BHARATIYA VIDYA BHAVAN'S
SARDAR PATEL INSTITUTE OF TECHNOLOGY
(Empowered Autonomous Institute Affiliated to University of Mumbai)
[Knowledge is Nectar]

**Department of Computer Engineering**

**Course – Data Analytics Open Elective**

| | |
|---|---|
| **UID** | 2021300126 |
| **Name** | Pranay Singhvi |
| **Class and Batch** | TE Computer Engineering - Batch A |
| **Date** | 23-01-2024 |
| **Lab #** | 2 |
| **Aim** | To find mean, median, mode, range, standard deviation, variance, first and third quartile, and correlation coefficient. |
| **Data Set** | https://www.kaggle.com/datasets/trolukovich/nutritional-values-for-common-foods-and-products |
| **Colab File Link** | https://colab.research.google.com/drive/1s0BvHZn_3K0UGE01jJlws0SM-cDl3z--?usp=sharing |
| **Purpose** | Nutritional Research: Analyzing relationships between different nutrients and exploring trends in food composition for scientific and academic research. |
| **Code** | Mean:<br>**Theory:** The mean is the average of a set of values. It is calculated by summing all the values and dividing by the number of observations.<br>**Application**: In a nutrients dataset, the mean can represent the average nutritional content for a specific nutrient, providing a central measure that reflects the typical value.<br>`def custom_mean(data):`<br>`  return sum(data) / len(data)`<br>Median:<br>**Theory:** The median is the middle value in a dataset when it is ordered. It is less sensitive to extreme values than the mean.<br>**Application**: In a nutrients dataset, the median can be used to understand the central tendency of the data, especially when there are outliers that might disproportionately influence the mean.<br>`def custom_median(data):`<br>`  sorted_data = sorted(data)`<br>`  n = len(sorted_data)`<br>`  if n % 2 == 0:`<br>`  return (sorted_data[n // 2 - 1] + sorted_data[n // 2]) / 2`<br>`  else:`<br>`  return sorted_data[n // 2]` |

## Mode:

**Theory**: The mode is the value that appears most frequently in a dataset.
**Application**: In a nutrients dataset, identifying the mode can highlight the most common nutrient content, aiding in understanding prevalent nutritional characteristics among foods.

```
def custom_mode(data):
    frequency = {}
    for value in data:
        frequency[value] = frequency.get(value, 0) + 1
    mode_values = [key for key, val in frequency.items() if val == max(frequency.values())]
    return mode_values
```

## Range:

**Theory:** The range is the difference between the maximum and minimum values in a dataset.
**Application**: In a nutrients dataset, the range provides insight into the spread of values, helping to identify the variability in nutrient content across different foods.

```
def range(column):
  return max(column)-min(column)
```

## Standard Deviation:

**Theory**: Standard deviation measures the average deviation of each data point from the mean.
**Application**: In a nutrients dataset, a higher standard deviation indicates greater variability in nutrient content, while a lower standard deviation suggests more consistency. Variance and standard deviation help quantify the degree of dispersion in nutrient values.

```
def custom_standard_deviation(data):
    mean_value = sum(data) / len(data)
    squared_diff = [(x - mean_value) ** 2 for x in data]
    variance = sum(squared_diff) / len(data)
    std_deviation = variance ** 0.5
    return std_deviation
```

## First and Third Quartile (Q1 and Q3):

**Theory**: Quartiles divide a dataset into four equal parts. Q1 is the median of the lower half, and Q3 is the median of the upper half.
**Application**: In a nutrients dataset, quartiles help identify the distribution of nutrient values, and the interquartile range (Q3 - Q1) gives a measure of the spread around the median.

```
def calculate_quartiles(numbers):


    sorted_numbers = sorted(numbers)
    n = len(sorted_numbers)
    mid = n//2
    if n % 2 ==0:
      lh = numbers[:mid]
      uh = numbers[mid:]

      first_q = (lh[mid//2] + lh[mid//2 -1]) / 2
      third_q = (uh[mid//2] + uh[mid//2 -1]) / 2

    else:
      lh = numbers[:mid]
      uh = numbers[mid+1:]

      first_q = lh[mid//2]
      third_q = uh[mid//2]

    return [first_q,third_q ]
```

## Correlation Coefficient:

**Theory**: The correlation coefficient measures the strength and direction of a linear relationship between two variables.

**Application**: In a nutrients dataset, the correlation coefficient can be used to explore relationships between different nutrients. For example, it can reveal whether foods high in one nutrient tend to be high or low in another, providing valuable insights into dietary patterns.

```
def correlation_coeff(col1,col2):
  mean1 = custom_mean(col1)
  mean2 = custom_mean(col2)

  sum_product_deviations = sum((xi - mean1)*(yi - mean2) for xi,yi in zip(col1,col2))
  sum_sq_dev1 =  sum((xi - mean1)**2 for xi in col1)
  sum_sq_dev2 =  sum((xi - mean2)**2 for xi in col2)

  coef = sum_product_deviations / ((sum_sq_dev1*sum_sq_dev2)**0.5)
  return coef
```

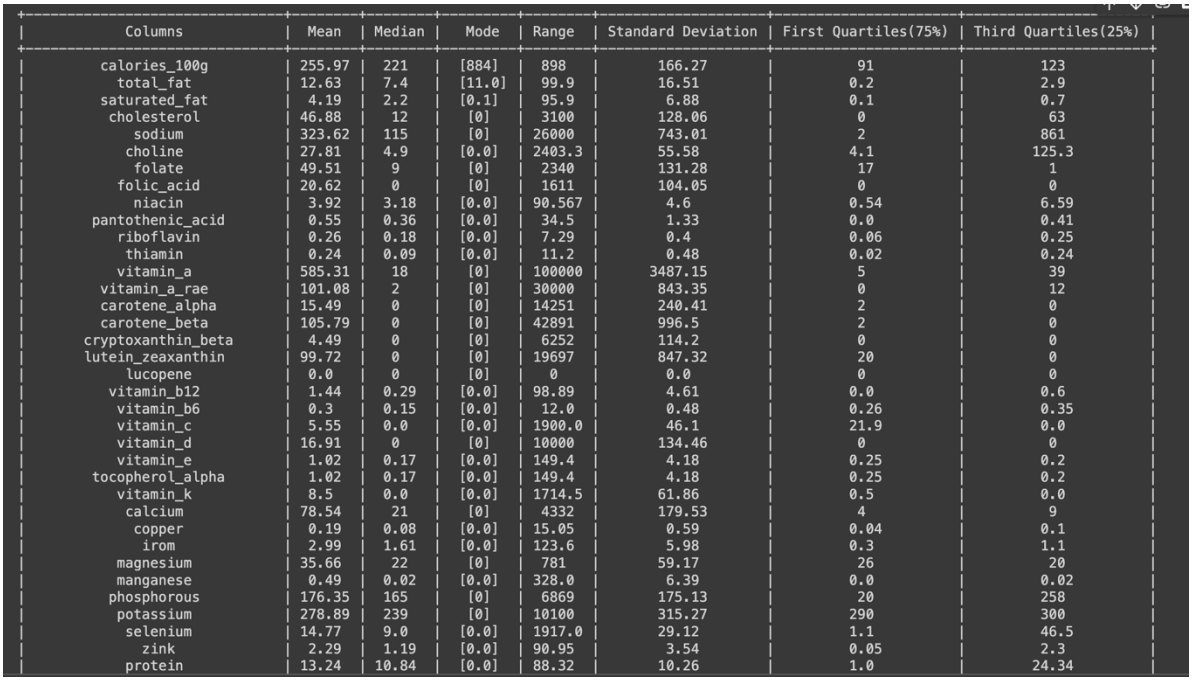
```
print(correlation_coeff(df["monounsaturated_fatty_acids_g"],df["saturated_fatty_acids_g"]))
```

```
0.4876755173136889
```

Now, Using all function given above in below code to print all mean, median, mode and standard deviation for each columns.

```
Table = PrettyTable()
Table.field_names = ["Columns","Mean", "Median", "Mode","Range","Standard Deviation","First Quartiles(75%)", "Third Quartiles(25%)"]
for col in columns:
  mean_value = round(custom_mean(df[col]),2)
  median_value = round(custom_median(df[col]),2)
  mode_value = custom_mode(df[col])
  range_value =  range(df[col])
  deviation_value = round(custom_standard_deviation(df[col]),2)
  result = calculate_quartiles(df[col].tolist())
  Table.add_row([col,mean_value,median_value,mode_value,range_value,deviation_value,round(result[0],2),round(result[1],2)])
```

Output:

| | |
|---|---|
| |  |

| Columns | Mean | Median | Mode | Range | Standard Deviation | First Quartiles(75%) | Third Quartiles(25%) |
|---|---|---|---|---|---|---|---|
| calories_100g | 255.97 | 221 | [884] | 898 | 166.27 | 91 | 123 |
| total_fat | 12.63 | 7.4 | [11.0] | 99.9 | 16.51 | 0.2 | 2.9 |
| saturated_fat | 4.19 | 2.2 | [0.1] | 95.9 | 6.88 | 0.1 | 0.7 |
| cholesterol | 46.88 | 12 | [0] | 3100 | 128.06 | 0 | 63 |
| sodium | 323.62 | 115 | [0] | 26000 | 743.01 | 2 | 861 |
| choline | 27.81 | 4.9 | [0.0] | 2403.3 | 55.58 | 4.1 | 125.3 |
| folate | 49.51 | 9 | [0] | 2340 | 131.28 | 17 | 1 |
| folic_acid | 20.62 | 0 | [0] | 1611 | 104.05 | 0 | 0 |
| niacin | 3.92 | 3.18 | [0.0] | 90.567 | 4.6 | 0.54 | 6.59 |
| pantothenic_acid | 0.55 | 0.36 | [0.0] | 34.5 | 1.33 | 0.0 | 0.41 |
| riboflavin | 0.26 | 0.18 | [0.0] | 7.29 | 0.4 | 0.06 | 0.25 |
| thiamin | 0.24 | 0.09 | [0.0] | 11.2 | 0.48 | 0.02 | 0.24 |
| vitamin_a | 585.31 | 18 | [0] | 100000 | 3487.15 | 5 | 39 |
| vitamin_a_rae | 101.08 | 2 | [0] | 30000 | 843.35 | 0 | 12 |
| carotene_alpha | 15.49 | 0 | [0] | 14251 | 240.41 | 2 | 0 |
| carotene_beta | 105.79 | 0 | [0] | 42891 | 996.5 | 2 | 0 |
| cryptoxanthin_beta | 4.49 | 0 | [0] | 6252 | 114.2 | 0 | 0 |
| lutein_zeaxanthin | 99.72 | 0 | [0] | 19697 | 847.32 | 20 | 0 |
| lucopene | 0.0 | 0 | [0] | 0 | 0.0 | 0 | 0 |
| vitamin_b12 | 1.44 | 0.29 | [0.0] | 98.89 | 4.61 | 0.0 | 0.6 |
| vitamin_b6 | 0.3 | 0.15 | [0.0] | 12.0 | 0.48 | 0.26 | 0.35 |
| vitamin_c | 5.55 | 0.0 | [0.0] | 1900.0 | 46.1 | 21.9 | 0.0 |
| vitamin_d | 16.91 | 0 | [0] | 10000 | 134.46 | 0 | 0 |
| vitamin_e | 1.02 | 0.17 | [0.0] | 149.4 | 4.18 | 0.25 | 0.2 |
| tocopherol_alpha | 1.02 | 0.17 | [0.0] | 149.4 | 4.18 | 0.25 | 0.2 |
| vitamin_k | 8.5 | 0.0 | [0.0] | 1714.5 | 61.86 | 0.5 | 0.0 |
| calcium | 78.54 | 21 | [0] | 4332 | 179.53 | 4 | 9 |
| copper | 0.19 | 0.08 | [0.0] | 15.05 | 0.59 | 0.04 | 0.1 |
| irom | 2.99 | 1.61 | [0.0] | 123.6 | 5.98 | 0.3 | 1.1 |
| magnesium | 35.66 | 22 | [0] | 781 | 59.17 | 26 | 20 |
| manganese | 0.49 | 0.02 | [0.0] | 328.0 | 6.39 | 0.0 | 0.02 |
| phosphorous | 176.35 | 165 | [0] | 6869 | 175.13 | 20 | 258 |
| potassium | 278.89 | 239 | [0] | 10100 | 315.27 | 290 | 300 |
| selenium | 14.77 | 9.0 | [0.0] | 1917.0 | 29.12 | 1.1 | 46.5 |
| zink | 2.29 | 1.19 | [0.0] | 90.95 | 3.54 | 0.05 | 2.3 |
| protein | 13.24 | 10.84 | [0.0] | 88.32 | 10.26 | 1.0 | 24.34 |

| | |
|---|---|
| **Conclusion** | In conclusion, I have learnt to calculate mean, median, mode and standard deviation. I also learnt about first and third quartiles and also learnt to find correlation between two attribute |