Python Chapter 5 Functions Activities

Contents

Time required: 90 minutes

Online Tutorials

Go through the following tutorials.

- LearnPython.org Functions
- Python Functions

Python Statistics Module

The Python statistics module is a part of the Python Standard Library, which means it comes pre-installed with Python. It provides a set of functions for working with statistical operations and calculations. This module is particularly useful for anyone working with data analysis, scientific computing, or even in fields like cybersecurity and IT where data analysis is crucial.

Mean (Average)

Calculates the arithmetic mean (average) of a list of numbers.

The term "mean" refers to the average of a set of numbers. It is calculated by adding up all the numbers in the set and then dividing that sum by the total number of values in the set. The mean is a measure of central tendency and is often used to represent the typical or average value in a dataset.

```
import statistics
data = [1, 2, 3, 4, 5]
avg = statistics.mean(data)
print(avg)  # Output: 3.0
```

Median

Finds the middle value of a data set. If the data set has an even number of elements, it returns the average of the two middle values.

The "median" is the middle value in a dataset when the values are arranged in ascending or descending order. If there is an even number of values, the median is the average of the two middle values. The median is a measure of central tendency and is less affected by extreme values (outliers) compared to the mean, making it useful for describing the center of a dataset.

```
import statistics
data = [1, 3, 5, 7]
median = statistics.median(data)
print(median)  # Output: 4.0
```

Mode

The most frequently occurring value(s) in a dataset.

The "mode" is the value that appears most frequently in a dataset. It is the number that occurs with the highest frequency, making it a measure of central tendency. A dataset can have no mode if all values occur with the same frequency (no value appears more often than others), or it can have multiple modes if two or more values have the same highest frequency. The mode is useful for identifying the most common or popular value in a dataset.

```
import statistics
data = [1, 2, 2, 3, 4]
mode = statistics.mode(data)
print(mode)  # Output: 2
```

Standard Deviation

Calculates the standard deviation, which measures the amount of variation or dispersion in a dataset.

The "standard deviation" is a statistical measure that quantifies the amount of variation or dispersion in a dataset. It indicates how spread out the values in a dataset are from the mean (average) of the dataset. A low standard deviation means that the values tend to be close to the mean, while a high standard deviation indicates that the values are more spread out.

Mathematically, the standard deviation is calculated by taking the square root of the average of the squared differences between each value in the dataset and the mean. This process allows you to understand the degree of variability or uncertainty in the data. In practical terms, a smaller standard deviation suggests that the data points are clustered closely around the mean, while a larger standard deviation implies that the data points are more widely dispersed from the mean. Standard deviation is a valuable tool for assessing the consistency or reliability of data in various fields, including statistics, finance, and science.

```
import statistics
data = [2, 4, 4, 4, 5, 5, 7, 9]
stdev = statistics.stdev(data)
print(stdev)  # Output: 2.0
```

Variance

Calculates the variance, which is the square of the standard deviation.

Variance measures how data points in a dataset deviate from the mean. It quantifies the spread or dispersion of data. A larger variance means greater variability, while a smaller variance implies less variability.

To calculate variance, you follow these steps:

1. Calculate the mean (average) of the dataset.
2. For each data point, subtract the mean, then square the result.
3. Find the average of these squared differences.

Variance is often used in conjunction with the standard deviation, which is simply the square root of the variance. Together, variance and standard deviation help assess the spread and consistency of data in various fields, including finance, science, and quality control.

```
import statistics
data = [2, 4, 4, 4, 5, 5, 7, 9]
variance = statistics.variance(data)
print(variance)  # Output: 4.0
```

Tutorial 5.1: Statistics Module

Python functions can be stored in an external file called a module.

Create a Python file named **stats_module.py** Notice that this module has a **main()** function. This is to test the module.

```
"""
    Name: stats_module.py
    Author:
    Created:
    Purpose: A module with a calculate common statistics function
"""
# Import Python statistics module
import statistics


# ----------------------- GET STATS -------------------------------------#
def get_stats(num):
    """Calculate and return common stats on a list"""
    # Calculate the mean (average)
    avg = statistics.mean(num)

    # Calculate the median (middle value)
    med = statistics.median(num)

    # Calculate the mode (central tendency)
    mde = statistics.mode(num)

    # Standard deviation is a measure of how spread out the numbers are
    stdv = statistics.stdev(num)

    # How much does the data spread out
    var = statistics.variance(num)
    return avg, med, mde, stdv, var
```

```
31  def main():
32      """Place code here to test the module"""
33      # Using the statistics library requires a collection
34      # In this case, a list
35      nums = [23.4, 46, 78, 202.5, 34, 42, 77.7, 42]
36      avg, med, mde, stdv, var = get_stats(nums)
37      print(nums)
38      print(f"Mean: {avg}")
39      print(f"Median: {med}")
40      print(f"Mode: {mde}")
41      print(F"Standard Deviation: {stdv:.2f}")
42      print(F"Variance: {var:.2f}")
43
44
45  # If a standalone program, call the main function
46  # Else, use as a module
47  if __name__ == '__main__':
48      main()
```

stats_main.py

```
"""
    Name: stats_main.py
    Written by:
    Written on:
    Purpose: Use statistics functions in an external module
"""
# Import external module
import stats_module


def main():
    # Using the statistics library requires a collection
    # In this case, a list of numbers
    nums = [23.4, 46, 78, 202.5, 34, 42, 77.7, 42]
    # Get statistics from external module
    avg, med, mde, stdv, var = stats_module.get_stats(nums)
    print(f"Numbers: {nums}")
    print(f"              Mean: {avg:.2f}")
    print(f"            Median: {med:.2f}")
    print(f"              Mode: {mde:.2f}")
    print(F"Standard Deviation: {stdv:.2f}")
    print(F"          Variance: {var:.2f}")


# If a standalone program, call the main function
# Else, use as a module
if __name__ == '__main__':
    main()
```

Example run:

```
Numbers: [23.4, 46, 78, 202.5, 34, 42, 77.7, 42]
              Mean: 68.20
            Median: 44.00
              Mode: 42.00
Standard Deviation: 57.65
          Variance: 3323.60
```

Assignment Submission

1. Use pseudocode or TODO.
2. Comment your code to show evidence of understanding.

3. Attach the program files.
4. Attach screenshots showing the successful operation of the program.
5. Submit in Blackboard.