

We'll be starting shortly!

To help us run the workshop smoothly, please kindly:

- Switch off screen sharing and mute your microphone
- Submit all questions using the Q&A function
- If you have an urgent request, please use the “Raise Hand” function

Thank you!

{<coding:lab>}





{<coding:lab>}

Data Science 101

Session 3

About Coding Lab

- Founded by an MIT Graduate who worked in Silicon Valley
- Global Tech advisory team based in New York, Japan and Singapore
- We have Campuses in Japan, Australia and Singapore
- We offer coding classes starting from age 4 to adulthood

{<coding:lab>}



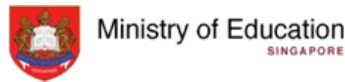
Features and Partners



{<coding:lab>}



Features and Partners



THE STRAITS TIMES



And many more...

{<coding:lab>}



Meet our Students



Sarah, 18
Honourable
Mention, NOI 2018



Team ajdisjd
1st Place, iCode
2019



Elijah, 14
Youngest Medalist,
NOI 2019



Surya, 14
Created a stock
rating algorithm



Team Trendsetter
Best Presentation
Award

More schools we have taught at:



EtonHouse
International School • Pre-School

MMI Modern
Montessori
International
SCHOOL
A MEMBER of the Modern Montessori International Group

{<oding:lab>}



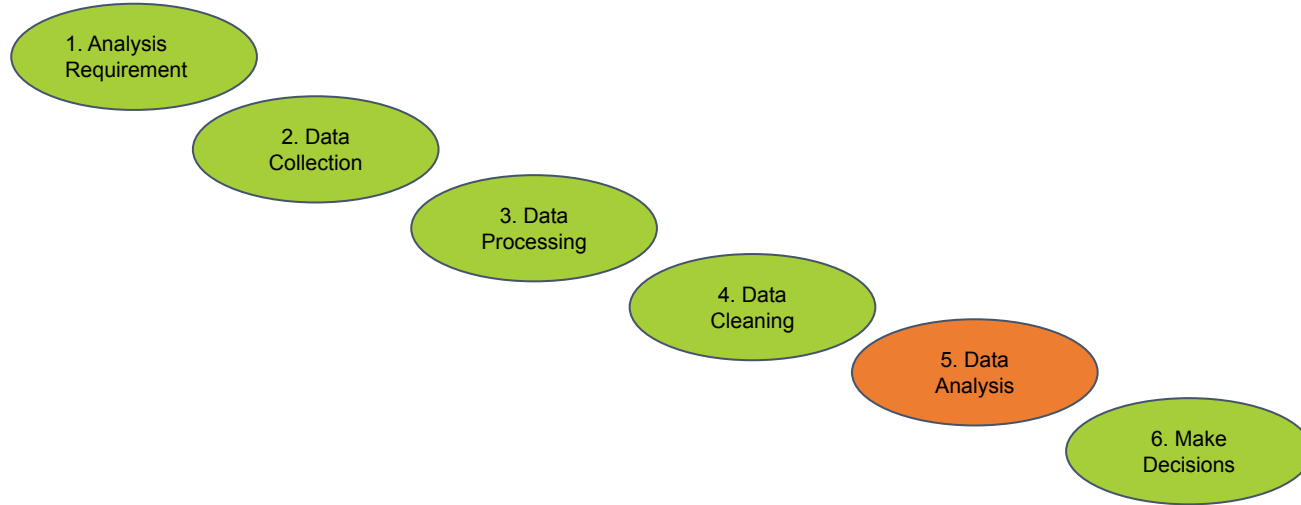
Let's get started - Session Overview

- Introduction to SciPy
- Statistical testing with Scipy
- Linear Regression

{<coding:lab>}



Data Analytics Process Overview



{<coding:lab>}



Introduction to Scipy



{<coding:lab>}

What is Scipy?

- An open source Python-based library
- Used in mathematics, scientific computing, Engineering, and technical computing
- There are many built in functions in Scipy
 - We will be focusing on scipy.stats
- Documentation: <https://docs.scipy.org/doc/>

{<coding:lab>}



Scipy.stats()

- Contains a large number of probability distributions and statistical functions
- Applying what we have learnt, we are going to use Scipy to conduct statistical tests for us

{<coding:lab>}



Hypothesis Testing with Scipy



{<coding:lab>}

Z-test in Scipy

- Scipy provides a method named `scipy.stats.zscore` that calculates the z score of each value in the sample, relative to the sample mean and standard deviation
 - However it does not perform the entire z-test
- For convenience, we will be using a module that has dependencies on `scipy.stats`
 - Statsmodels
 - Documentation: <https://www.statsmodels.org/>

{<coding:lab>}



When To Use a z-test

- Your population standard deviation is known
- Your data should be normally distributed
 - For sample sizes >30 , this does not always matter
 - Central Limit Theorem
- Your data is randomly selected from a population
 - Where there is an equal chance of selecting each item
- If there are multiple samples, the sample sizes should be roughly equal

{<coding:lab>}



One Sample z-test Scenario

- A scientist has collected the the shell size of 30 seashells and wants to compare the mean value with the given mean of 0.5
- We can use a one sample z-test to verify this

{<coding:lab>}



One Sample z-test: Sample Code

```
1  from statsmodels.stats.weightstats import ztest
2  import numpy as np
3
4  a = np.array([ 0.7972,  0.0767,  0.4383,  0.7866,
5  0.8091, 0.1954,  0.6307,  0.6599,  0.1065,  0.0508,
6  0.7971,  0.0768,  0.4382,  0.7867,  0.809, 0.1955,
7  0.6306,  0.66,  0.1066,  0.0507, 0.7973,  0.0766,
8  0.4384,  0.7867,  0.8092, 0.1953,  0.6308,  0.658,
9  0.1066,  0.0507])
10 mean = np.mean(a) #0.4550667
11 zset, pval = ztest(a, x2=None, value=0.5)
12 pval
```

{<coding:lab>}



One Sample z-test Hypothesis

- Our hypotheses for testing goes as follows
 - Null Hypothesis (H_0): The mean of the shell size is 0.5
 - Alternate Hypothesis (H_1): The mean of the shell size is not 0.5

{<coding:lab>}



How to Draw Conclusion

- We then look at the p-value from the test
 - A p-value of less than 0.05 will indicate that our data falls into the tail end extremes of the normal curve
 - Hence it is extremely unlikely for our null hypothesis to be true
 - A p-value of over 0.05 indicates that there is no evidence to say that the null hypothesis is not true

```
14 if pval < 0.05:  
15     print("reject null hypothesis")  
16 else:  
17     print("accept null hypothesis")
```

{<coding:lab>}



Conclusion on Shell Size

- Since our p-value was 0.425, this is greater than 0.05, hence this indicates that there is not enough evidence to conclude that the mean of the shell sizes is not the predicted value of 0.5
 - We do not reject the null hypothesis

{<coding:lab>}



Shell Size (Demo/Practice - 1)

- Using the given shell data, conduct a one-sample z-test
 - Raw data can be found in shelldata1.txt

{<coding:lab}



Checkpoint 1

- Every student must be able to:
 - Perform a one-sample Z-test
- For students who are waiting, try the following:
 - Read up on Z-test
 - Explore the scipy library

{<coding:lab>}



Two Sample z-test

- In two sample z-test, we are checking two independent data groups
 - Deciding whether sample mean of two group is equal or not
- A scientist has collected the the shell size of 15 seashells each from 2 beaches and wants to tell if the mean size are the same

{<coding:lab>}



Two Sample z-test Hypothesis

- Hypothesis:
 - H_0 : The mean shell size in both groups are the same
 - H_1 : The mean shell size in both groups are not the same

{<coding:lab}



Two Sample z-test (Demo/Practice - 41)

```
1  from statsmodels.stats.weightstats import ztest
2  import numpy as np
3
4  a = np.array([ 0.7972,  0.0767,  0.4383,  0.7866,
5  0.8091, 0.1954,  0.6307,  0.6599,  0.1065,  0.0508,
6  0.7971,  0.0768,  0.4382,  0.7867,  0.809])
7
8  b = np.array([0.1955, 0.6306, 0.66, 0.1066, 0.0507,
9  0.7973,  0.0766,  0.4384,  0.7867,  0.8092, 0.1953,
10 0.6308,  0.658,  0.1066,  0.0507])
11
12 mean = np.mean(a) #0.4550667
13 zset, pval = ztest(a, x2=b, alternative='two-sided')
14 pval
```

{<coding:lab>}



Conclusion on Two Sample Shell Size

- Since our p-value was 0.457, this is greater than 0.05, hence this indicates that there is not enough evidence to conclude that the mean of the shell sizes from the two groups are not the same
 - We do not reject the null hypothesis

{<coding:lab}



Shell Size (Demo/Practice - 2)

- Using the given shell data, conduct a two-sample z-test
 - Raw data can be found in shelldata2.txt

{<coding:lab}



Checkpoint 2

- Every student must be able to:
 - Perform a two-sample Z-test
- For students who are waiting, try the following:
 - Read up on Z-test
 - Explore the scipy library

{<coding:lab>}



T-test in Scipy

- `stats.ttest_1samp()` calculates for us whether the sample mean is statistically different from a known or hypothesised population mean

{<coding:lab>}



When To Use a t-test

- When we want to determine if there is a significant difference between the means of two groups which may be related in certain features
- It is mostly used when the dataset would follow a normal distribution but may have unknown variances
 - E.g. Dataset from recording the outcome of 100 coin flips
- Also works for small sample sizes

{<coding:lab>}



T-test Scenario

- Let's say the mean age in a population is 27
- Given the ages of 14 people, we want to find out the deviation from the population mean
- We will use the one sample t-test to determine whether or not the sample mean is statistically different from our hypothesised mean of 27

{<coding:lab>}



T-test in Scipy: Sample Code

```
1 from scipy.stats import ttest_1samp
2 import numpy as np
3 ages = [32,34,29,29,22,39,38,37,38,36,30,26,22,22]
4
5 ages_mean = np.mean(ages)
6
7 tset, pval = ttest_1samp(ages, 27)
8 pval
```

{<coding:lab>}



T-test Conclusion

- Since the p-value is 0.0327, it is less than 0.05 and we can say that based on our observed example, it is unlikely that the mean age of the population is 27
 - We can then reject our null hypothesis that says mean age = 27

{<coding:lab>}



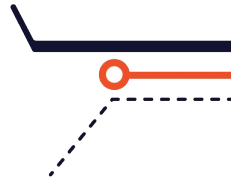
Average Age (Demo/Practice - 3)

- Using the given age data, conduct a one-sample t-test
 - Raw data can be found in ages.txt
- How likely is it for the mean age to be 27?
 - What about 30?

{<coding:lab>}



Checkpoint 3



- Every student must be able to:
 - Perform a one-sample t-test using scipy
- For students who are waiting, try the following:
 - Read up on t-test
 - Explore the scipy library



{<coding:lab>}



Other Kinds of t-test

- What we did in our example above was a one sample t-test
- There are also other kinds of t-test that you can read up on
 - Two sample t-test
 - Paired t-test
- The t-test works well when comparing data from two groups
 - But what if we want to compare more than two groups at the same time?

{<coding:lab>}



Anova Test in Scipy

- To perform one way ANOVA test, `stats.f_oneway()` is used for 2 or more sample groups
- The built in function returns the F-value of the test
 - E.g. If the shell measurements are collected from 5 locations and we want to test if they share the same sample mean
 - It is possible for the groups to have different sizes

{<coding:lab>}



Anova Test Example

- We have shell measurements collected from 5 locations
 - We want to test if they share the same sample mean
 - It is possible for the groups to have different sizes
- Hypothesis
 - H_0 : The shell sizes in the 5 locations have the same mean
 - H_1 : The shell sizes in the 5 locations do not have the same mean

{<coding:lab>}



Anova Test Example

- Using `f_oneway` from `scipy.stats`, we can conduct an ANOVA Test

```
1 import scipy.stats as stats
2 a = [0.0571, 0.0813, 0.0831, 0.0976, 0.0817, 0.0859, 0.0735, 0.0659, 0.0923, 0.0836]
3 b = [0.0873, 0.0662, 0.0672, 0.0819, 0.0749, 0.0649, 0.0835, 0.0725]
4 c = [0.0974, 0.1352, 0.0817, 0.1016, 0.0968, 0.1064, 0.105]
5 d = [0.1033, 0.0915, 0.0781, 0.0685, 0.0677, 0.0697, 0.0764, 0.0689]
6 e = [0.0703, 0.1026, 0.0956, 0.0973, 0.1039, 0.1045]
7 stats.f_oneway(a,b,c,d,e)
```

{<coding:lab>}



Anova Test Conclusion

- The p-value from our test was 0.000281
- Hence we conclude that it is highly unlikely for the samples to have the same mean
 - We reject the null hypothesis

{<coding:lab}



Anova Test in Scipy (Demo/Practice - 4)

- Using the raw data found in shelldata3.txt, conduct an ANOVA test

{<coding:lab}



Types of ANOVA Tests

- One-way ANOVA Test
- Two-way ANOVA Test
 - Extension of one-way ANOVA test
 - Used when we have 2 independent variable and 2 or more sample groups
- ANOVA Test is also known as F-test

{<coding:lab}



Checkpoint 4

- Every student must be able to:
 - Perform an Anova test using scipy
- For students who are waiting, try the following:
 - Read up on anova
 - Explore the scipy library

{<coding:lab>}



Categorical Testing

- Recall the 3 kinds of tests that we covered earlier
 - Z-test, t-test and ANOVA test
- These tests all apply to quantitative data
- How do we deal with categorical data?

{<coding:lab}



Chi-square Test in Scipy

- Chi-square test is used to test categorical datasets
- `Stats.chisquare()` performs a one way chi square test and returns the statistics

{<coding:lab>}



Chi-square Test Scenario


- Say we want to find out if a particular dice is fair
 - In a fair dice, each number has exactly a $\frac{1}{6}$ chance of showing up
- We collect a total of 160 dice rolls over 4 sessions of rolling the same dice 40 times
- Hypothesis
 - H_0 : The dice is fair
 - H_1 : The dice is not fair

{<coding:lab>}



Sample Data from 160 Dice Rolls

- The table below shows the results of all our dice rolls

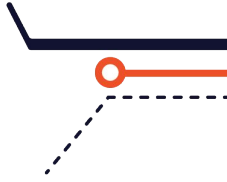


	set1	set2	set3	set4
1	7	5	6	8
2	9	6	5	3
3	5	5	8	4
4	5	11	7	10
5	6	10	7	6
6	8	3	7	9

{<coding:lab>}



Chi-square Test Scenario



- With 160 rolls, we expect each number to come up approximately 26-27 times
- We can see that this happened for some numbers, but not others
 - The numbers 1 and 6 came up 26 and 27 times respectively
 - The number 3 only came up 22 times
 - The number 4 came up 33 times



{<coding:lab>}



Chi-square Test in Scipy

- `Stats.chisquare()` performs a one way chi square test and returns the statistics
- The test is only valid if there are at least 5 observed frequencies
 - We have 6

{<coding:lab}



Chi-square Test in Scipy: Sample Code

```
1  import numpy as np
2  from scipy.stats import chi2_contingency
3  num1 = [7, 5, 6, 8]
4  num2 = [9, 6, 5, 3]
5  num3 = [5, 5, 8, 4]
6  num4 = [5, 11, 7, 10]
7  num5 = [6, 10, 7, 6]
8  num6 = [8, 3, 7, 9]
9  dice = np.array([num1, num2, num3, num4, num5, num6])
10
11  chi2_stat, pval, dof, ex = chi2_contingency(dice)
12  pval
```

{<coding:lab>}



Chi-square Test Conclusion

- Since the p-value is 0.604, we conclude that there is not enough evidence to suggest that the dice is not a fair dice
 - We do not reject the null hypothesis

{<coding:lab}



Chi-Square Test in Scipy

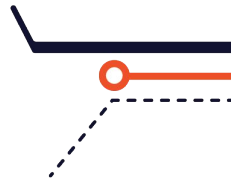
(Demo/Practice - 5)

- Using the raw data found in diceroll.txt, conduct a chi-square test

{<coding:lab>}



Checkpoint 5



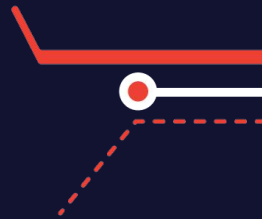
- Every student must be able to:
 - Perform a chi-square test using scipy
- For students who are waiting, try the following:
 - Read up on chi-square test
 - Explore the scipy library



{<coding:lab>}



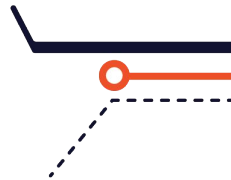
Linear Regression with Scipy



{<coding:lab>}



Linear Regression with Scipy



- Using `scipy.stats.linregress`
- We will use our height vs shoe size dataset from the previous session
 - Raw data can be found in `heightshoesize.txt`



{<coding:lab>}



Shoe Size Example

- We want to find out if there is a relationship between a person's height and their shoe size
 - Height is our explanatory variable
 - Shoe size is our response variable

{<coding:lab}



Shoe Size Linear Regression

- Sample Code

```
1 from scipy.stats import linregress
2 height = [157, 163, 179, 173, 174, 175, 175, 168, 180, 183]
3 shoesize = [4, 7, 10, 9, 9.5, 10.5, 6, 7.5, 11, 11.5]
4
5 linregress(height, shoesize)
```

{<coding:lab>}



Program Output

- Program Output
 - `LinregressResult(slope=0.2591882947221738, intercept=-36.16181849851941, rvalue=0.8579135245156916, pvalue=0.0014967544132328656, stderr=0.05488023917355253)`

{<coding:lab>}



How To Read Linear Regression Output

- Slope and intercept form our equation $y = mx + c$
 - In this case, shoe size = $0.259 * \text{height} - 36.1618$
- R-value shows how strongly your data is correlated
 - A value nearer to -1 indicates strong negative correlation
 - A value nearer to 1 indicates strong positive correlation
 - A value close to 0 indicates weak correlation

{<coding:lab>}



Linear Regression P-Value

- The p-value for each term tests the null hypothesis that the coefficient is equal to zero (no effect)
- A low p-value (< 0.05) indicates that you can reject the null hypothesis
 - In other words, a predictor that has a low p-value is likely to be a meaningful addition to your model because changes in the predictor's value are related to changes in the response variable

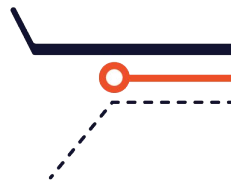
{<coding:lab>}



Chi-Square Test in Scipy

(Demo/Practice - 6)

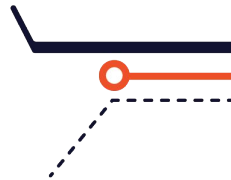
- Using the raw data found in heightshoesize.txt, create a linear regression model



{<coding:lab>}



Checkpoint 6



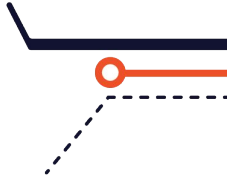
- Every student must be able to:
 - Create a simple linear regression model
- For students who are waiting, try the following:
 - Read up on linear regression
 - Explore the scipy library



{<coding:lab>}



Other Types of Regression



- In some cases, a simple linear regression may not be the best fit model
- There are other types of regression models you can explore
 - Multiple Linear Regression
 - Polynomial Regression



{<coding:lab>}



Summary

- Various statistical test in scipy
 - Z-test
 - T-test
 - ANOVA-test
- Categorical Testing in scipy
 - Chi-square Test
- Simple Linear Regression in scipy

{<coding:lab}



Your Feedback Matters!



LIKE and follow us for
more resources and tips!



@codinglabasia

{<coding:lab>}

