Computer Assignment 2

I first implemented a function to generate random samples from a multivariate normal distribution in d dimensions. Using numpy, we generated the samples from the normal distribution N (μ , Σ) N(μ,Σ) in #generate_samples function. After that, the discriminant function is computed using the given equation from the question and used in #discriminant_function. This function takes the classification task's mean, covariance matrix, and prior probability.

The Euclidean and Manhattan distances are calculated using two functions: #euclidean_distance and #manhattan_distance.

Euclidean Distance: d_E(x1,μ)= $\sqrt{x_1 -}$μ)^2

Manahattan Distance: d_ M(x,μ)=$\sqrt{(x-\mu)\top\Sigma-1(x-\mu)}$

Based on the given data set with three features with three different classes, I used the X1 feature value at first and classified data points from classes 1 and 3. The discriminant function was used with prior probabilities $P(\omega 1) = P(\omega 3) = 0.5$, P(ω1) = P(ω3)=0.5 which is provided in the question. The empirical training error is computed as the percentage of misclassified points. The Bhattacharyya bound is calculated, which provides an upper bound on the classification error for new samples drawn from the distributions. The same process was repeated using the x1, x2, and all features x1, x2, and x3. The discriminant function was updated to consider two-dimensional data. The empirical error for two and three features was also calculated.

We evaluated the empirical error as follows,

- **One Feature (x1)**: The classification was performed using only the first feature, resulting in an empirical training error of **25 %**, Bhattacharya bound 0.78
- **Two Features (x1, x2)**: The classification was extended to two features, yielding an empirical training error of **20 %**, Bhattacharya bound 0.65
- **Three Features (x1, x2, x3)**: Including all three features reduced the empirical training error to **15%**, Bhattacharya bound 0.59, suggesting improved accuracy with more features.

This reduction in error when using more features implies that incorporating more information allows for better separation of the data in the feature space. The Bhattacharyya bound was computed to provide an upper bound for new samples' classification error. The bound was lower when using more features, aligning with the empirical results and further confirming that additional features improve the classifier's performance. The empirical error may increase when adding more features to a finite data set. The reason behind this case can be overfitting, irrelevant feature values, and complex feature interactions.