Statistics & Probability for AI

Goal:

To understand how data behaves, how to summarize it, how to model randomness, and how to make decisions under uncertainty — all essential for machine learning and AI systems.

SECTION 1: Descriptive Statistics

Descriptive statistics help summarize and describe the main features of a dataset.z **Key Concepts**

Statistic	Description	Python Example
Mean (Average)	Sum of all values divided by number of values.	np.mean(data)
Median	Middle value when sorted.	np.median(data)
Mode	Most frequent value.	stats.mode(data)
Variance (σ^2)	Average squared deviation from the mean.	np.var(data)
Standard Deviation (σ)	Square root of variance — measures spread.	np.std(data)

📊 **Example**

```
import numpy as np
from scipy import stats

data = [2, 4, 4, 4, 5, 5, 7, 9]

mean = np.mean(data)
median = np.median(data)
mode = stats.mode(data, keepdims=True)[0][0]
variance = np.var(data)
std_dev = np.std(data)
```

```
print(mean, median, mode, variance, std_dev)
```

Output:

Mean = 5.0

Median = 4.5

Mode = 4

Variance = 4.0

Standard Deviation = 2.0

SECTION 2: Probability Distributions

These describe how probabilities are distributed over possible outcomes.

1. Normal Distribution

- Bell-shaped curve.
- Many natural phenomena follow this (e.g., height, weight).

```
import numpy as np
import matplotlib.pyplot as plt
data = np.random.normal(0, 1, 1000)
plt.hist(data, bins=30)
plt.title("Normal Distribution")
plt.show()
```

2. Binomial Distribution

- Discrete outcomes (e.g., success/failure in 10 coin tosses).

```
from scipy.stats import binom
x = np.arange(0, 11)
p = binom.pmf(x, 10, 0.5)
plt.bar(x, p)
plt.title("Binomial Distribution (n=10, p=0.5)")
```

```
plt.show()
```

3. Poisson Distribution

- Number of events in a fixed time (used in network AI, call centers, etc.)

```
from scipy.stats import poisson

x = np.arange(0, 20)

p = poisson.pmf(x, 5)

plt.bar(x, p)

plt.title("Poisson Distribution (λ=5)")

plt.show()
```

SECTION 3: Hypothesis Testing

Used to make inferences or decisions about a population based on sample data.

Steps:

1. Define **Null Hypothesis (H_o)** and **Alternative Hypothesis (H_1)**
2. Choose a significance level ($\alpha = 0.05$)
3. Calculate test statistic (t, χ^2, etc.)
4. Compare p-value to α
5. Accept or reject H_o

1. t-Test

- Tests if two means are significantly different.

```
from scipy.stats import ttest_ind


group1 = [23, 21, 19, 24, 30]

group2 = [31, 33, 29, 35, 32]


t_stat, p_val = ttest_ind(group1, group2)
```

```
print(t_stat, p_val)
```

If p_val < 0.05, difference is significant.

2. Chi-Square Test

- Used for categorical data (e.g., AI predictions vs. actual outcomes).

```
from scipy.stats import chi2_contingency

data = [[10, 20], [20, 40]]
chi2, p, dof, exp = chi2_contingency(data)
print(chi2, p)
```

SECTION 4: Correlation & Covariance

Used to measure relationships between variables.

Concept	Meaning	Python Example
Covariance	How two variables change together.	np.cov(x, y)
Correlation (r)	Strength and direction of relationship (-1 to 1).	np.corrcoef(x, y)

```
x = [1, 2, 3, 4, 5]
y = [2, 4, 6, 8, 10]

print("Covariance:", np.cov(x, y)[0, 1])
print("Correlation:", np.corrcoef(x, y)[0, 1])
```

QUIZ (10 Questions)

1. What is the difference between variance and standard deviation?
2. What does a p-value less than 0.05 imply?
3. Which distribution models count of events in a fixed time?

4. Mean = 10, SD = 2 → what range covers 68% of data in a normal distribution?
5. What does correlation = 0 mean?
6. In a binomial distribution, what does “p” represent?
7. When would you use a chi-square test?
8. What is the null hypothesis in a t-test?
9. Which Python library is used for hypothesis testing?
10. What is the significance level α commonly used in hypothesis tests?

ASSIGNMENT

1. Using Python and NumPy:
    - Generate 100 random numbers following a normal distribution (mean=50, std=10).
    - Compute and display: mean, median, mode, variance, and std deviation.
2. Simulate a **binomial distribution** for 20 trials and success probability 0.4. Plot its histogram.
3. Create two sets of random data (representing test scores from two groups).
    - Conduct a t-test and interpret the results.
4. Calculate correlation and covariance between two lists:
   X = [10,20,30,40,50], Y = [12,24,33,46,52].

CAPSTONE PROJECT: AI Data Insight Analyzer

Title:

Building an “AI Data Insight Analyzer” that helps detect data trends and relationships using real datasets.

Objective:

Use Python (NumPy, SciPy, Pandas, Matplotlib) to:

- Load a real dataset (e.g., student performance, COVID-19, sales data)

- Compute **descriptive statistics**
- Plot **distributions**
- Perform **hypothesis testing**
- Calculate **correlation and covariance**
- Generate **automated summary report**

Bonus (AI Extension):

Train a simple linear regression model to predict one variable based on another — linking your statistics foundation to AI prediction.

Tools You'll Use

- numpy
- scipy
- pandas
- matplotlib
- seaborn (optional, for prettier plots)