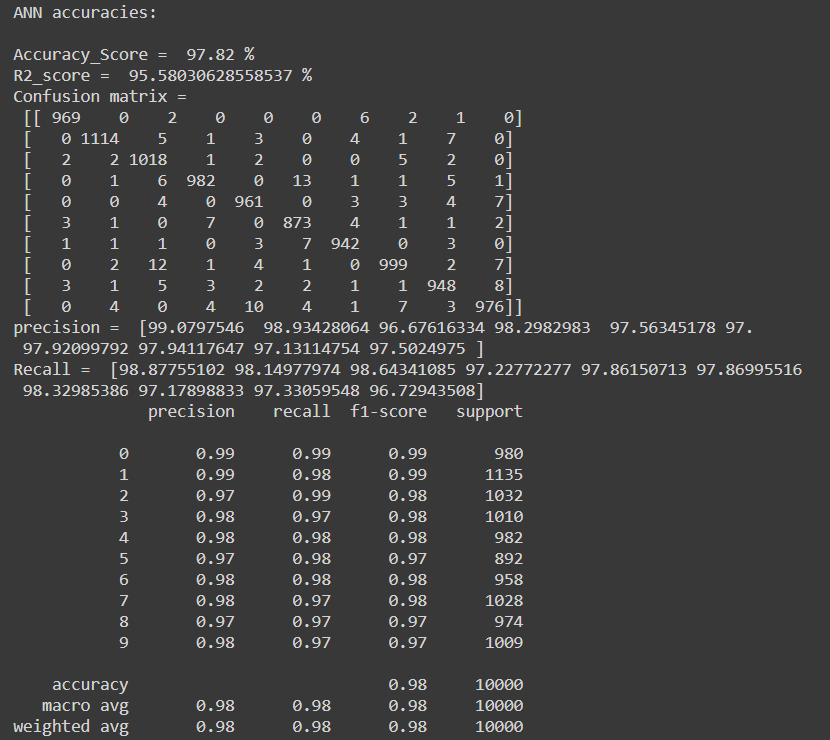
An artificial neural network (ANN) is the piece of a computing system designed to simulate the way the human brain analyzes and processes information. It is the foundation of artificial intelligence (AI) and solves problems that would prove impossible or difficult by human or statistical standards. ANNs have self-learning capabilities that enable them to produce better results as more data becomes available.

**Experiment the model with different parameters**

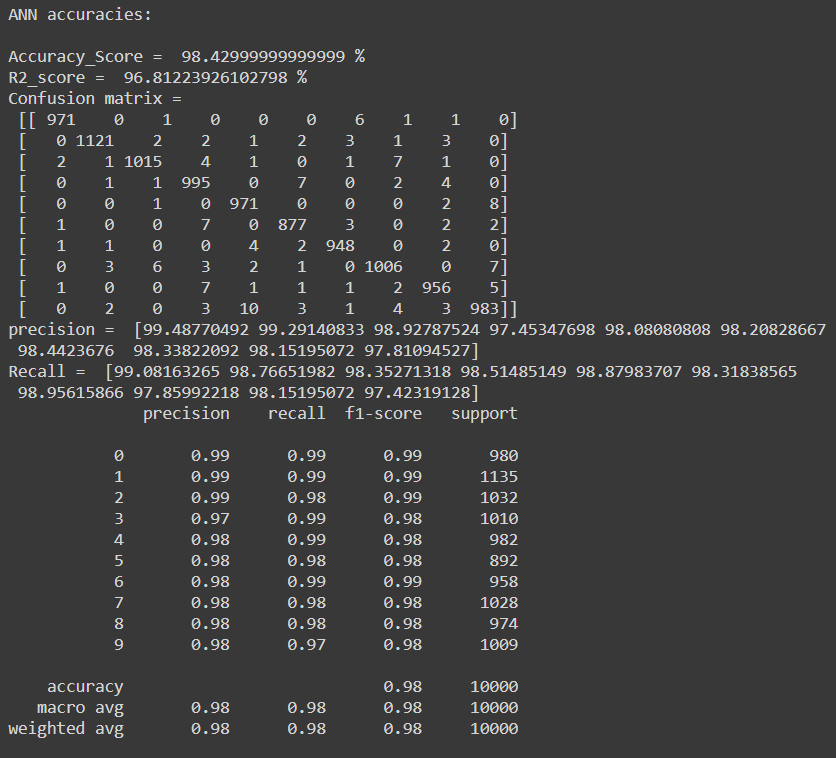
**ann with two hidden layers**

****

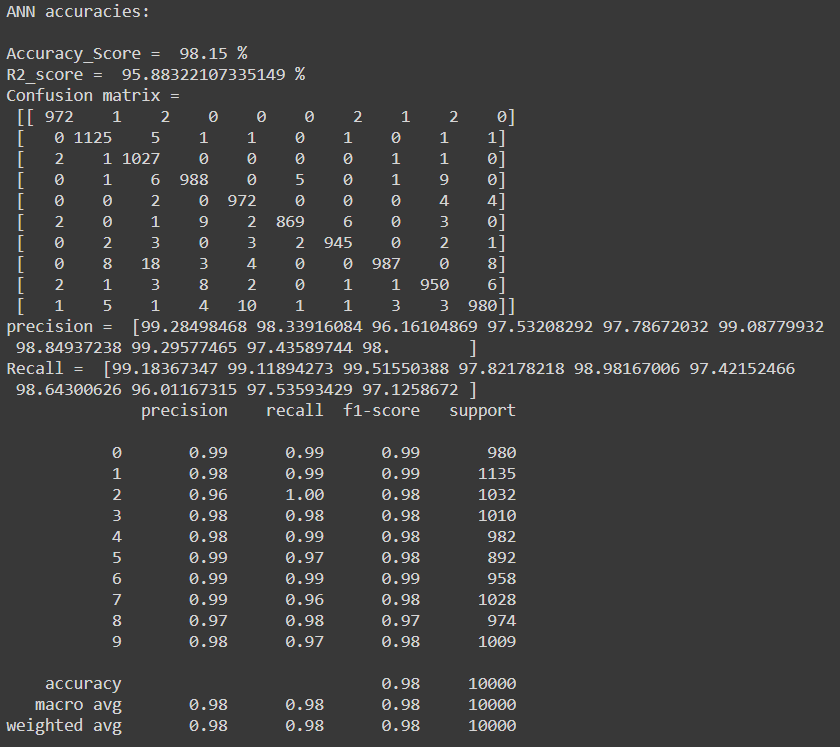
**ann with 'RMSprop' optimizer and two 16 nurons hidden layers**

****

**ann with 'adam' optimizer and three (64, 32, 16)nurons hidden layers**



**ann with 'adam' optimizer and three (56,28,14)nurons hidden layers**

****

**Comparisons between highest accuracy in each model**

Accuracy\_Score

|  |  |  |
| --- | --- | --- |
| KNN | SVM | ANN |
| 96.6199999999 % | 98.61 % |  |

R2\_score

|  |  |  |
| --- | --- | --- |
| KNN | SVM | ANN |
| 92.56785449858 % | 96.93865251891 % |  |

Confusion matrix

|  |  |  |
| --- | --- | --- |
| KNN | SVM | ANN |

Precision

|  |  |  |
| --- | --- | --- |
| KNN | SVM | ANN |

Recall

|  |  |  |
| --- | --- | --- |
| KNN | SVM | ANN |

classification\_report

|  |  |  |
| --- | --- | --- |
| KNN | SVM | ANN |