Statistics Tutorial 1 : Mean, Median, Mode, Standard Deviation, Coefficient Variation

1. Statistics is the science of collecting and analyzing data taken from a sample of the population

2. The Population represents all items or people of interest. *A Sample is a subset of the population that we can analyze. We mainly focus on Successes, or results we are looking for in a sample. Examples being Age, Car Owner, College Graduates, Sex, Home Owner, etc. *Here M represents Successes in the Population. N the Total Population. x successes in the sample. And, n the total sample from the population.

3. There are many types of data. Categorical Data describes what makes a thing unique like (Age, Car Owner, Sex, Graduate, or the answer to a Yes or No Question)

4. Numerical Data is either Finite meaning that it has an ending value, or infinite meaning the opposite.

5. Continuous Data is data that can be broken down into infinitely smaller amounts. Think of things like distance, height, weight, etc.

6. Qualitative Data can be either Nominal (Named Data). It is data used for naming something which doesn't have an order. Race would be an example because there are many races, but there is no order to them. Ordinal data is also named but it has an order like (Bad, OK, Good, Great)

7. Quantitative Data is like a ratio or interval being an amount between 2 defined amounts like (Numbers between 8 and 16).

8. There are many ways to visualize data. A Cross Table shows relationships between rows and columns of data. Frequency shows how often something happens. Here we can see that when we sampled 100 random men that 78 were men that didn't exercise.

9. With Pie Charts each slice represents a category and the size of the slice represents its frequency. What differentiates it from other charts is that it must always equal 100%.

10. Bar Charts have bars that represent the categories and the bar lengths represent the frequency

11. A Pareto Chart lists categories in descending order and includes a line that represents the cumulative frequency or the sum of all other frequencies.

12. A Frequency Distribution Table focuses on the number of occurrences or the frequency. Here we list a range of test scores and how many students scored in that range. *A Histogram differs from a bar graph in that histograms show the distribution of grades in a range in this example versus using categories like a bar graph. Also Histograms are drawn with the bars touching.

13. The Mean or average provides an average value by summing all values and dividing by the number of components. μ is used to represent the mean of the population. $\bar{x}$ represents the mean of a sample. While it can be very useful often outliers dramatically effect results. For example 1, 2, 3, 4, 5 has a mean of 3. 1, 2, 3, 4, 100 has a mean of 22.

14. Median tries to eliminate the influence of outliers by returning the number at the center of the data set. If you have an even number of components instead take the center 2 values and return the average.

15. The Mode returns the value that occurs most often. If components occur at an equal rate there is no mode. If there are multiple values then you will have more than 1 mode.

16. The Variance measures how data is spread around the mean. There is both a symbol for variance of the population and the sample. To find it we first calculate the mean. Then we sum all sample values minus the mean squared. Then we divide by the number of samples minus 1 in the case of a variance of a sample which is what well use.

17. Because we square values with variance that gives extra weight to outliers. For this reason we find the square root of the variance to find the Standard Deviation. The Standard Deviation is large if the numbers are more spread out and lower if they are closer to the mean.

18. The coefficient of variation is used to compare 2 measurements that operate on different scales. Here I'm comparing miles to kilometers. Even though they measure the same distance because they use different units that is not seen when calculating standard deviation. By dividing by the mean however we can see that they actually have the same dispersion.

19. Covariance tells us if 2 groups of data are moving in the same direction. Here I'll compare whether earnings effect the market cap of a corporation. The market cap of a corporation is the total value of all that corporations stock. You make this calculation by plugging in the values minus their mean and then multiply. If I do that I get a value of 5803.2. If the value is greater than 0 that means those values are moving together. If less than 0 they are moving in opposite directions. Zero means that they are independent.

20. The Correlation Coefficient adjust the covariance so that it is easier to see the relationship between x and y. Its value can't be greater than 1 or less than -1. The closer you get to 1 the closer the relationship between the values. In this example we plug in the standard deviations of the market cap and earnings. When we do this we get a value of .6601 which means they are correlated. Perfect correlation would have a value of 1. 0 shows independence. Negative values show an inverse correlation.

```
import math


def mean(*args):
    val_sum = sum(args)
    return val_sum / len(args)


def median(*args):
    if len(args) % 2 == 0:
        i = round((len(args) + 1) / 2)
        j = i - 1
        return (args[i] + args[j]) / 2
    else:
        k = round(len(args) / 2)
        return args[k]


def mode(*args):
    # Count how many times values show up in
    # the list and put it in a dictionary
    dict_vals = {i: args.count(i) for i in args}
    # Create a list of keys that have the maximum
    # number of occurrence in the list
    max_list = [k for k, v in dict_vals.items() if v == max(dict_vals.values())]
    return max_list


def variance(*args):
    mean_val = mean(*args)
    numerator = 0
    for i in args:
        numerator += (i - mean_val) ** 2
    denominator = len(args) - 1
    return numerator / denominator


def standard_deviation(*args):
    return math.sqrt(variance(*args))


def coefficient_variation(*args):
    return standard_deviation(*args) / mean(*args)


def covariance(*args):
    # Use a list comprehension to get all values
    # stored in the 1st & 2nd list
    list_1 = [i[0] for i in args]
    list_2 = [i[1] for i in args]
    # Pass those lists to get their means
    list_1_mean = mean(*list_1[0])
    list_2_mean = mean(*list_2[0])
    numerator = 0
```

```
    # We must have the same number of elements
    # in both lists
    if len(list_1[0]) == len(list_2[0]):
        for i in range(len(list_1[0])):
            # FInd xi - x mean * yi - y mean
            numerator += (list_1[0][i] - list_1_mean) * (list_2[0][i] - list_2_mean)
        denominator = len(list_1[0]) - 1
        return numerator / denominator
    else:
        print("Error : You must have the same number of values in both lists")


def correlation_coefficient(*args):
    list_1 = [i[0] for i in args]
    list_2 = [i[1] for i in args]
    # Pass those lists to get their standard deviations
    list_1_sd = standard_deviation(*list_1[0])
    list_2_sd = standard_deviation(*list_2[0])
    print(f"L1 SD : {list_1_sd}")
    print(f"L2 SD : {list_2_sd}")
    denominator = list_1_sd * list_2_sd
    # Get the covariance
    numerator = covariance(*args)
    return numerator / denominator


print(f"Mean : {mean(1, 2, 3, 4, 5)}")
print(f"Median : {median(1, 2, 3, 4, 5)}")
print(f"Median : {median(1, 2, 3, 4, 5, 6)}")
print(f"Mode : {mode(1, 2, 3, 4, 5, 4, 5)}")
print(f"Variance : {variance(4, 6, 3, 5, 2)}")
print(f"Standard Deviation : {standard_deviation(4, 6, 3, 5, 2)}")
print(f"Coefficient Variation (miles): {coefficient_variation(3, 4, 4.5, 3.5)}")
print(f"Coefficient Variation (kms): {coefficient_variation(4.828, 6.437, 7.242, 5.632)}")

# List that contains market cap in 1st list
# and earnings in the 2nd list
m_d_list = [[1532, 1488, 1343, 928, 615], [58, 35, 75, 41, 17]]
print(f"Stock Covariance : {covariance(m_d_list)}")

# Get the Correlation Coefficient
print(f"Correlation Coefficient : {correlation_coefficient(m_d_list)}")
```