Machine Learning

Dr. Thyagaraju G S

Dr.Thyagaraju G S, Professor and HoD, Department of CSE, SDM Institute Of Technology,Ujire-574240.
Source Book : S. Sridhar, M Vijayalakshmi “Machine Learning”. Oxford, 2021

Module1: Syllabus

Introduction: Need for Machine Learning, Machine Learning Explained, Machine Learning in Relation to other Fields, Types of Machine Learning, Challenges of Machine Learning, Machine Learning Process, Machine Learning Applications.

Understanding Data – 1: Introduction, Big Data Analysis Framework, Descriptive Statistics, Univariate Data Analysis and Visualization.

Chapter-1, 2 (2.1-2.5)

Module2: Syllabus

Understanding Data – 2: Bivariate Data and Multivariate Data, Multivariate Statistics, Essential Mathematics for Multivariate Data, Feature Engineering and Dimensionality Reduction Techniques.

Basic Learning Theory: Design of Learning System, Introduction to Concept of Learning, Modelling in Machine Learning.

Chapter-2 (2.6-2.8, 2.10), Chapter-3 (3.3, 3.4, 3.6)

Module 2.1: Understanding Data-2

1. Mean, Variance and Standard Deviation
2. Bivariate Data and Multivariate Data
3. Multivariate Statistics,
4. Essential Mathematics for Multivariate Data,
5. Feature Engineering and Dimensionality Reduction Techniques.

2.1.1: Mean, Variance and Standard Deviation

Example Dataset

- Consider the following five numbers representing the scores of five students in a test:
- **X=[10,20,30,40,50]**

1. Mean (Average)

The **mean** is the average of all values in the dataset. It is calculated as:

$$\text{Mean}(\mu) = \frac{\sum X_i}{N}$$

where:

- X_i are the individual values,
- N is the number of values.

Calculation for our dataset:

$$\mu = \frac{10 + 20 + 30 + 40 + 50}{5} = \frac{150}{5} = 30$$

So, the Mean is 30.

2. Variance (σ^2)

The **variance** measures how much the data points deviate from the mean. It is calculated as:

$$\sigma^2 = \frac{\sum(X_i - \mu)^2}{N}$$

Calculation:

1. Find the difference from the mean for each value:
   - $10 - 30 = -20$
   - $20 - 30 = -10$
   - $30 - 30 = 0$
   - $40 - 30 = 10$
   - $50 - 30 = 20$

2. Square each difference:
   - $(-20)^2 = 400$
   - $(-10)^2 = 100$
   - $(0)^2 = 0$
   - $(10)^2 = 100$
   - $(20)^2 = 400$
3. Compute the average of these squared differences:

$$\sigma^2 = \frac{(400 + 100 + 0 + 100 + 400)}{5} = \frac{1000}{5} = 200$$

So, the Variance is 200.

3. Standard Deviation (σ)

The **standard deviation** is the square root of the variance:

$$\sigma = \sqrt{\sigma^2}$$

Calculation:

$$\sigma = \sqrt{200} \approx 14.14$$

So, the Standard Deviation is approximately 14.14.

Summary of Results

For the dataset $[10, 20, 30, 40, 50]$:

- **Mean** = 30
- **Variance** = 200
- **Standard Deviation** ≈ 14.14

Arithmetic Mean (AM):

The **Arithmetic Mean** is the most common type of average. It is calculated by adding up all the values in a data set and then dividing the sum by the number of values.

Formula:

$$\text{Arithmetic Mean (AM)} = \frac{\sum x_i}{n}$$

Where:

- $\sum x_i$ is the sum of all values
- n is the number of values

Arithmetic Mean (AM):

Example: Suppose we have the data set: 5, 10, 15, 20.

- Sum = $5 + 10 + 15 + 20 = 50$
- Number of values $n = 4$

So, the Arithmetic Mean is:

$$\text{AM} = \frac{50}{4} = 12.5$$

Weighted Mean

The **Weighted Mean** is similar to the Arithmetic Mean but is used when some values contribute more than others. Each value in the data set has a weight, and these weights are used to calculate the mean.

Formula:

$$\text{Weighted Mean} = \frac{\sum w_i x_i}{\sum w_i}$$

Where:

- x_i are the data values
- w_i are the weights associated with each value
- $\sum w_i x_i$ is the sum of the products of each value and its weight
- $\sum w_i$ is the sum of the weights

Weighted Mean

Example: Let's say we have the data set: 5, 10, 15, 20 with corresponding weights: 1, 2, 3, 4.

- Sum of the weighted values: $5 \times 1 + 10 \times 2 + 15 \times 3 + 20 \times 4 = 5 + 20 + 45 + 80 = 150$
- Sum of the weights: $1 + 2 + 3 + 4 = 10$

So, the Weighted Mean is:

$$\text{Weighted Mean} = \frac{150}{10} = 15$$

Geometric Mean (GM):

The **Geometric Mean** is used when we want to calculate the average of data that involves rates, percentages, or values that are multiplied together. The Geometric Mean is found by multiplying all the values together and then taking the n-th root (where n is the number of values).

Formula:

$$\text{Geometric Mean (GM)} = \left(\prod_{i=1}^{n} x_i \right)^{1/n}$$

Where:

- x_i are the data values
- n is the number of values
- $\prod_{i=1}^{n} x_i$ is the product of all the values

Geometric Mean (GM):

Example: Let's say we have the data set: 4, 16, 64.

- Multiply the values: $4 \times 16 \times 64 = 4096$
- Take the cube root (since there are 3 values): $\sqrt[3]{4096} = 16$

So, the Geometric Mean is:

$$\text{GM} = 16$$

Harmonic Mean (HM):

The **Harmonic Mean** is used when the data consists of rates or ratios. It is calculated as the reciprocal of the Arithmetic Mean of the reciprocals of the data values.

Formula:

$$\text{Harmonic Mean (HM)} = \frac{n}{\sum \frac{1}{x_i}}$$

Where:

- x_i are the data values
- n is the number of values

Harmonic Mean (HM):

Example: Let's say we have the data set: 4, 6, 12.

- Reciprocal of each value: $\frac{1}{4}, \frac{1}{6}, \frac{1}{12}$
- Sum of reciprocals: $\frac{1}{4} + \frac{1}{6} + \frac{1}{12} = \frac{3}{12} + \frac{2}{12} + \frac{1}{12} = \frac{6}{12} = 0.5$
- Number of values $n = 3$

So, the Harmonic Mean is:

$$\text{HM} = \frac{3}{0.5} = 6$$

Variance (Measure of Spread)

Definition

Variance (σ^2) measures how far individual data points deviate from the mean of a dataset. It quantifies the spread of data.

Formula

$$\text{Variance}(\sigma^2) = \frac{\sum(X_i - \mu)^2}{N}$$

Where:

- X_i = individual values
- μ = mean (average) of values
- N = total number of values

Covariance (Measure of Relationship)

Definition

Covariance measures the relationship between two variables—whether they increase or decrease together.

Formula

For two variables X and Y, covariance is given by:

$$\text{Cov}(X, Y) = \frac{\sum(X_i - \mu_X)(Y_i - \mu_Y)}{N}$$

Where:

- X_i and Y_i are data points of two variables
- μ_X and μ_Y are means of X and Y
- N is the number of data points

Interpreting Covariance

1. **Positive Covariance (> 0):**
    - When **one variable increases, the other also increases.**
    - Example: **Hours Studied & Test Score** → More studying leads to higher scores.
2. **Negative Covariance (< 0):**
    - When **one variable increases, the other decreases.**
    - Example: **Sleep Hours & Stress Levels** → More sleep reduces stress.
3. **Zero Covariance (≈ 0):**
    - No relationship between the variables.
    - Example: **Height & Favorite Color** → No connection.

Difference Between Variance and Covariance

Feature	Variance	Covariance
Measures	Spread of one variable	Relationship between two variables
Formula	$\frac{\sum(X_i-\mu)^2}{N}$	$\frac{\sum(X_i-\mu_X)(Y_i-\mu_Y)}{N}$
Result	Always positive	Can be positive, negative, or zero
Example	Spread of exam scores	Relationship between study hours and scores

Bivariate Data

- Bivariate data involves **two variables** and is used to examine relationships between them.
- The key focus is to **see how one variable changes** in response to another.

Example of Bivariate Data

Let's consider a dataset that shows the number of hours studied and the corresponding test scores of five students.

Hours Studied (X)	Test Score (Y)
1	50
2	55
3	65
4	70
5	80

Here:

- **X (Independent Variable)**: Hours studied
- **Y (Dependent Variable)**: Test scores

Multivariate Data

- Multivariate data involves **more than two variables** and is used when analyzing multiple factors affecting an outcome.

Example of Multivariate Data

Let's consider a dataset that tracks three variables: hours studied, test scores, and sleep hours for five students.

Hours Studied (X1)	Test Score (Y)	Sleep Hours (X2)
1	50	8
2	55	7
3	65	6
4	70	5
5	80	4

Here:

- **X1 (Independent Variable 1)**: Hours studied
- **X2 (Independent Variable 2)**: Sleep hours
- **Y (Dependent Variable)**: Test scores

Key Differences

Feature	Bivariate Data	Multivariate Data
Number of Variables	2	More than 2
Example Use Case	Correlation, Simple Regression	Multiple Regression, Machine Learning Models
Example Variables	Hours studied & test scores	Hours studied, sleep hours & test scores

2.1.2.1 Bivariate Data Analysis

- Bivariate data involves two variables and is used to explore relationships between them.
- The goal is to identify patterns and causes of relationships in the data.
- Consider **Table** , which presents temperature data from a shop alongside sweater sales figures.
- To understand the relationship between temperature and sweater sales, **graphical visualization** is useful. One such method is the **scatter plot**.

Table: Temperature and Sales Data

Temperature (°C)	Sales of Sweaters (in thousands)
5	200
10	150
15	140
20	75
22	60
23	55
25	20

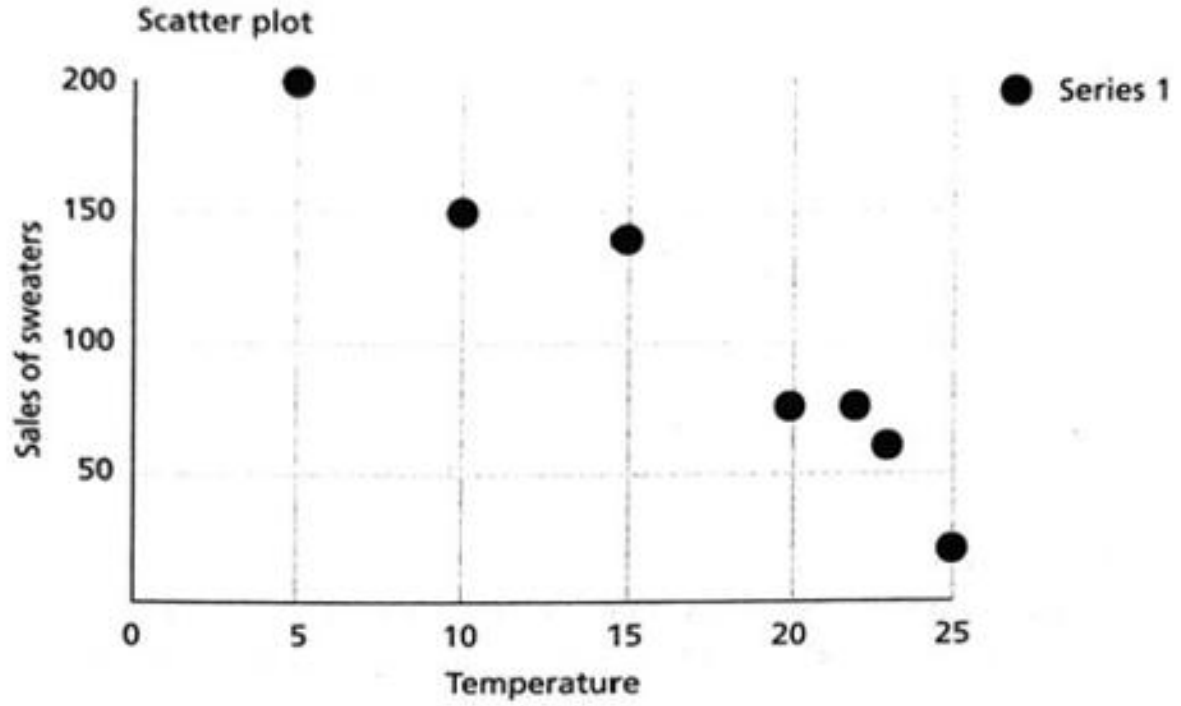
Scatter Plot and Its Importance

A scatter plot visually represents **bivariate data** by plotting two variables on a **2D graph**.

It helps in:

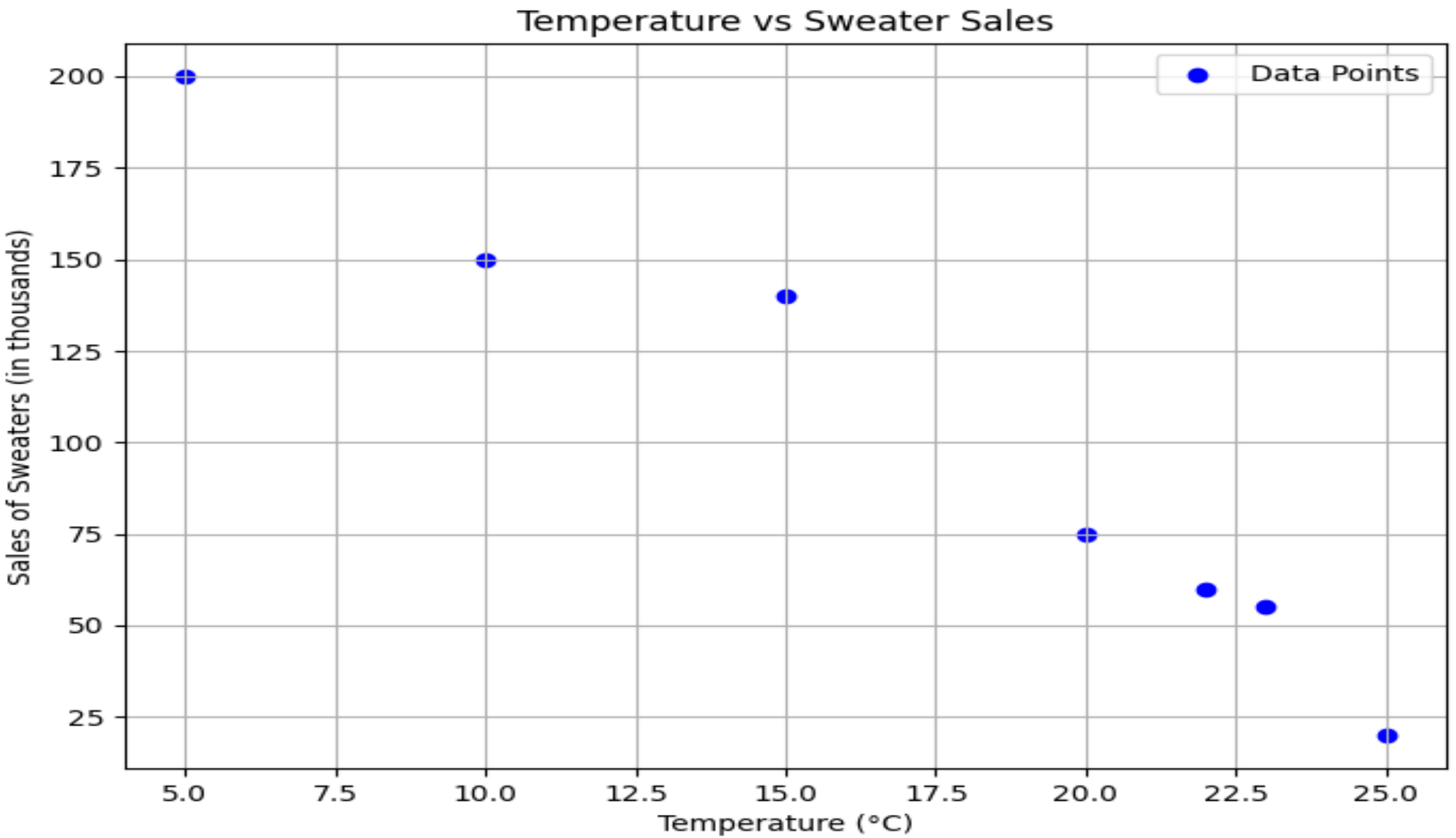
- Identifying trends or patterns.
- Observing relationships between variables.
- Detecting outliers.
- Evaluating the **strength, shape, and direction** of the relationship

Figure presents a scatter plot of temperature against sweater sales



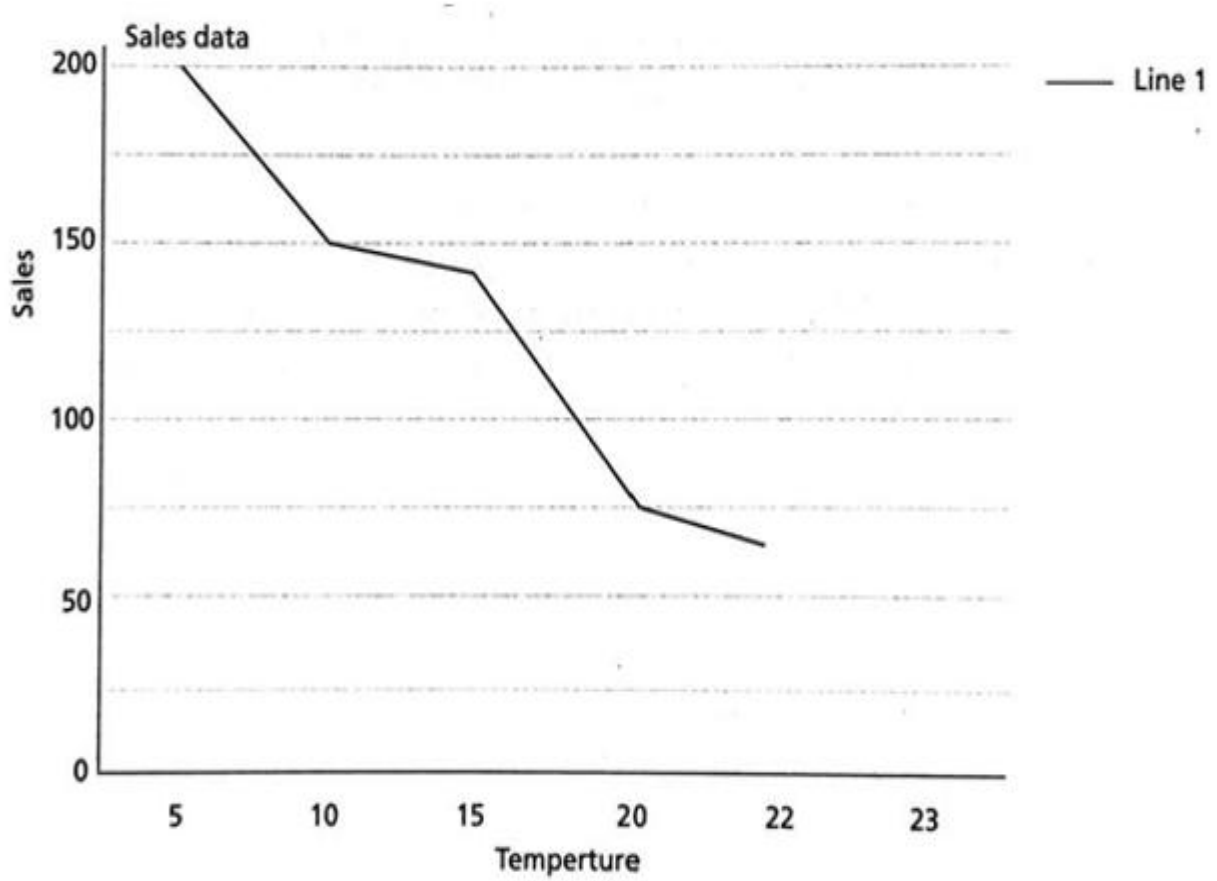
The scatter plot illustrates a **negative correlation** between temperature and sweater sales—**sales decline as temperature rises**.

```python
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# Data
temperature = np.array([5, 10, 15, 20, 22, 23, 25])
sales = np.array([200, 150, 140, 75, 60, 55, 20])

# Scatter Plot
plt.figure(figsize=(8,6))
plt.scatter(temperature, sales, color='blue', label='Data Points')
plt.xlabel('Temperature (°C)')
plt.ylabel('Sales of Sweaters (in thousands)')
plt.title('Temperature vs Sweater Sales')
plt.grid(True)
plt.legend()
plt.show()
```


Temperature vs Sweater Sales
Data Points
Sales of Sweaters (in thousands)
200
175
150
125
100
75
50
25
5.0
7.5
10.0
12.5
15.0
17.5
20.0
22.5
25.0
Temperature (°C)

Line Chart for Sales Data

- Line graphs are similar to scatter plots but **connect data points with lines,** making trends more visible.
- The graph clearly demonstrates the downward trend, confirming that sweater sales decrease as temperature increases.
- By **analyzing** these graphical representations, businesses can make data-driven decisions, such as adjusting stock based on seasonal demand



Bivariate Statistics

- **Covariance and correlation** are key concepts in bivariate statistics.
- **Covariance measures** the joint variability **of two random variables**, **such as X and Y,** which are typically represented by capital letters.
- Covariance, denoted as **COV(X,Y)**, indicates how changes in one variable correspond to changes in another.
- Since **covariance** can take any value, it is often normalized to fall within the range of **-1 to +1** using the **Pearson correlation** coefficient.

Bivariate Statistics

The formula for calculating covariance between two data sets, x and y, is:

$$\mathrm{COV}(X,Y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(X))(y_i - E(Y))$$

where:

- x_i and y_i are individual data values of X and Y,
- $E(X)$ and $E(Y)$ represent the mean values of X and Y, respectively,
- N is the total number of data points.

Example : Finding Covariance

Given two sets of data:

$$X = \{1, 2, 3, 4, 5\}, \quad Y = \{1, 4, 9, 16, 25\}$$

Solution:

Calculate the mean values:

$$E(X) = \frac{1+2+3+4+5}{5} = 3, \quad E(Y) = \frac{1+4+9+16+25}{5} = 11$$

Example : Finding Covariance

Compute covariance using the formula:

$$\frac{(1-3)(1-11)+(2-3)(4-11)+(3-3)(9-11)+(4-3)(16-11)+(5-3)(25-11)}{5}$$

$$=\frac{(1-3)(1-11)+(2-3)(4-11)+(3-3)(9-11)+(4-3)(16-11)+(5-3)(25-11)}{5}=12$$

Thus, the covariance between X and Y is **12**.

If normalization is required, dividing by the correlation of the variables gives the **Pearson correlation coefficient**. In some cases, $N-1$ is used instead of N, yielding:

$$\frac{60}{4}=15$$

This approach standardizes the covariance measure for easier interpretation.

Correlation

- The **Pearson correlation coefficient** is a widely used statistical measure for determining the relationship between two variables. It quantifies the strength and direction of a **linear relationship** between x and y.
- The **sign** of the correlation coefficient is more significant than its actual value, as it indicates the nature of the relationship between the variables:
  - A **positive** correlation means that both variables increase together.
  - A **negative** correlation indicates that as one variable increases, the other decreases.
  - A **zero** correlation suggests that the two variables are independent of each other.

Correlation

If two dimensions are highly correlated, one of them may be redundant and can be removed in certain applications.

For given datasets:

$$X = (x_1, x_2, \ldots, x_N), \quad Y = (y_1, y_2, \ldots, y_N)$$

the **Pearson correlation coefficient**, denoted as r, is calculated as:

$$r = \frac{\mathrm{COV}(X, Y)}{\sigma_X \sigma_Y}$$

where σ_X and σ_Y represent the standard deviations of X and Y, respectively.

Example: Finding the Correlation Coefficient

Given the data sets:

$$X = \{1, 2, 3, 4, 5\}, \quad Y = \{1, 4, 9, 16, 25\}$$

Solution:

The mean values of X and Y are:

$$E(X) = \frac{15}{5} = 3, \quad E(Y) = \frac{55}{5} = 11$$

Example: Finding the Correlation Coefficient

The standard deviations of X and Y are **1.41** and **8.6486**, respectively. The correlation coefficient is calculated using the formula:

$$r = \frac{\mathrm{COV}(X, Y)}{\sigma_X \sigma_Y}$$

Substituting the covariance (**12**) and standard deviations:

$$r = \frac{12}{1.41 \times 8.6486} \approx 0.984$$

2.1.3 Multivariate Statistics

- In **machine learning**, most datasets are **multivariable**, meaning they contain multiple observable variables. These datasets often involve thousands of measurements for one or more subjects.
- Multivariate data are similar to **bivariate data** but may include more than two dependent variables.
- Some common **multivariate analyses** include:
  - **Regression analysis**
  - **Principal component analysis (PCA)**
  - **Path analysis**

Regression Analysis

- Regression analysis is a **statistical method** used to model the relationship between a **dependent variable** and one or more **independent variables**.
- It helps **predict outcomes and determine** the influence of predictor variables.

Example of Regression Analysis with Dataset

- Let's consider a simple **Linear Regression** example where we predict a student's exam score based on the number of study hours.

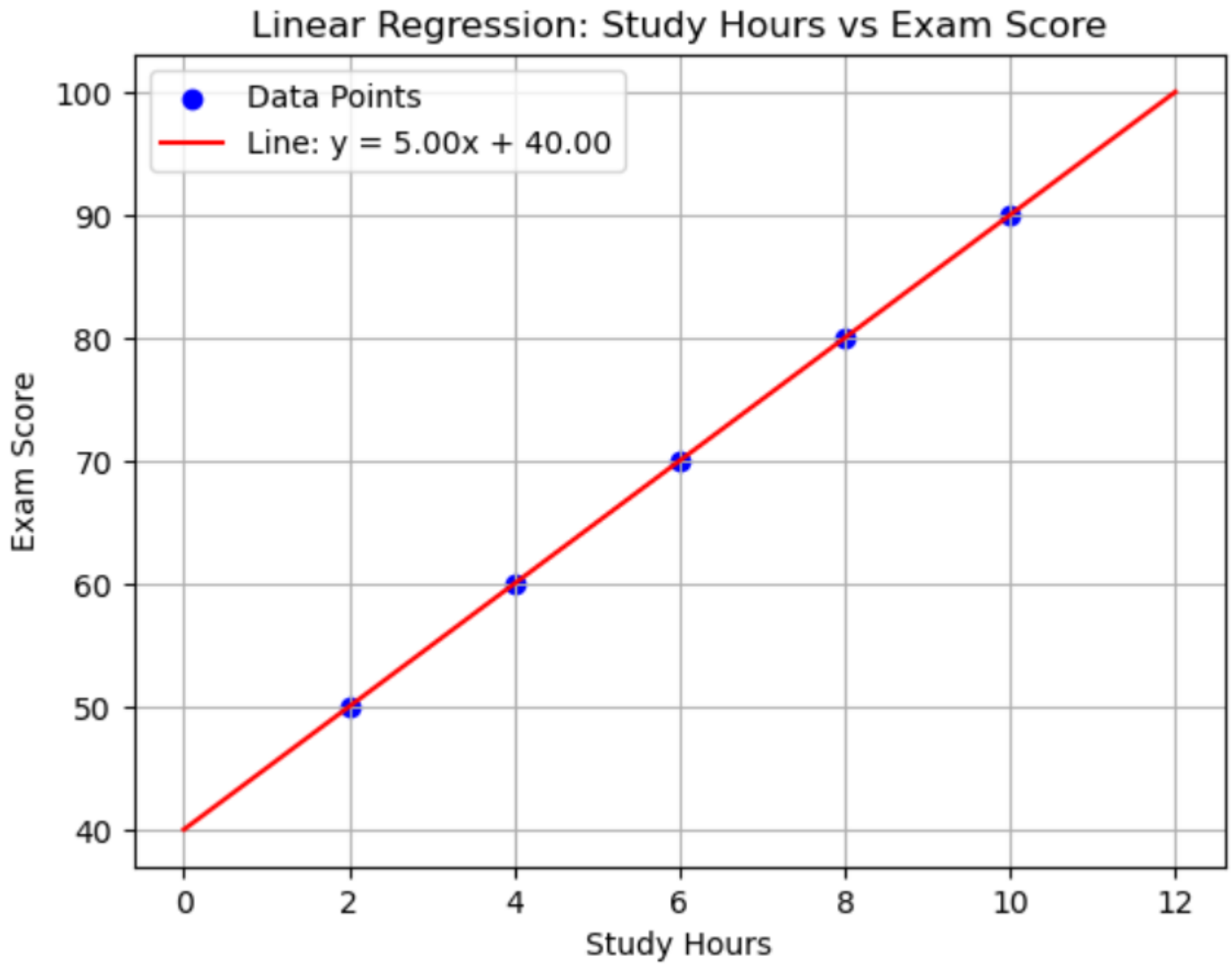
Step 1: Original Dataset (Before Applying Regression)

Student	Study Hours (X)	Exam Score (Y)
1	2	50
2	4	60
3	6	70
4	8	80
5	10	90

Here, we assume there is a **linear relationship** between the number of study hours and the exam score.

```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import linregress

# Given dataset
study_hours = np.array([2, 4, 6, 8, 10])
exam_scores = np.array([50, 60, 70, 80, 90])

# Perform linear regression
slope, intercept, r_value, p_value, std_err = linregress(study_hours, exam_scores)

# Define the regression equation
def regression_line(x):
    return slope * x + intercept
```

```
# Generate predictions
x_values = np.linspace(0, 12, 100)
y_values = regression_line(x_values)

# Plot the data points
plt.scatter(study_hours, exam_scores, color='blue', label='Data Points')

# Plot the regression line
plt.plot(x_values, y_values, color='red', label=f'Line: y = {slope:.2f}x + {intercept:.2f}')

# Labels and title
plt.xlabel('Study Hours')
plt.ylabel('Exam Score')
plt.title('Linear Regression: Study Hours vs Exam Score')
plt.legend()
plt.grid()

# Show the plot
plt.show()

# Print the equation of the regression line
print(f"Linear Regression Equation: y = {slope:.2f}x + {intercept:.2f}")
```




```
Linear Regression Equation: y = 5.00x + 40.00
```

Step 2: Applying Simple Linear Regression

The general formula for **linear regression** is:

$$Y = \beta_0 + \beta_1 X + \varepsilon$$

where:

- Y is the dependent variable (Exam Score)
- X is the independent variable (Study Hours)
- β_0 is the intercept
- β_1 is the slope (coefficient)
- ε is the error term

Using regression analysis, we get the best-fit equation:

$$Y = 5X + 40$$

where:

- Intercept (β_0) = 40
- Slope (β_1) = 5

Step 3: Predictions (After Applying Regression)

If we use this equation, we can predict exam scores for new study hours:

Study Hours (X)	Predicted Exam Score (Y)
3	$(5 \times 3) + 40 = 55$
7	$(5 \times 7) + 40 = 75$
9	$(5 \times 9) + 40 = 85$

Thus, if a student studies **7 hours**, their predicted exam score is **75**.

Step 4: Regression Interpretation

- The **slope (5)** means that for every **1 additional study hour**, the exam score increases by **5 points**.
- The **intercept (40)** means that if a student studies **0 hours**, the predicted score is **40**.

Visualization

- If we plot Study Hours (X) vs. Exam Score (Y), a **straight-line trend** appears.
- This confirms a **positive correlation** (more study leads to higher scores).

Types of Regression:

- **Linear Regression**: Relationship between dependent (Y) and independent (X) variables using a straight line ($Y = \beta_0 + \beta_1 X + \varepsilon$).
- **Multiple Regression**: Extends linear regression to multiple independent variables ($Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta n Xn + \varepsilon$).
- **Logistic Regression**: Used for binary outcomes (e.g., Yes/No, 0/1).
- **Polynomial Regression**: Models nonlinear relationships using higher-degree terms.
- **Ridge and Lasso Regression**: Regularized regression techniques to prevent overfitting.

Applications:

- Predicting sales, stock prices, or medical conditions.
- Analyzing customer behavior.
- Forecasting trends based on past data.

Principal Component Analysis (PCA)

- PCA is a **dimensionality reduction technique** that transforms **correlated variables into a smaller set of uncorrelated variables** (principal components) while retaining most of the data's variance.

Steps in PCA:

1. **Standardize the data**: Ensure all variables are on the same scale.
2. **Compute the covariance matrix**: Understand relationships among variables.
3. **Calculate eigenvalues and eigenvectors**: Identify the directions of maximum variance.
4. **Select principal components**: Choose components that explain the most variance.
5. **Transform the data**: Project the original data onto the new principal components.

Example of Principal Component Analysis (PCA) with Dataset

Step 1: Original Dataset (Before Applying PCA)

Consider a dataset with three features: **Height (cm), Weight (kg), and Age (years)** of individuals.

Person	Height (cm)	Weight (kg)	Age (years)
1	170	70	25
2	160	65	30
3	175	80	35
4	180	85	28
5	165	72	40

Here, the dataset has **three correlated variables** (Height, Weight, and Age). PCA will reduce dimensionality while preserving most of the variance.

Step 2: After Applying PCA

PCA transforms the original dataset into a new set of **Principal Components (PCs)**. These PCs are uncorrelated and capture the maximum variance in the data.

Person	PC1	PC2
1	1.2	0.4
2	0.8	-0.2
3	1.5	0.5
4	1.8	0.6
5	0.9	-0.3

- **PC1 (Principal Component 1)** captures the highest variance (e.g., a mix of Height, Weight, and Age).
- **PC2 (Principal Component 2)** captures the second highest variance.
- The third component (PC3) might have minimal variance and can be ignored, reducing the dataset from 3D to 2D.

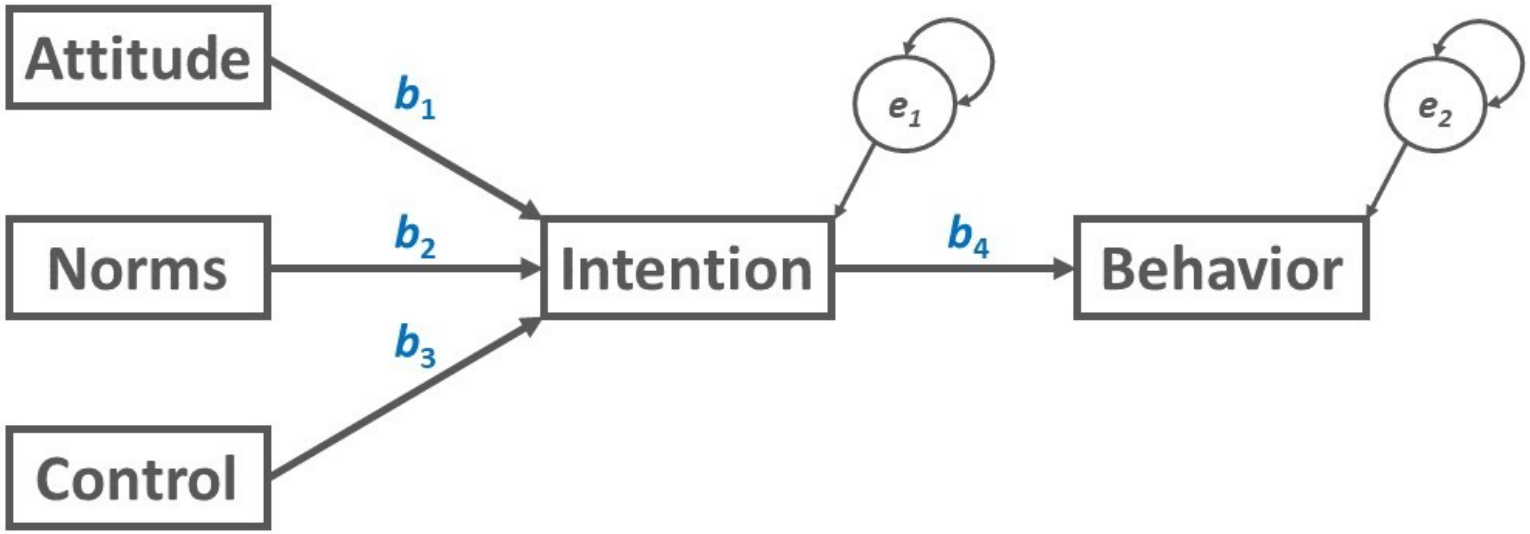
Applications:

1. **Reducing complexity** in large datasets (e.g., image processing, genetics).
2. **Identifying patterns** in high-dimensional data.
3. **Feature extraction** for machine learning models.

Path Analysis

- Path analysis is an extension of multiple regression that examines **causal relationships between variables using a path diagram**. It helps understand direct and indirect effects among variables.
- Causal Relationships: A **causal relationship** refers to a cause-and-effect connection between two or more variables, where one variable (the cause) directly influences another variable (the effect). In simple terms, if changing one variable leads to a change in another, there is a causal relationship between them.
- **Examples**: Smoking causes lung cancer, Higher interest rates reduce consumer spending, More study hours lead to better exam performance

Example: Path Diagram



Key Components:

- **Exogenous Variables**: Independent variables with no direct cause in the model.
- **Endogenous Variables**: Dependent variables influenced by other variables.
- **Path Coefficients**: Standardized regression weights indicating relationships.
- **Direct and Indirect Effects**: Effects that occur directly or via other variables.

Applications:

- Social and behavioral sciences (e.g., analyzing factors affecting student performance).
- Economics and business research (e.g., studying relationships between customer satisfaction and loyalty).
- Biological and medical research (e.g., examining disease risk factors).

Multivariate Statistics:

A sample dataset with multivariate data is structured as follows:

Id	Attribute 1	Attribute 2	Attribute 3
1	1	4	1
2	2	5	2
3	3	6	1

The **mean** of multivariate data is represented as a **mean vector**. For the dataset above, the mean values of the three attributes are: $(2.00, 5.00, 1.33)$

Multivariate Statistics:

- The **variance** of multivariate data forms the **covariance matrix**, which is a key concept in multivariate statistics. The mean vector is also known as the **centroid**, and variance is represented in a **dispersion matrix**.
- Multivariate data involve three or more variables. The purpose of **multivariate analysis** is broad and includes:
  - **Regression analysis**
  - **Factor analysis**
  - **Multivariate analysis of variance (MANOVA)**

Factor Analysis:

- Factor analysis is a technique used to reduce the complexity of data by identifying underlying relationships or factors that explain the patterns of correlations among multiple observed variables.
- The goal is to identify the "latent" variables (unobserved factors) that influence the observed variables.

Multivariate Analysis of Variance (MANOVA)

- MANOVA is an extension of Analysis of Variance (ANOVA) that allows researchers to examine the **effect of one or more independent variables on multiple dependent variables simultaneously**.
- This technique is useful when there are **multiple response variables**, and you want to see if the independent variable(s) have a joint effect on them.

Covariance Matrix (Dispersion Matrix)

- In **multivariate statistics**, variance and covariance play a crucial role in understanding the relationships between multiple variables. These concepts are represented using **the covariance matrix (dispersion matrix)**.
- The **covariance matrix** is a square matrix that contains the **variances** of each variable **along the diagonal** and the **covariances** between the variables in the **off-diagonal elements**.

For a dataset with p variables, the covariance matrix Σ is:

$$\Sigma = \begin{bmatrix} \mathrm{Var}(X_1) & \mathrm{Cov}(X_1, X_2) & \dots & \mathrm{Cov}(X_1, X_p) \\ \mathrm{Cov}(X_2, X_1) & \mathrm{Var}(X_2) & \dots & \mathrm{Cov}(X_2, X_p) \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{Cov}(X_p, X_1) & \mathrm{Cov}(X_p, X_2) & \dots & \mathrm{Var}(X_p) \end{bmatrix}$$

- **Diagonal elements** represent the variance of each variable.
- **Off-diagonal elements** represent the covariance between pairs of variables.

2. Example Dataset for Covariance Matrix

Let's consider three variables:

- X_1 (Hours Studied)
- X_2 (Test Scores)
- X_3 (Sleep Hours)

Student	Hours Studied (X_1)	Test Score (X_2)	Sleep Hours (X_3)
1	1	50	8
2	2	55	7
3	3	65	6
4	4	70	5
5	5	80	4

3. Step-by-Step Calculation of the Covariance Matrix

Step 1: Compute the Mean Vector (Centroid)

The mean of each variable is:

$$\mu_1 = \frac{1+2+3+4+5}{5} = 3$$

$$\mu_2 = \frac{50+55+65+70+80}{5} = 64$$

$$\mu_3 = \frac{8+7+6+5+4}{5} = 6$$

Thus, the **mean vector** (centroid) is:

$$\mu = \begin{bmatrix} 3 \\ 64 \\ 6 \end{bmatrix}$$

Step 2: Compute Variance and Covariance

Variance of Each Variable

Variance is calculated as:

$$\text{Var}(X) = \frac{1}{n}\sum(X_i - \mu)^2$$

- **Variance of X_1 (Hours Studied):**

$$\sigma_{11} = \frac{(1-3)^2 + (2-3)^2 + (3-3)^2 + (4-3)^2 + (5-3)^2}{5}$$

$$= \frac{4+1+0+1+4}{5} = \frac{10}{5} = 2$$

- Variance of X_2 (Test Scores):

$$\sigma_{22} = \frac{(50-64)^2 + (55-64)^2 + (65-64)^2 + (70-64)^2 + (80-64)^2}{5}$$

$$= \frac{196 + 81 + 1 + 36 + 256}{5} = \frac{570}{5} = 114$$

- Variance of X_3 (Sleep Hours):

$$\sigma_{33} = \frac{(8-6)^2 + (7-6)^2 + (6-6)^2 + (5-6)^2 + (4-6)^2}{5}$$

$$= \frac{4 + 1 + 0 + 1 + 4}{5} = \frac{10}{5} = 2$$

Covariance Between Variables

Covariance is calculated as:

$$\text{Cov}(X, Y) = \frac{1}{n} \sum (X_i - \mu_X)(Y_i - \mu_Y)$$

- **Covariance between X_1 and X_2:**

$$\sigma_{12} = \frac{(1-3)(50-64) + (2-3)(55-64) + (3-3)(65-64) + (4-3)(70-64) + (5-3)(80-64)}{5}$$

$$= \frac{(-2)(-14) + (-1)(-9) + (0)(1) + (1)(6) + (2)(16)}{5}$$

$$= \frac{28 + 9 + 0 + 6 + 32}{5} = \frac{75}{5} = 15$$

- Covariance between X_1 and X_3:

$$\sigma_{13} = \frac{(1-3)(8-6)+(2-3)(7-6)+(3-3)(6-6)+(4-3)(5-6)+(5-3)(4-6)}{5}$$

$$= \frac{(-2)(2)+(-1)(1)+(0)(0)+(1)(-1)+(2)(-2)}{5}$$

$$= \frac{-4-1+0-1-4}{5} = \frac{-10}{5} = -2$$

- Covariance between X_2 and X_3:

$$\sigma_{23} = \frac{(50-64)(8-6)+(55-64)(7-6)+(65-64)(6-6)+(70-64)(5-6)+(80-64)(4-6)}{5}$$

$$= \frac{(-14)(2)+(-9)(1)+(1)(0)+(6)(-1)+(16)(-2)}{5}$$

$$= \frac{-28-9+0-6-32}{5} = \frac{-75}{5} = -15$$

Step 3: Form the Covariance Matrix

$$\Sigma = \begin{bmatrix} 2 & 15 & -2 \\ 15 & 114 & -15 \\ -2 & -15 & 2 \end{bmatrix}$$

This **dispersion matrix (covariance matrix)** provides:

- Variances along the diagonal.
- Covariances in the off-diagonal elements, showing relationships between variables.

Heatmap

- A **heatmap** is a graphical representation of a **2D matrix**, where colors are used to represent data values.
  - **Darker colors** indicate larger values, while **lighter colors** represent smaller values.
  - The advantage of heatmaps is that they allow humans to quickly **visualize data patterns**.
- For example:
  - In **traffic analysis**, heatmaps help distinguish between **high-traffic and low-traffic** areas.
  - In **health data visualization**, a heatmap can represent the relationship between **patients' weight and health status** (e.g., X-axis: weight, Y-axis: patient count).

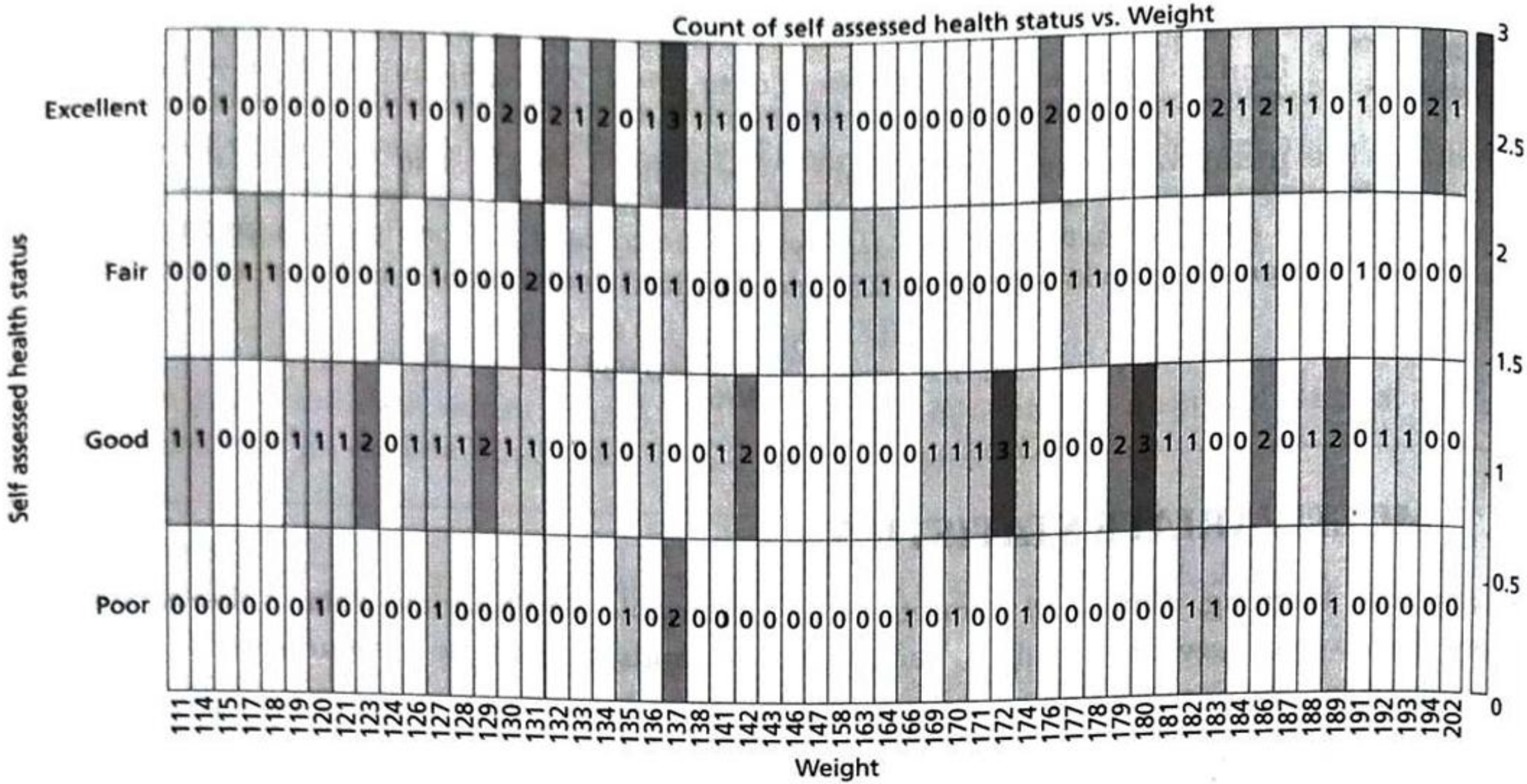
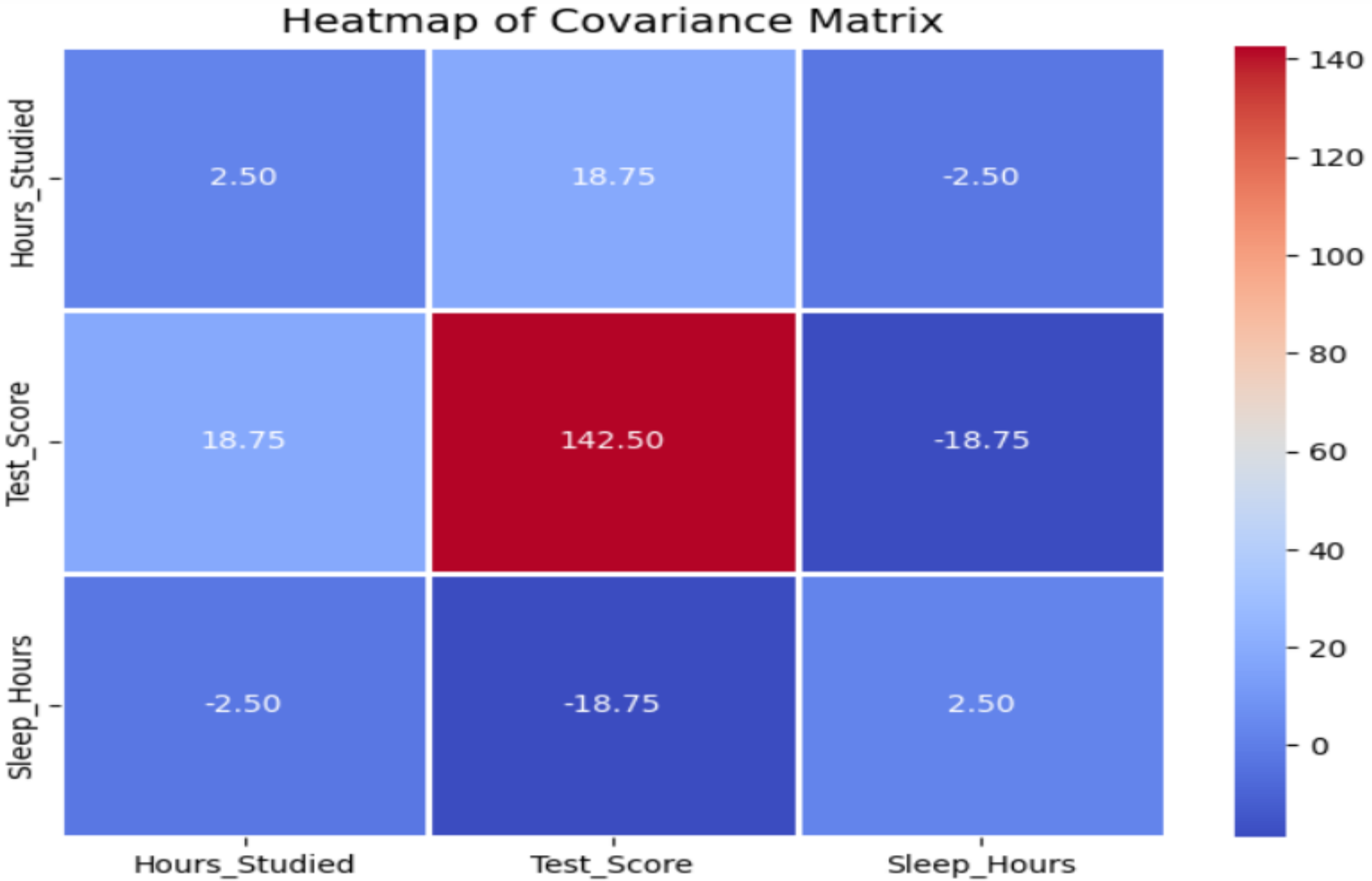


Figure 2.13: Heatmap for Patient Data

```python
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Sample dataset (Hours Studied, Test Score, Sleep Hours)
data = {
    'Hours_Studied': [1, 2, 3, 4, 5],
    'Test_Score': [50, 55, 65, 70, 80],
    'Sleep_Hours': [8, 7, 6, 5, 4]
}

# Convert to DataFrame
df = pd.DataFrame(data)

# Compute the Covariance Matrix
cov_matrix = df.cov()

# Create Heatmap
plt.figure(figsize=(8,6))
sns.heatmap(cov_matrix, annot=True, cmap='coolwarm', fmt=".2f", linewidths=2, cbar=True)

# Add Title
plt.title("Heatmap of Covariance Matrix", fontsize=14)

# Show the heatmap
plt.show()
```

Covariance Matrix

	Hours_Studied	Test_Score	Sleep_Hours
Hours_Studied	2.50	18.75	-2.50
Test_Score	18.75	142.50	-18.75
Sleep_Hours	-2.50	-18.75	2.50


Heatmap of Covariance Matrix
Hours_Studied
Test_Score
Sleep_Hours
2.50
18.75
-2.50
18.75
142.50
-18.75
-2.50
-18.75
2.50
Hours_Studied
Test_Score
Sleep_Hours
140
120
100
80
60
40
20
0

1. `sns.heatmap(cov_matrix, ...)`

- `sns.heatmap()` is a function from the **Seaborn** library used to visualize 2D matrices like **correlation matrices, covariance matrices, or confusion matrices**.
- Here, `cov_matrix` is a **Pandas DataFrame** containing the covariance values.

2. `annot=True`

- `annot=True` means **display numeric values inside the heatmap cells**.
- The values will be printed on each grid of the heatmap.

✅ **Without** `annot=True` → Only colors appear.

✅ **With** `annot=True` → Numbers are displayed inside the heatmap.

3. `cmap='coolwarm'`

- `cmap` (**colormap**) defines the **color scheme** used for the heatmap.
- `coolwarm` is a diverging color palette:
  - **Blue (cool)** → Represents lower values.
  - **Red (warm)** → Represents higher values.
  - **White** → Represents values near zero.

✅ Other colormap options: `'viridis'`, `'Blues'`, `'Greens'`, `'magma'`, `'cividis'`.

4. `fmt=".2f"`

- `fmt=".2f"` sets the format of displayed numbers inside the heatmap.
- `".2f"` means **2 decimal places**.

✅ Example:

- `12.3456` → `12.35`
- `7.8912` → `7.89`

5. `linewidths=2`

- Defines the **thickness of lines** between heatmap cells.
- **Bigger values** = More separation between grid cells.

✅ `linewidths=0` → No separation between cells.
✅ `linewidths=2` → Clear white grid lines.

6. `cbar=True`

- Adds a **color bar** (legend) to the side of the heatmap.
- The color bar helps interpret the scale of values.

✅ **With** `cbar=True` → A side color bar is displayed.

✅ **With** `cbar=False` → No color bar is shown.

Pair Plot

- A **pairplot**, also known as a **scatter matrix**, is a visual technique used to analyze **multivariate data**.
- It consists of multiple **pairwise scatter plots** that display relationships between different variables in a dataset.
- The results are arranged in a **matrix format**, making it easy to identify patterns such as **correlations** between variables.
- By examining the **pairplot**, one can quickly **observe trends, clusters, and relationships among variables**.

- In the example below, a **random matrix** with three columns is selected, and the relationships among these columns are visualized using a **pairplot (scatter matrix)**, as shown in **Figure:**

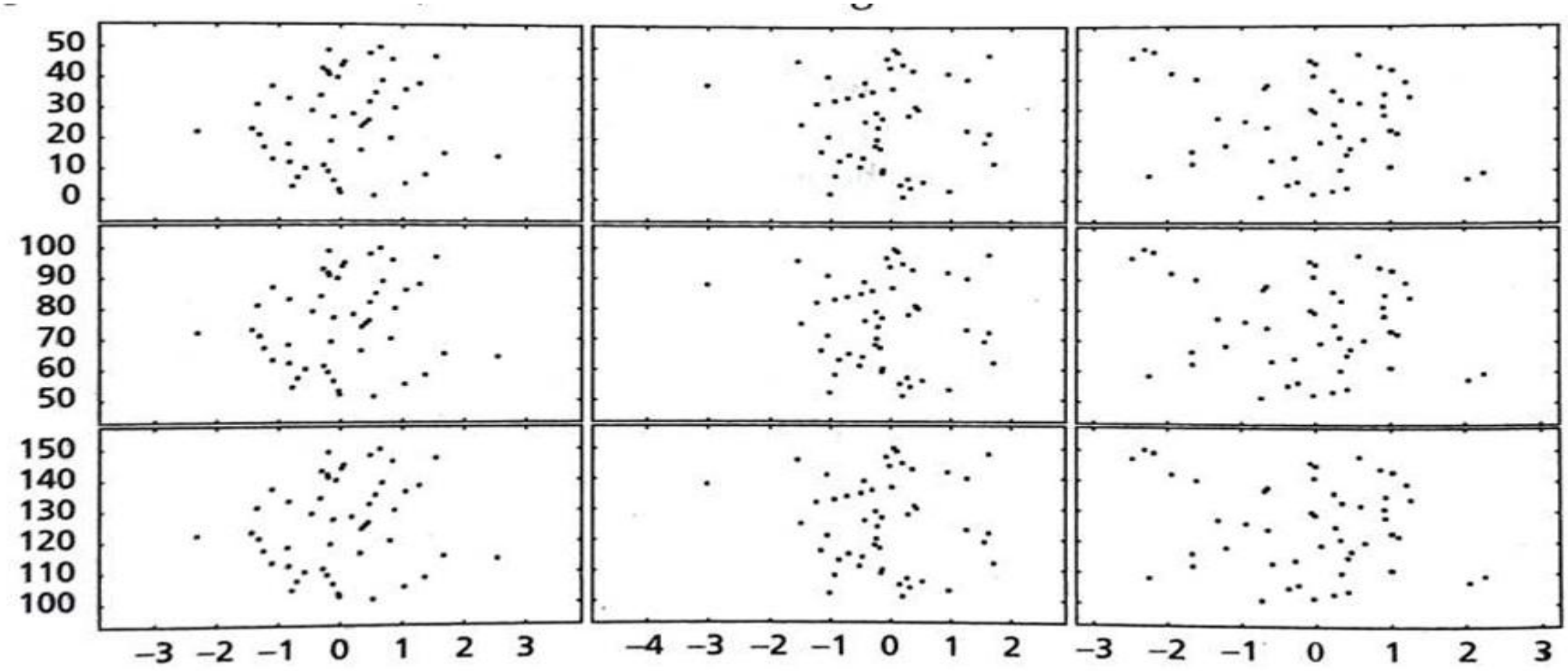


Figure 2.14: Pairplot for Random Data

2.1.3 Essential Mathematics for Multivariate Data

- Machine learning relies on several mathematical concepts, including **linear algebra, statistics, probability, and information theory**.
- This section explores key aspects of **linear algebra and probability** that are fundamental to understanding multivariate data.

Linear Algebra in Machine Learning

- **Linear algebra** is a crucial branch of mathematics widely used in scientific applications and other mathematical fields.
- While all areas of mathematics contribute to machine learning, **linear algebra plays a fundamental role** as it provides the mathematical framework for working with **linear equations, vectors, matrices, vector spaces, and transformations**.
- These structures form the **foundation of machine learning**, making it impossible to develop machine learning models without them.
- Now, let's explore some essential concepts of linear algebra.

1.Linear Systems and Gaussian Elimination for Multivariate Data

A **linear system of equations** is a set of equations with **unknown variables**.

Given a system represented as:

$$Ax = y$$

The solution for **x** is given by:

$$x = y/A = A^{-1}y$$

- This holds true if **y** is nonzero and **A** is an invertible (nonzero) matrix.

1.Linear Systems and Gaussian Elimination for Multivariate Data

For a system with **N equations** and **n unknown variables**, if **A** is represented as:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}$$

And **y** is represented as:

$$y = (y_1, y_2, \dots, y_n)$$

Then, the unknown variable **x** can be computed as:

$$x = y/A = A^{-1}y$$

Types of Solutions in a Linear System

1. If there is a **unique solution**, the system is called **consistent independent.**
2. If there are **multiple solutions**, the system is **consistent dependent**.
3. If there is **no solution** or the equations are **contradictory**, the system **is inconsistent**.

Gaussian Elimination for Solving Large Systems of Equations

- Gaussian elimination is an efficient method used to solve **large systems of equations**.
- The step-by-step procedure is as follows:

1. **Write the given matrix** representing the system of equations.
2. **Augment the matrix** by appending vector **y** to **A**, forming an **augmented matrix**.
3. **Perform row operations**:

- Use the first element a_{11} as a **pivot**.
- Eliminate all elements below a_{11} in the second row using the matrix operation:

$$R_2 \leftarrow R_2 - \left(\frac{a_{21}}{a_{11}}\right) R_1$$

Here, R_2 represents the second row, and (a_{21} / a_{11}) is the **multiplier**. The same logic applies to remove a_{11} in other rows.

4. **Reduce the matrix** to row echelon form.
5. **Solve for unknown variables**:

- The first unknown variable is found using:

$$x_n = \frac{y_m}{a_{nn}}$$

- The remaining unknowns are determined using **back-substitution**:

$$x_{n-1} = \frac{y_{n-1} - a_{n-1,n}x_n}{a_{(n-1)(n-1)}}$$

This process is known as back-substitution, and it efficiently finds solutions to systems of linear equations.

- To effectively apply the **Gaussian elimination method**, the following **row operations** are used:
  - **Swapping rows**
  - **Multiplying or dividing a row by a constant**
  - **Replacing a row by adding or subtracting a multiple of another row**
- These operations help reduce a system of equations to its **row echelon form**, as demonstrated in the following example.

Example :
Solving a System Using Gaussian Elimination

Solve the given system of equations using **Gaussian elimination**:

$$2x_1 + 4x_2 = 6$$

$$4x_1 + 3x_2 = 7$$

Solution:

1. **Convert the system into an augmented matrix:**

$$\begin{bmatrix} 2 & 4 & |6 \\ 4 & 3 & |7 \end{bmatrix}$$

2. **Transform the matrix by dividing the first row by 2:**

$$\begin{bmatrix} 1 & 2 & |3 \\ 4 & 3 & |7 \end{bmatrix}$$

3. **Eliminate the first column entry in the second row using row operations:**

$$R_2 \leftarrow R_2 - 4R_1$$

Resulting in:

$$\begin{bmatrix} 1 & 2 & |3 \\ 0 & -5 & |-5 \end{bmatrix}$$

4. **Divide the second row by -5:**

$$R_2 \leftarrow R_2/-5$$

$$\begin{bmatrix} 1 & 2 & |3 \\ 0 & 1 & |1 \end{bmatrix}$$

5. **Use back-substitution to find the values of x_1 and x_2:**

   - From the second row:

$$x_2 = 1$$

   - Substituting $x_2 = 1$ into the first equation:

$$x_1 + 2(1) = 3$$

Thus, the final solution is:

$$x_1 = 1, \quad x_2 = 1$$

Matrix Decompositions

- In many cases, it is beneficial to **decompose a matrix** into its **fundamental components** to simplify complex matrix operations. These techniques are known as **matrix factorization methods**.
- One of the most widely used methods is **Eigen decomposition**, which **reduces a matrix into its eigenvalues and eigenvectors**. This decomposition is represented as:

$$A = Q \Lambda Q^T$$

Matrix Decompositions

$$A = Q\Lambda Q^T$$

Where:

- **Q** is the matrix of **eigenvectors**
- **Λ (Lambda)** is the **diagonal matrix** of **eigenvalues**
- Q^T is the **transpose of matrix Q**

Matrix decomposition techniques are fundamental in various applications, including **machine learning, data compression, and optimization**.

LU Decomposition

One of the fundamental matrix decomposition techniques is **LU decomposition**, where a matrix **A** is expressed as the product of two matrices:

$$A = LU$$

where:

- **L** is a **lower triangular matrix**
- **U** is an **upper triangular matrix**

LU decomposition can be performed using **Gaussian elimination**, as discussed earlier. The process involves augmenting **A** with an **identity matrix**, then applying row operations to transform **A** into an upper triangular matrix while keeping track of the multipliers used, which form the **L** matrix.

Example : Finding the LU Decomposition

Given the matrix:

$$A = \begin{bmatrix} 1 & 2 & 4 \\ 3 & 3 & 2 \\ 3 & 4 & 2 \end{bmatrix}$$

Find its **LU decomposition**.

Solution: First, augment an identity matrix and apply Gaussian elimination. The steps are as shown in:

$$\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}\begin{bmatrix}1&2&4\\3&3&2\\3&4&2\end{bmatrix}$$ Initial Matrix

$$\begin{bmatrix}1&0&0\\3&1&0\\0&0&1\end{bmatrix}\begin{bmatrix}1&2&4\\0&-3&-10\\3&4&2\end{bmatrix}$$ $R_2 = R_2 - 3R_1$

$$\begin{bmatrix}1&0&0\\3&1&0\\3&0&1\end{bmatrix}\begin{bmatrix}1&2&4\\0&-3&-10\\0&-2&-10\end{bmatrix}$$ $R_3 = R_3 - 3R_1$

$$\begin{bmatrix}1&0&0\\3&1&0\\3&\frac{2}{3}&1\end{bmatrix}\begin{bmatrix}1&2&4\\0&-3&-10\\0&0&\frac{-10}{3}\end{bmatrix}$$ $R_3 = R_3 - \frac{2}{3}R_2$

Now, it can be observed that the first matrix is L as it is the lower triangular matrix whose values are the determiners used in the reduction of equations above such as 3, 3 and 2/3. The second matrix is U, the upper triangular matrix whose values are the values of the reduced matrix because of Gaussian elimination.

$$L = \begin{pmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 3 & \frac{2}{3} & 1 \end{pmatrix} \text{ and } U = \begin{pmatrix} 1 & 2 & 4 \\ 0 & -3 & -10 \\ 0 & 0 & -\frac{10}{3} \end{pmatrix}.$$

Verification: Compute LU and Check if $LU = A$

$$L \times U = \begin{bmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 3 & \frac{2}{3} & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & 4 \\ 0 & -3 & -10 \\ 0 & 0 & -\frac{10}{3} \end{bmatrix}$$

Performing matrix multiplication:

$$\begin{bmatrix} (1 \cdot 1 + 0 \cdot 0 + 0 \cdot 0) & (1 \cdot 2 + 0 \cdot (-3) + 0 \cdot 0) & (1 \cdot 4 + 0 \cdot (-10) + 0 \cdot (-10/3)) \\ (3 \cdot 1 + 1 \cdot 0 + 0 \cdot 0) & (3 \cdot 2 + 1 \cdot (-3) + 0 \cdot 0) & (3 \cdot 4 + 1 \cdot (-10) + 0 \cdot (-10/3)) \\ (3 \cdot 1 + \frac{2}{3} \cdot 0 + 1 \cdot 0) & (3 \cdot 2 + \frac{2}{3} \cdot (-3) + 1 \cdot 0) & (3 \cdot 4 + \frac{2}{3} \cdot (-10) + 1 \cdot (-10/3)) \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 2 & 4 \\ 3 & 3 & 2 \\ 3 & 4 & 2 \end{bmatrix}$$

Since **LU = A**, the decomposition is verified as correct. ✅

Conclusion

- Through **Gaussian elimination**, we decomposed matrix **A** into **L** and **U**, where **L** contains the multipliers used in elimination, and **U** is the transformed upper triangular matrix.
- **LU** yield original Matrix A
- **LU d**ecomposition is widely used in solving linear systems, inverting matrices, and numerical computations.

Feature Engineering and Dimensionality Reduction Techniques

- Feature engineering involves selecting important attributes (features) that enhance model performance in machine learning. It includes two main tasks:
- **Feature Transformation** – Creating new features from existing ones to improve performance (e.g., height and weight forming Body Mass Index - BMI).
- **Feature Selection** – Choosing the most relevant features while minimizing dataset size without compromising reliability.

Feature Removal and Selection Techniques

Feature removal is based on two key aspects:

- **Feature Relevancy** – Some features contribute more to classification than others. Relevant features are determined using statistical measures like **mutual information, correlation coefficients, and distance measures**.
- **Feature Redundancy** – Redundant features provide duplicate information. For example, if a dataset includes "Date of Birth," the "Age" field becomes unnecessary since it can be derived, reducing dimensionality.

Feature Selection Techniques

1. Stepwise Forward Selection
2. **Stepwise Backward Elimination**
3. Combined Approach
4. **Principal Component Analysis (PCA)**

1. Stepwise Forward Selection

- Starts with an **empty** set of attributes.
- Iteratively adds attributes that **improve statistical significance**.
- Process continues until an optimal subset of features is selected.

2. Stepwise Backward Elimination

1. Starts with a **full** set of attributes.
2. Iteratively removes the **least significant** attribute.
3. Process continues until an optimal subset is reached.

3. Combined Approach

1. Uses both **forward selection** and **backward elimination** together.
2. Adds the best attribute while removing the worst attribute at each step.

4. Principal Component Analysis (PCA)

1. Transforms the dataset into a new set of **compact** and **informative** features.
2. Reduces dimensionality by **eliminating redundant information**.
3. Ensures new features contain **maximum variance** from the original data.

$$m_x = \frac{1}{M}\sum_{k=1}^{M} x_k \qquad (2.53)$$

$$A = \frac{1}{M}\sum_{k=1}^{M} x_k x_k^T - m_x m_x^T \qquad (2.54)$$

Example 2.12: Let the data points be $\begin{pmatrix}2\\6\end{pmatrix}$ and $\begin{pmatrix}1\\7\end{pmatrix}$. Apply PCA and find the transformed data.

Again, apply the inverse and prove that PCA works.

Solution: One can combine two vectors into a matrix as follows:

The mean vector can be computed as Eq. (2.53) as follows:

$$\mu = \begin{pmatrix}\frac{2+1}{2}\\ \frac{6+7}{2}\end{pmatrix} = \begin{pmatrix}1.5\\6.5\end{pmatrix}$$

$$m_x = \frac{1}{M}\sum_{k=1}^{M} x_k$$

As part of PCA, the mean must be subtracted from the data to get the adjusted data:

$$x_1 = \begin{pmatrix} 2 - 1.5 \\ 6 - 6.5 \end{pmatrix} = \begin{pmatrix} 0.5 \\ -0.5 \end{pmatrix}$$

$$x_2 = \begin{pmatrix} 1 - 1.5 \\ 7 - 6.5 \end{pmatrix} = \begin{pmatrix} -0.5 \\ 0.5 \end{pmatrix}$$

x – m = [x1,x2] = $\begin{pmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{pmatrix}$

One can find the covariance for these data vectors. The covariance can be obtained using Eq. (2.54):

$$m_1 = \begin{pmatrix} 0.5 \\ -0.5 \end{pmatrix} \begin{pmatrix} 0.5 & -0.5 \end{pmatrix} = \begin{pmatrix} 0.25 & -0.25 \\ -0.25 & 0.25 \end{pmatrix}$$

$$m_2 = \begin{pmatrix} -0.5 \\ 0.5 \end{pmatrix} \begin{pmatrix} -0.5 & 0.5 \end{pmatrix} = \begin{pmatrix} 0.25 & -0.25 \\ -0.25 & 0.25 \end{pmatrix}$$

$$A = \frac{1}{M} \sum_{k=1}^{M} x_k x_k^T - m_x m_x^T$$

The final covariance matrix is obtained by adding these two matrices as:

$$C = \begin{pmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{pmatrix}$$

The eigen values and eigen vectors of matrix C can be obtained (left as an exercise) as $\lambda_1 = 1$, $\lambda_1 = 0$. The eigen vectors are $\begin{pmatrix} -1 \\ 1 \end{pmatrix}$ and $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$. The matrix A can be obtained by packing the eigen vector of these eigen values (after sorting it) of matrix C. For this problem, $A = \begin{pmatrix} -1 & 1 \\ 1 & 1 \end{pmatrix}$. The transpose of A, $A^T = \begin{pmatrix} -1 & 1 \\ 1 & 1 \end{pmatrix}$ is also the same matrix as it is an orthogonal matrix. The matrix can be normalized by diving each elements of the vector, by the norm of the vector to get:

$$A = \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}$$

One can check that the PCA matrix A is orthogonal. A matrix is orthogonal is $A^{-1} = A$ and $AA^{-1} = I$.

$$AA^T = \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

The transformed matrix y using Eq. (2.55) is given as:

$$y = A \times (x - m)$$

Recollect that $(x-m)$ is the adjusted matrix.

$$y = A(x-m) = \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{pmatrix}$$

$$= \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{pmatrix} \left(\textit{for convenience } 0.5 = \frac{1}{2} \right)$$

$$= \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & 0 \end{pmatrix}$$

One can check the original matrix can be retrieved from this matrix as:

$$\{(A)^T \times y\} + m$$

$$x = A^T y + m = \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 1.5 \\ 6.5 \end{pmatrix}$$

$$= \begin{pmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{pmatrix} + \begin{pmatrix} 1.5 \\ 6.5 \end{pmatrix} = \begin{pmatrix} 2 & 1 \\ 6 & 7 \end{pmatrix}$$

Therefore, one can infer the original is obtained without any loss of information.

Determining Eigen Values and Eigen Vectors.

The given matrix is:

$$C = \begin{bmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{bmatrix}$$

Step 1: Find the Eigenvalues

The eigenvalues λ satisfy the characteristic equation:

$$\det(C - \lambda I) = 0$$

where I is the identity matrix:

$$C - \lambda I = \begin{bmatrix} 0.5 - \lambda & -0.5 \\ -0.5 & 0.5 - \lambda \end{bmatrix}$$

Taking the determinant:

$$\begin{vmatrix} 0.5-\lambda & -0.5 \\ -0.5 & 0.5-\lambda \end{vmatrix} = (0.5-\lambda)(0.5-\lambda) - (-0.5)(-0.5)$$

$$= (0.5-\lambda)^2 - 0.25$$

$$= 0.25 - \lambda + \lambda^2 - 0.25$$

$$= \lambda^2 - \lambda$$

Setting the determinant to zero:

$$\lambda^2 - \lambda = 0$$

$$\lambda(\lambda - 1) = 0$$

Thus, the eigenvalues are:

$$\lambda_1 = 0, \quad \lambda_2 = 1$$

Principal Component Analysis (PCA)

Concept of PCA

- PCA is used to transform a dataset into a **lower-dimensional representation** while preserving the most **important variance**.
- It relies on the **mean vector** and **covariance matrix** to determine principal components.

Principal Component Analysis (PCA) and Transformation of Random Vectors

Consider a set of random vectors represented as:

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

The **mean vector** of these random vectors is defined as:

$$m_x = E[x]$$

where E represents the expected value of the population, computed using probability density functions (PDFs) of the elements of x and joint PDFs between elements x_i and x_j. The **covariance matrix** is given by:

$$C = E[(x - m)(x - m)^T]$$

For a large enough number M of random vectors, the mean vector and covariance matrix can be approximated as:

$$m_x = \frac{1}{M}\sum_{i=1}^{M} x_i$$

$$A = \frac{1}{M}\sum_{i=1}^{M}(x_i - m_x)(x_i - m_x)^T$$

This covariance matrix is **real and symmetric**, making it possible to compute its **eigenvalues** and **eigenvectors**. The eigenvalues (λ_i) are arranged in descending order ($\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$), and the corresponding eigenvectors are calculated to construct a **transform kernel (A)**. The **transform** of the original data is then performed using:

$$y = A(x - m)$$

This transformation is also known as **Karhunen-Loève (KL) transform** or **Hotelling transform**. The original data can be reconstructed as:

$$x = A^T y + m$$

Goal and Algorithm of PCA

PCA reduces the set of attributes into a **smaller, meaningful subset** that captures most of the data variance. Instead of using all eigenvectors of the covariance matrix, PCA selects only a small subset with the highest variance, optimizing information compression. If **K largest eigenvalues** are used, the **recovered information** is given by:

$$x = A_K^T y + m$$

where A_K is the matrix containing the selected eigenvectors.

Steps in the PCA Algorithm:

1. Compute the **mean** vector (m).
2. Subtract the mean from the dataset to obtain a dataset centered at zero.
3. Compute the **covariance matrix** (C).
4. Calculate **eigenvalues** and **eigenvectors** of C.
5. Select the eigenvectors corresponding to the **largest eigenvalues** (principal components).
6. Form a **feature vector matrix** from these selected eigenvectors.
7. Apply the **PCA transformation** using:

$$y = A(x - m)$$

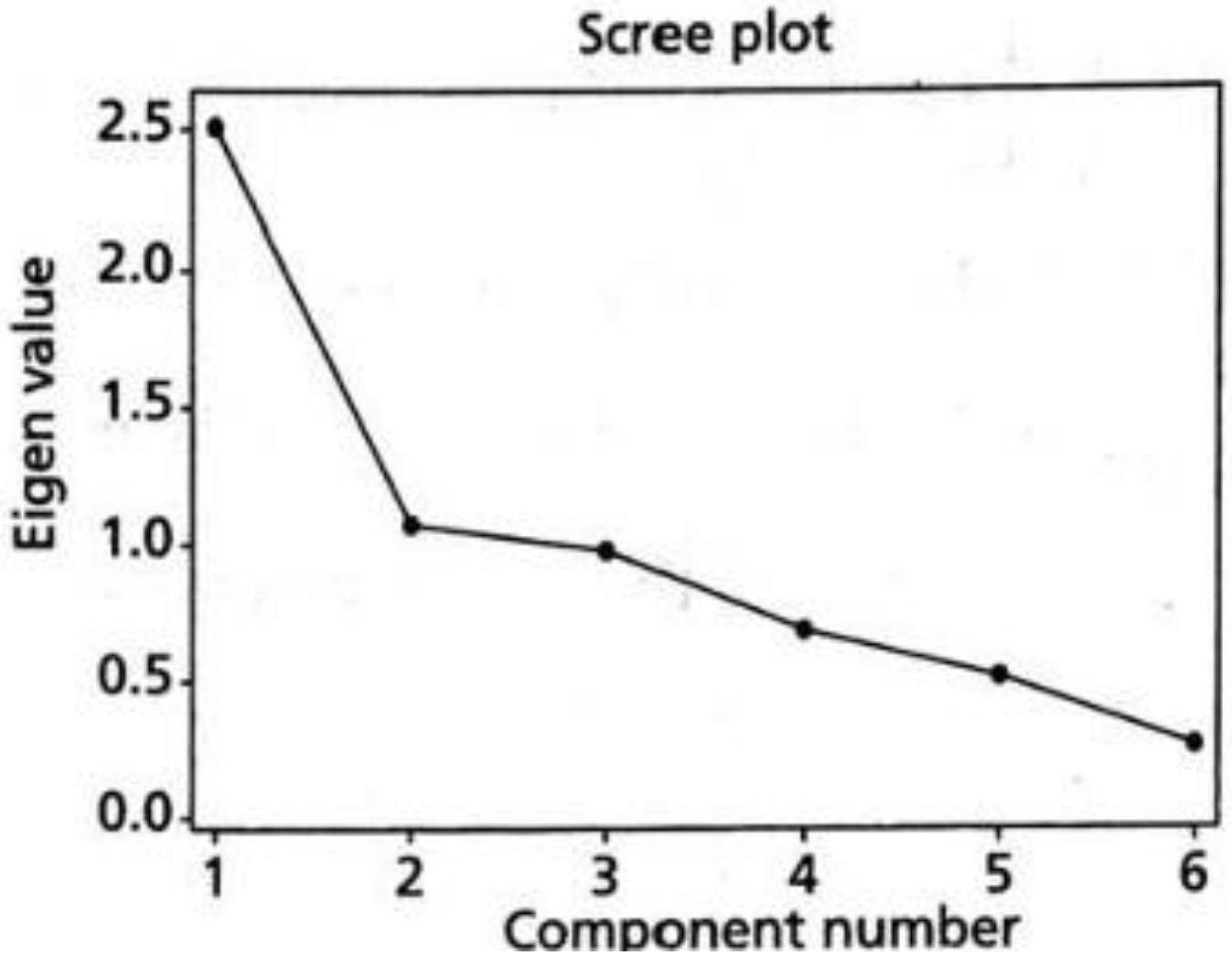
where A is the transpose of the feature vector matrix.

The **original data** can be reconstructed as:

$$x = A^T y + m$$

Importance of PCA and Scree Plot

- PCA effectively eliminates irrelevant attributes while preserving data structure. If required, the original data can be reconstructed **without loss**.
- A **Scree Plot** is a visualization technique used to identify **important components**. It displays **eigenvalues** against their corresponding **principal components** to determine which components contribute most to variance.
- From the example **scree plot** (Figure), it is evident that **only 6 out of 246 attributes are significant**, with the first attribute being the most important.


Scree plot
Eigen value
2.5
2.0
1.5
1.0
0.5
0.0
1
2
3
4
5
6
Component number

Example 2.12: Let the data points be $\begin{pmatrix}2\\6\end{pmatrix}$ and $\begin{pmatrix}1\\7\end{pmatrix}$. Apply PCA and find the transformed data.

Again, apply the inverse and prove that PCA works.

Solution: One can combine two vectors into a matrix as follows:

The mean vector can be computed as Eq. (2.53) as follows:

$$\mu = \begin{pmatrix}\frac{2+1}{2}\\ \frac{6+7}{2}\end{pmatrix} = \begin{pmatrix}1.5\\6.5\end{pmatrix}$$

As part of PCA, the mean must be subtracted from the data to get the adjusted data:

$$x_1 = \begin{pmatrix} 2 - 1.5 \\ 6 - 6.5 \end{pmatrix} = \begin{pmatrix} 0.5 \\ -0.5 \end{pmatrix}$$

$$x_2 = \begin{pmatrix} 1 - 1.5 \\ 7 - 6.5 \end{pmatrix} = \begin{pmatrix} -0.5 \\ 0.5 \end{pmatrix}$$

One can find the covariance for these data vectors. The covariance can be obtained using Eq. (2.54):

$$m_1 = \begin{pmatrix} 0.5 \\ -0.5 \end{pmatrix} \begin{pmatrix} 0.5 & -0.5 \end{pmatrix} = \begin{pmatrix} 0.25 & -0.25 \\ -0.25 & 0.25 \end{pmatrix}$$

$$m_2 = \begin{pmatrix} -0.5 \\ 0.5 \end{pmatrix} \begin{pmatrix} -0.5 & 0.5 \end{pmatrix} = \begin{pmatrix} 0.25 & -0.25 \\ -0.25 & 0.25 \end{pmatrix}$$

The final covariance matrix is obtained by adding these two matrices as:

$$C = \begin{pmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{pmatrix}$$

$$m_x = \frac{1}{M} \sum_{i=1}^{M} x_i$$

$$A = \frac{1}{M} \sum_{i=1}^{M} (x_i - m_x)(x_i - m_x)^T$$

The eigen values and eigen vectors of matrix C can be obtained (left as an exercise) as $\lambda_1 = 1$, $\lambda_1 = 0$. The eigen vectors are $\begin{pmatrix} -1 \\ 1 \end{pmatrix}$ and $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$. The matrix A can be obtained by packing the eigen vector of these eigen values (after sorting it) of matrix C. For this problem, $A = \begin{pmatrix} -1 & 1 \\ 1 & 1 \end{pmatrix}$. The transpose of A, $A^T = \begin{pmatrix} -1 & 1 \\ 1 & 1 \end{pmatrix}$ is also the same matrix as it is an orthogonal matrix. The matrix can be normalized by diving each elements of the vector, by the norm of the vector to get:

$$A = \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}$$

One can check that the PCA matrix A is orthogonal. A matrix is orthogonal is $A^{-1} = A$ and $AA^{-1} = I$.

$$AA^T = \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

The transformed matrix y using Eq. (2.55) is given as:

$$y = A \times (x - m)$$

Recollect that $(x-m)$ is the adjusted matrix.

$$y = A(x-m) = \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{pmatrix}$$

$$= \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{pmatrix} \left(\textit{for convenience } 0.5 = \frac{1}{2} \right)$$

$$= \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & 0 \end{pmatrix}$$

One can check the original matrix can be retrieved from this matrix as:

$$\{(A)^T \times y\} + m$$

$$x = A^T y + m = \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 1.5 \\ 6.5 \end{pmatrix}$$

$$= \begin{pmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{pmatrix} + \begin{pmatrix} 1.5 \\ 6.5 \end{pmatrix} = \begin{pmatrix} 2 & 1 \\ 6 & 7 \end{pmatrix}$$

Therefore, one can infer the original is obtained without any loss of information.

Machine Learning and the Importance of Probability and Statistics

- Machine learning is deeply connected to **statistics and probability**. **Statistics** is the heart of data analysis and it is used for , **understanding and interpreting** data.
- **Probability** plays a crucial role in machine learning as well. Any dataset can be assumed to be generated using appropriate single or multiple **probability distributions.**

The Role of Probability and Statistics in Machine Learning

- **Hypothesis testing**
- **Model building** and **Model Evaluation**
- **Sampling theory** is crucial for **creating datasets** and **ensuring robust model** performance.

Probability Distributions and its Types

- A **probability distribution** describes the likelihood of different outcomes for a given **random variable** (e.g., X).

Probability distributions are classified into **two main types**:

1. **Continuous Probability Distribution**
2. **Discrete Probability Distribution**

PDF and CDF

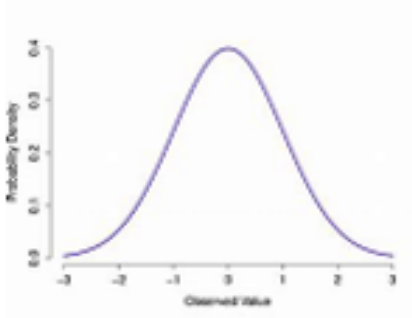
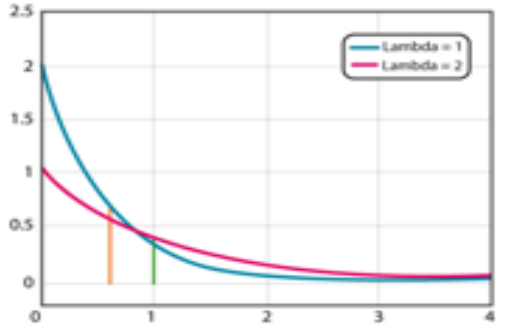
Continuous Probability Distribution is characterized by:

- **Probability Density Function (PDF)**: Determines the probability of a **specific outcome** occurring.
- **Cumulative Distribution Function (CDF)**: Computes the probability that a variable takes on a **value less than or equal to a given point**.

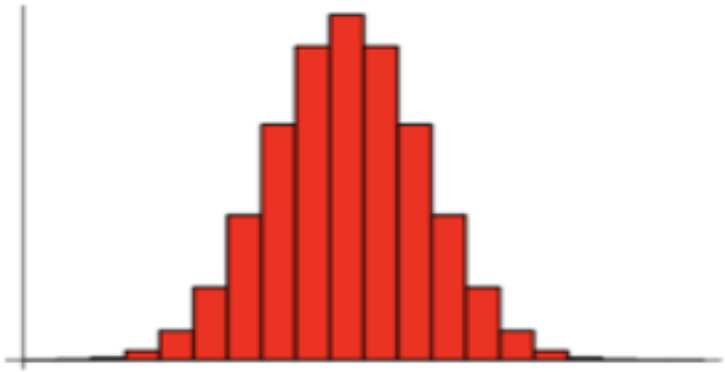
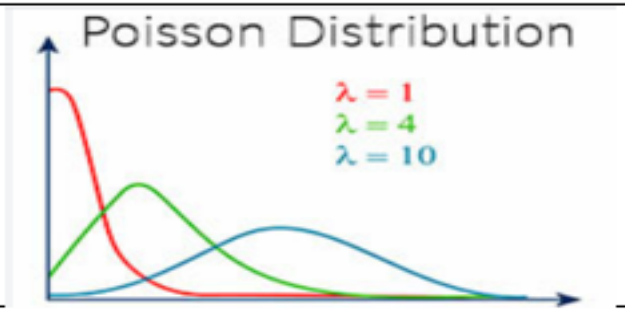
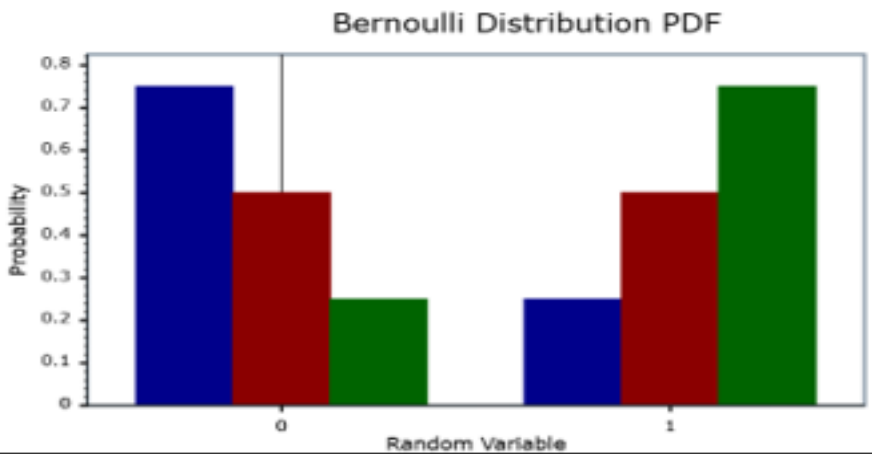
Common Continuous Probability Distributions in Machine Learning

1. **Normal Distribution** (Gaussian)
2. **Rectangular Distribution** (Uniform)
3. **Exponential Distribution**

Common Continuous Probability Distributions

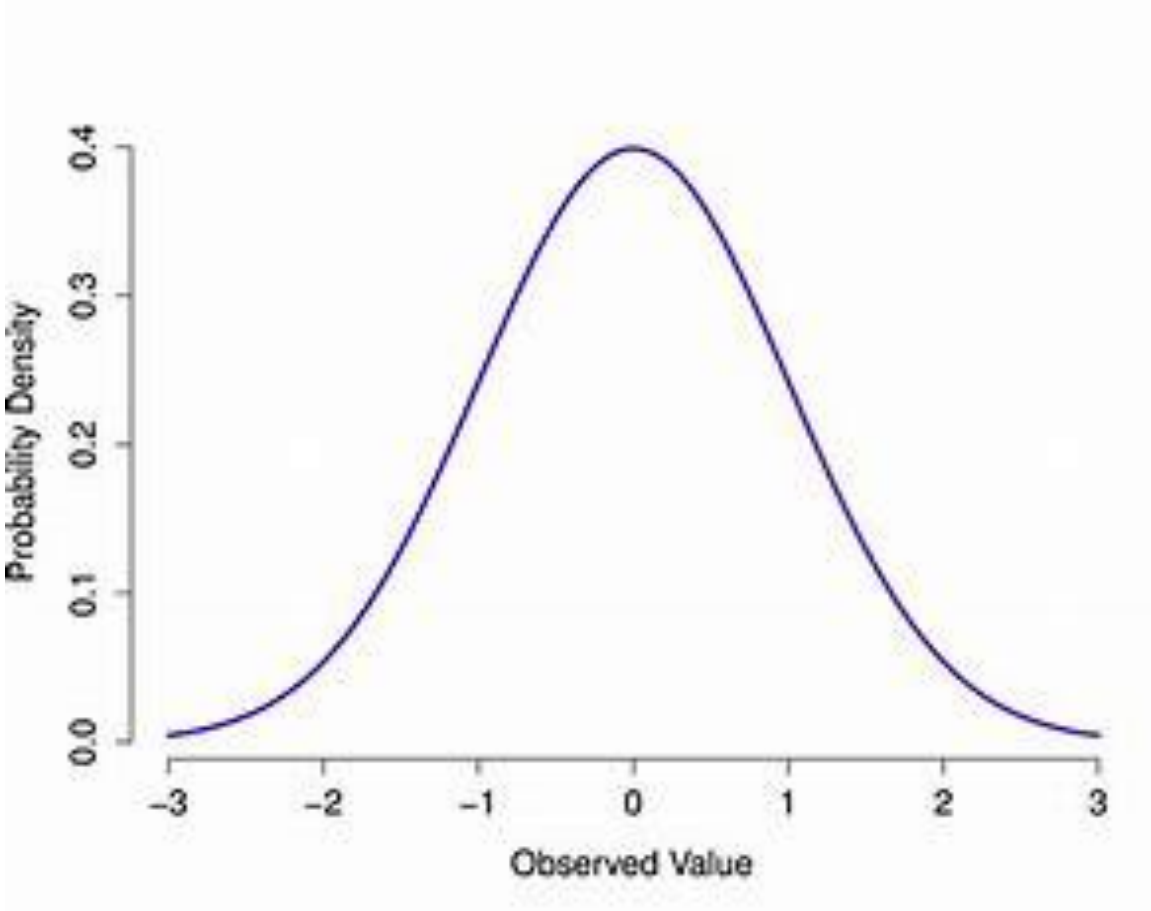
Type	PDF	
Normal Distribution (Gaussian)	$f(x, \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$	
Rectangular Distribution (Uniform)	$P(X = x) = \begin{cases} \frac{1}{b-a}, & \text{for } a \leq x \leq b \\ 0, & \text{otherwise} \end{cases}$	
Exponential Distribution	$f(x, \lambda) = \begin{cases} \lambda e^{-\lambda x}, & x \geq 0, \lambda > 0 \\ 0, & x < 0 \end{cases}$	

Common Discrete Probability Distributions

Type	PDF	
Binomial	$P(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$	
Poisson	$P(X = x) = \frac{e^{-\lambda}\lambda^x}{x!}$	
Bernoulli	$P(X = k) = \begin{cases} q = 1-p, & \text{if } k = 0 \\ p, & \text{if } k = 1 \end{cases}$	

1. Normal Distribution

- The **normal distribution**, also known as the **Gaussian distribution** or **bell-shaped curve**, is a widely used **continuous probability distribution**.
- Many real-world phenomena, such as **heights of individuals, blood pressure, and exam scores**, follow a normal distribution.



1. Normal Distribution

The **Probability Density Function (PDF)** of a normal distribution is given by:

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

where:

- μ is the **mean** (central value).
- σ is the **standard deviation**, representing the spread of data.
- σ^2 is the **variance**.

The normal distribution is often standardized to have $\mu = 0$ and $\sigma = 1$, which simplifies calculations and comparisons. In this case, **mean, median, and mode** are the same.

Z-Score and Normalization

Z-Score and Normalization

A key concept related to the normal distribution is the **z-score**, which helps standardize values for comparison. It is calculated as:

$$z = \frac{x - \mu}{\sigma}$$

- When $\mu = 0$ and $\sigma = 1$, the z-score is simply **x**.
- The z-score helps **normalize** data for analysis.

Checking for Normality

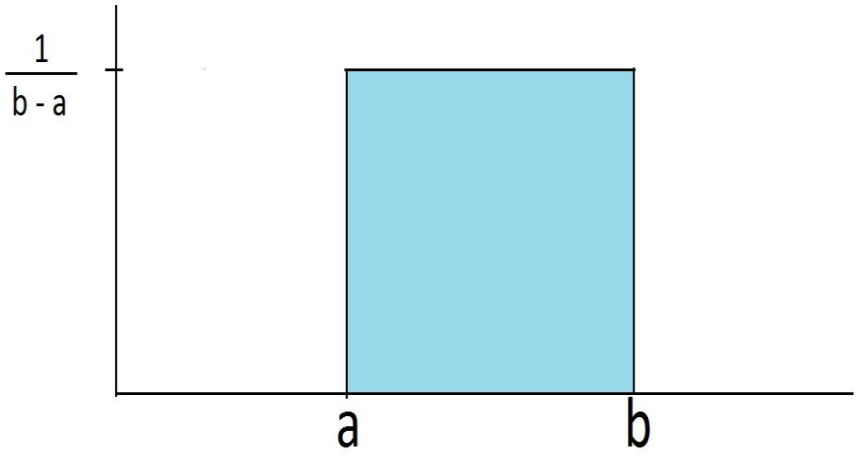
- Many statistical tests assume that data follows a normal distribution. To verify this, **normality tests** such as the **Q-Q plot** can be used.
- In a Q-Q plot, if data follows a normal distribution, the plot will align closely with a straight diagonal line.

2. Rectangular Distribution (Uniform Distribution)

The **rectangular distribution**, also known as the **uniform distribution**, is characterized by **equal probabilities** for all values within a given range $[a, b]$.

The **Probability Distribution Function (PDF)** is:

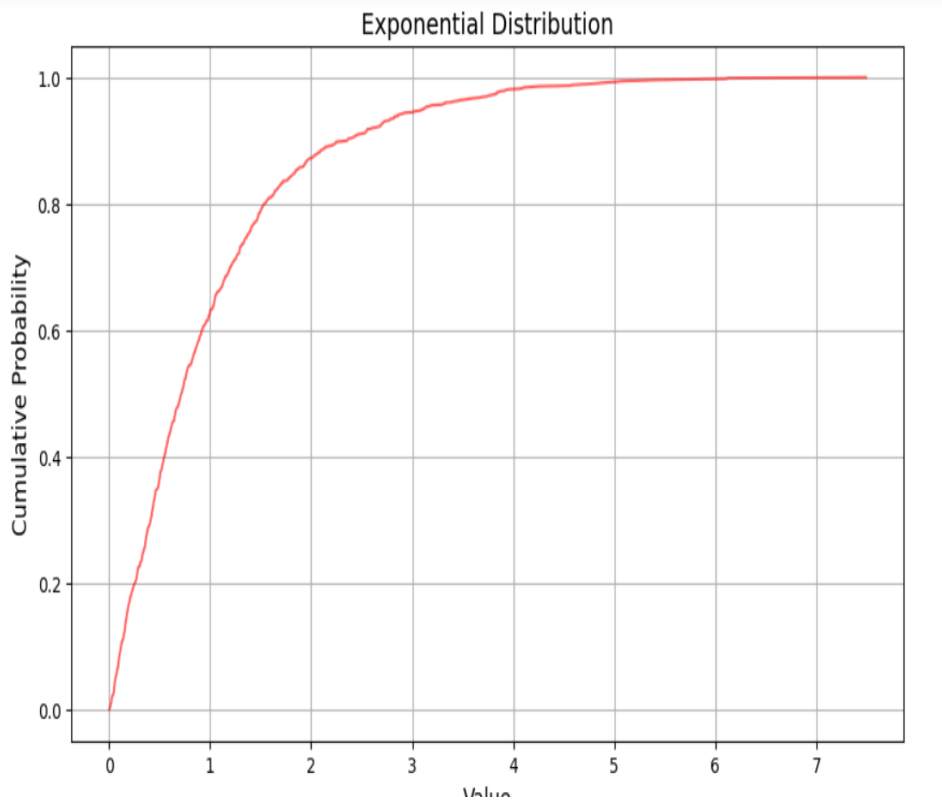
$$P(X = x) = \begin{cases} \frac{1}{b-a}, & \text{for } a \leq x \leq b \\ 0, & \text{otherwise} \end{cases}$$



This distribution is commonly used when all outcomes in a given interval are equally likely.

3. Exponential Distribution

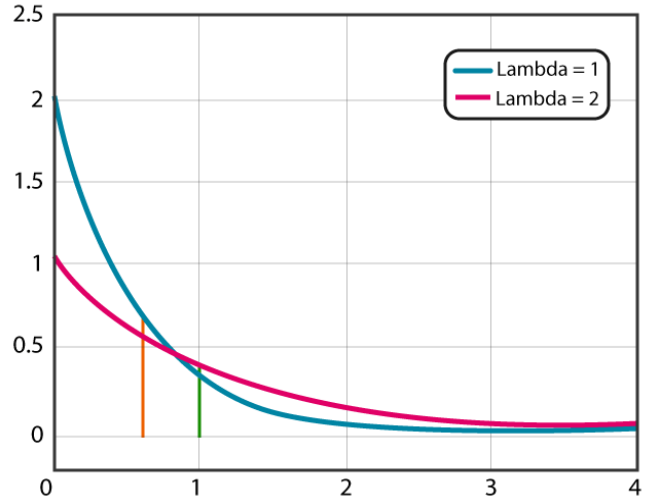
- The **exponential distribution** is a **continuous probability distribution** used to model the time between events in a **Poisson process** and special case of the **Gamma distribution** with a shape parameter of 1.
- This distribution is widely applied in fields such as **queueing theory, reliability analysis, and survival modeling**.
- The exponential distribution is particularly useful in modeling **waiting times** or **time until an event occurs**.



3. Exponential Distribution

The **Probability Density Function (PDF)** is:

$$f(x,\lambda) = \begin{cases} \lambda e^{-\lambda x}, & x \geq 0, \lambda > 0 \\ 0, & x < 0 \end{cases}$$



where:

- **x** is a random variable.
- **λ** is the **rate parameter**.

The **mean** and **standard deviation** of the exponential distribution are both given by **β**, where:

$$\beta = \frac{1}{\lambda}$$

Discrete Distributions

1. **Binomial**,
2. **Poisson**, and
3. **Bernoulli distributions**

1. Binomial Distribution

- The binomial distribution is frequently encountered in machine learning. It represents experiments with only two possible outcomes: success or failure. This distribution is also known as the **Bernoulli trial**.
- The purpose of the binomial distribution is to determine the probability of obtaining exactly **k** successes in **n** trials. The probability of achieving **k** successes from **n** trials is given by the binomial coefficient:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

The probability distribution function for a binomially distributed random variable is given by:

$$p^k(1-p)^{n-k} \quad \text{or} \quad p^k q^{n-k}$$

where **p** is the probability of success, and **q = 1 - p** is the probability of failure.

Combining both, the probability density function (PDF) of the binomial distribution is:

$$P(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$$

where:

- **p** is the probability of success per trial,
- **k** is the number of successes, and
- **n** is the total number of trials.

The mean of a binomial distribution is:

$$\mu = n \times p$$

The variance is given by:

$$\sigma^2 = np(1-p)$$

Hence, the standard deviation is:

$$\sigma = \sqrt{np(1-p)}$$

2. Poisson Distribution

- The **Poisson distribution** is another important probability distribution. It models the probability of a given number of events occurring within a fixed time interval.
- The key parameter, **λ (lambda)**, represents the mean number of occurrences over the interval.

Some practical applications of the Poisson distribution include:

- Modeling the number of emails received per hour,
- Estimating customer arrivals at a shop,
- Counting the number of phone calls received at an office.

The probability density function (PDF) of a Poisson-distributed variable is:

$$P(X = x) = \frac{e^{-\lambda}\lambda^{x}}{x!}$$

where:

- **x** is the number of events occurring, and
- **λ** is the expected number of occurrences over a given period.

The **mean** of a Poisson distribution is **λ**, and its **standard deviation** is **√λ**.

3. Bernoulli Distribution

- The **Bernoulli distribution** models experiments with a **binary** outcome (i.e., either success or failure).
- The probability of success is **p**, while the probability of failure is **1 - p**.

3. Bernoulli Distribution

The **Probability Mass Function (PMF)** of a Bernoulli-distributed random variable is:

$$P(X = k) = \begin{cases} q = 1 - p, & \text{if } k = 0 \\ p, & \text{if } k = 1 \end{cases}$$

The **mean** of a Bernoulli distribution is:

$$\mu = p$$

The **variance** is:

$$\sigma^2 = p(1 - p) = q$$

Density Estimation

- Density estimation involves determining the probability distribution of a dataset based on observed values. Given a set of observed values $x_1, x_2, \ldots, x_n$ from a larger dataset with an unknown distribution, density estimation aims to approximate the underlying distribution.
- The estimated density function, denoted as **p(x)**, can be used to evaluate the probability of any unknown data point **x**.
- If **p(x)** is lower than a predefined threshold **ε**, then **x** is likely an anomaly or outlier.
- Otherwise, **x** is considered normal. This concept is often used in **anomaly detection**.

There are two primary methods for density estimation:

- Parametric Density Estimation
- Non-Parametric Density Estimation

Parametric Density Estimation

- This method assumes that the data follows a **known** probability distribution.
- The density function can be expressed as **$p(x \mid \Theta)$**, where **Θ** represents the parameters of the distribution.
- A widely used parametric method is **Maximum Likelihood Estimation (MLE)**.

Maximum Likelihood Estimation (MLE)

- Maximum Likelihood Estimation (MLE) is a probabilistic framework used for density estimation. It involves defining a **likelihood function**, which represents the probability of observing the given data under a particular distribution with specific parameters.
- For instance, if we have a set of observations $\mathbf{X = \{x_1, x_2, ..., x_n\}}$, density estimation involves selecting a probability density function (PDF) with suitable parameters to model the data. MLE formulates this as an **optimization problem**, aiming to maximize the likelihood of observing **X** given the parameters **Θ**.

Maximum Likelihood Estimation (MLE)

- Mathematically, the likelihood function is expressed as:

$$L(X;\Theta) = P(X|\Theta)$$

where **L(X; Θ)** represents the probability of observing **X** given the parameters **Θ**. The goal of MLE is to find **Θ** such that **L(X; Θ)** is maximized.

The joint probability of observing all data points can be written as:

$$\prod_{i=1}^{n} p(x_i;\Theta)$$

Maximum Likelihood Estimation (MLE)

Since direct computation of this formula can be unstable, the problem is typically restated as the **maximum log-likelihood** function:

$$\sum_{i=1}^{n} \log p(x_i; \Theta)$$

Rather than maximizing, one can equivalently **minimize** the **negative log-likelihood function**:

$$-\sum_{i=1}^{n} \log p(x_i; \Theta)$$

Minimization is often preferred in optimization problems.

Relevance of MLE in Machine Learning

- MLE plays a crucial role in **predictive modeling** within machine learning. It is particularly relevant to **regression problems**, which are often solved using the **least-squares** approach.
- From the MLE perspective, if a regression model is framed as predicting **y** given **x**, then the MLE framework can be applied as:

$$\max \sum \log g(y|x,h)$$

- where **g(y | x, h)** represents the conditional probability of **y** given **x** with model parameters **h**.

Module 2.2: Basic Learning Theory

1. Design of Learning System,
2. Introduction to Concept of Learning,
3. Modelling in Machine Learning.

End of Module2