**KNN**

**What is KNN**

KNN stands for "k-nearest neighbors". It is a type of supervised machine learning algorithm used for classification and regression. Given a new observation, it finds the k-number of training examples that are closest to it in feature space, and then it assigns the class label based on the majority class among those k-nearest examples. It's a simple yet powerful algorithm that can be used for both classification and regression problems.

**Application**

1. Collect and preprocess the data: This step involves collecting the data that will be used to train and test the algorithm. The data may need to be cleaned and preprocessed to ensure that it is in a format that can be used by the algorithm.

Graphical user interface, text, application, email

Description automatically generated

1. Assing Age, Sex, BP, Cholesterol and Na\_to\_K columns to X and Drug column to Y.

Table

Description automatically generated with medium confidence

1. Split the data into training and test sets: The data is typically split into two sets, one for training and one for testing. The training set is used to train the algorithm, while the test set is used to evaluate the performance of the trained model. Then StandardScaler is used to standardize the features of a dataset by removing the mean and scaling to unit variance.

Text, table

Description automatically generated with medium confidence

1. Define the distance metric: The distance metric is a function that is used to calculate the distance between two observations. Common distance metrics include Euclidean distance, Manhattan distance and Minkowski distance. After trying this 3 distances it has been seen that manhattan is the best distance metric out of them.

After choosing the metric knn score is printed for train and test sets. It’s seen that knn score is 0.857 for train and 0.85 for test set.

Make predictions: Given a new observation, the KNN algorithm finds the k-nearest training examples to it in feature space, and then it assigns the class label based on the majority class among those k-nearest examples.

There is a prediction example for 52 year old male with BP levels are low, normal cholesterol and 32.922 Na\_to\_K which preditcs DrugY

Graphical user interface, text, application, email

Description automatically generated

1. The sensitivity of k-NN classification accuracy to the choice of the 'k' parameter value is determined and displayed in the graph below.

Chart, scatter chart

Description automatically generated

1. The max KNN Acccuracy on every ‘k’ value is determined and displayed in the graph below.

Chart, line chart

Description automatically generated

1. A classification report is used for displaying performance evaluation metric in machine learning that is used to evaluate the performance of a classification algorithm. It provides a summary of the performance of the algorithm on a test set by computing several evaluation metrics such as precision, recall, f1-score, and support.

The classification report contains the following metrics:

Precision: It is the ratio of correctly predicted positive observations to the total predicted positive observations. It measures the ability of the classifier not to label as positive a sample that is negative.

Recall: It is the ratio of correctly predicted positive observations to the all observations in actual class. It measures the ability of the classifier to find all the positive observations.

F1-Score: It is the harmonic mean of precision and recall. The F1-Score reaches its best value at 1 (perfect precision and recall) and worst at 0.

Support: It is the number of samples of the true response that lie in that class.

The confusion matrix is used to show the number of correct predictions and incorrect predictions made by the classifier. It is typically represented as a table, with the true class labels on one axis and the predicted class labels on the other axis.

A confusion matrix allows to understand how well the classifier is doing by providing a clear picture of how many observations are being correctly classified and how many are not. It also helps to identify common mistakes made by the classifier.

Table

Description automatically generated

1. ROC AUC (Receiver Operating Characteristic - Area Under the Curve) is a performance evaluation metric for binary classification problems. It is implemented for measure of the trade-off between the true positive rate (sensitivity) and the false positive rate (1-specificity) of a classifier.

The predict\_proba() method is used to generate the predicted class probabilities for the test data.

Then, the roc\_auc\_score() method is provided by the scikit-learn library to calculate the AUC-ROC score. The method takes two arguments: the true labels of the test set and the predicted class probabilities generated by the classifier.

The predict\_proba() method is used to generate the predicted class probabilities for the test data.

As we can see below ROC AUC score for our multiclass classification is 0.964

Graphical user interface, text, application

Description automatically generated

1. The ROC curve is a graphical representation of the performance of a classifier, where the true positive rate is plotted on the y-axis and the false positive rate is plotted on the x-axis. The ideal classifier would have a ROC curve that hugs the top left corner of the plot, indicating that it has a high true positive rate and a low false positive rate. The area under the ROC curve (AUC) is a single number that summarizes the performance of a classifier across all possible threshold

Used the roc\_curve() method provided by the scikit-learn library to calculate the true positive rate and false positive rate for each class.

Plot the ROC curve for each class by plotting the true positive rate against the false positive rate. Use different colors for each class to distinguish them.

Graphical user interface, text, application, email

Description automatically generated

1. The plot shows the trade-off between the true positive rate and the false positive rate for each class. The closer the ROC curve is to the top left corner of the plot, the better the classifier is at distinguishing between the positive and negative classes. The AUC score is a single number that summarizes the performance of the classifier across all possible thresholds. A score of 1 represents a perfect classifier, and a score of 0.5 represents a classifier that performs no better than random guessing.

Diagram

Description automatically generated with low confidence