COMP534 – APPLIED ARTIFICIAL INTELLIGENCE

Supervised learning methods for solving a classification problem – Report.

Bharathidasan Ilango- 201670038

Introduction:

In this assignment, we will develop 3 different classification models to predict the class of a given sample based on its features. And compare its performance of each model. We will be using a dataset containing samples with 9 features, and the class variable indicates whether the sample belongs to class 0 or class 1.

To build our model, we will use Python programming language and several libraries such as numpy, pandas, sklearn, and matplotlib. The various classification methods implemented are Naive Bayes, K-Nearest Neighbours (KNN), and Support Vector Machine (SVM). The development process involves the following steps, Data loading and exploration, Data pre-processing, Model training and hyperparameter tuning and Model evaluation.

We will split the dataset into training and testing sets and use cross-validation to select the best hyperparameters for each classification method. Finally, we will evaluate the performance of each model using metrics such as accuracy, precision, recall, and F1-score.

In the following sections, we will discuss each step in more detail, including the libraries used, the classification methods used, and the training and testing process.

Libraries used:

We will be using the following libraries for our project:

1. numpy: Is implemented for numerical operations.
2. pandas: Is implemented for data manipulation and analysis.
3. scikit-learn (sklearn): Is implemented for machine learning tasks such as classification, cross-validation, and hyperparameter tuning.
4. matplotlib: Is implemented for data visualization.
5. seaborn: Is implemented for advanced data visualization.

Classification methods used:

We will use various classification methods including Naive Bayes, K-Nearest Neighbours (KNN), and Support Vector Machine (SVM) to build our classification models. Each method has its own strengths and weaknesses, and we will evaluate the performance of each method using various metrics such as accuracy, precision, recall, and F1 score.

Naive Bayes: A probabilistic classifier based on Bayes' theorem. We will use Gaussian, Multinomial, Complement, and Bernoulli Naive Bayes classifiers.

K-Nearest Neighbours (KNN): A non-parametric classifier that classifies the data based on the majority vote of its k-nearest neighbours. We will use the KNN classifier with different values of k.

Support Vector Machine (SVM): A linear or non-linear classifier that separates the data into different classes by finding the hyperplane that maximizes the margin between the classes. We will use the SVM classifier with different kernels such as linear, polynomial, and radial basis function (RBF).

For each classification method, we will use cross-validation to select the best hyperparameters. We will use the GridSearchCV function in sklearn to perform hyperparameter tuning.

Training and testing process:

We will split the dataset into training and testing sets using the train_test_split function in sklearn. We will use an 80/20 split ratio, where 80% of the data is used for training and 20% for testing. We will use the confusion matrix to visualize the performance of each model.

We will also use K-fold cross-validation to select the best hyperparameters for each classification method. We will use the KFold function in sklearn to create the cross-validation object.

Finally, we will use the fit function to train the model on the training set, and the predict function to predict the class of the testing set.

Evaluation Section:

Naïve Bayes classification model:

As per the evaluation of the test results the best version of the classification method of Naive Bayes is Gaussian model, with an accuracy of 97.14%. The precision, recall, and F1 score are also high, indicating good overall performance of the model.

Looking at the confusion matrix Figure 1: Naive Bayes Confusion Matrix , we can see that there are 85 true negatives (TN), 51 true positives (TP), 3 false positives (FP), and 1 false negative (FN). This indicates that the model correctly predicted 85 negative cases and 51 positive cases, but made 3 false positive errors (i.e., predicted a positive case when it was actually negative) and 1 false negative error (i.e., predicted a negative case when it was actually positive).

Precision is defined as TP / (TP + FP), which is the proportion of true positives among all positive predictions. In this case, the precision is 94.44%, which means that 94.44% of the predicted positive cases are actually positive. Recall is defined as TP / (TP + FN), which is the proportion of true positives among all actual positive cases. In this case, the recall is 98.08%, which means that the model correctly identified 98.08% of the actual positive cases. F1 score is the harmonic mean of precision and recall, which provides a proportionate measure of both metrics. In this case, the F1 score is 96.23%, indicating adequate equilibrium between precision and recall.

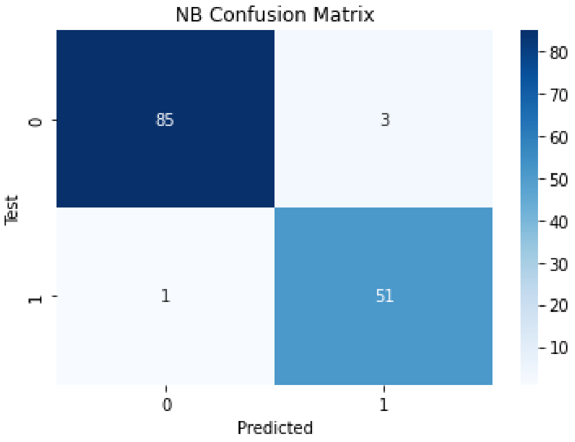

Figure 1: Naive Bayes Confusion Matrix

K-Nearest Neighbour Classification Model:

KNN is a non-parametric classifier that uses the k-nearest neighbours to classify new instances based on the majority class of its neighbour's. The best hyperparameter found by tuning method GridSearchCV is 'neighbours': 7. After evaluation using best hyperparameter and test data, the performance of the model is evaluated as follows.

As per the confusion matrix Figure 2: KNN Confusion Matrix shows that there are 87 true positives and 1 false positive in the first column, indicating that 87 samples are correctly classified as positive and 1 sample is incorrectly classified as positive. In the second column, there are 2 false negatives and 50 true negatives, indicating that 2 samples are incorrectly classified as negative, and 50 samples are correctly classified as negative.

Based on the confusion matrix, we can calculate various evaluation metrics for the KNN model. The precision of the model is 98.04%, which indicates that out of all the predicted positive samples, 98.04% were actually positive. The recall of the model is 96.15%, which indicates that out of all the actual positive samples, 96.15% were correctly classified as positive by the model. The F1 score of the model is 97.09%, which is a weighted average of precision and recall, and provides a single measure of the model's performance.

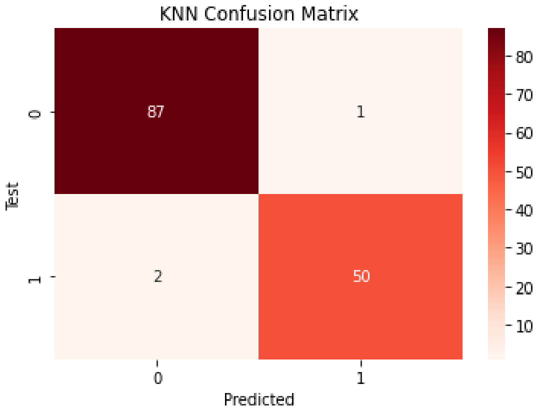

Figure 2: KNN Confusion Matrix

Support Vector Machine Classifier Model:

In SVM the hyperparameters used is 'kernel': ['linear', 'poly', 'rbf', 'sigmoid'] by passing the hyperparameters in GridsearchCV cross validation method which returns the best parameter as 'kernel': 'rbf'. After evaluating using the best hyperparameter and test data, the performance of the model is evaluated as follows.

There are 86 true positives, meaning that the model correctly predicted 86 instances as positive out of a total of 88 actual positives. There are 2 false positives, meaning that the model incorrectly predicted 2 instances as positive when they were actually negative. There is 1 false negative, meaning that the model incorrectly predicted 1 instance as negative when it was actually positive. There are 51 true negatives, meaning that the model correctly predicted 51 instances as negative out of a total of 52 actual negatives.

Based on the confusion matrix Figure 3: SVM Confusion Matrix, we can calculate several performance metrics. The overall accuracy of the model is 97.86%, which is the sum of true positives and true negatives divided by the total number of instances. It measures the proportion of correct predictions out of all predictions made by the model. Precision is a metric that measures the proportion of true positives among the instances that the model predicted as positive. In this case, the precision is 96.23%, indicating that 96.23% of the instances predicted as positive were actually positive. Recall is a metric that measures the proportion of true positives among the instances that are actually positive. In this case, the recall is 98.08%, indicating that 98.08% of the actual positives were correctly identified as positive by the model. The F1-score is the harmonic mean of precision and recall, and it balances both metrics. In this case, the F1-score is 97.14%.

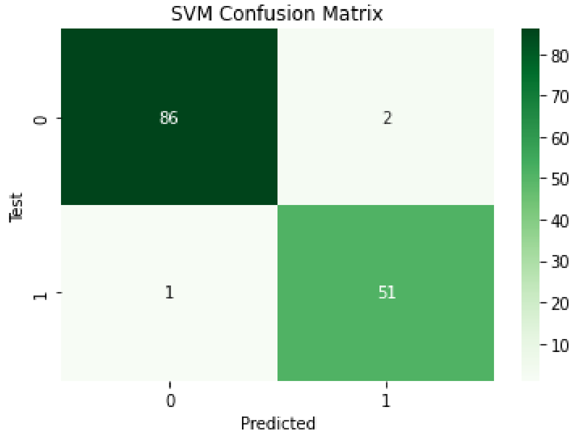

Figure 3: SVM Confusion Matrix

Conclusion:

Based on the performance table Figure 4: Performance Metrices of Classifier, we can see that all three classifiers: Naive Bayes, KNN, and SVM, have high accuracy, precision, recall, and F1 score.

The Naive Bayes classifier has a slightly lower precision and F1 score compared to KNN and SVM, but it still has high accuracy and recall. Naive Bayes is a probabilistic classifier that assumes that the features are independent, and it works well when the assumption holds true.

The KNN classifier has the highest precision among the three classifiers, indicating that it has the highest proportion of true positives among the instances that it predicted as positive.

The SVM classifier has the highest F1 score among the three classifiers, indicating that it has a good balance between precision and recall. SVM is a linear classifier that separates the classes with a hyperplane in a high-dimensional feature space.

Overall, the choice of a classification method depends on the specific problem at hand, the size and complexity of the dataset, and the resources available. It is important to test multiple classifiers and evaluate their performance using appropriate metrics before selecting the best one for a particular problem**.**

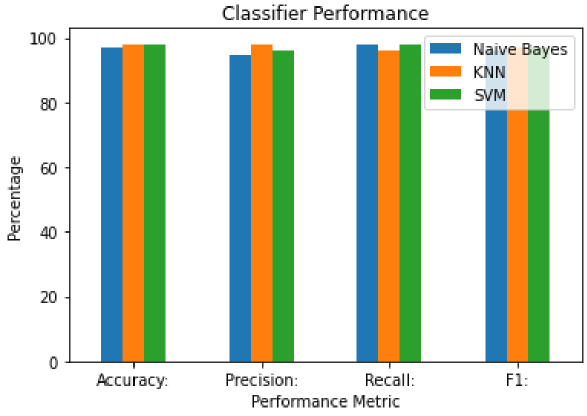

Figure 4: Performance Metrices of Classifiers