

Lab Assignment 19

Student Name: Chauhan Vandana Ramdayal

Student Id: AF0411629

Topic: Numpy Statical Function

NumPy Statistical Functions

Statistics involves gathering data, analyzing it, and drawing conclusions based on the information collected.

NumPy provides us with various statistical functions that can perform statistical data analysis.

Common NumPy Statistical Functions

Here are some of the statistical functions provided by NumPy:

Functions	Descriptions
<code>median()</code>	return the median of an array
<code>mean()</code>	return the mean of an array
<code>std()</code>	return the standard deviation of an array
<code>percentile()</code>	return the nth percentile of elements in an array
<code>min()</code>	return the minimum element of an array
<code>max()</code>	return the maximum element of an array

1.How to find the mean of every NumPy array in the given list?

Input: list = [np.array([3, 2, 8, 9]), np.array([4, 12, 34, 25, 78]), np.array([23, 12, 67])]

Code:

```
import numpy as np
list_of_arrays = [np.array([3, 2, 8, 9]), np.array([4, 12, 34, 25, 78]), np.array([23, 12, 67])] # Input list of NumPy arrays
means = [np.mean(array) for array in list_of_arrays] # Calculate the mean of every NumPy array in the list
print(means)
```

Output:-

```
vscode/1X1.py
[np.float64(5.5), np.float64(30.6), np.float64(34.0)]
```

2. Compute the median of the flattened NumPy array Input: x_odd = np.array([1, 2, 3, 4, 5, 6, 7])

Code:-

```
import numpy as np
x_odd = np.array([1,2,3,4,5,6,7])
print("Printing the array --",x_odd)
Med_value = np.median(x_odd)
print("Median of the array is...",Med_value)
```

Output:-

```
Printing the array -- [1 2 3 4 5 6 7]
Median of the array is... 4.0
```

3. Compute the standard deviation of the NumPy array Input: arr = [20, 2, 7, 1, 34]

Code:-

```
import numpy as np
arr = [20,2,7,1,34]
print("arr:",arr)
print("std of array:",np.std(arr))
print("less precision with float32",np.std(arr,dtype="float32"))
```

Output:-

```
arr: [20, 2, 7, 1, 34]
std of array: 12.576167937809991
less precision with float32 12.576168
```

4. Suppose you have a CSV file named 'house_prices.csv' with price information, and you want to perform the following

operations:

- 1. Read the data from the CSV file into a NumPy array.
- 2. Calculate the average of house prices.
- 3. Identify house price above the average.
- 4. Save the list of high prices to a new CSV file. Note: Download 'house_prices.csv' file from LMS.

Code:-

```
import numpy as np
# Step 1: Read the data from the CSV file into a NumPy array
file_name = 'house_prices.csv'
# # Load only the house prices column into a NumPy array
house_prices = np.loadtxt(file_name, delimiter=',', skiprows=1)
# # Step 2: Calculate the average of house prices
average_price = np.mean(house_prices)
print(f"Average House Price: {average_price}")
# # Step 3: Identify house prices above the average
high_prices = house_prices[house_prices > average_price]
print(f"House Prices Above Average: {high_prices}")
# # Step 4: Save the list of high prices to a new CSV file
np.savetxt('high_prices.csv', high_prices, delimiter=',', header="High Prices", fmt="%s", comments='')
print("High prices saved to 'high_prices.csv'.")
```

Output:-

```
Average House Price: 50374.32274881517
House Prices Above Average: [ 75000.  52778.  58500. ... 187528. 187529. 187530.]
High prices saved to 'high_prices.csv'.
```