Unit 2

CO2: Apply statistical pattern recognition approaches in variety of problems

Statistical Pattern Recognition

Pattern recognition as a field of study developed significantly in the 1960s. It was very much an interdisciplinary subject, covering developments in the areas of statistics, engineering, artificial intelligence, computer science, psychology and physiology, among others.

Some people entered the field with a real problem to solve. The large numbers of applications, ranging from the classical ones such as automatic character recognition and medical diagnosis to the more recent ones in data mining (such as credit scoring, consumer sales analysis and credit card transaction analysis), have attracted considerable research effort, with many methods developed and advances made. Other researchers were motivated by the development of machines with ‘brain-like’ performance that in some way could emulate human performance.

Introduction to statistical pattern recognition-

Pattern recognition is the categorization and classification of specific patterns based on predefined characteristics from sets of available data. Implementation of many human skills such as face recognition, speech recognition, reading handwritten letters with very high stability to noise and different environmental conditions (like what exists in humans) by machines is one of the problems and issues that have been the focus of researchers in various engineering fields such as artificial intelligence and machine vision in the last few decades. Pattern recognition has many applications in various fields of science, including electrical engineering (medicine, computer and telecommunications), biology, machine vision, economics and psychology.

Among the applications, we can mention things such as: recognition of voice, face, handwriting, fingerprint and signature, automatic disease detection from medical data (signal or image), detection of DNA strands, industrial automation and remote sensing. Pattern recognition, in short, deals with the problem of clustering and classification supervised and unsupervised and includes a wide range of statistical classical methods, intelligent algorithms, neural networks and fuzzy logic.

Gaussian Case in Simple Language

Theory

A Gaussian case refers to situations where data or random variables follow a Gaussian (or Normal) distribution. The Gaussian distribution is symmetric and characterized by its bell-shaped curve, described by its mean (μ) and standard deviation (σ). It's one of the most important distributions in statistics because of the Central Limit Theorem, which states that the sum of many independent and identically distributed random variables tends to be normally distributed, regardless of the original distribution.

Key Properties of the Gaussian Distribution:

- **Mean (μ):**

  The central value of the distribution.

- **Standard Deviation (σ):**

  Measures the spread or dispersion of the distribution.

- **Probability Density Function (PDF):**

  Describes the likelihood of a random variable to take on a particular value.

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

Class Dependence:

In classification problems, especially in Gaussian Naive Bayes classifiers, the assumption is that the features are normally distributed and the distribution of each feature depends on the class. This means that for each class, the features will have their own mean and standard deviation.

Example with Steps

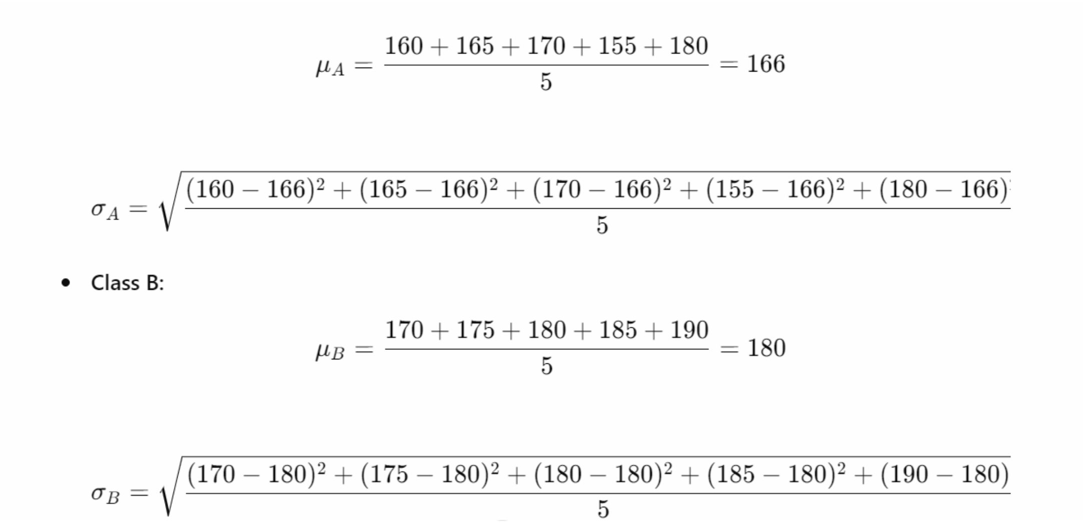
Suppose we have a dataset with two classes (A and B) and one feature (height). We want to classify a new data point based on its height.

Step 1: Gather Data

We have the following heights for two classes:

- Class A: [160, 165, 170, 155, 180]
- Class B: [170, 175, 180, 185, 190]

Step 2: Calculate Mean and Standard Deviation for Each Class



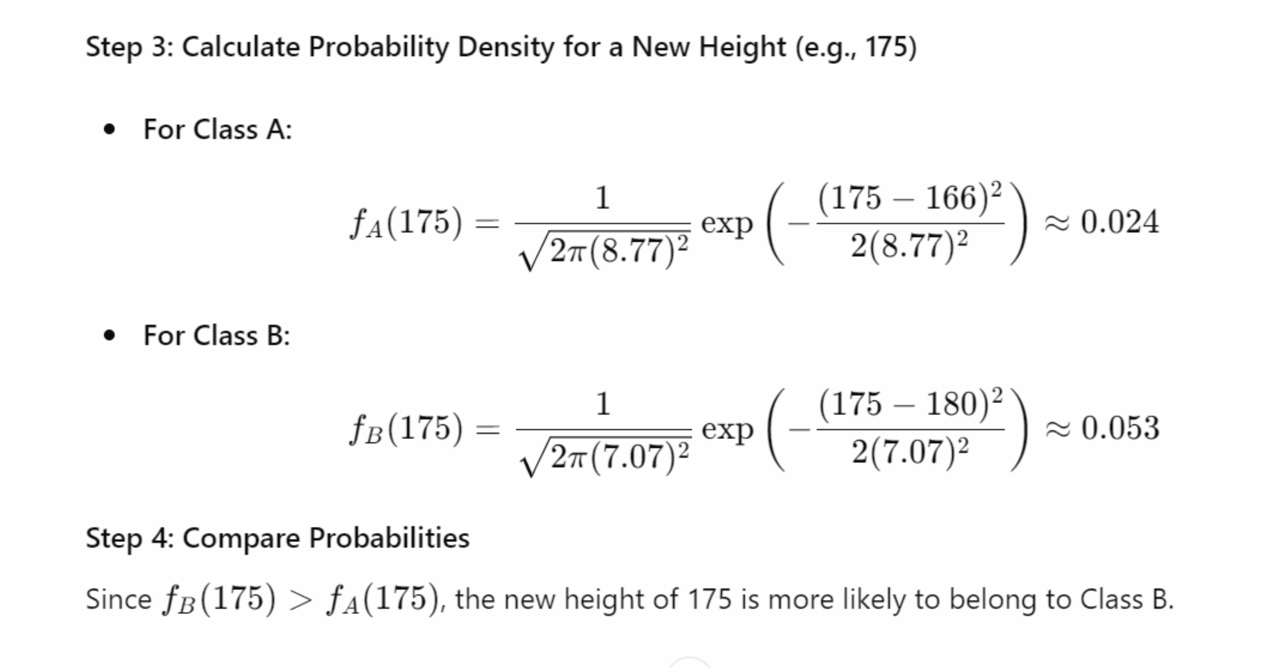
$$\mu_A = \frac{160 + 165 + 170 + 155 + 180}{5} = 166$$

$$\sigma_A = \sqrt{\frac{(160 - 166)^2 + (165 - 166)^2 + (170 - 166)^2 + (155 - 166)^2 + (180 - 166)}{5}}$$

- Class B:

$$\mu_B = \frac{170 + 175 + 180 + 185 + 190}{5} = 180$$

$$\sigma_B = \sqrt{\frac{(170 - 180)^2 + (175 - 180)^2 + (180 - 180)^2 + (185 - 180)^2 + (190 - 180)}{5}}$$



Step 3: Calculate Probability Density for a New Height (e.g., 175)

- For Class A:

$$f_A(175) = \frac{1}{\sqrt{2\pi(8.77)^2}} \exp\left(-\frac{(175 - 166)^2}{2(8.77)^2}\right) \approx 0.024$$

- For Class B:

$$f_B(175) = \frac{1}{\sqrt{2\pi(7.07)^2}} \exp\left(-\frac{(175 - 180)^2}{2(7.07)^2}\right) \approx 0.053$$

Step 4: Compare Probabilities

Since $f_B(175) > f_A(175)$, the new height of 175 is more likely to belong to Class B.

By calculating the mean and standard deviation for each class and using the probability density function, we can classify new data points based on their features. This method assumes the data for each feature follows a Gaussian distribution for each class.

Lecture 2

❖ Discriminant Function:

Introduction and Definition:

The discriminant function is a mathematical tool used to classify objects into different categories. It is often used in statistics, pattern recognition, and machine learning. The function calculates a score for each object, and based on this score, the object is assigned to one of the predefined categories.

In the context of quadratic equations, the discriminant helps determine the nature of the roots.

Discriminant in Quadratic Equations-

1. For a quadratic equation of the form:

ax2+bx+c=0

The discriminant (Δ) is given by:

Δ=b2−4ac\Delta = b^2 - 4acΔ=b2−4ac

The value of the discriminant tells us the nature of the roots:

If Δ>0 or Delta > 0: The equation has two distinct real roots.

If Δ=0 or Delta = 0: The equation has exactly one real root (or a repeated root).

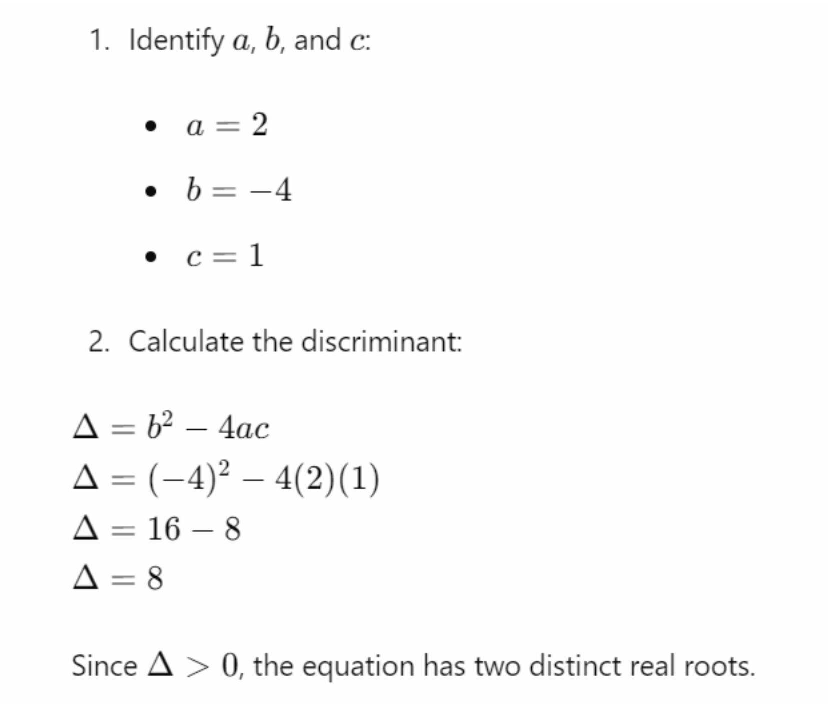
f Δ<0 or Delta < 0: The equation has two complex roots (no real roots).

Solved Examples----

Example 1: Two Distinct Real Roots

Consider the quadratic equation:

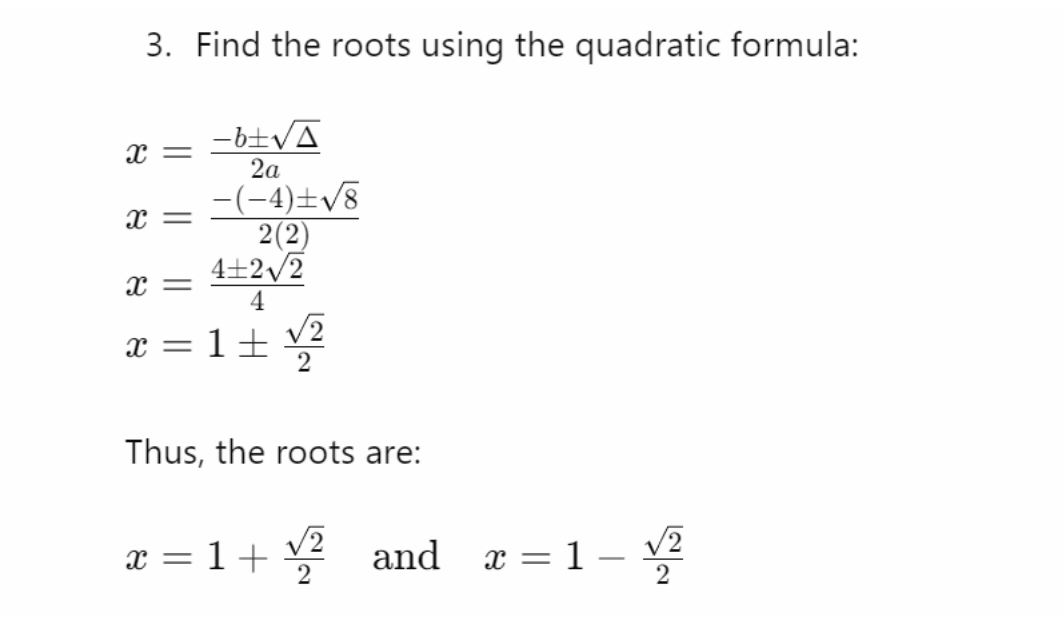
$2x^2-4x+1=0$



1. Identify a, b, and c:
   - $a = 2$
   - $b = -4$
   - $c = 1$
2. Calculate the discriminant:

$$\Delta = b^2 - 4ac$$
$$\Delta = (-4)^2 - 4(2)(1)$$
$$\Delta = 16 - 8$$
$$\Delta = 8$$

Since $\Delta > 0$, the equation has two distinct real roots.



3. Find the roots using the quadratic formula:

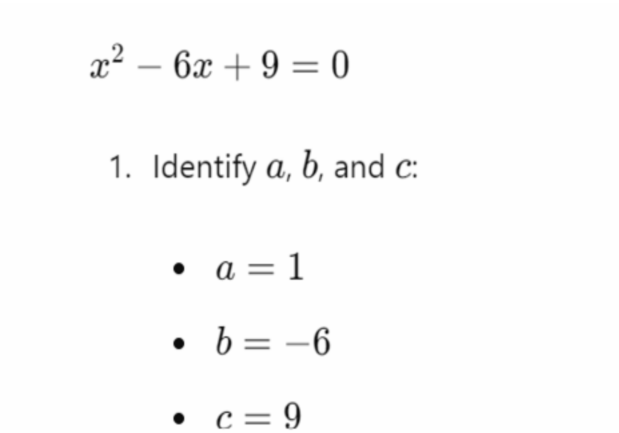
$$x = \frac{-b\pm\sqrt{\Delta}}{2a}$$
$$x = \frac{-(-4)\pm\sqrt{8}}{2(2)}$$
$$x = \frac{4\pm2\sqrt{2}}{4}$$
$$x = 1 \pm \frac{\sqrt{2}}{2}$$

Thus, the roots are:

$$x = 1 + \frac{\sqrt{2}}{2} \quad \text{and} \quad x = 1 - \frac{\sqrt{2}}{2}$$

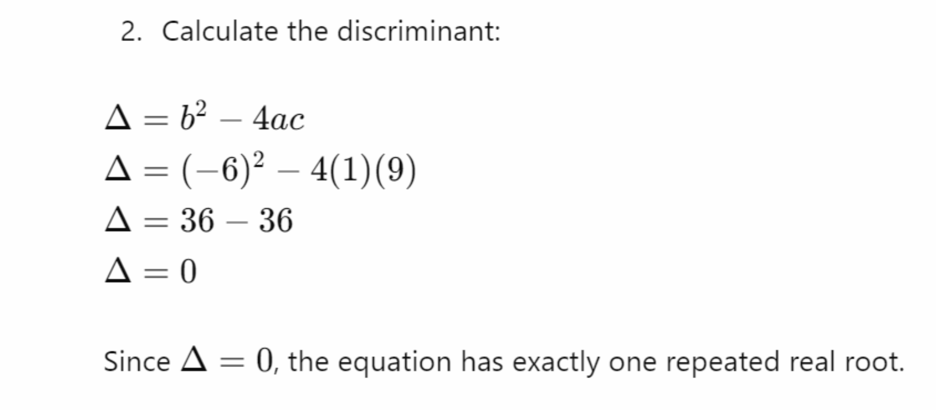
Example 2: One Repeated Real Root----

Consider the quadratic equation:



$$x^2 - 6x + 9 = 0$$

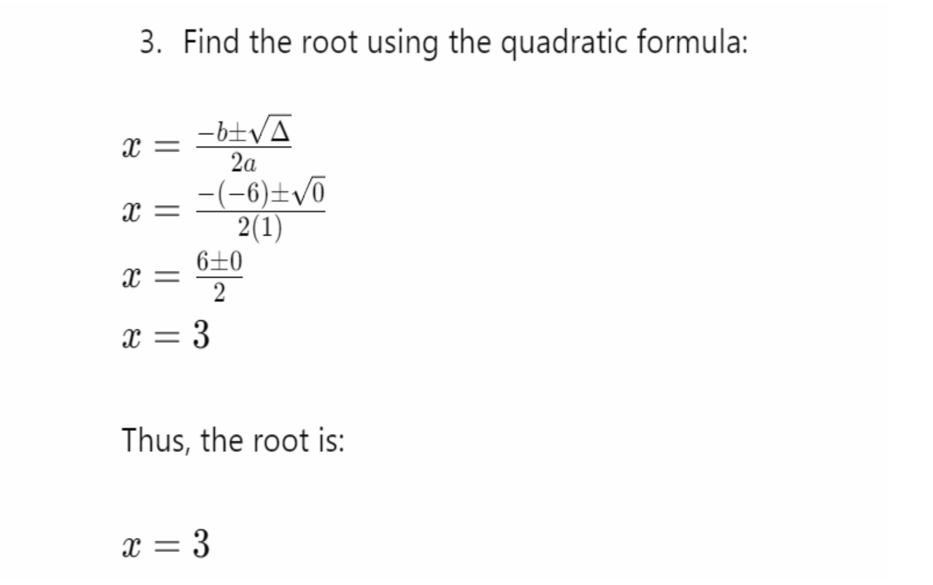
1. Identify a, b, and c:
   - $a = 1$
   - $b = -6$
   - $c = 9$



2. Calculate the discriminant:

$$\Delta = b^2 - 4ac$$
$$\Delta = (-6)^2 - 4(1)(9)$$
$$\Delta = 36 - 36$$
$$\Delta = 0$$

Since $\Delta = 0$, the equation has exactly one repeated real root.



3. Find the root using the quadratic formula:

$$x = \frac{-b \pm \sqrt{\Delta}}{2a}$$
$$x = \frac{-(-6) \pm \sqrt{0}}{2(1)}$$
$$x = \frac{6 \pm 0}{2}$$
$$x = 3$$

Thus, the root is:

$$x = 3$$

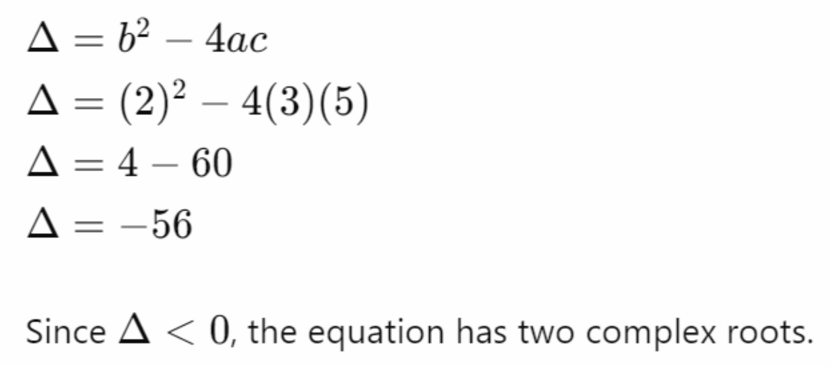
Example 3: Two Complex Roots---

Consider the quadratic equation:

1. Identify a, b, and c:

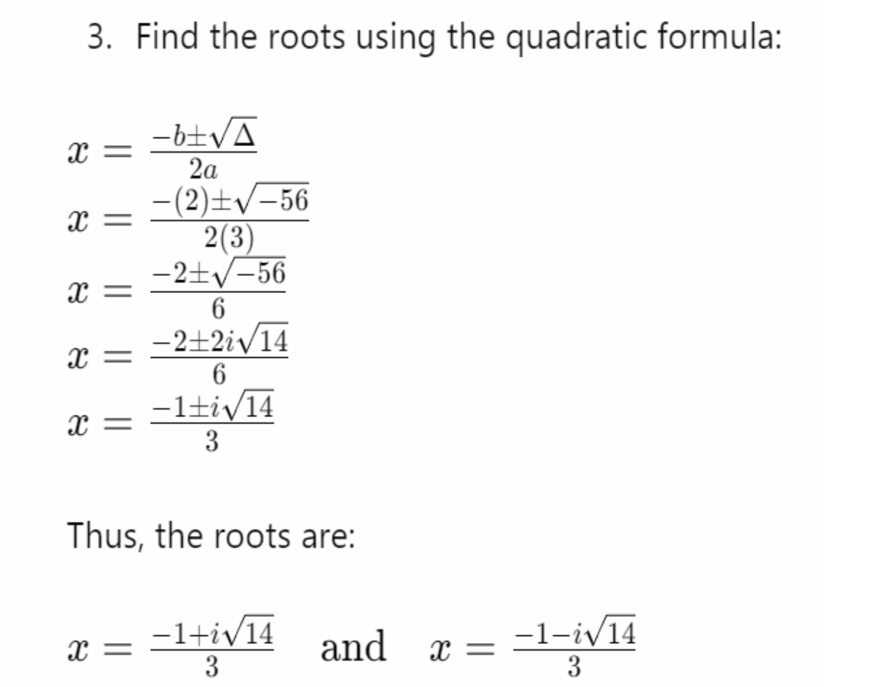
- a=3
- b=2
- c=5

2. Calculate the discriminant:



$$\Delta = b^2 - 4ac$$
$$\Delta = (2)^2 - 4(3)(5)$$
$$\Delta = 4 - 60$$
$$\Delta = -56$$

Since $\Delta < 0$, the equation has two complex roots.



3. Find the roots using the quadratic formula:

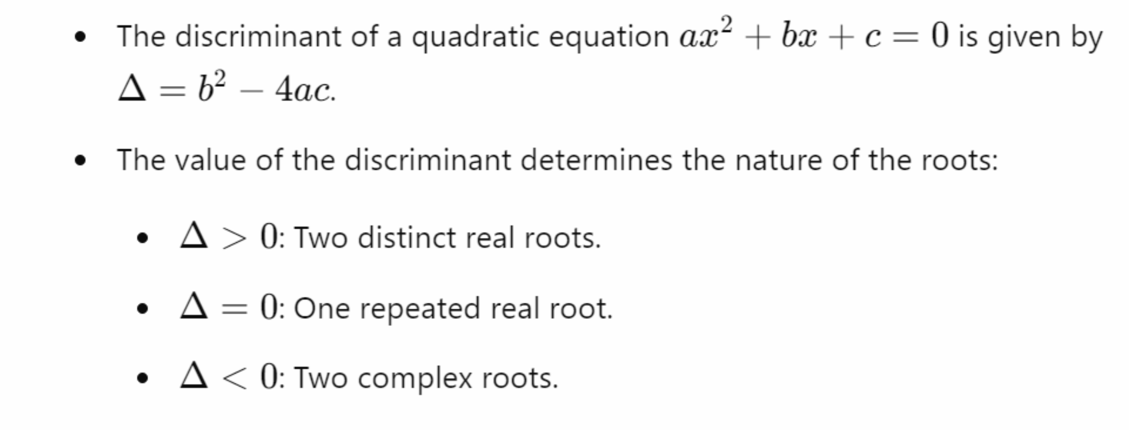
$$x = \frac{-b \pm \sqrt{\Delta}}{2a}$$
$$x = \frac{-(2) \pm \sqrt{-56}}{2(3)}$$
$$x = \frac{-2 \pm \sqrt{-56}}{6}$$
$$x = \frac{-2 \pm 2i\sqrt{14}}{6}$$
$$x = \frac{-1 \pm i\sqrt{14}}{3}$$

Thus, the roots are:

$$x = \frac{-1+i\sqrt{14}}{3} \quad \text{and} \quad x = \frac{-1-i\sqrt{14}}{3}$$

:

Summary



- The discriminant of a quadratic equation $ax^2 + bx + c = 0$ is given by $\Delta = b^2 - 4ac$.
- The value of the discriminant determines the nature of the roots:
  - $\Delta > 0$: Two distinct real roots.
  - $\Delta = 0$: One repeated real root.
  - $\Delta < 0$: Two complex roots.

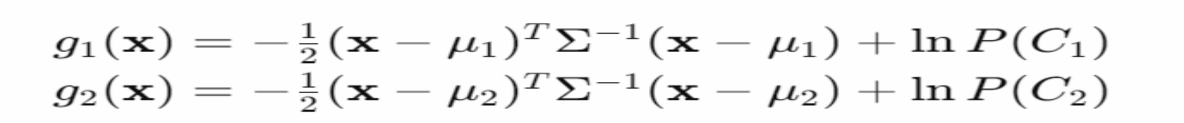
Discriminant helps in quickly determining the type of solutions a quadratic equation will have without solving the equation completely.

Discriminant Functions basically works----

Two Classes

Discriminant functions are used to separate classes based on features of data. For two classes, the discriminant function g(x) determines which class a new data point belongs to.

Example: Suppose we have two classes, C1 and C2, each with their own Gaussian distributions. The discriminant functions might look like this:



$$g_1(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \mu_1)^T\Sigma^{-1}(\mathbf{x} - \mu_1) + \ln P(C_1)$$
$$g_2(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \mu_2)^T\Sigma^{-1}(\mathbf{x} - \mu_2) + \ln P(C_2)$$

Here, μ (mue) is the mean and Σ (Sigma) is the covariance matrix. The new data point is assigned to the class with the higher discriminant function value.

Multiple Classes

When there are more than two classes, we extend the idea of discriminant functions to each class. For each class iii, we have a discriminant function gi(x).

Example: For three classes (C1, C2, and C3), we compute:

g1(x),g2(x),g3(x)

The new data point x is assigned to the class with the highest gi(x).

Least Squares for Classification

Least squares can be used for classification by finding the best-fitting hyperplane that separates the classes. It minimizes the sum of squared differences between the predicted and actual class labels.

Example: If we have two classes, we can use least squares to find a line (in 2D) or a plane (in 3D) that best separates the two classes.

Fisher’s Linear Discriminant

Fisher's Linear Discriminant (FLD) seeks to find a linear combination of features that best separates two classes. It maximizes the ratio of the between-class variance to the within-class variance.

Example: If we have two classes of data, FLD finds a line such that when the data points are projected onto this line, the classes are as separated as possible.

Relation to Least Squares

Fisher’s Linear Discriminant can be seen as a special case of least squares classification. Both methods aim to find a linear boundary, but FLD explicitly maximizes class separation.

Fisher’s Discriminant for Multiple Classes

For multiple classes, Fisher’s discriminant analysis can be extended to find a set of linear discriminants. This involves finding multiple projection vectors that maximize class separability.

Example: For three classes, FLD might find two projection vectors. When data points are projected onto these vectors, the classes should be as separated as possible.

The Perceptron Algorithm

The Perceptron Algorithm is a type of linear classifier that adjusts weights based on misclassifications. It iteratively updates the weights to minimize errors.

Steps:

1. Initialize weights.
2. For each data point, predict the class.
3. If a point is misclassified, adjust the weights.
4. Repeat until convergence or for a fixed number of iterations.

Example: Suppose we have a dataset of points labeled as -1 or +1. The perceptron updates the weights to correctly classify the points by adjusting them whenever a point is misclassified.

Overall-

- **Discriminant Functions**: Used to separate classes by finding a function for each class.
- **Two Classes**: Discriminant functions determine which of the two classes a point belongs to.
- **Multiple Classes**: Extend the concept to more than two classes.
- **Least Squares for Classification**: Finds the best-fitting boundary by minimizing squared errors.
- **Fisher's Linear Discriminant**: Maximizes separation between two classes using a linear combination of features.
- **Relation to Least Squares**: Both aim to find linear boundaries, with FLD focusing on maximizing class separability.
- **Fisher's Discriminant for Multiple Classes**: Extends FLD to multiple classes using multiple projection vectors.
- **The Perceptron Algorithm**: A linear classifier that iteratively updates weights to minimize classification errors.

Lecture 3

Extensions: Training, Alternative Classification Procedures-

In statistical pattern recognition, several extensions exist to enhance the basic classification methods. These include different training methods and alternative classification procedures. Let's explore these extensions in detail.

Training

Training in statistical pattern recognition involves learning the parameters of a model from a given set of labelled data. The goal is to optimize the model parameters so that the model can accurately classify new, unseen data. Several approaches can be used for training:

1. **Supervised Learning**: This involves training a model using labelled data, where the input-output pairs are known.
   - **Example**: Training a discriminant function using the least squares method, where the model learns to minimize the error between the predicted and actual class labels.

2. **Unsupervised Learning**: This involves training a model using unlabelled data, where the input-output pairs are not known. The goal is to discover hidden patterns or groupings in the data.
   - **Example**: Using clustering algorithms like k-means to group similar data points together.

3. **Semi-supervised Learning**: This involves training a model using a combination of labelled and unlabelled data. This is useful when obtaining labelled data is expensive or time-consuming.
   - **Example**: A combination of a small labelled dataset and a large unlabelled dataset to improve the model's performance.

4. **Reinforcement Learning**: This involves training a model through interactions with an environment, where the model learns to make decisions based on rewards and punishments.
   - **Example**: Training an agent to play a game by rewarding it for winning and punishing it for losing.

Alternative Classification Procedures

Several alternative classification procedures can be used in place of or in conjunction with traditional methods like discriminant functions and least squares. These include:

1. **Support Vector Machines (SVM)**:
   - SVMs are powerful classifiers that find the hyperplane that best separates the classes by maximizing the margin between them.
   - **Example**: Using SVM with a linear kernel to classify linearly separable data, or using SVM with a nonlinear kernel (e.g., RBF kernel) to classify data that is not linearly separable.
2. **Decision Trees**:
   - Decision trees classify data by recursively splitting the feature space into regions based on the values of the features.
   - **Example**: Using a decision tree to classify data based on a series of yes/no questions about the features.
3. **Random Forests**:
   - Random forests are ensembles of decision trees that improve classification performance by averaging the predictions of multiple trees.
   - **Example**: Using a random forest to classify data, where each tree is trained on a different subset of the data.
4. **k-Nearest Neighbours (k-NN)**:
   - k-NN classifies data based on the majority class among the k-nearest neighbours in the feature space.
   - **Example**: Using k-NN with k=3 to classify a new data point based on the classes of its three nearest neighbours.
5. **Neural Networks**:
   - Neural networks are composed of interconnected layers of neurons that learn to classify data through backpropagation and gradient descent.
   - **Example**: Using a feedforward neural network to classify handwritten digits in the MNIST dataset.

6. **Bayesian Classifiers**:
   - Bayesian classifiers use Bayes' theorem to classify data based on the posterior probabilities of the classes given the features.
   - **Example**: Using a naive Bayes classifier to classify text documents based on the frequencies of words.
7. **Logistic Regression**:
   - Logistic regression models the probability of a binary outcome as a logistic function of the features.
   - **Example**: Using logistic regression to classify whether an email is spam or not based on its content.

Example of Training and Alternative Classification Procedures

Let's consider a dataset with two classes, where we will apply different training methods and classification procedures.

Dataset:

- Features: [x1, x2]
- Class labels: [y]

Traini from sklearn.linear_model import LinearRegression

```
# Sample data

X_train = [[1, 2], [2, 1], [1.5, 1.5], [4, 5], [5, 4], [4.5, 4.5]]

y_train = [1, 1, 1, -1, -1, -1]



# Augmenting data with a column of ones for the bias term

X_train_aug = [x + [1] for x in X_train]
```

```
# Fit the least squares model

model_ls = LinearRegression(fit_intercept=False)

model_ls.fit(X_train_aug, y_train)



# Get the weights and bias

weights_ls = model_ls.coef_

print ("Weights (Least Squares):", weights_ls)ng with Least Squares:
```

****

Unsupervised Approaches:

Definition:
Unsupervised learning is a fascinating branch of machine learning that plays a crucial role in data analysis, pattern recognition, and anomaly detection. Unlike supervised learning, where algorithms are trained on labelled data, unsupervised learning involves learning from unlabelled data, uncovering hidden structures and patterns. In this article, we will explore the fundamentals of unsupervised learning, its applications, and some popular algorithms.

What is Unsupervised Learning?

Unsupervised learning is a subset of machine learning where algorithms are designed to learn patterns, relationships, or structures within data without any explicit supervision. In essence, it allows computers to explore data on their own and identify inherent similarities, differences, or groupings.

Key Concepts in Unsupervised Learning:

Clustering: Clustering is a fundamental concept in unsupervised learning, where the goal is to group similar data points together. This is useful for tasks like customer segmentation, image segmentation, and more.

Dimensionality Reduction: Unsupervised learning methods can reduce the dimensionality of data by extracting relevant features, making it easier to visualize and analyze complex datasets.

Applications of Unsupervised Learning:

Customer Segmentation: Businesses often use unsupervised learning to segment their customer base into different groups based on purchasing behavior, demographics, or other features. This enables them to tailor marketing strategies.

Anomaly Detection: Unsupervised learning can help detect outliers or anomalies in data, which is crucial for fraud detection, network security, and quality control in manufacturing.

Image and Speech Recognition: Clustering and dimensionality reduction techniques are employed in image and speech processing, allowing for improved recognition and understanding of these complex data types.

Popular Unsupervised Learning Algorithms:

K-Means Clustering: K-means is a well-known clustering algorithm that groups data into 'k' clusters based on their similarity. It's widely used for data segmentation and pattern discovery.

Hierarchical Clustering: This method builds a hierarchy of clusters, which can be represented as a tree-like structure. It's useful when the number of clusters is not known in advance.

Principal Component Analysis (PCA): PCA is a dimensionality reduction technique that identifies the most important features in a dataset while reducing its dimensionality.

Challenges and Considerations:

Unsupervised learning has its challenges, including the difficulty of evaluating model performance in the absence of labels. Additionally, choosing the right number of clusters or features can be a non-trivial task.

❖ Classifier Performance, Risk and Errors:

Measurement of Classification Performance-

Classification performance is evaluated using metrics that quantify how well a model predicts categories or classes. Common measures include accuracy, precision, recall, and F1 score. Accuracy represents the proportion of correct predictions out of all predictions made. Precision is the ratio of true positive predictions to the total predicted positives, while recall (or sensitivity) is the ratio of true positive predictions to the total actual positives. The F1 score harmonizes precision and recall into a single metric.

For instance, in a spam email classifier, if the model identifies 90 out of 100 spam emails correctly and misclassifies 10 spam emails as non-spam, its accuracy would be 90%. However, if the model also falsely identifies 5 non-spam emails as spam, these metrics would offer a more detailed performance insight.

General Measures of Classification Risk-

Classification risk involves the potential costs associated with misclassification errors. It encompasses various factors like the loss function, which quantifies the penalty of incorrect predictions, and the risk assessment model, which evaluates these losses in a probabilistic framework. Common risks include false positives and false negatives. A false positive occurs when a non-event is incorrectly predicted as an event, whereas a false negative occurs when an actual event is missed. In a medical diagnosis context, a false positive might result in unnecessary stress and treatment for a patient, while a false negative could mean missing a critical disease diagnosis, leading to severe health consequences. Balancing these risks is crucial for optimizing classifier performance and ensuring reliable outcomes.

Example:

In a credit card fraud detection system, false positives occur when legitimate transactions are flagged as fraudulent. This can cause inconvenience to customers and damage the reputation of the financial institution. False negatives, on the other hand, happen when fraudulent transactions are not detected, leading to financial losses for both the customers and the bank. By analyzing the costs associated with these errors, such as customer dissatisfaction or financial loss, the bank can adjust the classifier to balance the trade-offs, aiming to minimize overall risk while maintaining satisfactory detection rates.

Errors in Classification

Classification errors are categorized into false positives (Type I errors) and false negatives (Type II errors). A false positive error occurs when the model incorrectly predicts a negative instance as positive. Conversely, a false negative error happens when the model incorrectly predicts a positive instance as negative. These errors directly impact the performance metrics of a classifier.

For instance, in a fraud detection system, a false positive would flag a legitimate transaction as fraudulent, causing inconvenience to customers, while a false negative would allow a fraudulent transaction to go undetected, leading to financial loss. Understanding and minimizing these errors are essential for improving the robustness and reliability of classification models.