Graphical user interface, text, application

Description automatically generatedGraphical user interface, application

Description automatically generated

Student Name

***Sanatkumar Rajmogali Ippalpalli***

Title of Project Report

***Guided Project 15 – Handwritten Digit Recognition***

Table of Contents

[Introduction 4](#_Toc83577626)

[Eckovation theme & Question 5](#_Toc83577627)

[Prerequisites before starting coding 6](#_Toc83577628)

[program DEVELOPMENT steps 8](#_Toc83577629)

[Dataset/Image requirements 8](#_Toc83577630)

[Technique – All Algorithms 8](#_Toc83577631)

[PROGRAM / CODE DEVELOPMENT 9](#_Toc83577632)

[Analysis 15](#_Toc83577633)

[CONCLUSION 16](#_Toc83577634)

[Figure 1 Import libraries and datasets/modules 9](#_Toc83577635)

[Figure 2 Load Digits 9](#_Toc83577636)

***EXECUTIVE SUMMARY***

Science and technology improved many technologies and has guided numerous innovative features which advanced several classification algorithms of recognizing numbers from photographic images or with manual input, namely: support vector machine (SVM), Knearest neighbors (KNN), random forest (RF) and several variants of neural networks. The success rates of the algorithms in the field of handwriting recognition were compared.

Handwriting recognition is the ability of a computer or device to accept handwritten text as input from sources such as printed documents, photographs, images, or input stream from other devices.

An example is smartphones or tablets with a touchscreen that can use handwriting (by finger or stylus) input as text or number input. There are many applications and technologies currently available for handwriting recognition and optical character recognition (OCR).

To construct and train neural networks or train classifiers, a well-known and rather complete base of handwritten digits MNIST was chosen.

As part of guided project, one of the technologies use of MNIST for handwritten digit recognitions as the details mentioned in the question.

# Introduction

Science and technology improved many technologies and has guided numerous innovative features which advanced the techniques in deep learning impacting computer vision, image processing.

Handwriting recognition is the ability of a computer or device to accept handwritten text as input from sources such as printed documents, photographs, images, or input stream from other devices. An example is smartphones or tablets with a touchscreen that can use handwriting (by finger or stylus) input as text or number input. There are many applications and technologies currently available for handwriting recognition and optical character recognition (OCR).

Hence Eckovation includes this guided project in the courseware for students to understand, implementation / execute the code themselves.

This report includes the 5W1H about the theme of development of code and running the code with database available over the internet. At the end of the report, the conclusions share the handwritten digits recognition using MNIST features extracted and useful for next course of activities to gain advantages in the edge detection activities development.

# Eckovation theme & Question

**Theme : Handwritten Digit Recognition using MNIST**

Digit recognition system is the working of a machine to train itself for recognizing the digits from different sources like emails, bank cheque, papers, images, etc. and in different real-world scenarios for online handwriting recognition on computer tablets or system. Developing such a system includes a machine to understand and classify the images of handwritten digits as 10 digits (0–9). Handwritten digits from the MNIST database has been one of the most famous databases among the machine learning community for many recent decades.

**Question:**

Use MNIST dataset to create a classifier for all the 10 digits. First implement the classifier by squeezing the image into a vector and then using a MLP. Now, try the same task using a different machine learning classifier such as an SVM to check the gain in performance by using perceptrons as compared to conventional machine learning techniques.

# Prerequisites before starting coding

1. Who - Software needed?
2. What - Version / Release of software?
3. Any Prerequisites
4. How - to install the software
5. Which -libraries are needed to execute the problem statement
6. Where – dataset requirements, path location to include in the code
7. When – to use the above feature extraction
8. Who – Software neeed?

Python

1. What- Version / Release of software?

Python version 3.6 (latest version of python)

1. Any Prerequisites

RAM space availability & hard disk space availability

Admin rights to install the software

1. How - to install the software
2. The following url <https://www.python.org/downloads/>can be referred to download python.
3. Second and easier option is to download anaconda and use its anaconda prompt to run the commands. To install anaconda check this url <https://www.anaconda.com/download/>
4. Which -libraries are needed to execute the problem statement
5. Numpy (pip install numpy)
6. Matplotlib (pip install matplotlib)
7. Pandas
8. Seaborn
9. Sklearn datasets
10. Sklearn metrics
11. Sklearn clusters
12. Sklearn preprocessing
13. Collections
14. Skimage.feature
15. Where – dataset requirements, path location to include in the code
16. Once you have python downloaded and installed, you will need to setup PATH variables (if you want to run python program directly, detail instructions are below in how to run software section). To do that check this: [https://www.pythoncentral.io/add-python-to-path-python-is-not- recognized-as-an-internal-or-external-](https://www.pythoncentral.io/add-python-to-path-python-is-not-recognized-as-an-internal-or-external-command/) [command/](https://www.pythoncentral.io/add-python-to-path-python-is-not-recognized-as-an-internal-or-external-command/).
17. Setting up PATH variable is optional as you can also run program without it and more instruction are given below on this topic.
18. When – to use the above feature extraction
19. When – to use the above technique

Handwriting recognition is the ability of a computer or device to accept handwritten text as input from sources such as printed documents, photographs, images, or input stream from other devices.

# program DEVELOPMENT steps

* Dataset/Image requirement
* Technique selections
* Program / code development
* Analysis

### Dataset/Image requirements

To build and train the model, a well-known and fairly complete base of handwritten digits MNIST was selected.

### Technique – All Algorithms

Let us hop to the inscribing carving!

### PROGRAM / CODE DEVELOPMENT

As explained step by step during the lecture by mentor, we would approach steps and understand the basics with brief explanation as needed.

#### Step 1: Import the relevant libraries and applicable datasets/modules

Graphical user interface, text, application

Description automatically generated

Figure Import libraries and datasets/modules

#### Step 2: Load Digits

Qr code

Description automatically generated with medium confidence

Figure Load Digits

#### Step 3: MLPClassifier

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

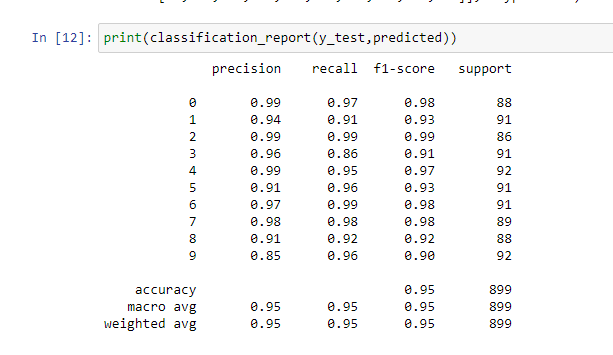
Description automatically generated

Graphical user interface, text, application

Description automatically generated

A picture containing text

Description automatically generated



Qr code

Description automatically generated with low confidence

Calendar

Description automatically generated

#### Step 4: SVC Algorithm

Graphical user interface, text, application

Description automatically generated

Table

Description automatically generated

Qr code

Description automatically generated

#### Step 5: Decision Tree Classifier

Graphical user interface, text, application, email

Description automatically generated

Table

Description automatically generated

#### Step 6: Random Forest Classifier

A picture containing text

Description automatically generated

#### Step 7 : Logistic Regression

Table

Description automatically generated with medium confidence

### Analysis

The algorithms success rates in the field of handwriting recognition were compared. These methods are Support Vector Machine (SVM), Logistic Regression, Decision Tree, Random Forest (RF).

This entire program runs within few seconds.

# CONCLUSION

In this guided project, we built MNIST digits and Improving the accuracy of handwritten digit recognition is achieved by increasing the complexity of the used deep learning neural networks.

This is done in first attempt. Hence, the improvements in the code with time with multiple attempts may be checked and justified for the accuracy score.

This entire program runs within few seconds.

references:

1. <http://ceur-ws.org/Vol-2864/paper44.pdf>