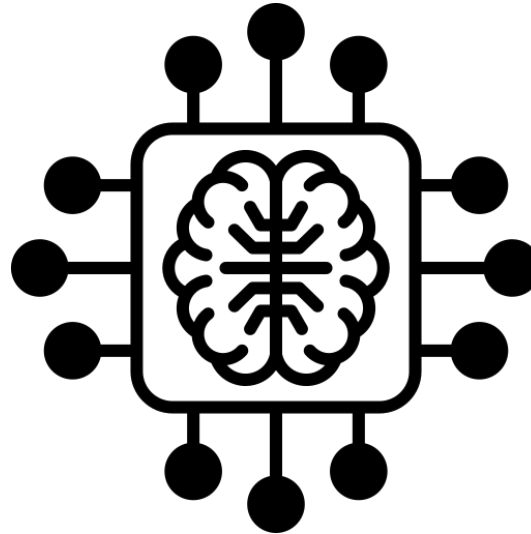


# SBS4115 Fundamentals of AI & Data Analytics



## Statistical Analysis for Data Analytics

Lecturer: Ir Dr Kelvin K. W. Siu  
email: [kelvinsiu@thei.edu.hk](mailto:kelvinsiu@thei.edu.hk)



Department of Construction,  
Environment and Engineering

# Intended Learning Outcomes



- By the end of this lecture, you will be able to...
  - Explain what descriptive statistics is
  - Calculate arithmetic mean, median and mode
  - Suggest the dispersion or variability in a set of data
  - Standardize the data by the standardization formulae
  - Conduct sampling process.

# Introduction to Statistics



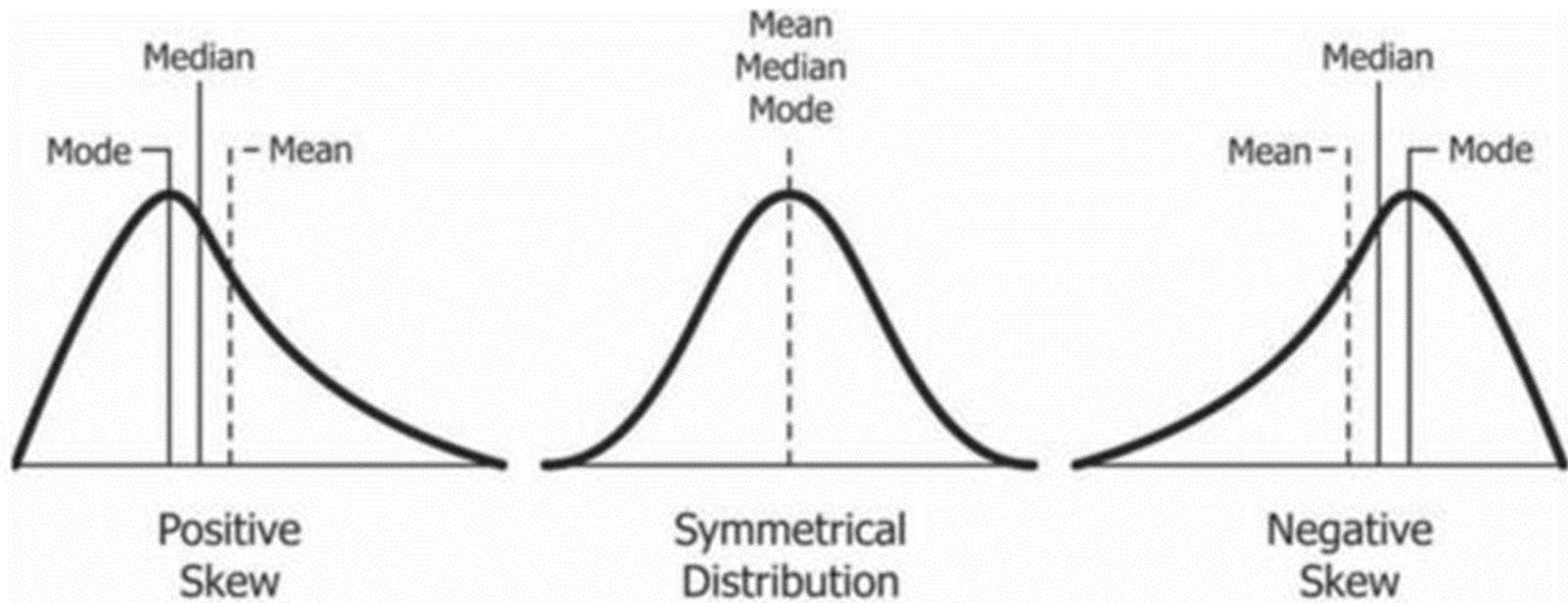
- **Statistics** refers to the scientific method by which data is collected, organized, analyzed and interpreted for the purpose of description and decision making, and therefore it is **the foundation knowledge of data science**.
- Statistical methods can be further divided into two subdivisions:
  1. **Descriptive statistics** deals with the presentation of numerical facts, or data, in either tables or graphs form, and with the methodology of analyzing the data.
  2. Inferential statistics involves techniques for making inferences about the whole population on the basis of observations obtained from samples.
- In this course, we will focus on descriptive statistics which can be applied to data analytics.

# Central Tendency



- It seems apparent that in most set of numerical data there is a tendency for the observed values to group themselves about some interior values; **some central values seem to be the characteristics of the data.**
- This phenomenon is referred to as **central tendency**.
- For a given set of data, the measure of location we use depends on what we mean by middle; different definitions give rise to different results.
- We shall consider some more commonly used measures, namely **arithmetic mean, median and mode**.
- The formulas in finding these values depend on whether they are ungrouped data or grouped data.

# Central Tendency



# Central Tendency



- **Arithmetic mean** (or simply mean) is obtained by adding together all of the measurements and dividing by the total number of measurements taken.
- **The population mean  $\mu$  is the mean of all the  $N$  measurements from the whole population.**
- Mathematically it is given by the following formula:

$$\mu = \frac{\sum x_i}{N}$$

where  $x_1, x_2, x_3, \dots$  are the measurements.

# Central Tendency



- Arithmetic mean can be used to calculate any numerical data and it is **always unique**.
- It is obvious that extreme values affect the mean.
- Also, arithmetic mean ignores the degree of importance in different categories of data.
- Consider the following set of data:

2, 2, 2, 2, 8, 10, 15, 17, 20, 99

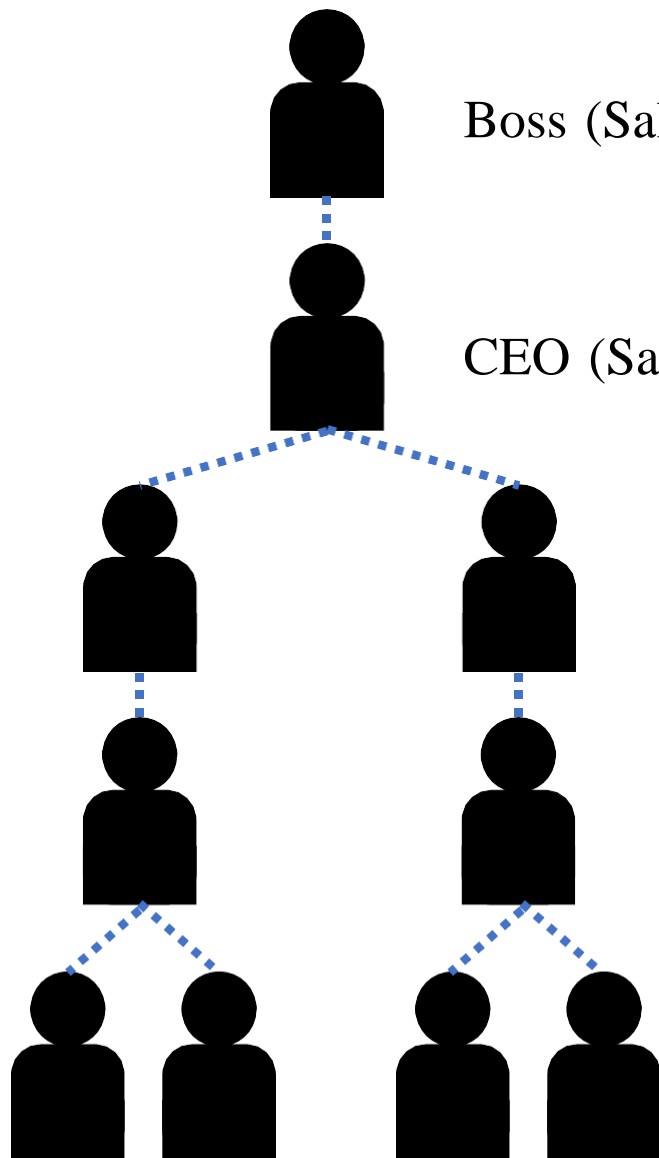
- The mean is:

$$\frac{2+2+2+2+8+10+15+17+20+99}{10} = \frac{177}{10} = 17.7$$

# Central Tendency

- **Median** is defined as the middle item (or 50<sup>th</sup> percentile) of all given observations arranged in order.
  - In case of the number of measurements is even, the median is obtained by taking the average of the middle.
  - Median is unique and it is not affected by a few extreme values.
- **Mode** is the value which occurs most frequently.
  - Given a set of data, we simply count the frequency of each value.
  - If more than one values have the same largest frequency, then the mode is not unique.
- The median is:  $\frac{8+10}{2} = 9$
- The mode is: 2 (*freq*: 4)





Boss (Salary: 99)

CEO (Salary: 20)

Manager (Hong Kong) (Salary: 17)

Manager (Kowloon) (Salary: 15)

Vice Manager (Hong Kong) (Salary: 10)

Vice Manager (Kowloon) (Salary: 8)

Officer (Salary: 2)

How would you describe the central tendency of the salary of this company?

Unit: HK\$10,000

# Central Tendency



- The example above illustrates using different measures of central tendency might give different results.
- **Depending on the situation, we need to choose a suitable method.**
- For example, to reflect the salary level of employees in Hong Kong, median is more suitable than mean and mode.
- The reason is that mean salary is greatly affected by a small fraction of employees with extremely high salary (e.g. CEO of big firms), while the mode might be just equal to the minimum wage.

# Central Tendency

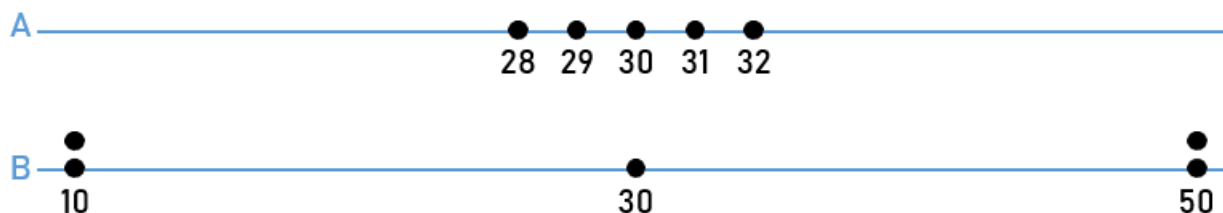
Table 1: Overall Monthly Wage Distribution of Employees, May – June 2020

Percentile	Monthly wage (HK\$)	
10 <sup>th</sup>	9,600	(-3.9%)
25 <sup>th</sup>	13,200	(+1.4%)
50 <sup>th</sup> (median)	18,400	(+1.5%)
75 <sup>th</sup>	28,800	(+2.2%)
90 <sup>th</sup>	45,300	(+1.6%)

Note: Monthly wage figures are rounded to the nearest hundred of Hong Kong dollar. Figures in brackets represent percentage changes over May – June 2019, which are calculated based on unrounded monthly wage figures.

# Dispersion

- Central tendency including mean, median and mode **may not be able to reflect the true picture of some data.**
- Compare the two sets of data below:  
A: 28, 29, 30, 31, 32  
B: 10, 10, 30, 50, 50
- Both the mean and median of A are 30, which is equal to that of B.
- However, it is obvious that the structure of A and B are quite different in the sense that the values in A are close, meanwhile those in B are more extreme.



# Dispersion

- It is necessary to set up some measures to study the dispersion or variability in a set of data.
- Range** is the difference between the maximum and the minimum values.



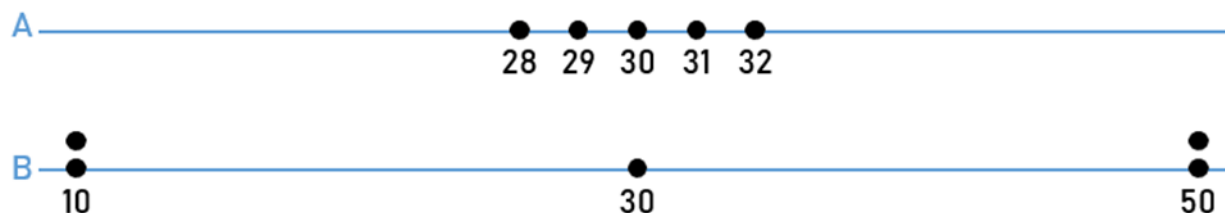
- For A, the range is  $32 - 28 = 4$ .



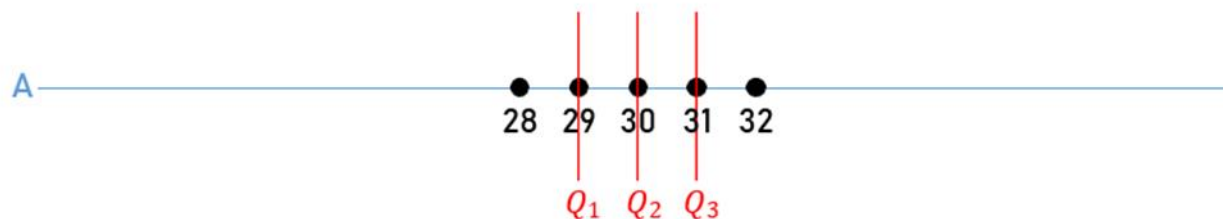
- For B, the range is  $50 - 10 = 40$ .

# Dispersion

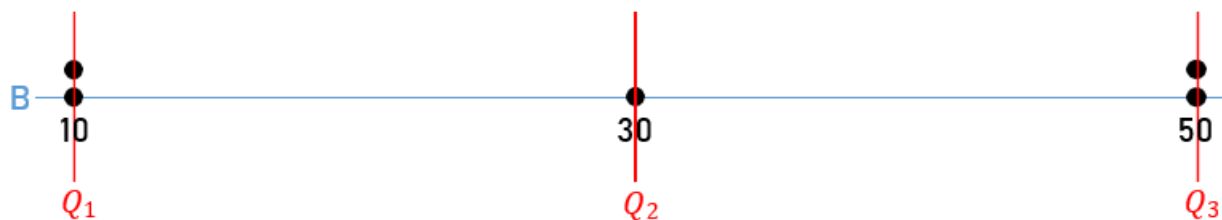
- **Quartiles** are the most commonly used values of position which divides distribution into four equal parts such that:
  - 25% of the data are  $\leq Q_1$ ;
  - 50% of the data are  $\leq Q_2$  (or median);
  - 75% of the data are  $\leq Q_3$ .
- It is also denoted the value  $(Q_3 - Q_1)/2$  is the quartile deviation, QD, or the semi-interquartile range.



# Dispersion



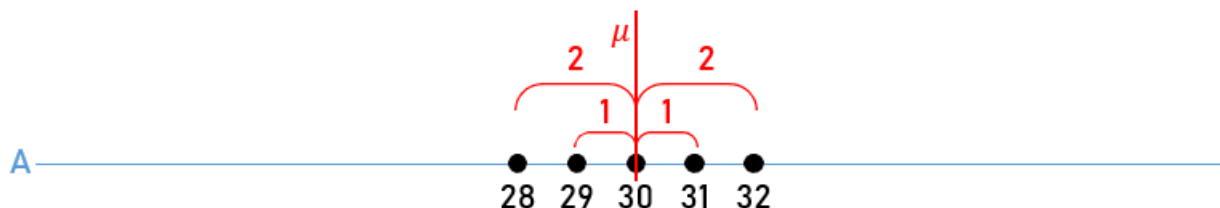
- For A,  $Q_1 = 29$ ,  $Q_2 = 30$ ,  $Q_3 = 31$ .



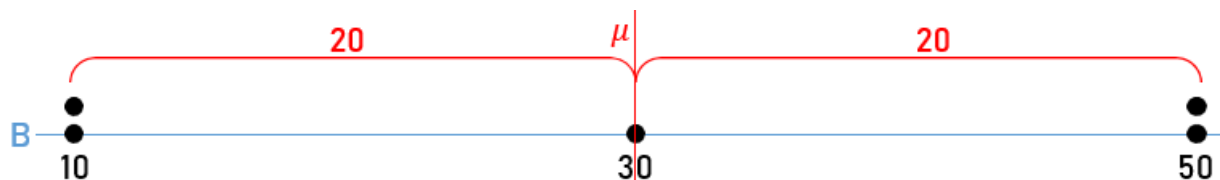
- For B,  $Q_1 = 10$ ,  $Q_2 = 30$ ,  $Q_3 = 50$ .

# Dispersion

- **Mean absolute deviation (MAD)** is the mean of the absolute values of all deviations from the mean, so it takes every item into account.
- Mathematically it is given as ( $\mu$  is the population mean):  $\frac{\sum |x_i - \mu|}{N}$



$$MAD_A = \frac{|28 - 30| + |29 - 30| + |30 - 30| + |31 - 30| + |32 - 30|}{5} = 1.2$$



$$MAD_B = \frac{|10 - 30| + |10 - 30| + |30 - 30| + |50 - 30| + |50 - 30|}{5} = 16$$



# Dispersion

- **Variance and standard deviation** can also be used to measure variation.
- **The population variance  $\sigma^2$  is the mean of the square of all deviations from the mean.**
- Mathematically it is given as:  $\sigma^2 = \frac{\sum (x_i - \mu)^2}{N}$
- And **the population standard deviation  $\sigma$  is defined as the square root of variance:**
$$\sigma = \sqrt{\frac{\sum (x_i - \mu)^2}{N}}$$
- Notice that  $\sigma$  has the same unit as the data values  $x_i$  but  $\sigma^2$  does not.

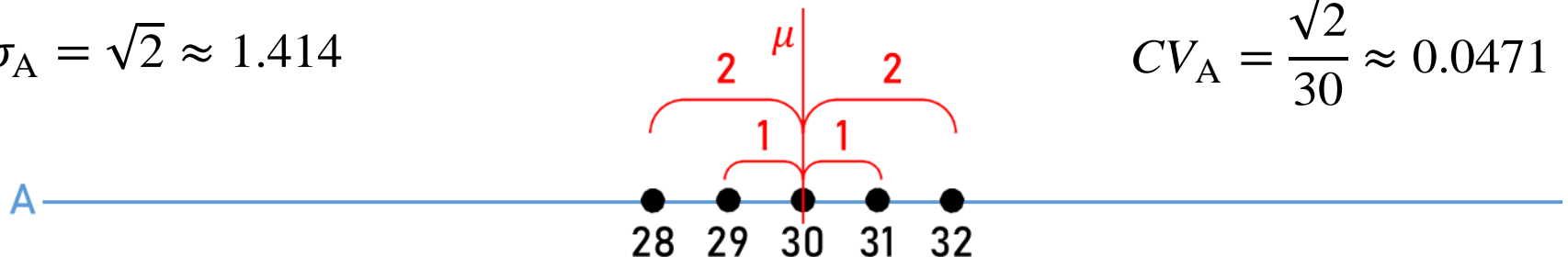
# Dispersion

- **Coefficient of variation (CV)** is a measure of relative importance.
- It does not depend on unit and can be used to make comparison even two samples differ in means or relate to different types of measurements.
- The coefficient of variation gives:  $\frac{\sigma}{\mu}$

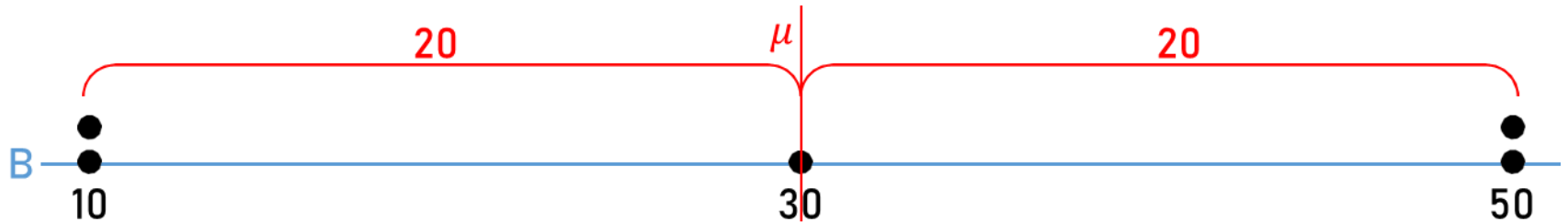
$$\sigma_A^2 = \frac{(28 - 30)^2 + (29 - 30)^2 + (30 - 30)^2 + (31 - 30)^2 + (32 - 30)^2}{5} = 2$$

$$\sigma_A = \sqrt{2} \approx 1.414$$

$$CV_A = \frac{\sqrt{2}}{30} \approx 0.0471$$



# Dispersion



$$\sigma_B^2 = \frac{(10 - 30)^2 + (10 - 30)^2 + (30 - 30)^2 + (50 - 30)^2 + (50 - 30)^2}{5} = 320$$

$$\sigma_B = \sqrt{320} \approx 17.89$$

$$CV_B = \frac{\sqrt{320}}{30} \approx 0.596$$

- We can see that the dispersion in B is much greater than that of A with either method.

# Standardization



- Due to the nature of the measurement or the unit used, different sets of data might have different scales.
- Since many machine learning algorithms require **feature scaling for optimal performance**, there is a need to **transform the data sets with a uniform scale**.
- This process is called **standardization**.
- Suppose a set of data  $x_1, x_2, \dots, x_N$  has mean  $\mu$  and standard deviation  $\sigma$ , we can subtract  $\mu$  from each data and then divide the difference by  $\sigma$ .

# Standardization



- In other words, the standardized data is given by the **standardization formula**:

$$z = \frac{x - \mu}{\sigma}$$

- This value of  $z$  is also called **standard score of the data**.
- This process is invertible.
- In other words, if we are given a standard score  $z$ , we can retrieve the original data value by:

$$x = \mu + z\sigma$$

if mean  $\mu$  and standard deviation  $\sigma$  is known to us.

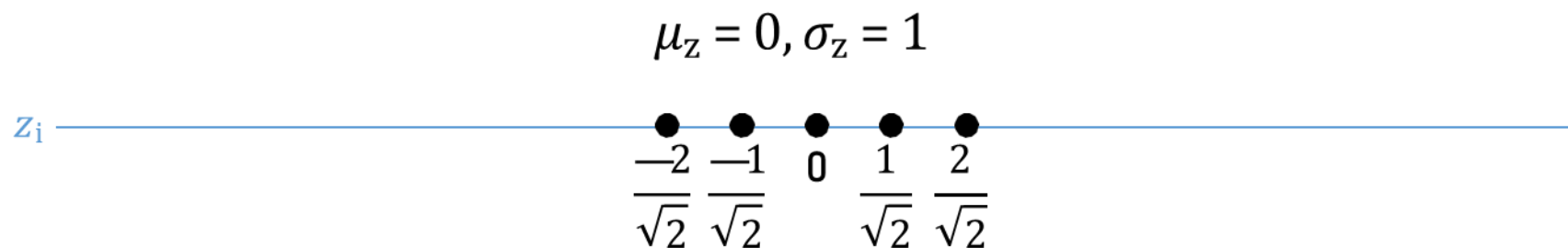
# Standardization

- Consider the set of data A in the previous example.
- We have evaluated  $\mu = 30$ ,  $\sigma = \sqrt{2}$ .
- Using standardization formula, we can evaluate the standardized data as follows:

Original value	Standard score
$x_1 = 28$	$z_1 = \frac{28 - 30}{\sqrt{2}} = -\frac{2}{\sqrt{2}} \approx -1.414$
$x_2 = 29$	$z_2 = \frac{29 - 30}{\sqrt{2}} = -\frac{1}{\sqrt{2}} \approx -0.707$
$x_3 = 30$	$z_3 = \frac{30 - 30}{\sqrt{2}} = 0$
$x_4 = 31$	$z_4 = \frac{31 - 30}{\sqrt{2}} = \frac{1}{\sqrt{2}} \approx 0.707$
$x_5 = 32$	$z_5 = \frac{32 - 30}{\sqrt{2}} = \frac{2}{\sqrt{2}} \approx 1.414$

# Standardization

- Notice that the standardized set of data  $z_1, z_2, \dots, z_N$  has mean 0 and standard deviation 1, despite of the mean, standard deviation or even the unit used in the original set of data.



# Sampling

- The population mean  $\mu$  and population variance  $\sigma^2$  introduced before can be evaluated from the whole set of population data.
- **However, if the population is too large, data collection and evaluation would be practically difficult.**
- Instead, we can **use a sample with relatively small size to estimate the population parameters.**

## Sampling

['sam-plin]

A process used in statistical analysis in which a predetermined number of observations are taken from a larger population.

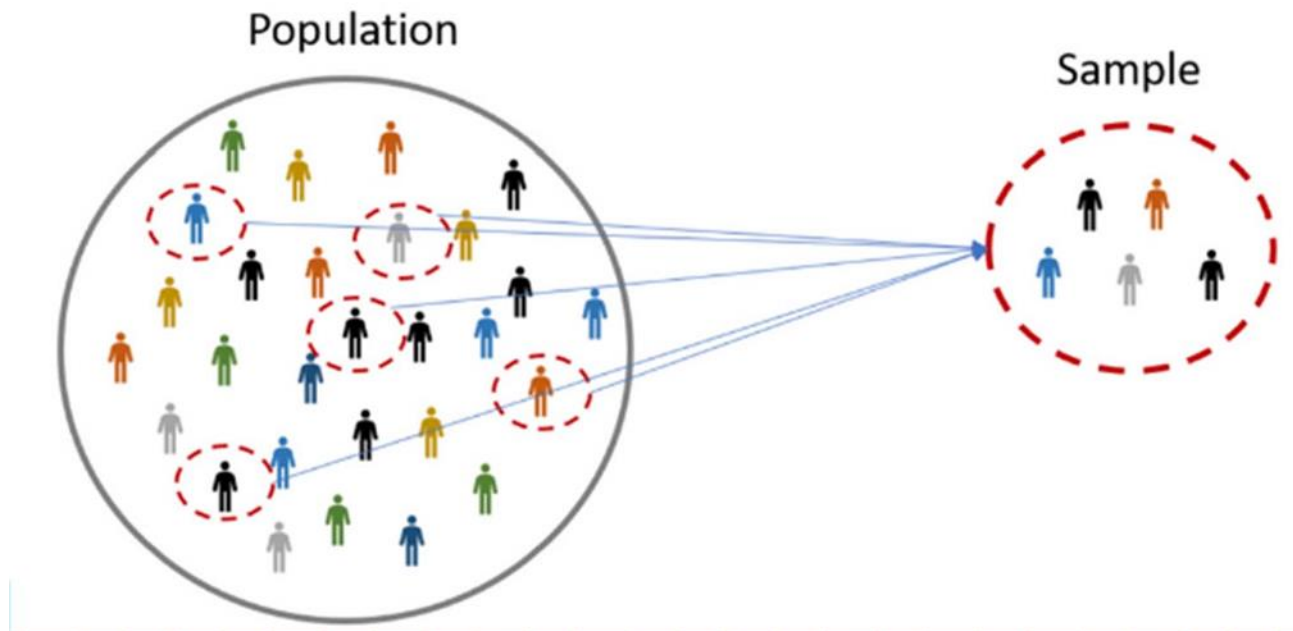


➤ This process is called **sampling**.



# Sampling

- Imagine if we ask to find the population mean height of all citizen in Hong Kong, we can randomly choose 100 citizens and measure their height.
- The estimation, however, will be **dependent on the random sample**.



# Sampling



- Consider a set of sample data  $x_1, x_2, \dots, x_n$  with sample size  $n$ .
- We can use sample mean  $\bar{x}$  to estimate the population mean.

$$\bar{x} = \frac{\sum x_i}{n}$$

- We can also use sample variance  $s^2$  to estimate population variance.

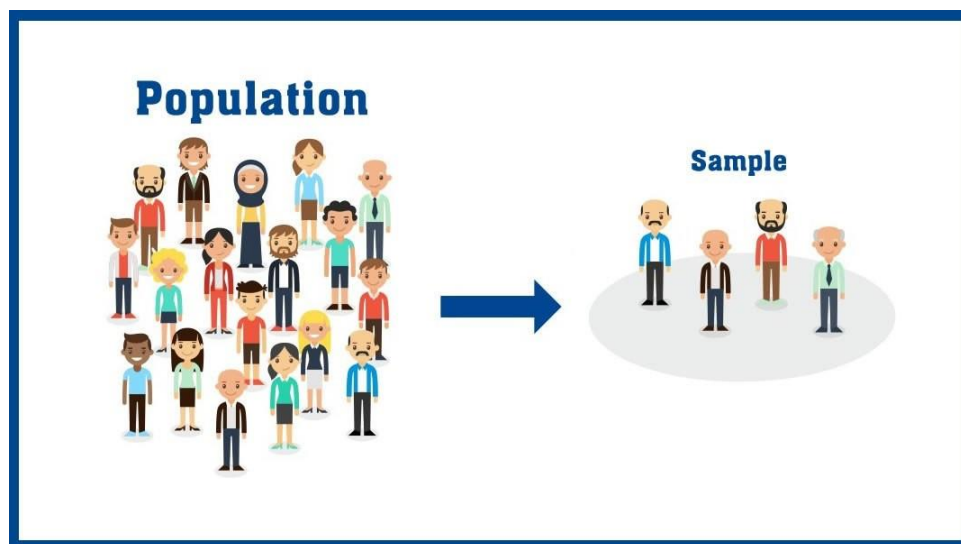
$$s^2 = \frac{\sum (x_i - \bar{x})^2}{n - 1}$$

- Sample standard deviation  $s$  is just the square root of sample variance.

$$s = \sqrt{\frac{\sum (x_i - \bar{x})^2}{n - 1}}$$

# Sampling

- Notice that **sample variance involves division by  $n - 1$** , while **population variance involves division by  $N$** .
- We can formulate this into  $N - ddof$  (or  $\Delta df$ ) where *ddof* refers to “**delta degree of freedom**”.
- For sample variance,  $ddof = 1$ ; for population variance,  $ddof = 0$ .



# Comparison operators in Python

	Operator	Description	Syntax
<a href="#">Python Equality Operators</a>	==	Equal to: True if both operands are equal	a == b
<a href="#">Inequality Operators</a>	!=	Not equal to: True if operands are not equal	a != b
<a href="#">Greater than Sign</a>	>	Greater than: True if the left operand is greater than the right	a > b
<a href="#">Less than Sign</a>	<	Less than: True if the left operand is less than the right	a < b
<a href="#">Greater than or Equal to Sign</a>	>=	Greater than or equal to: True if left operand is greater than or equal to the right	a >= b
<a href="#">Less than or Equal to Sign</a>	<=	Less than or equal to: True if left operand is less than or equal to the right	a <= b

# If-else statements in Python



## The usage of if...elif....else:

elif means “else if”

```
s = float(input("Enter the score of student: "))
```

```
if s >= 40 and s <= 100:
```

```
    print("Pass")
```

```
elif s >= 0 and s < 40:
```

```
    print("Failed")
```

```
else:
```

```
    print("Invalid input")
```

Try The above commands

# Exercise 1:



Covert hour in 24-hour format to 12-hour format using if-else statement

Steps:

Use “input” to ask for input (you learned it in Chapter 2) of hour in 24-hour format, **the input should be an integer.**

If the integer input  $n$  is between 12 and 24, then the hour should be  $n-12$  (e.g. Hour 20 means 8pm)

If the integer input is out of the range of 0 to 24, output “invalid input”

# For loop



```
m = int(input("Please enter an integer "))
```

```
for i in range(1, m ):
```

```
    print("The number i is = ", i)
```

```
    print("The number i time 2 = ", i*2)
```

```
for i in range(1, m + 1):
```

```
    print("The number i is = ", i)
```

```
    print("The number i time 2 = ", i*2)
```

Compare the difference of the above two for loops

# Exercise 2:



Calculate the sum of  $1+2+3+\dots+m$

Steps:

Use “input” to ask for input of number  $m$ , **the input should be an integer.**

Use for loop to add up the numbers from 1 to  $m$ . For example, if  $m$  equals to 5, do the calculation  $1+2+3+4+5$

Output the result



# While loop



```
m = 100
```

```
r = 1
```

```
while r <= m:
```

```
    print("The number of r = ", r)
```

```
    r = r + 1
```

As long as  $r$  is smaller than or equal to  $m$ , the number will be printed out and the value of  $r$  will be increased by 1.

# Exercise 3:



Calculate the sum of  $1 \times 2 \times 3 \dots \times m$

Steps:

Use “input” to ask for input of number  $m$ , **the input should be an integer.**

Use while loop to multiply the numbers from 1 to  $m$ . For example, if  $m$  equals to 5, do the calculation  $1 \times 2 \times 3 \times 4 \times 5$

Output the result

# Different formats of data



Text files (.txt)

CSV (comma separated values) files (.csv)

Excel files (.xls)

JSON (JavaScript Object Notation)

SQLite

We will learn about .txt .csv and .xls in this lecture

# Create text files (.txt)



```
content=""Hello Python
```

```
Trying to output text in a file called file.txt
```

```
Welcome""
```

```
f=open(r'C:\Users\User user\Desktop\filename.txt', 'w', encoding='utf-8')
```

```
f.write(content)
```

```
f.close()
```

## Be careful about the path of the file

# Read text files (.txt)



```
with open(r'C:\Users\User user\Desktop\filehello.txt', 'r', encoding='utf-8') as f:  
    output_str=f.read()  
    print(output_str) # Hello
```

# Create CSV files (.csv)



```
import csv
```

```
csvtable = [  
    ['Name', 'Height', 'Weight'],  
    ['Peter', 170, 65],  
    ['David', 183, 78],  
]
```

```
with open(r'C:\Users\User user\Desktop\test2.csv', 'w', newline='') as csvfile:  
    writer = csv.writer(csvfile)  
    writer.writerows(csvtable)
```

The indents in the above lines are very important. Use the key “TAB” to indent.

# Read CSV files (.csv)



```
import csv
```

```
with open(r'C:\Users\User user\Desktop\test2.csv', newline='') as csvfile:
```

```
    rows = csv.reader(csvfile)
```

```
    for row in rows:
```

```
        print(row)
```

# Create Excel files (.xls)



```
import openpyxl
```

```
# Create a workbook
```

```
workbook=openpyxl.Workbook()
```

```
# Create a worksheet
```

```
sheet = workbook.worksheets[0]
```

```
# Store data in cells
```

```
sheet['A1'] = 'Class 1A'
```



# Create Excel files (.xls)



```
# write data as strings
listtitle=['Seat number', 'Name', 'Chinese', 'English', 'Mathematics']
sheet.append(listtitle)
listdatas=[[1, 'Peter', 65, 62, 40],
            [2, 'Mary', 85, 90, 87],
            [3, 'John', 92, 90, 95]]
for listdata in listdatas:
    sheet.append(listdata)

# Save the file
workbook.save(r'C:\Users\User user\Desktop\Class_info.xlsx')
```

# Read Excel files (.xls)



```
import openpyxl
# read the Excel file
workbook = openpyxl.load_workbook(r'C:\Users\User user\Desktop\Class_info.xlsx')

# get the first workbook
sheet = workbook.worksheets[0]

# Get the cell
print(sheet['A1'].value)
```

# Read Excel files (.xls)



```
# Get the total number of rows and column
print(sheet.max_row, sheet.max_column)

# Show all the data in cells
for i in range(1, sheet.max_row+1):
    for j in range(1, sheet.max_column+1):
        print(sheet.cell(row=i, column=j).value, end=" ")
    print()
```

# Simplest way to plot a graph

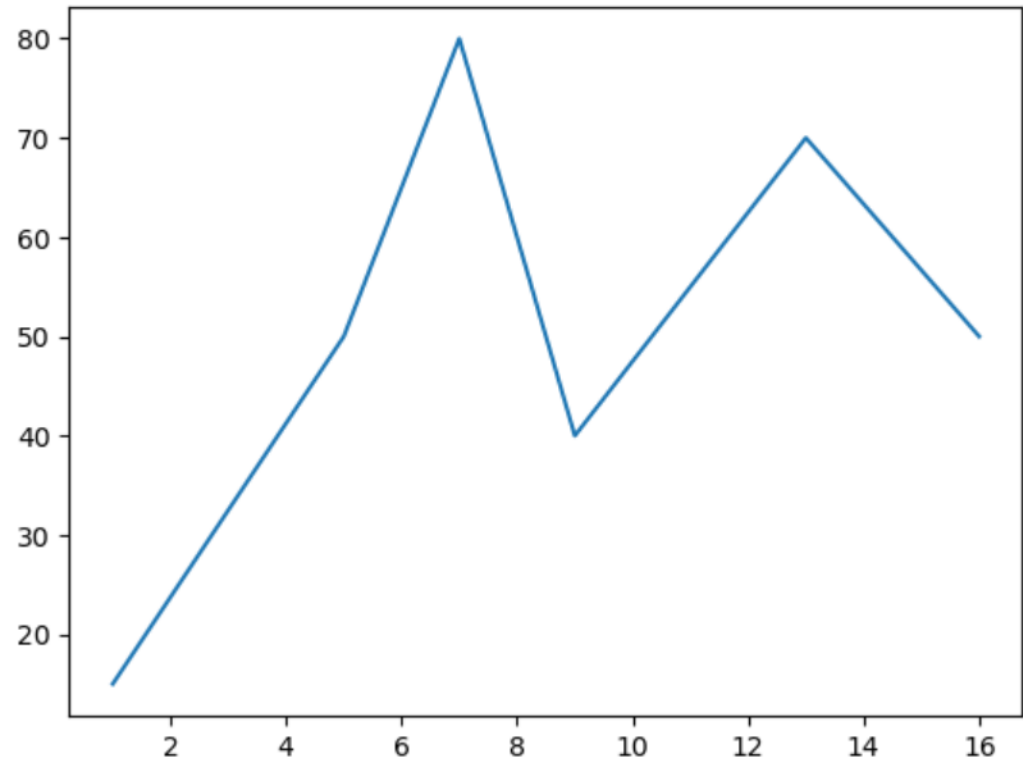
```
import matplotlib.pyplot as plt
```

```
listx = [1,5,7,9,13,16]
```

```
listy = [15,50,80,40,70,50]
```

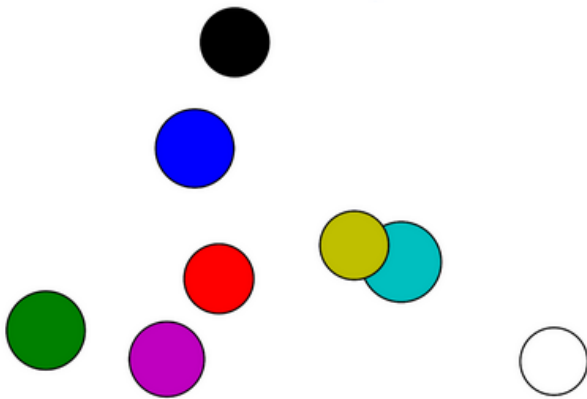
```
plt.plot(listx, listy)
```

```
plt.show()
```



# Different color codes in plots


















Base colors in matplotlib



Let's start simple: there are some base colors in matplotlib, as can be seen from the left plot. But they're not very attractive, at least for me. These are named:

- 'b' - blue
- 'c' - cyan
- 'g' - green
- 'k' - black
- 'm' - magenta
- 'r' - red
- 'w' - white
- 'y' - yellow

# Different markers in plots

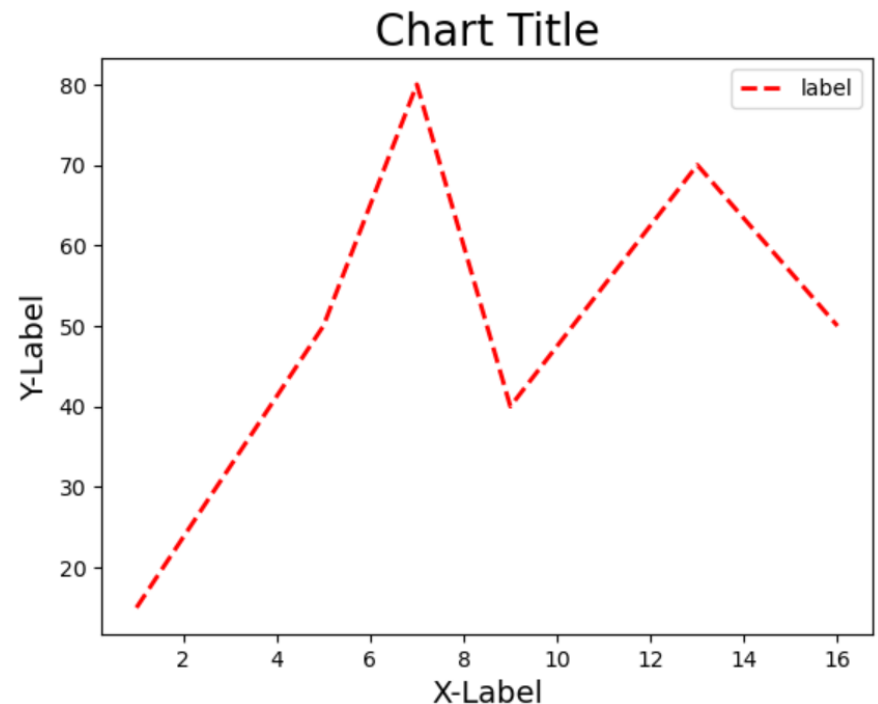
marker	symbol	description
"."		point
"."		pixel
"o"		circle
"v"		triangle_down
"^"		triangle_up
"<"		triangle_left
">"		triangle_right
"1"		tri_down
"2"		tri_up
"3"		tri_left
"4"		tri_right
"8"		octagon
"s"		square
"p"		pentagon
"P"		plus (filled)
"*"		star
"h"		hexagon1

# Try again with different styles

```
import matplotlib.pyplot as plt
listx = [1,5,7,9,13,16]
listy = [15,50,80,40,70,50]
plt.plot(listx, listy, color="red", lw="2.0", ls="--", label="label")
plt.legend()
plt.title("Chart Title", fontsize=20)
plt.xlabel("X-Label", fontsize=14)
plt.ylabel("Y-Label", fontsize=14)
plt.show()
```

Also try to set the range of the plot:

```
plt.xlim(0, 20)
plt.ylim(0, 100)
```



# Analyze data of different cars



**Read the csv file “car.csv” and do some simple data analysis**

```
import pandas as pd
import matplotlib.pyplot as plt

# Load the data
df = pd.read_csv(r'C:\Users\User user\Desktop\cars.csv')

# Display the first few rows
print("First ten rows of the dataset:")
print(df.head(10))

# Identify missing values
print("\nMissing values in each column:")
print(df.isnull().sum())
```



# Analyze data of different cars



# Calculate the mean

```
mean_horsepower = df['Horsepower'].mean()  
print("The average horsepower is ", mean_horsepower)
```

# Calculate the median

```
median_city_mpg = df['City mpg'].median()  
print("The median of City mpg is ", median_city_mpg)
```

# Calculate the standard deviation

```
std_highway_mpg = df['Highway mpg'].std()  
print("The standard deviation of highway mpg is ", std_highway_mpg)
```

# Analyze data of different cars



```
# Calculate the minimum and maximum
min_model_year = df['Model Year'].min()
max_model_year = df['Model Year'].max()
print("The maximum model year is ", max_model_year)
print("The minimum model year is ", min_model_year)

# Calculate the mode
mode_Length = df['Length'].mode()[0]
print("The mode of length of cars is ", mode_Length))
```

# Exercise 4:



Use Python to calculate:

1. The average width of cars
2. The median of the horsepower of the cars
3. The standard deviation of the torque
4. The mode of Highway mpg

# Usage of “groupby”



The “groupby” function in pandas is a powerful tool for grouping data based on one or more columns and then performing operations on each group. This is particularly useful for data aggregation, transformation, and analysis.

Try below:

```
mean_horsepower_by_year=df.groupby('Year')['Horsepower'].mean()  
print(mean_horsepower_by_year)
```

# Plotting a bar chart



```
# Convert the result to lists for plotting
years = mean_horsepower_by_year.index.tolist()
mean_horsepower = mean_horsepower_by_year.tolist()

# Plot the bar chart
plt.figure(figsize=(10, 6))
plt.bar(years, mean_horsepower, color='skyblue')
plt.title('Mean Horsepower by Model Year')
plt.xlabel('Model Year')
plt.ylabel('Mean Horsepower')
plt.xticks(ticks=years, labels=[int(year) for year in years], rotation=45)
plt.show()
```

# Assignment

The assignment is about data analytics.

You can treat it as an individual project.

Please select any dataset from the governments open data platform.

(<https://data.gov.hk/en/>)



Download Queue Bookmarks Text Size 繁 | 簡

Datasets Providers Help Developer Center Community

## Explore Open Data on DATA.GOV.HK

Innovate with Open Data. Drive change through Information.

Learn More >



Bookmarks



Download Queue



Feedback

# Assignment



After selecting a dataset, perform analysis using the Python skills taught in the class. You are encouraged to do more with skills not mentioned in the notes.

Save your work in the form of `.ipynb` format (or `.py` format)

Submit your **`.ipynb` file (or `.py` file)** and the **data file in `.csv` format** to [kelvinsiu@thei.edu.hk](mailto:kelvinsiu@thei.edu.hk) by **25 Nov 2024**, remember to state your full name and student number in your email. (Please make sure your code can be run with no problems.)

The baseline score is 70, you will get extra marks if you can do more analysis using skills not mentioned in the notes.

# Checklist

- Can you:
  1. Explain what descriptive statistics is?
  2. Calculate arithmetic mean, median and mode?
  3. Suggest the dispersion or variability in a set of data?
  4. Standardize the data by the standardization formulae?
  5. Conduct sampling process?

