BITS F464 - Machine Learning

Assignment – 2

Submission Time & Date: 2359Hrs on 28th Apr, 2023

Marks: 20

In this assignment, you will learn to build the following models:

- Naïve Bayes classifier to predict whether income is more than $50k per year.
- ANN to classify handwritten digits

Part A - Naive Bayes Classifier to predict income

Data:

This dataset contains over 32,000 instances of individuals, with 14 features describing demographic and employment information. The target variable is whether the individual earns over $50k per year. The features are assumed to be independent, making it a good fit for Naive Bayes classification.

Link: https://archive.ics.uci.edu/ml/datasets/Adult

Task 1: Data Preprocessing

1. Use the dataset shared.
2. Load the dataset into a pandas DataFrame.
3. Check for missing values and handle them appropriately.
4. Split the dataset into training and testing sets (80% for training, 20% for testing).

Task 2: Naive Bayes Classifier Implementation

1. Implement a function to calculate the prior probability of each class (benign and malignant) in the training set.
2. Implement a function to calculate the conditional probability of each feature given to each class in the training set.
3. Implement a function to predict the class of a given instance using the Naive Bayes algorithm.
4. Implement a function to calculate the accuracy of your Naive Bayes classifier on the testing set.

Task 3: Evaluation and Improvement

1. Evaluate the performance of your Naive Bayes classifier using accuracy, precision, recall, and F1-score.

2. Experiment with different smoothing techniques to improve the performance of your classifier. You need to study and understand the different smoothing techniques on your own.
3. Compare the performance of your Naive Bayes classifier with other classification algorithms like logistic regression and k-nearest neighbors.

Implementation Requirements:

1. Libraries like Sckitlearn, Tensorflow, Keras and PyTorch should not be used to build Naïve Bayes classifier.
2. Usage of Numpy, Pandas and Matplotlib libraries is allowed to build Naïve Bayes classifier.

Part B: Building a Basic Neural Network for Image Classification

The goal is to build a basic neural network that can classify images of handwritten digits from the MNIST dataset.

Data:

You will be using the MNIST dataset for this assignment, which contains 70,000 images of handwritten digits (0-9). The dataset is available on the internet, or you can use any library that has the MNIST dataset.

Implementation Requirements:

1. You will use a Python-based framework like TensorFlow, Keras, or PyTorch to build and train your neural network.

Architecture:

In general, a basic neural network architecture can be considered that consists of an input layer, one or more hidden layers, and an output layer.

You are supposed to build 15 distinct artificial neural network classifiers by varying one or more paramours from the following list:

(i). Number of hidden layers – 2 or 3

(ii) Total number of neurons in the hidden layer is 100 or 150

(iii) Activation function is from any of the following functions: tanh, sigmoid, ReLu

Training & Testing:

You need to train your network on the MNIST dataset. You can use any optimization algorithm like stochastic gradient descent or Adam optimizer. You need to evaluate your network's performance on a test set of images from the MNIST dataset. You can calculate the accuracy and confusion matrix to measure your network's performance.

Perform a comparative study of these 15 models and figure out the best classifier. Do you have a classifier that is not statistically significant from the best classifier? Detail the results with all explanations.

Important instructions for Part A and Part B:

- ✓ The final deliverables of the assignment should include the code and a detailed report illustrating all the details.
- ✓ You should make use of random 67% of the data to train the model and 33% of the data to test the model.
- ✓ As and when applicable, perhaps at all places, you should have several (at least 10) random training and testing splits of the data and put up results of models of all these splits along with the average and variance of performance metrics.

Contact for clarifications

In case of any queries, please contact the Teaching Assistants (TAs) of the course by email, and any other communication is invalid. You should write a mail to all the following TAs for any clarification.

1. Mr. Akshat Agrawal, f20190264@hyderabad.bits-pilani.ac.in
2. Ms. Chavali Lalita, p20190423@hyderabad.bits-pilani.ac.in
3. Ms. Chaitra C R, p20210024@hyderabad.bits-pilani.ac.in
4. Ms. P.Priyanka Chowdary, p20210022@hyderabad.bits-pilani.ac.in