



## Measures of Central Tendency

1 # Median of voltage data ignoring any NaN values
2 voltage\_median = np.nanmedian(voltage,axis=0)

3 voltage\_median
[] 娜Open 'voltage\_median' in Data Wrangler

reference: L3-Measures of Central Tendency

## Fruit Price List

```
1 # Fruit price list
      2 fruits = np.array([120,60,85,150,200])
      3 fruits
      1 # Mean of fruit prices
      2 fruits_mean = np.mean(fruits)
      3 fruits_mean
      1 # Median of fruit prices
      2 fruits_median = np.median(fruits)
      3 fruits_median
    1 # Sort fruit prices
2 fruits_sorted = np.sort(fruits)
      3 fruits_sorted

    Open 'fruits_sorted' in Data Wrangler

      1 # Mode of fruit prices
      2 stats.mode(fruits)
  Voltage Response
     1 # Voltage response data
      voltage = np.array([
           [1,2,3,4,5,6,7,8],
            [12,5,9.1,3.3,24,18.5,15.2,np.nan],
           [2.8,4.5,6,9,11.7,14.8,17.3,20]
      6 ])
7 voltage
1 # Size of array
      2 np.shape(voltage)
      1 # Mean of voltage data
      voltage_mean = np.mean(voltage,axis=1)
voltage_mean
1 # Mean of voltage data ignoring any NaN values
      voltage_mean = np.nanmean(voltage,axis=1)
      3 voltage_mean
1 # Transpose voltage data
      voltage = voltage.T
      3 voltage
1 # Size of array
      1 # Mean of voltage data ignoring any NaN values
      2 voltage_mean = np.nanmean(voltage,axis=0)
3 voltage_mean
□ ■ Open 'voltage_mean' in Data Wrangler
```



Gyro A. Madrona Electronics Engineer

```
1 # Sort voltage data
2 voltage_sorted = np.sort(voltage,axis=0)
3 voltage_sorted
[1] 哪Open Voltage_sorted in Data Wrangler

1 # Mode of voltage data
2 voltage_mode = stats.mode(voltage,axis=0)
3 voltage_mode

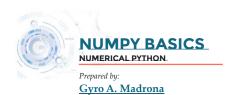
Pythor
```

## Measures of Variability

reference: L4-Measures of Variability

## **Exam Performance**

```
1 # Exam performance data
       2 grade = np.array([3.5,6.7,7,7.4,7.8,8.2,8.5,8.8,9,9.1,9.4,9.8])
       3 grade
圆 Open 'grade' in Data Wrangler
       1 # Maximum grade of exam data
       2 grade_max = np.max(grade)
3 grade_max
       1 # Minimum grade of exam data
       2 grade_min = np.min(grade)
       3 grade_min
      1 # Range of exam data
       2 grade_range = grade_max - grade_min
       3 grade_range
      1 # First quartile (Q1) of exam data
       2 grade_q1 = np.percentile(grade,25) # 25%
       3 grade_q1
      1 # Second quartile (Q2) of exam data
       2 grade_q3 = np.percentile(grade,75) # 75%
       3 grade_q3
      1 # Interquartile range (IQR) of exam data
      2 grade_iqr = grade_q3 - grade_q1
3 grade_iqr
D ~ 1 # Population variance
       2 grade_var = np.var(grade)
3 grade_var
      1 # Sample variance
       2 grade_var = np.var(grade,ddof=1)
       3 grade_var
                                                                                                                                             1 # Population standard deviation
       2 grade_std = np.std(grade)
3 grade_std
       1 # Sample standard deviation
       2 grade_std = np.std(grade,ddof=1)
       3 grade_std
```



Electronics Engineer

Ice Cream Price List

```
1 price = np.array([
      5 price = price.T
      6 price
1 # Mean of USD
      2 usd_mean = np.mean(price,axis=0)[0] # [0] 1st column
      3 usd_mean
     1 # Standard deviation of USD
      2 usd_std = np.std(price,ddof=1,axis=0)[0] # [0] 1st column
      3 usd_std
1 # USD Coefficient of variation
      2 usd_cv = usd_std/usd_mean
3 usd_cv
                                                                                                                                       Python
                                                                                                                          1 # Mean of PHP
      2 php_mean = np.mean(price,axis=0)[1] # [0] 2nd column
      3 php_mean
     1 # Standard deviation of PHP
      2 php_std = np.std(price,ddof=1,axis=0)[1] # [1] 2nd column
      3 php_std
      1 # PHP Coefficient of variation
     2 php_cv = php_std/php_mean
     3 php_cv
  Pooled Standard Deviation
      1 battery = np.array([
      5 battery = battery.T
      6 battery
□ ■ Open 'battery' in Data Wrangler
     1 # Extract rows where the 1st column is 'A'
     2 model_a = battery[battery[:,0]=='A']
     3 model_a
のpen 'model_a' in Data Wrangler
    1 # Extract rows where the 1st column is 'B'
2 model_b = battery[battery[:,0]=='B']
3 model_b
1 # Extract rows where the 1st column is 'C'
    1 # Extract rows where the 1st column i
2 model_c = battery[battery[:,0]=='C']
      3 model_c
```



Prepared by:

Gyro A. Madrona

Electronics Engineer

```
1 # Convert string to float
              voltage_a = model_a[:,1].astype(float)
voltage_b = model_b[:,1].astype(float)
voltage_c = model_c[:,1].astype(float)
1 # Average variance
2 a_var = np.var(voltage_a, ddof=1)
3 b_var = np.var(voltage_b, ddof=1)
4 c_var = np.var(voltage_c, ddof=1)
             6 ave_var = np.mean([a_var,b_var,c_var])
7 ave_var
        1 # Pooled standard deviation is the square root of average variance
2 pooled_std = np.sqrt(ave_var)
3 pooled_std
```