Data Mining Midterm Project

Participants:

1-Sherwan Ali/220201945

2-Ipek Uçar/210201059

3-Yaprak Sevinç Aldoğan/220201032

November 2024

A screen shot of a computer code

Description automatically generated

In figure1:

This code sets up the basic tools for analyzing and predicting diabetes using machine learning. It first imports important libraries like pandas for working with data, numpy for numerical calculations, and scikit-learn for machine learning . Then, it loads the diabetes dataset from a file called diabetes.csv into a DataFrame using pandas. A random seed (42) is set to make sure the results are consistent every time the code is run. These steps prepare the data and tools needed to process the dataset and train machine learning models in later parts of the code.

A close-up of a computer screen

Description automatically generated

In figure 2:

This part loads the diabetes dataset (diabetes.csv) into a Pandas Data Frame called data. It then prints the first 5 rows to give an overview of the dataset.

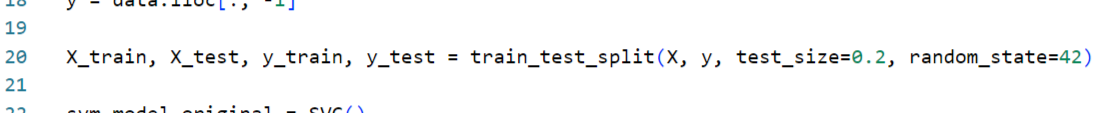
A number on a white background

Description automatically generated

In figure 3:

We split the data set into: Features (X): Contains predictors like Glucose, BMI, etc.

Target (y): The Outcome column, indicating whether the person has diabetes (1) or not (0).



In figure 4:

We divide the dataset into:

Training Set (80%): Used to train the machine learning models.

Testing Set (20%): Used to evaluate the models’ performance on unseen data.

random\_state=42 ensures the same split every time the code runs.

A computer code on a white background

Description automatically generated

In figure 5: we used The Support Vector Machine (SVM) model:

Is trained using the X\_train and y\_train data.

Makes predictions on the test data (X\_test).

Outputs a confusion matrix, which shows the counts of true positives, true negatives, false positives, and false negatives.

Calculates the accuracy score, which measures how well the model predicted.

A screenshot of a computer code

Description automatically generated

In figure 6:

The K-Nearest Neighbors (KNN) model:

Works similarly to SVM but predicts based on the nearest data points in the feature space.

Is trained using the X\_train and y\_train data.

Makes predictions on the test data (X\_test).

Outputs a confusion matrix and accuracy score.

A computer code on a white background

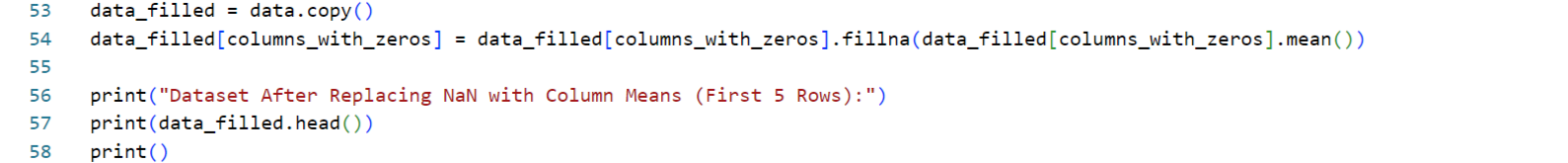
Description automatically generated

In figure 7:

In this step, we identify certain columns (Glucose, BloodPressure, SkinThickness, Insulin, and BMI) where a value of 0 doesn’t make sense , These 0s are considered missing data, so we replace them with NaN. This allows us to handle these missing values appropriately later.

Replacing 0s with NaN makes it clear which data points are incomplete and need to be fixed.

After replacing the 0s, the dataset is updated, and missing values are clearly marked as NaN



In figure 8:

Once the missing values (NaN) are identified, they are replaced with the mean of the corresponding column. For example:

If the BMI column has missing values, they are replaced by the average BMI value from the rest of the data.

Filling missing values with the mean is a simple and effective method to ensure that the dataset is complete and can be used for machine learning models.

The dataset now has no missing values, making it ready for further analysis. This step ensures consistency and avoids errors when training the models.

A computer code with text

Description automatically generated

In figure 9:

After cleaning the data, the Support Vector Machine (SVM) model is trained again using the updated dataset where missing values have been replaced.

A confusion matrix is generated, showing how many predictions were correct or incorrect.

The accuracy of the SVM model is calculated to measure its performance.

The performance of the SVM model is analyzed and compared to its performance on the original dataset.

A close-up of a computer code

Description automatically generated

The K-Nearest Neighbors (KNN) model is also retrained and evaluated on the cleaned dataset. The steps are similar to those for the SVM model:

A confusion matrix is created to show the model’s performance.

The accuracy is calculated to measure how well the KNN model predicts outcomes.

The performance of the KNN model is analyzed and compared to its performance on the original dataset.