**Report on Digit Recognition Using Naive Bayes Classifier**

**Introduction**

This report outlines the process and results of a digit recognition project utilizing a Naive Bayes classifier. The project involved handling image data for digits '0' and '1', feature extraction, parameter estimation, classification, and evaluation of the model's performance.

**Data Preparation**

**Initial Code**

The original data preparation code provided was designed to handle four .mat files containing images of digits '0' and '1':

* digit0\_stu\_train7302.mat
* digit1\_stu\_train7302.mat
* digit0\_testset.mat
* digit1\_testset.mat

This code successfully loaded these files using the scipy.io.loadmat function and extracted image arrays representing handwritten digits. However, it did not account for labels associated with these images.

**Modifications Made**

To address the need for labels and enhance the dataset preparation, we modified the code to manage a total of eight files, incorporating both images and labels for comprehensive data handling

Updated Data Loading Code: We expanded the data loading script to also load label files for both training and test datasets

Gene Data Code Modification: We also modified the geneNewData script to generate these additional files.

* digit0\_testset\_label
* digit1\_testset\_label
* digit0\_stu\_train7302\_label
* digit1\_stu\_train7302\_label

**Feature Extraction**

Feature extraction was performed to compute two key statistics for each image:

1. **Average Brightness**: The mean pixel value across each image.
2. **Standard Deviation of Brightness**: The variance in pixel values, providing insight into the image's contrast.

The extract features function was used to compute these statistics for both training and testing datasets.

**Naive Bayes Parameter Calculation**

The Naive Bayes classifier requires estimating the parameters of the Gaussian distributions for each feature. For each digit, the mean and variance of the average brightness and standard deviation of brightness were computed. These parameters are crucial for the likelihood estimation in the Naive Bayes classification.

The parameters were calculated using the following formulas:

* **Mean**: Average of the feature values.
* **Variance**: Measure of the spread of feature values.

These parameters were saved into a .npz file for later use in classification.

**Classification and Prediction**

The Naive Bayes classifier was employed to predict the labels of test images. The predict class function computes the likelihood of each class based on the Gaussian distributions estimated from the training data. The predictions for the test datasets were saved into .npy files for further analysis.

**Observations and Analysis**

1. **Data Loading**: The initial code provided ensured that datasets were loaded correctly, and the training and test datasets were successfully generated.
2. **Feature Statistics**: The average brightness and standard deviation provided clear distinctions between the digit classes, which are essential for accurate classification.
3. **Parameter Calculation**: The computed means and variances for the Gaussian distributions were consistent with the expected values for the given digit images.
4. **Prediction Accuracy**: The classifier's performance was assessed based on the predictions saved for test datasets. The next step involves evaluating these predictions against the true labels to determine the accuracy of the model.

**Provided Code Integration**

The provided code, including the geneNewData module, facilitated the generation of the required datasets. This module was called to ensure that the datasets were correctly prepared for training and testing. The main function in the provided code successfully loaded the datasets and printed the number of images in each set, verifying that the datasets were correctly generated.

**Conclusion**

The project demonstrated the application of a Naive Bayes classifier for digit recognition. Through the processes of feature extraction, parameter estimation, and classification, the model was able to make predictions based on the features of digit images. The integration of provided code with custom modifications ensured accurate data preparation and feature extraction, laying the groundwork for evaluating the model's performance in future steps.

RESULTS

ID: 7302

Mean of Feature 1 for Digit 0: 44.27157015306123

Variance of Feature 1 for Digit 0: 116.49525481420339

Mean of Feature 2 for Digit 0: 87.48470097019501

Variance of Feature 2 for Digit 0: 101.66337135046739

Mean of Feature 1 for Digit 1: 19.34709693877551

Variance of Feature 1 for Digit 1: 31.43803227207414

Mean of Feature 2 for Digit 1: 61.31701520189509

Variance of Feature 2 for Digit 1: 82.65110960215664

Accuracy for Digit 0: 0.9173469387755102

Accuracy for Digit 1: 0.9233480176211454