**Nazmul Hasan Rabbi Report for Task 3:** **Customer Segmentation**

Our objective is to develop a model that can assign labels to customers based on the feature values.

**Task 3.1 Evaluate classifiers using *5*-fold cross-validation on training data** [15 points]**.**

Only the following 5 features were used for classification:

Relevant features: x20, x48, x7, x15, x16

|  |  |
| --- | --- |
| **Model** | **Mean cross-validation Accuracy** |
| **Gaussian Naïve Bayes** | 0.8868 |
| **Decision Tree** | 0.9274 |
| **Random Forest** | 0.9642 |
| **K Nearest Neighbors** | 0.9668 |
| **Logistic Regression** | 0.8936 |
| **Support Vector Classifier** | 0.9650 |
| **Artificial Neural Network** | 0.9660 |

**Task 3.2 Choose a model** [10 Points]

Based on cross-validation accuracy, I chose a K-Nearest Neighbors classifier with n\_neighbors=5 and default values for all other hyper-parameters.

**CHOSEN\_MODEL:**

KNeighborsClassifier(algorithm='auto', leaf\_size=30, metric='minkowski', metric\_params=None, n\_jobs=None, n\_neighbors=5, p=2, weights='uniform')

Accuracy of **CHOSEN\_MODEL** with the validation examples = 0.9656

Confusion matrix of **CHOSEN\_MODEL** for the validation examples in the table below.

|  |  |  |  |
| --- | --- | --- | --- |
| **Actual class** | **Predicted class** | | |
| **0** | **1** | **2** |
| **0** | 1605 | 25 | 30 |
| **1** | 24 | 1603 | 28 |
| **2** | 30 | 35 | 1620 |

Precision for class 0 = 0.9670

Recall for class 2 = 0.9610

**Task 3.3 Classify new examples**. [5 Points]

Classify the 60 unlabeled examples in the file “customer\_segmentation.unlabeled.csv” using **CHOSEN\_MODEL** and present your results in the table below.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **label** |  | **ID** | **label** |  | **ID** | **label** |
| 1 | 0 |  | 21 | 1 |  | 41 | 2 |
| 2 | 2 |  | 22 | 1 |  | 42 | 2 |
| 3 | 0 |  | 23 | 1 |  | 43 | 2 |
| 4 | 0 |  | 24 | 1 |  | 44 | 2 |
| 5 | 0 |  | 25 | 1 |  | 45 | 2 |
| 6 | 0 |  | 26 | 1 |  | 46 | 2 |
| 7 | 0 |  | 27 | 2 |  | 47 | 2 |
| 8 | 0 |  | 28 | 1 |  | 48 | 2 |
| 9 | 0 |  | 29 | 1 |  | 49 | 2 |
| 10 | 0 |  | 30 | 1 |  | 50 | 2 |
| 11 | 0 |  | 31 | 1 |  | 51 | 2 |
| 12 | 0 |  | 32 | 1 |  | 52 | 2 |
| 13 | 0 |  | 33 | 1 |  | 53 | 2 |
| 14 | 0 |  | 34 | 1 |  | 54 | 2 |
| 15 | 0 |  | 35 | 1 |  | 55 | 2 |
| 16 | 0 |  | 36 | 1 |  | 56 | 2 |
| 17 | 0 |  | 37 | 1 |  | 57 | 2 |
| 18 | 0 |  | 38 | 2 |  | 58 | 2 |
| 19 | 0 |  | 39 | 2 |  | 59 | 2 |
| 20 | 1 |  | 40 | 2 |  | 60 | 2 |