INFO 7390

Advances in Data Sciences and Architecture Assignment 2

Student Name: ___________Zixuan Yu____________________________
Professor: Nik Bear Brown

Due: Sunday May 27, 2018

Q1 (5 Points) Given a normal distribution with a mean to 33, a standard deviation of 11, and the sample size to 100. What is the probability of finding a value:

a. less than 11 (2 points)

p ((X-μ) /σ/√n)

Answer1: P(x<11) = (11-33)/11/√100=-20

P(z=-20) = 2.7536241186061556e-89

b. greater than 55 (2 points)

Answer2: P(x>55) = (55-33)/11/√100=20

P = 1-0.9772=2.7536241186061556e-89

c. less than 11 or greater than 55 (1 point)

Answer3: P (x<11 or x>55)

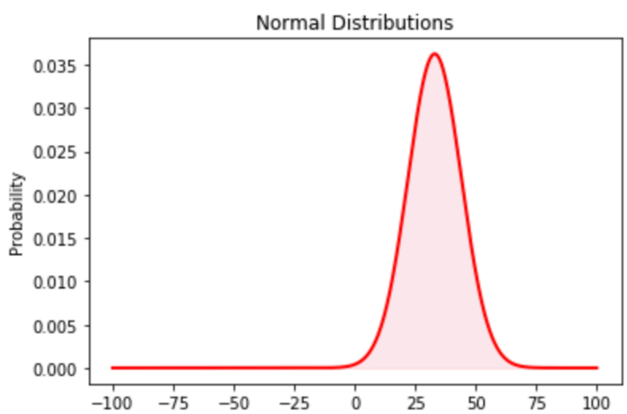
P(z=20) +P(z=-20) =5.507248237212311e-89

Show the calculation as done by hand.

Q2 (5 Points) Write python code to plot Q1 and calculate Q1.

```python
import matplotlib.pyplot as plt
import numpy as np
import math
from scipy import stats
domain = np.arange(-100, 100, 0.1)
values = stats.norm(33, 11).pdf(domain)
plt.plot(domain, values, color='r', linewidth=2)
plt.fill_between(domain, 0, values, color='#ffb6c1', alpha=0.3)
plt.ylabel("Probability")
plt.title("Normal Distributions")
plt.show()
def prob(x,u,stu,n):
    return stats.norm.cdf((x-u)/(stu/math.sqrt(n)))
def prob1(x,u,stu,n):
    return stats.norm.sf((x-u)/(stu/math.sqrt(n)))
print("prob1",prob(11,33,11,100))
print("prob2",prob1(55,33,11,100))
print("prob3",2*prob(11,33,11,100))
```


```python
plt.plot(domain, values, color='r', linewidth=2)
plt.fill_between(domain, 0, values, color='#ffb6c1', alpha=0.3)
plt.ylabel("Probability")
plt.title("Normal Distributions")
plt.show()
def prob(x,u,stu,n):
    return stats.norm.cdf((x-u)/(stu/math.sqrt(n)))
def prob1(x,u,stu,n):
    return stats.norm.sf((x-u)/(stu/math.sqrt(n)))
print("prob1",prob(11,33,11,100))
print("prob2",prob1(55,33,11,100))
print("prob3",2*prob(11,33,11,100))
```


```
prob1 2.7536241186061556e-89
prob2 2.7536241186061556e-89
prob3 5.507248237212311e-89
```

Q3 (5 Points) Given a normal distribution with a mean to 33, a standard deviation of 11, and the sample size to 1000. What is the probability of finding a value:

a. less than 11 (2 points)

p $((X-\mu)/\sigma/\sqrt{n})$

Answer1: $P(x<11) = (11-33)/11/\sqrt{1000}=-63$

$P(z=-63) = 0.0$

b. greater than 55 (2 points)

Answer2: $P(x>55) = (55-33)/11/\sqrt{1000}=63$

$P =0.0$

c. less than 11 or greater than 55 (1 point)

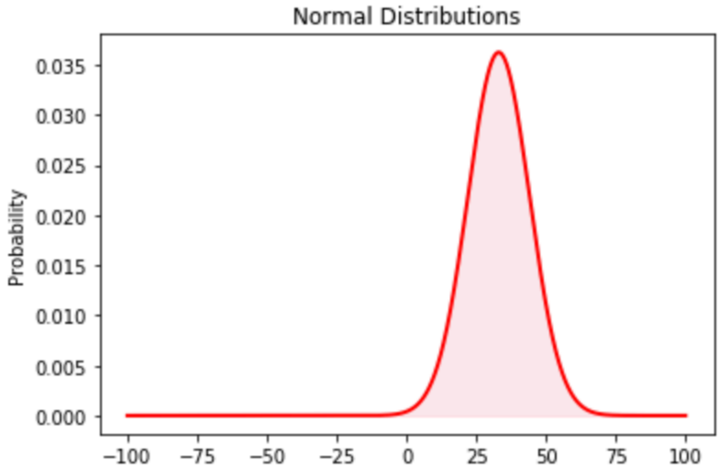
Answer3: $P (x<11 \text{ or } x>55)$

$P=0.0$

Show the calculation as done by hand.

Q4 (5 Points) Write python code to plot Q3 and calculate Q3.

```
plt.fill_between(domain, 0, values, color='#ffb6c1', alpha=0.3)
plt.ylabel("Probability")
plt.title("Normal Distributions")
plt.show()
def prob(x,u,stu,n):
    return stats.norm.cdf((x-u)/(stu/math.sqrt(n)))
def prob1(x,u,stu,n):
    return stats.norm.sf((x-u)/(stu/math.sqrt(n)))
print("prob1",prob(11,33,11,1000))
print("prob2",prob1(55,33,11,1000))
print("prob3",2*prob(11,33,11,1000))
```




```
prob1 0.0
prob2 0.0
prob3 0.0
```

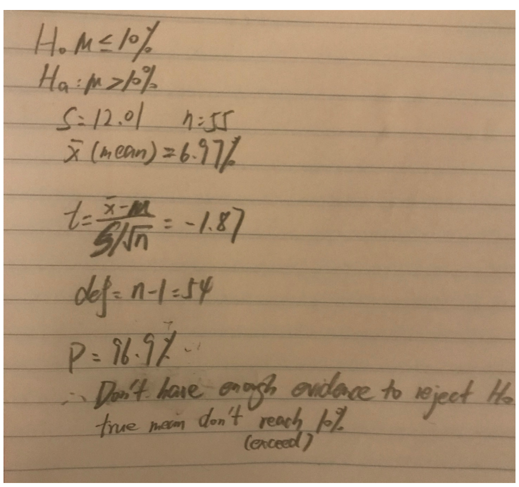
Q5 (5 Points)

The one-year rate of return to shareholders was calculated in a sample of 55 tech stocks. The data, is below and in the file tech_stocks.csv.

```
[23.72502353842273, 21.62401646603374, -
0.7463274288122257, 1.717883045082 8002, -
2.634776050958738, -2.792138753758266, -
10.514395560878746, 8.72052 9920419578,
18.782813772780308, 5.825456165455785,
11.172228117978728, 11. 9703296292814 6, -
30.981624884074883, 8.428109006257554,
13.715597227579686 , -7.14438096845215,
35.38150590002323, 5.951675701660346, -
```

2.128337264991 565, 12.952160066221724, -
9.52841782146271, 9.27768703224383, -10.48902962
5059331, 1.7170477394203232, 11.717280979491225,
18.84977052950971, 12.645 227894971965, -
2.444524930791145, -4.870684454119193,
9.384408019477661, 1 3.450953108385315,
23.714466213916317, 5.7140681189301255, -
14.73667486810 843, 6.455693762385872,
9.715370033540502, 11.133859293104898,
5.125843059 42378, -3.6547977197096486,
15.65791149754521, 17.045514919166266, 20.8641
8259486488, 28.498593533062984,
15.689734619702122, 7.954721816163218, -3.
113512775937407, 12.86046371264133,
2.467429173851536, -2.682786932363779, -
1.9362359856511269, 5.912048015521583,
24.003261208189425, 9.708478961113 5, -
6.91532401310932, 21.426117689357]

A. (1 point) Specify the null and alternative hypotheses tested for determining whether the true mean one-year rate of return for tech stocks exceeded 10%.


$H_0\ \mu \le 10\%$

$H_a: \mu > 10\%$

$S = 12.01 \quad n = 55$

$\bar{x}\ (mean) = 6.97\%$

$t = \frac{\bar{x} - \mu}{S/\sqrt{n}} = -1.87$

$def = n - 1 = 54$

$P = 96.9\%$

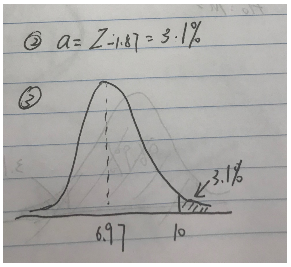
∴ Don't have enough evidence to reject H_0

true mean don't reach 10% (exceed)

B (3 points) Calculate the observed significance level of the test.

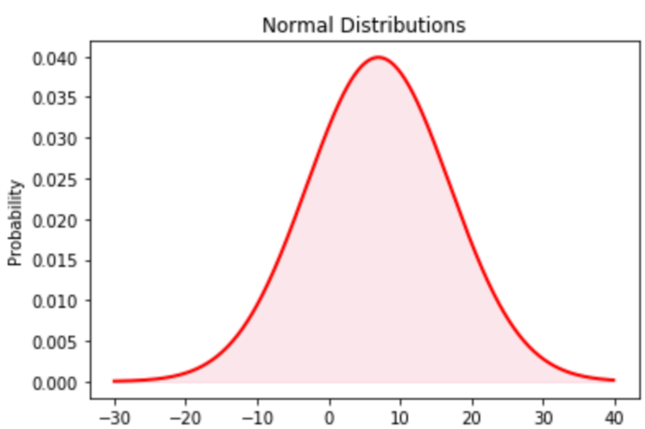
C. (1 point) Interpret the result.

Show the calculation as done by hand.



Q6 (5 Points) Write python code to plot Q5 and calculate Q5.

```
import math
from scipy import stats
domain = np.arange(-30, 40, 0.1)
values = stats.norm(6.98, 10).pdf(domain)
plt.plot(domain, values, color='r', linewidth=2)
plt.fill_between(domain, 0, values, color='#ffb6c1', alpha=0.3)
plt.ylabel("Probability")
plt.title("Normal Distributions")
plt.show()
def prob(x,u,stu,n):
    return stats.norm.cdf((x-u)/(stu/math.sqrt(n)))
print("prob:",1-prob(6.98,10,12.01,55))
print("significance level:",prob(6.98,10,12.01,55))
```




```
prob: 0.9688991977563493
significance level: 0.031100802243650777
```

```python
import pandas as pd
df=pd.read_csv(r'/Users/james/Desktop/tech_stocks.csv')
print(df)
print(df.describe())
```

```
54        54  21.426118
       Unnamed: 0          0
count    55.00000  55.000000
mean     27.00000   6.975336
std      16.02082  12.013618
min       0.00000 -30.981625
25%      13.50000  -2.286431
50%      27.00000   8.428109
75%      40.50000  13.583275
max      54.00000  35.381506
```

Q7 (5 Points) A company has placed an order for 5,000 laptops with a supplier on the condition that no more than 1% of the devices will be defective. To check the shipment, the company tests a random sample of 100 laptops and finds that 2 are defective. Standard deviation is 5%.

Does this provide sufficient evidence to indicate that the proportion of defective can laptops in the shipment exceeds 1%? Explicitly state your null and alternative hypothesis.

$H_0 = \mu \leq 1\%$

$H_a = \mu > 1\%$

$s = 5\%$

$n = 100$

$\bar{x} = 2\%$

$$t = \frac{\bar{x} - \mu}{s/\sqrt{n}}$$

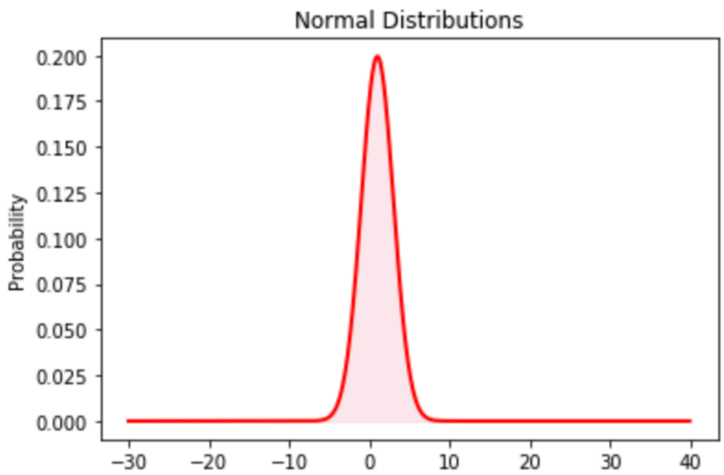
$$= 2$$

$P = 0.977$

$\therefore H_0$ don't reject.

Q8 (5 Points) Write python code to plot Q3 and conduct a hypothesis test on Q7.

```python
import matplotlib.pyplot as plt
import numpy as np
import math
from scipy import stats
domain = np.arange(-30, 40, 0.1)
values = stats.norm(1, 2).pdf(domain)
plt.plot(domain, values, color='r', linewidth=2)
plt.fill_between(domain, 0, values, color='#ffb6c1', alpha=0.3)
plt.ylabel("Probability")
plt.title("Normal Distributions")
plt.show()
def prob(x,u,stu,n):
    return stats.norm.sf((x-u)/(stu/math.sqrt(n)))
print("prob:",1-prob(2,1,5,100))
```




```
prob: 0.9772498680518208
```

Q9 (5 Points) An ultra-marathon runner ran 103 miles per week as reported by runner's world. A random sample of 500 ultra-marathon runners had a mean of 101 miles per week ran when asked.

Let m denote mean distance for all ultra-marathon runners.

A (3 Points). Perform the hypothesis test

Ho: m=103 miles per week ran

Ha: m ≠103 miles per week ran at the 5% significance level. Assume the standard deviation is 60 miles.

$H_0: m = 103$ $\bar{x} = 101$

$H_a: m \neq 103$

$s = 60$

$n = 500$

$t = \frac{\bar{x} - m}{s / \sqrt{n}} = -0.745$

$df = n - 1 = 499$

$P = 0.772$

$\because$ $5\% < P < 95\%$

$\therefore$ H_0 don't reject

B (2 Points). Find a 95% confidence interval for m.

$a = 5\%$, $\frac{a}{2} = 0.025$, $Z_{0.025} = 1.96$

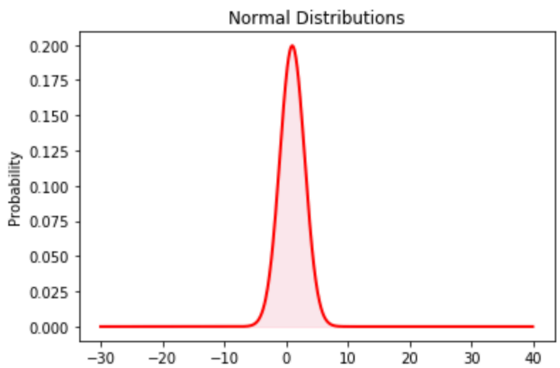
$m = M \pm Z_{\frac{a}{2}} \times \frac{6}{\sqrt{n}}$

$95.75 < m < 106.25$

Show the calculation as done by hand.

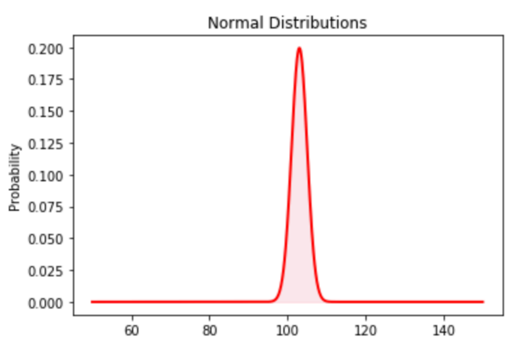
Q10 (5 Points) Write python code to plot Q9 and conduct a hypothesis test on Q9.

```python
import matplotlib.pyplot as plt
import numpy as np
import math
from scipy import stats
domain = np.arange(-30, 40, 0.1)
values = stats.norm(1, 2).pdf(domain)
plt.plot(domain, values, color='r', linewidth=2)
plt.fill_between(domain, 0, values, color='#ffb6c1', alpha=0.3)
plt.ylabel("Probability")
plt.title("Normal Distributions")
plt.show()
def prob(x,u,stu,n):
    return stats.norm.sf((x-u)/(stu/math.sqrt(n)))
print("prob:",1-prob(2,1,5,100))
```




```
prob: 0.9772498680518208
```

```python
plt.show()
def prob(x,u,stu,n):
    return stats.norm.sf((x-u)/(stu/math.sqrt(n)))
def asi(a,u,stu,n):
    return u-(stats.norm.ppf(1-(0.05/2))*(stu/math.sqrt(n)))
def bsi(a,u,stu,n):
     return u+(stats.norm.ppf(1-(0.05/2))*(stu/math.sqrt(n)))
print("prob:",prob(101,103,60,500))
print(asi(0.05,101,60,500))
print(bsi(0.05,101,60,500))
```




```
prob: 0.7719717298748721
95.7408647565405
106.2591352434595
```