

Topic 1 Weekly Exercise

Each question have only one correct answer, you can try multiple times before the deadline.

Stimulus

1

Multiple Choice 1 point Q1

Practice 1

- Input Your Last Name AAA
- Print Mr/Ms AAA

(AAA is a placeholder and you need to change it to your last name)

-----Below are Stimulus Part-----

Suppose Mr. Chan is enrolled in this course, and
The following lines of code are provided, we
want to display the result as below

Mr. Chan

-----Below are codelines-----

```
lastname='Chan'  
title=(1)  
nameprint=(2)
```

What shall be filled in (1)

- ☒ 'Mr. '
- ☐ 'Mr.'
- ☐ 'Mr '
- ☐ 'Mr'

2

Multiple Choice 1 point Q2

To print title and name, what shall be filled in (2)?

- ☐ lastname+title
- ☒ title+lastname
- ☐ lastname
- ☐ title

Stimulus

Practice 2

Instructions

Check if Formulas and Graph are correctly displayed

- (1) Find 95% VaR: $P(X > VaR) = 0.95$, here Wealth X is normal with mean = 1M and std = 0.5M (M stands for one million)
- (2) Given a statistic \bar{X} , and the standardized statistic \hat{z} , in upper tail hypothesis testing, find p-value = $P(\hat{z} > 2.3)$

-----Below are
codelines-----

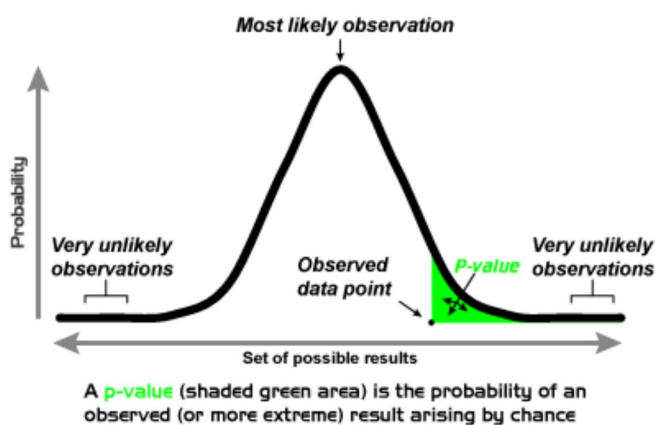
VaR=(i)
VaR
pv=(ii)
pv

-----Below are information for
Reference Only-----

$$H_0 : \mu \leq 100$$

$$H_1 : \mu > 100$$

$$\hat{z} = \frac{\bar{x} - \mu}{\frac{\sigma}{\sqrt{n}}} = 2.3$$



To understand what each of the functions work under `scipy.stats.norm`, you can refer to this page: <https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.norm.html>

3

Multiple Choice 1 point

What shall be filled in (i) to achieve (1)?

- ☐ `scipy.stats.norm.ppf(0.95,1000000,500000)`
- ☒ `scipy.stats.norm.ppf(0.05,1000000,500000)`
- ☐ `scipy.stats.norm.ppf(0.95,500000,1000000)`
- ☐ `scipy.stats.norm.ppf(0.05,500000,1000000)`

4

Multiple Choice 1 point

What shall be filled in (ii) to achieve (2)?

- ☒ `1-scipy.stats.norm.cdf(2.3,0,1)`
- ☐ `scipy.stats.norm.cdf(2.3,0,1)`
- ☐ `scipy.stats.norm.cdf(2.3,1,0)`
- ☐ `1-scipy.stats.norm.cdf(2.3,1,0)`

Stimulus

Practice 3

- We will use random number generator to generate daily changes of stock price(252 days). For simplicity, we assume that the daily change follows a standard normal distribution.
- Apply cumulative sum method of numpy array to compute accumulative sum of daily change, which is used to mimic stock price.
- Plot stock price.

-----Below are codelines-----

```
import numpy as np
import matplotlib.pyplot as plt
prc_chg=(1)
stockprice=(2)
plt.figure(figsize=(20,10))
(3)
plt.show()
```

5

Multiple Choice 1 point

What shall be filled in (1)?

- ☒ `np.random.normal(0,1,252)`
- ☐ `np.random.normal(1,0,252)`
- ☐ `np.random.normal(252,0,1)`
- ☐ `np.random.normal(252,1,0)`

6

Multiple Choice 1 point

What shall be filled in (2)?

- ☒ `np.cumsum(prc_chg)`
- ☐ `np.cumprod(prc_chg)`
- ☐ `np.cumprod(1+prc_chg)`
- ☐ `np.cumsum(1+prc_chg)`

7

Multiple Choice 1 point

Which of the code line below plots the stock prices in (3)?

- ☒ `plt.plot(stockprice,color='purple')`
- ☐ `plt.figure(stockprice,color='purple')`
- ☐ `plt.show(stockprice,color='purple')`
- ☐ `plot(stockprice,color='purple')`

