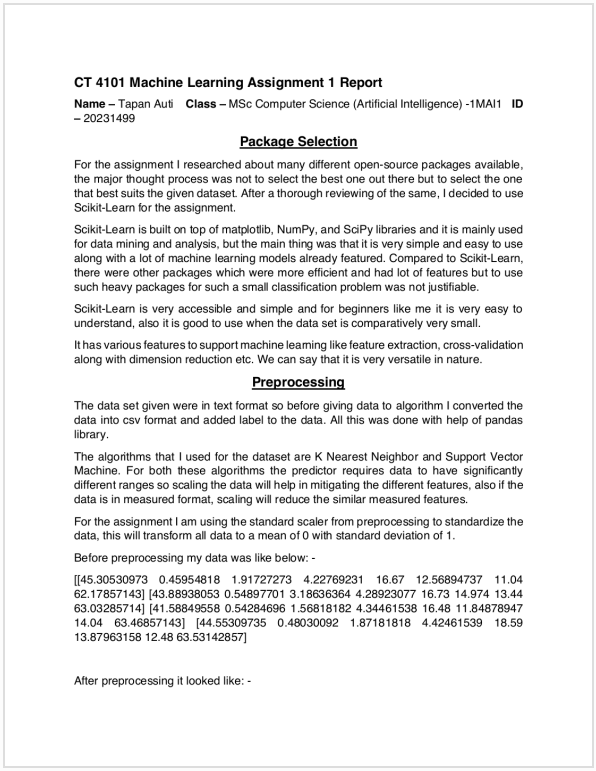
Digital Receipt

This receipt acknowledges that Turnitin received your paper. Below you will find the receipt information regarding your submission.

The first page of your submissions is displayed below.

Submission author:	Tapan Vivek Auti
Assignment title:	Submit Assignment 1
Submission title:	ML_20231499
File name:	ML_20231499.pdf
File size:	173.57K
Page count:	4
Word count:	1,173
Character count:	5,851
Submission date:	01-Nov-2020 12:07PM (UTC+0000)
Submission ID:	1432606930



CT 4101 Machine Learning Assignment 1 Report

Name – Tapan Auti **Class** – MSc Computer Science (Artificial Intelligence) -1MAI1 **ID** – 20231499

Package Selection

For the assignment I researched about many different open-source packages available, the major thought process was not to select the best one out there but to select the one that best suits the given dataset. After a thorough reviewing of the same, I decided to use Scikit-Learn for the assignment.

Scikit-Learn is built on top of matplotlib, NumPy, and SciPy libraries and it is mainly used for data mining and analysis, but the main thing was that it is very simple and easy to use along with a lot of machine learning models already featured. Compared to Scikit-Learn, there were other packages which were more efficient and had lot of features but to use such heavy packages for such a small classification problem was not justifiable.

Scikit-Learn is very accessible and simple and for beginners like me it is very easy to understand, also it is good to use when the data set is comparatively very small.

It has various features to support machine learning like feature extraction, cross-validation along with dimension reduction etc. We can say that it is very versatile in nature.

Preprocessing

The data set given were in text format so before giving data to algorithm I converted the data into csv format and added label to the data. All this was done with help of pandas library.

The algorithms that I used for the dataset are K Nearest Neighbor and Support Vector Machine. For both these algorithms the predictor requires data to have significantly different ranges so scaling the data will help in mitigating the different features, also if the data is in measured format, scaling will reduce the similar measured features.

For the assignment I am using the standard scaler from preprocessing to standardize the data, this will transform all data to a mean of 0 with standard deviation of 1.

Before preprocessing my data was like below: -

[[45.30530973 0.45954818 1.91727273 4.22769231 16.67 12.56894737 11.04 62.17857143] [43.88938053 0.54897701 3.18636364 4.28923077 16.73 14.974 13.44 63.03285714] [41.58849558 0.54284696 1.56818182 4.34461538 16.48 11.84878947 14.04 63.46857143] [44.55309735 0.48030092 1.87181818 4.42461539 18.59 13.87963158 12.48 63.53142857]

After preprocessing it looked like: -