Project 1: Density Estimation and Classification

Submitted By - Setu Parekh
Student ID - 1218691890

Table of Contents

Introduction

The goal of this project is to apply the concept of Naive Bayes classification technique on the given subset of MNIST dataset. For the purpose of this project, only two digits - 1 and 0 are considered for classification. This project is divided into four phases. First phase uses the training dataset for extracting features for each image. Second phase mainly deals with calculation of density parameters - Mean and Variance for each of the features extracted for digits 0 and 1. Third phase uses the testing dataset and the density parameters obtained from the previous phase to classify the data as digit 0 or 1 using the Naive Bayes classifier. Fourth phase calculates the accuracy for the predictions made by the classifier for digit 0 and 1.

First, the given 3D dataset in Numpy format is converted into 2D Pandas dataframe for further use in the project.

Phase - 1: Feature Extraction

This phase is mainly focussed on extracting features from the given training and testing dataset of digits - 0 and 1. Each row of the dataset represents an image. Following two features are extracted for each image:

1. **The average brightness of each image**: This is found by calculating the mean of all the pixel values in a row which represents a single image.
2. **The standard deviation of each image**: This is found by calculating standard deviation of all the pixel values in a row which represents a single image.

Following four data frames are generated at the end of this phase. A class vector column is also added to the dataframe.

1. Extracted features for training dataset of digit - 0
2. Extracted features for training dataset of digit - 1
3. Extracted features for testing dataset of digit - 0
4. Extracted features for testing dataset of digit - 1

Phase - 2: Density Estimation for Naive Bayes Classifier

We make an assumption that both the features extracted in phase-1 are independent and each of the images follows normal distribution. Normal distribution is characterized by two parameters: Mean and Variance. This phase mainly involves calculation of these density parameters for the training set of digit 0 and 1.

For this purpose, the extracted features dataframe for the training set of digit 0 and 1 is merged and aggregate function is used to calculate Mean and Variance of each feature grouped by class vector.

Following 8 parameters are obtained (Feature 1 is Average Brightness and Feature 2 is Standard Deviation):

1. Mean of feature1 for digit - 0 = ***44.25***
2. Variance of feature1 for digit - 0 = ***114.59***
3. Mean of feature2 for digit - 0 = ***87.53***
4. Variance of feature2 for digit - 0 = ***101.00***
5. Mean of feature1 for digit - 1 = ***19.35***
6. Variance of feature1 for digit - 1 = ***31.98***
7. Mean of feature2 for digit - 1 = ***61.34***
8. Variance of feature2 for digit - 1 = ***83.92***

	averageBrightness		averageVariance	
	mean	var	mean	var
classVector				
0	44.253333	114.593190	87.530562	101.008413
1	19.352751	31.981927	61.345655	83.919046

Figure 1: Model output screenshot of the 8 parameters obtained in phase-2

Phase - 3: Implementing Naive Bayes Classifier

In this phase, Naive Bayes classifier is implemented to predict the class vector of the previously unknown dataset. Extracted features dataframe for test data of digit 0 and 1 is used for this purpose.

We find the probability of the class for the given dataset i.e P(class = 0 | X) and P(class = 1| X). The previously unknown dataset from the test data is classified as digit 0 or 1 based on the highest probability out of two.

Probability calculation for continuous attributes -

It is estimated using Probability Density Function (PDF) by assuming normal distribution. The formula is given as -

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

Figure 2: Probability Density Function[1]

Where,

μ - mean of the distribution

σ - standard deviation of the distribution

σ^2 - variance of the distribution

Density parameters obtained in phase-2 are used to find the probability of the continuous attributes (Feature 1 and Feature 2 are continuous attributes in our case).

According to Naive Bayes classifier -

1. **P(class = 0 | X)** = P(X | class = 0) * P(class = 0)
   = P(feature1 | class = 0) * P(feature2 | class = 0) * P(class = 0)
2. **P(class = 1 | X)** = P(X | class = 1) * P(class = 1)
   = P(feature1 | class = 1) * P(feature2 | class = 1) * P(class = 1)

For the purpose of class prediction -

1. Each row of the test data is fed into the classifier and the probabilities are calculated.
2. Values of P(class = 0 | X) and P(class = 1 | X) are compared and the highest among the two is considered as the class label for that particular dataset, X.

Phase-4: Calculating model accuracy

This phase calculates the accuracy of the predictions made by the classifier for the test data of digit 0 and 1 respectively.

Accuracy is calculated as follows -

1. Model prediction for digit 0 and 1 are compared with the ground truth respectively.
2. Number of correct predictions is counted against the total number of datasets.
3. Accuracy, % is given by = (Number of correct predictions / Total number of observations) * 100

Result -

1. Accuracy for digit - 0 test set = ***91.73%***
2. Accuracy for digit - 1 test set = ***92.33%***

Summary

This project was based on building a Naive Bayes classification model for digit 0 and 1 on the subset of MNIST data. Training data consisted of 5000 images of digit - 0 and 5000 images of digit - 1. Testing data consisted of 980 images of digit - 0 and 1135 images of digit - 1. Two features - Average and Standard deviation were extracted for each image. These features in turn were used to calculate the density estimation parameters - Mean and Variance for calculating posterior probabilities in Naive Bayes classification model. After predicting the class

vector for the previously unknown test data of digit 0 and 1, model accuracy was also calculated.

References

[1] https://en.wikipedia.org/wiki/Normal_distribution
[2] https://www.coursera.org/learn/cse575-statistical-machine-learning/supplement/VciWU/project-overview