Experiment 04 - Classification using Python

| Roll No. | 65 |
| --- | --- |
| Name | Shreyansh Singh |
| Class | D15 |
| Subject | Business Intelligence Lab |
| LO Mapped | LO2:  Organize and prepare the data needed for data mining algorithms in terms of attributes and class inputs, training, validating, and testing files.  LO4: Implement various data mining algorithms from scratch using languages like Python/ Java etc. |
|  |  |

**Aim**: To implement the classification algorithm using Python.

**Classification Algorithm**:

A classification algorithm is a type of machine learning algorithm used to categorize data into predefined classes or categories based on the characteristics of the data. The goal of classification is to learn a function that can map input variables to output variables in a way that accurately predicts the correct class label for new, unseen data.

There are several types of classification algorithms, including decision trees, logistic regression, support vector machines, naive Bayes, k-nearest neighbors, and neural networks. Each algorithm has its own strengths and weaknesses, and the choice of algorithm depends on the specific problem at hand.

In general, classification algorithms work by first training on a dataset that has already been labeled with class labels. The algorithm learns to identify patterns in the data that are associated with each class, and these patterns are used to make predictions on new, unseen data. The algorithm can then be tested on a separate dataset to evaluate its performance and accuracy.

The process of training a classification algorithm typically involves several steps, including data preprocessing (such as scaling or normalizing the data), feature selection or extraction, and model training and evaluation. The choice of features and model hyperparameters can greatly affect the performance of the algorithm, and tuning these parameters is an important part of the modeling process.

Overall, classification algorithms are powerful tools for automating the process of categorizing data, and they have a wide range of applications in fields such as computer vision, natural language processing, and bioinformatics.

**Python Library Function Used**:

There are many libraries available for classification in various programming languages, but some of the most popular libraries are:

Scikit-learn: Scikit-learn is a popular machine learning library in Python that provides various classification algorithms like logistic regression, decision trees, random forests, SVMs, and more. It also includes functions for data preprocessing, feature selection, and model evaluation.

TensorFlow: TensorFlow is a popular open-source library for building and training machine learning models, including classifiers. It provides a high-level API for building neural networks, as well as pre-trained models for image classification and natural language processing tasks.

Keras: Keras is another popular deep learning library in Python that provides a user-friendly API for building and training neural networks. It includes various built-in classifiers like convolutional neural networks (CNNs), recurrent neural networks (RNNs), and more.

PyTorch: PyTorch is a popular deep learning library in Python that provides a flexible and efficient framework for building and training machine learning models. It includes built-in functions for building classifiers using deep learning models like CNNs, RNNs, and more.

MATLAB: MATLAB is a popular programming language and environment used for scientific computing and data analysis. It includes functions for building and training classifiers like logistic regression, SVMs, decision trees, and more.

These libraries provide a convenient way to implement and experiment with various classification algorithms and models, and they can greatly reduce the amount of code needed to develop a classification system.

**Code and Observation**:

GitHub Link *:* [*Github Repo contain notebook and dataset*](https://github.com/SinghShreyansh/DBMI/tree/main/Exp-4)

**Conclusion**:

In conclusion, classification is an important task in machine learning, and Python provides various libraries and tools for implementing and experimenting with classification algorithms. Scikit-learn is a popular library for implementing classifiers in Python, providing a wide range of algorithms and tools for data preprocessing, feature selection, and model evaluation. Other popular libraries like TensorFlow, Keras, and PyTorch provide powerful frameworks for building and training deep learning models for classification tasks.

When performing experiments on classification, it is important to carefully preprocess the data, select appropriate features, and evaluate the performance of the models using appropriate metrics like accuracy, precision, recall, and F1-score. It is also important to properly split the data into training and testing sets to avoid overfitting and ensure the generalization of the model.

Overall, classification is a powerful tool for automating the process of categorizing data, and Python provides a rich ecosystem of libraries and tools for building and evaluating classifiers for various applications.