PRML LAB 4

Vansh Agarwal
B21AI042

Q1. Scratch Implementation

Data Preprocessing & Visualisation

The imported dataset has been visualised with the distribution plots of each feature for each class (i.e. Class conditional densities) as shown in Fig 1. There are no missing value entries in the dataset. The distributions follow almost Gaussian distribution.

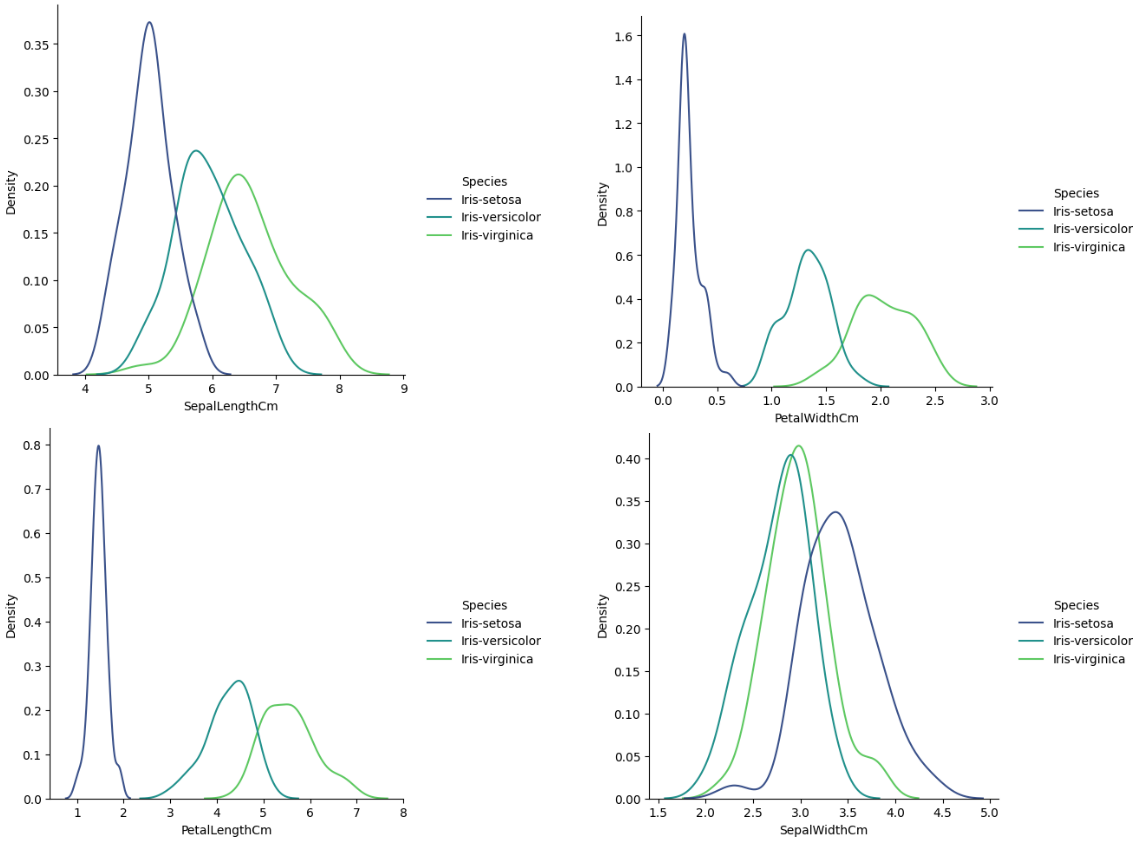

Fig 1

Scratch Implementation

Gaussian Naive Bayes is a model in which the probability of each feature given it belongs to a certain class follows a Normal Distribution. In case of multivariate the mean vector is μ_i and the covariance matrix is Σ_i for the distribution of a feature given a class i.e. $P(\mathrm{x}\,|\,\omega_i)$. The goal is to assign the class ω_i to a feature vector x if the posterior probability i.e. $P(\omega_i|x)$ is maximum out of all classes.

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}, \qquad f_{\mathbf{X}}(x_1, \ldots, x_k) = \frac{\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)}{\sqrt{(2\pi)^k|\boldsymbol{\Sigma}|}}$$

Testing & Decision boundaries for all cases

Case 1: $\Sigma_i = \sigma^2 I$
Assumes that each feature is independent of the other. The covariance matrix for each class is diagonal i.e. features have 0 covariance. This implies that for each class the contour plots formed will have the same shape and size and will be spherical since the variance of each feature is same as well. Training the model with this gives a linear discriminant function which gives rise to linear decision boundaries as can be seen in Fig 2a.
The accuracy for this case on test data is: 0.9556

Case 2: $\Sigma_i = \Sigma$
Assumes that each feature has same but arbitrary covariance matrix. This matrix is known as the pooled matrix according to linear discriminant analysis. Since all the classes share the same pooled covariance matrix, the contours of each class have same shape and size. However they might not be spherical since non diagonal entries are not necessarily zero. This again gives rise to linear discriminant function and linear decision boundaries as can be seen in Fig 2b. The pooled covariance matrix is calculated as:

$$\frac{\Sigma_{i=1}^{c}(n_i - 1)\Sigma_i}{\Sigma_{i=1}^{c}(n_i - 1)}$$

The accuracy for this case on test data is: 0.9879

Case 3: $\Sigma_i =$ Actual
Here we calculate the actual covariance matrix for each class. This gives rise to quadratic discriminant functions and the decision boundaries can be complex shapes such as parabolas, ellipses, hyperbolas etc as seen in Fig 2c.
The accuracy for this case on test data is: 0.9778

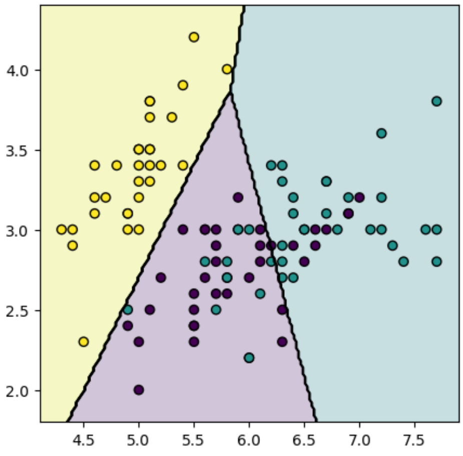

Fig 2a

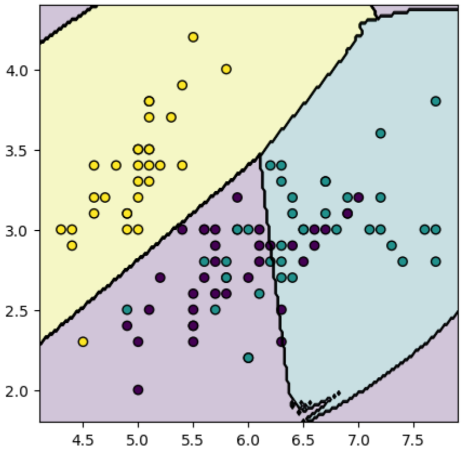

Fig 2b

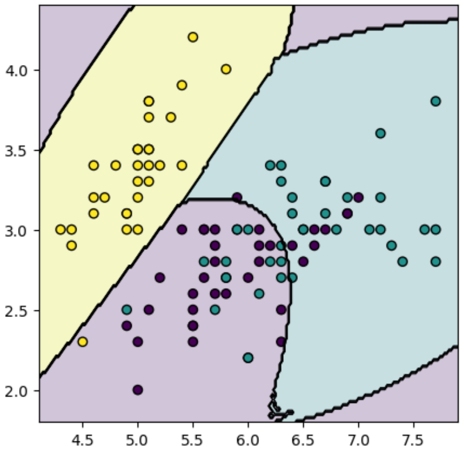

Fig 2c

5 Fold Cross Validation

The 5 fold cross validation scores are: **[0.9524, 1.0, 0.8571, 1.0, 1.0]**
And the **mean is 0.9619**

Synthetic Circular dataset

The synthetic dataset has been generated as shown in Fig 3a. The purple points represent class 1 and yellow represent class 2. Gaussian Niave Bayes case 3 is required to fit the data as this is linearly inseparable. No single straight line can separate the dataset perfectly and a curved boundary is required. Hence we choose case 3 for the model training. The decision Boundary has been plotted as shown in Fig 3b.

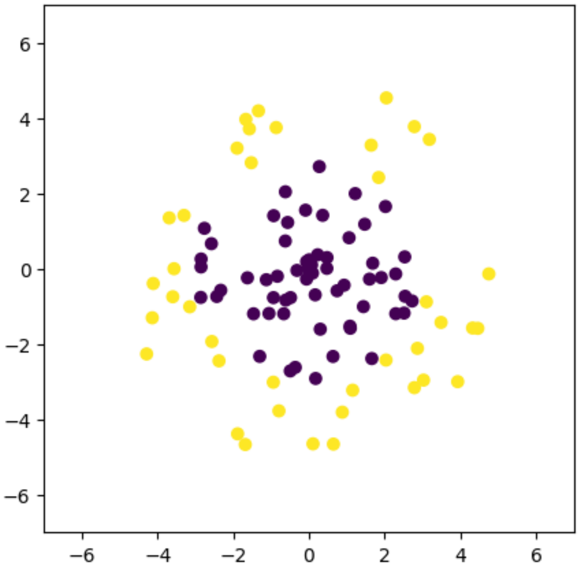

Fig 3a

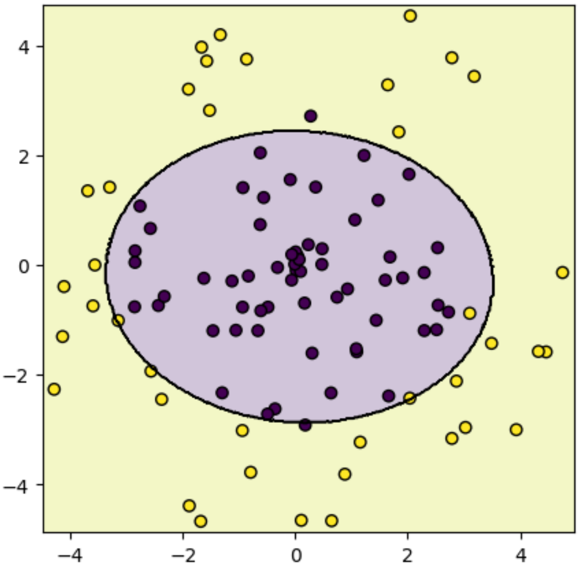

Fig 3b

Q2. Mahalanobis Distance

1.

The eigenvectors of covariance matrix signify the directions of maximum and minimum spreads of datapoints and their corresponding eigenvalues denote the magnitude of spread in these directions.
The matrix Σ_s is found to show correlation between features. That means the contour will be elliptical in shape and have some inclination with the axes as shown in fig 4.

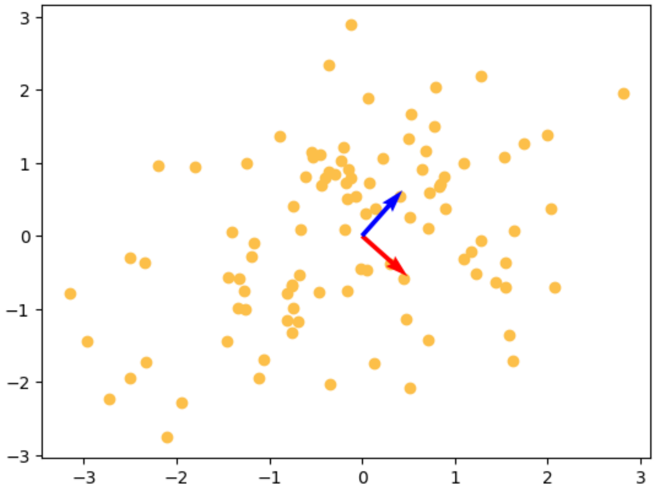

Fig 4

The eigenvectors are:
e_1 **= [0.7475, -0.6642]**
e_2 **= [0.6642, 0.7475]**
The corresponding eigenvalues are: 0.984 and 0.4539.
The dot product of e_1 **and** e_2 **is 0 which symbolizes these two eigenvectors are orthogonal.**

2.

The transformation performed is **whitening transform.** The covariance matrix now comes out to be almost identity matrix. This means that features show zero correlation. That means the initial covariance matrix has been decorelated. This means that the eigenvectors are now axis aligned and the non diagonal entries of covariance matrix are 0.

Also the dataset has been scaled such that the contour forms a circular shape indicating that the spread along both eigenvectors is same. i.e. **isotropic covariance matrix** as in Fig5.

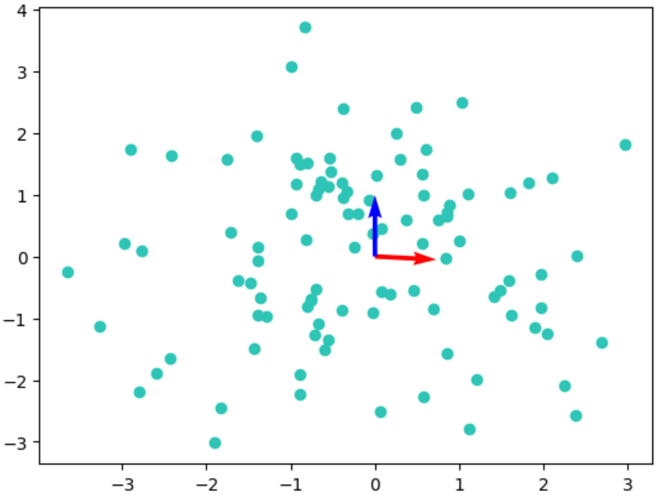

Fig 5

The eigenvectors are close to:
$e_1 = [1, 0]$
$e_2 = [0, 1]$
The corresponding eigenvalues are close to: 1 and 1.
The dot product of e_1 and e_2 is 0 which symbolizes these two eigenvectors are orthogonal.

3 & 4.

Fig 6 shows the scatter of points and the barplots of euclidian distance before the transformation.
Fig 7 shows the scatter of points and the barplots of euclidian distance after the transformation.

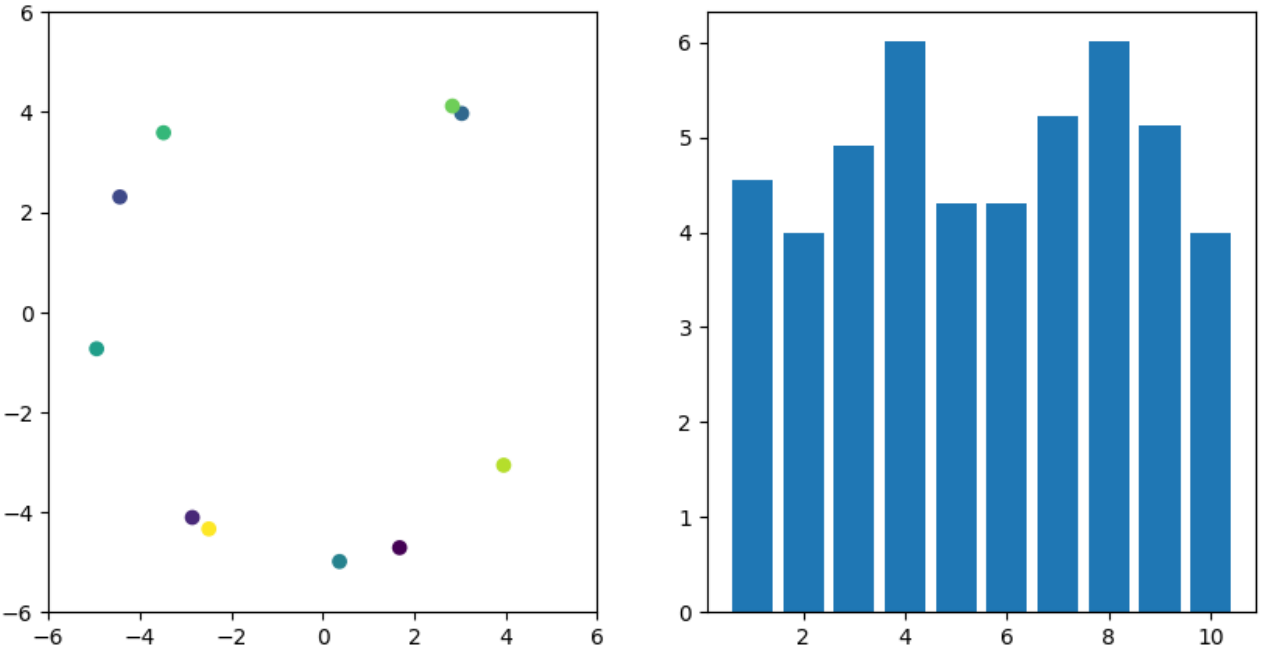

Fig 6

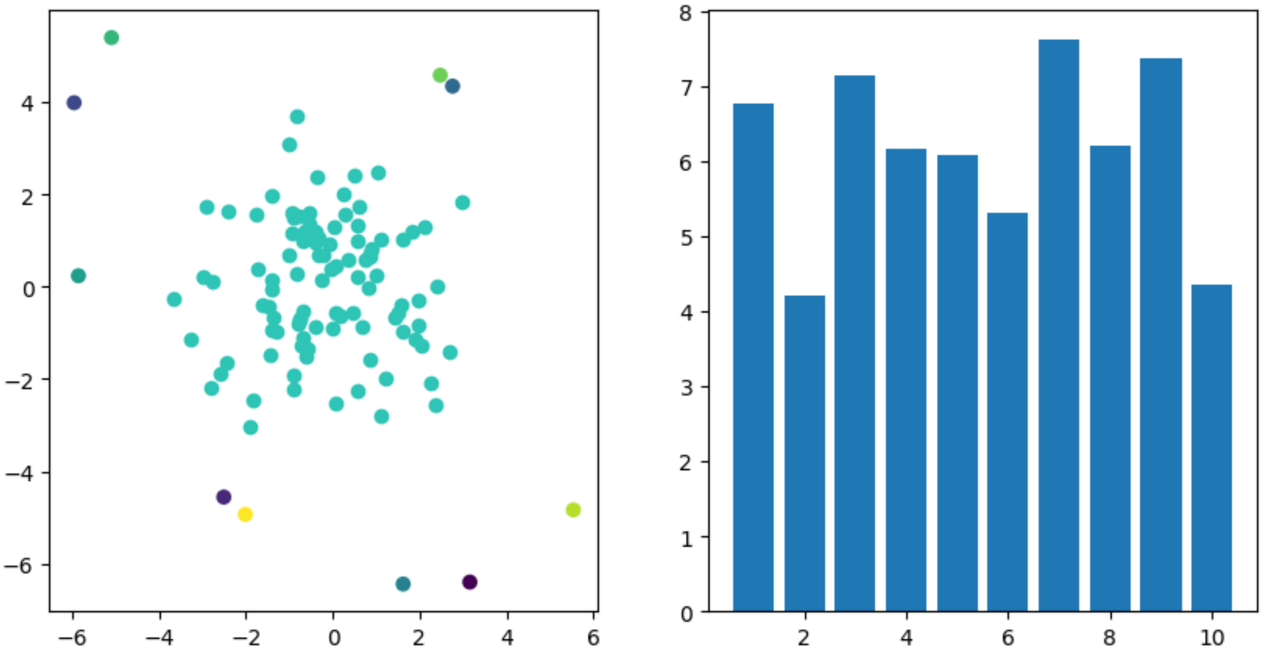

Fig 7