STAT2201, Semester 1, 2017

Solution for Assignment 4

Questions marked for grade (each 20%): Q1, Q2, Q5, Q7

Question 1 - Seeing the CLT with Simulation

Consider the following random variables:

$$U \sim \text{Uniform}(5, 10)$$
$$V \sim \text{Exponential}(5)$$
$$W \sim \text{Binomial}(10, 0.2)$$

(a) What is the mean and variance of each?

(b) Consider now,

$$X_n = \sum_{i=1}^{n} X_i,$$

where X is either U, V or W and different X_i are assumed indepdnent. What is the mean and variance of this random sum (a function of n)?

(c) For X either U, V or W, define,

$$\tilde{X}_n = \frac{X_n - E(X_n)}{\sqrt{var(X_n)}}.$$

Use the CLT to postulate the distribution of $\tilde{X}_n$ for non-small n.

(d) Generate Monte Carlo estimates of $P(|\tilde{X}_n| > 2.0)$ using no less than 10^6 generations of $\tilde{X}_n$ for every n, (separately for each U, V or W). Compare your results to $P(|Z| > 2.0)$ taken from a normal distribution table, where Z is a standard normal random variable. Do this for $n = 5, 10, 20$. Tabulate and explain your results.

Solution:

(a)

$E[U] = 7.5$, $E[V] = 0.2$, $E[W] = 2$,

$var(U) = 25/12 = 2.0833$, $var(V) = 1/5^2 = 0.04$, $var(W) = 0.16$.

(b)

$\mu_U(n) = 7.5n$, $\mu_V(n) = 0.2n$, $\mu_W(n) = 2n$,

$\sigma_U^2(n) = (25/12)n$, $\sigma_V^2(n) = 0.04n$, $\sigma_W^2(n) = 1.6n$.

(c)

Observe that $\tilde{U}_n$, $\tilde{V}_n$, $\tilde{W}_n$ are all a mean 0, variance 1 random variables. This exactly holds for any n. The CLT asserts that as n grows the distribution of this random variable gets close to a standard normal distribution.

(d)

In [1]:

```
#For Uniform(a,b) notice we can generate it by (b-a)*rand()+a
function Un(n)
    return sum([5*rand() + 5 for _ in 1:n])
end

for n in [5,10,20]
    data = [(Un(n) - 7.5n)/sqrt((25/12)n) for _ in 1:10^6];
    prop = sum([abs(x) > 2 for x in data])/10^6;
    println("n=",n,":  ",prop)
end
```

```
n=5:  0.043341
n=10:  0.044433
n=20:  0.044994
```

In [1]:

```
using Distributions
function Vn(n)
    return sum([rand(Exponential(1/5)) for _ in 1:n])
end

for n in [5,10,20]
    data = [(Vn(n) - 0.2*n)/sqrt(0.04*n) for _ in 1:10^6];
    prop = sum([abs(x) > 2 for x in data])/10^6;
    println("n=",n,":  ",prop)
end
```

```
n=5:  0.040847
n=10:  0.04149
n=20:  0.04342
```

In [3]:

```
using Distributions
function Wn(n)
    return sum([rand(Binomial(10,0.2)) for _ in 1:n])
end

for n in [5,10,20]
    data = [(Wn(n) - 2*n)/sqrt(1.6*n) for _ in 1:10^6];
    prop = sum([abs(x) > 2 for x in data])/10^6;
    println("n=",n,":  ",prop)
end
```

```
n=5:  0.049101
n=10:  0.032603
n=20:  0.041457
```

The probability under a standard Normal:

In [3]:

```
2*cdf(Normal(),-2)
```

Out[3]:

```
0.04550026389635841
```

Conclusion: We see that for all three initial disributions, centered random sums have "tail probabilities" that are similar to the standard normal.

Question 2 - Sample Mean

Suppose that samples of size $n = 25$ are selected at random from a normal population with mean 100 and standard deviation 10. What is the probability that the sample mean falls in the interval from $\mu_{\overline{X}} - 1.8\,\sigma_{\overline{X}}$ to $\mu_{\overline{X}} - 1.0\,\sigma_{\overline{X}}$.

Solution:

$X_i \sim N(100, 10^2)$, i.i.d.

$\overline{X} = \frac{1}{25}\sum_{i=1}^{25} X_i \sim N(100, (10/5)^2)$, that is, the standard deviation of the sample mean is 2. But the actual values of the mean and standard deviation of $\overline{X}$ are not needed for answering the question.

$P(\mu_{\overline{X}} - 1.8\sigma_{\overline{X}} \leq \overline{X} \leq \mu_{\overline{X}} - 1.0\sigma_{\overline{X}}) = P(-1.8 \leq Z \leq -1.0)$ where Z is a standard normal random variable. Hence this is the answer:

In [2]:

```
using Distributions
cdf(Normal(),-1.0)-cdf(Normal(),-1.8)
```

Out[2]:

```
0.12272493481853122
```

Question 3 - Choice of Sample Size

A normal population has a mean 100 and variance 25. How large must the random sample be if you want the standard error of the sample average to be 1.5?

Solution:

This question is actually the same as Question 6 of the previous assignment. See solution there.

Question 4 - Polymer Elasticity

The elasticity of a polymer is affected by the concentration of a reactant. When low concentration is used, the true mean elasticity is 55, and when high concentration is used, the mean elasticity is 60. The standard deviation of elasticity is 4 regardless of concentration. If two random samples of size 16 are taken, find the probability that $\overline{X}_{high} - \overline{X}_{low} \geqslant 2$.

Solution:

Even though not explicitly stated, we assume a Normal distribution here:

$\overline{X}_h \sim N(60, (\frac{4}{\sqrt{n}})^2)$ or $N(60, 1)$ with $n = 16$

$\overline{X}_l \sim N(55, (\frac{4}{\sqrt{n}})^2)$ or $N(55, 1)$ with $n = 16$

It is assumed the samples are independent hence the sample means are independent as well. Hence,

$\Delta = \overline{X}_h - \overline{X}_l \quad \sim \quad N(60 - 55, 1 + 1)$ or $N(5, \sqrt{2}^2)$

(the independence assumption is needed for adding up the variances without a covariance term).

$P(\Delta \geq 2) = P(Z > \frac{2-5}{2}) = 1 - \Phi(-1.5) = \Phi(1.5) = 0.9332$

In [4]:

```
cdf(Normal(),1.5)
```

Out[4]:

```
0.9331927987311419
```

Question 5 - Building up Confidence

For a normal population with known variance σ^2, answer the following questions:

(a) What is the confidence level for the interval $\bar{x} - 2.14\sigma/\sqrt{n} \leqslant \mu \leqslant \bar{x} + 2.14\sigma/\sqrt{n}$?

(b) What is the confidence level for the interval $\bar{x} - 2.49\sigma/\sqrt{n} \leqslant \mu \leqslant \bar{x} + 2.49\sigma/\sqrt{n}$?

(c) What is the confidence level for the interval $\bar{x} - 1.85\sigma/\sqrt{n} \leqslant \mu \leqslant \bar{x} + 1.85\sigma/\sqrt{n}$?

(d) What is the confidence level for the interval $\bar{x} - 1.96\sigma/\sqrt{n} \leqslant \mu \leqslant \bar{x} + 1.96\sigma/\sqrt{n}$?

Solution:

In all cases these are confidence intervals of the form, $\overline{x} \pm z_{1-\alpha/2}\sigma/\sqrt{n}$. So we just need to say what is $1-\alpha/2$ when $z_{1-\alpha/2} = 2.14, 2.49, 1.85, 1.96$.

Here is one way to do it (you should also do it with a table for practice):

In [9]:

```
using Distributions
for z in [2.14, 2.49, 1.85, 1.96]
    println(2*(1-cdf(Normal(),z)))
end
```

```
0.03235476674433224
0.012774309529886452
0.06431354959122748
0.04999579029644097
```

Question 6 - Beverage Machine

A postmix beverage machine is adjusted to release a certain amount of syrup into a chamber where it is mixed with carbonated water. A random sample of 25 beverages was found to have a mean syrup content of $\bar{x} = 1.10$ fluid ounce and a standard deviation of $s = 0.015$ fluid ounce. Find a 95% CI on the mean volume of syrup dispensed. State any assumptions that are made.

Solution:

We use a t-distribution with 24 degrees of freedom here $1-\alpha/2 = 0.975$ and $t_{0.975,24} = 2.064$

In [1]:

```
using Distributions
quantile(TDist(24),0.975)
```

Out[1]:

```
2.0638985616280254
```

The Standard error is $s/2.064 = 0.0072674$. Half the width of the confidence intervel is $2.064 * 0.0072674 = 0.015$ So the confidence interval is 1.10 ± 0.015 or $[0.95, 1.25]$.

Question 7 - P-Value

For the hypothesis test $H_0 : \mu = 10$ against $H_1 : \mu > 10$ and variance known, calculate the P-value for each of the following test statistics:

(a) $z = 2.05$

(b) $z = -1.84$

(c) $z = 0.4$

Solution:

(a)

In [11]:

```
1-cdf(Normal(),2.05)
```

Out[11]:

```
0.020182215405704418
```

(b)

In [12]:

```
1-cdf(Normal(),-1.84)
```

Out[12]:

```
0.9671158813408361
```

(c)

In [13]:

```
1-cdf(Normal(),0.4)
```

Out[13]:

```
0.3445782583896758
```

Question 8 - Sodium Content in Organic Cornflakes

The sodium content of twenty 300-gram boxes of organic cornflakes was determined. The data (in milligrams) is contained in (*9-65.csv*).

(a) Can you support a claim that mean sodium content of this brand of cornflakes differs from 130 milligrams? use $\alpha = 0.05$, state your hypothesis clearly, find the P-value and make a conclusion.

(b) Check that sodium content is normally distributed (e.g. using the code for Normal probability plots from Assignment 3).

(c) Compute the power of the test if the true mean sodium content is 130.5 milligrams.

(d) What sample size would be required to detect a true mean sodium content of 130.1 milligrams if you wanted the power of the test to be at least 0.75? Explain your answer.

(e) Explain how the question in part (a) could be answered by constructing a two-sided confidence interval on the mean sodium content.

Solution:

In [10]:

```
using DataFrames
table = readtable("9-65.csv",header=false)
```

Out[10]:

	x1
1	131.15
2	130.69
3	130.91
4	129.54
5	129.64
6	128.77
7	130.72
8	128.33
9	128.24
10	129.65
11	130.14
12	129.29
13	128.71
14	129.0
15	129.39
16	130.42
17	129.53
18	130.12
19	129.78
20	130.92

In [9]:

```
data = table[1]
```

Out[9]:

```
20-element DataArrays.DataArray{Float64,1}:
 131.15
 130.69
 130.91
 129.54
 129.64
 128.77
 130.72
 128.33
 128.24
 129.65
 130.14
 129.29
 128.71
 129.0
 129.39
 130.42
 129.53
 130.12
 129.78
 130.92
```

In [16]:

```
xBar = mean(data)
s = std(data)
n = length(data)
se = s/sqrt(n)
(xBar,s,n,se)
```

Out[16]:

```
(129.747,0.8764287583261093,20,0.1959754281052915)
```

(a) $H_0 : \mu = 130$ vs. $H_1 : \mu \neq 130$

In [17]:

```
tStat = (xBar-130)/se
```

Out[17]:

```
-1.2909781723454476
```

The critical value (see how to obtain it from a t-table also):

In [24]:

```
using Distributions
quantile(TDist(19),0.975)
```

Out[24]:

```
2.093024054408309
```

Under H_0 the T stastistic is distributed according to a t-distribution with 19 degrees of freedom.

In [19]:

```
using Distributions
pValue = 2*(1-cdf(TDist(19),abs(tStat)))
```

Out[19]:

```
0.21219878697735228
```

Conclusion: Since the p-value is 0.21 > 0.05, we do not reject H_0.

Here it is computed using the Hypothesis Tests package

In [20]:

```
using HypothesisTests
```

In [21]:

```
OneSampleTTest(data,130)
```

Out[21]:

```
One sample t-test
-----------------
Population details:
    parameter of interest:   Mean
    value under h_0:         130
    point estimate:          129.747
    95% confidence interval: (129.33681871490268,130.15718128509735)

Test summary:
    outcome with 95% confidence: fail to reject h_0
    two-sided p-value:           0.21219878697735228 (not significant)

Details:
    number of observations:   20
    t-statistic:              -1.2909781723454476
    degrees of freedom:       19
    empirical standard error: 0.1959754281052915
```

(b)

In [22]:

```
using PyPlot,Distributions,StatsBase

function NormalProbabilityPlot(data)
    mu = mean(data)
    sig = std(data)
    n = length(data)
    p = [( i -0.5)/ n for i in 1: n]
    x = quantile(Normal(),p)
    y = sort([( i - mu)/sig for i in data])
    PyPlot.scatter(x,y)
    xRange = maximum(x) - minimum(x)
    PyPlot.plot([minimum(x) - xRange /8 , maximum(x) + xRange /8] ,[minimum(x) - xRange
 /8 , maximum(x)+ xRange/8] ,color="red", linewidth =0.5)
    xlabel("Theoretical quantiles")
    ylabel("Quantiles of data");
    return
end
```

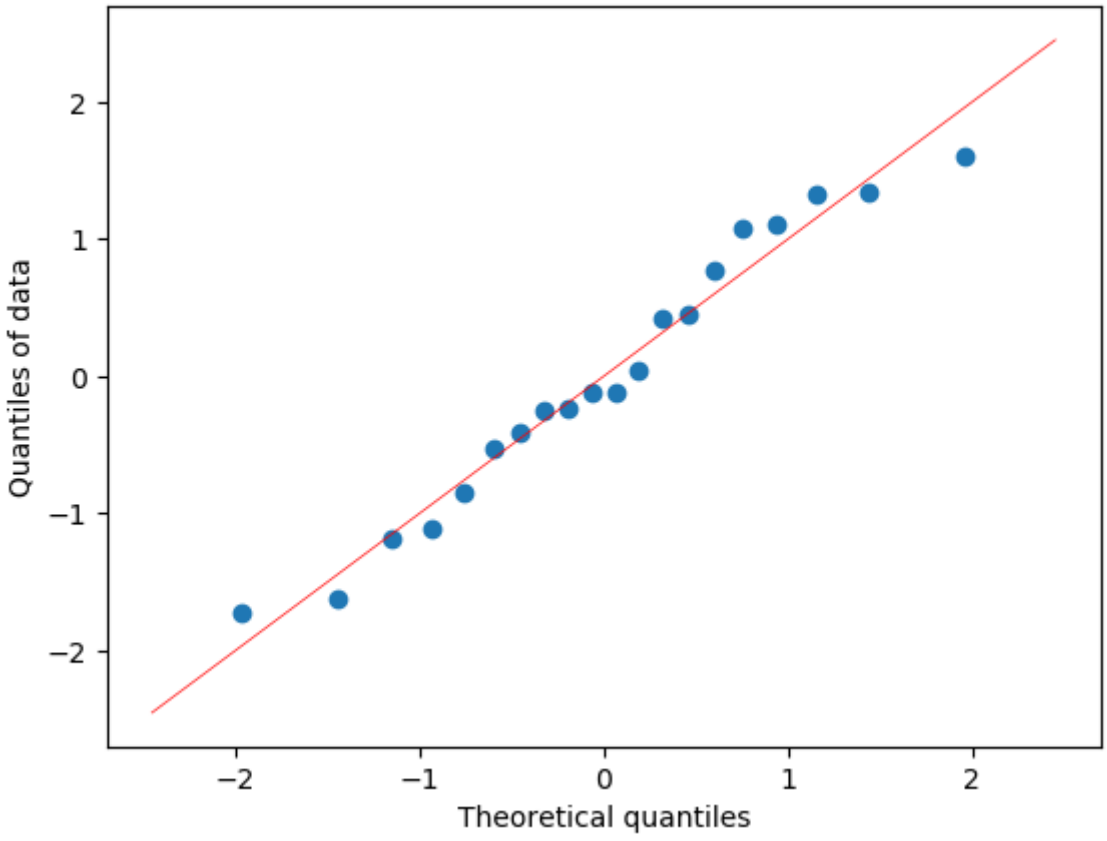
```
WARNING: using PyPlot.table in module Main conflicts with an existing iden
tifier.
```

Out[22]:

```
NormalProbabilityPlot (generic function with 1 method)
```

In [23]:

```
NormalProbabilityPlot(data)
```



The normal probability plot does not give a strong indication for deviation from the normallity assumption.

(c) Computing the power of the test in this way can be done in several ways. None of them are very simple. We'll use Monte-Carlo Simulation under the point (in the parameter space) of H_1. where $\mu = 130.5$ and $\sigma = 0.8764$. The goal is to look at the distribution of the test statistic in this case and see the chance of rejecting H_0

In [30]:

```
using Distributions
function tStatisticUnderH1()
    #This simulates a random sample under the specified point in H1
    data = [rand(Normal(130.5,0.8764)) for _ in 1:20]
    xBar = mean(data)
    s = std(data)
    #this is the t statistic. Notice that it is based on the random mean and random sam
ple variance we get. It still
    #uses 130 as mu_0
    tStatistic = (xBar - 130)/(s/sqrt(20))
    return tStatistic
end

#Repeat a simulation of the hypothesis test a million times, each time seeing if reject
ing or not
sum([abs(tStatisticUnderH1()) > 2.093 for _ in 1:10^6])/10^6
```

```
WARNING: Method definition tStatisticUnderH1() in module Main at In[29]:4
 overwritten at In[30]:4.
```

Out[30]:

0.678344

Hence our Monte Carlo estimate for the power is $1 - \beta = 0.68$.

(d) Now we can repeat the above mechanism for different sample sizes and seeing how it affects power. For this modify the code a bit:

In [42]:

```
using Distributions
function tStatisticUnderH1(n)
    data = [rand(Normal(130.1,0.8764)) for _ in 1:n]
    xBar = mean(data)
    s = std(data)
    tStatistic = (xBar - 130)/(s/sqrt(n))
    return tStatistic
end

#notice we are looping here over n, and for each n using a (slightly different) critica
l value
[(n, sum([abs(tStatisticUnderH1(n)) > quantile(TDist(n-1),0.975) for _ in 1:10^6])/10^6
   )   for  n in 20:25]
```

```
WARNING: Method definition tStatisticUnderH1(Any) in module Main at In[4
1]:3 overwritten at In[42]:3.
```

Out[42]:

```
6-element Array{Tuple{Int64,Float64},1}:
 (20,0.077353)
 (21,0.078838)
 (22,0.080237)
 (23,0.081493)
 (24,0.083776)
 (25,0.084775)
```

We see in the above how power behaves as a function of n. But since the difference between the mean under H0 and the mean under H1 is so small, we need much bigger samples to reach a power of 75.

In [43]:

```
[(n, sum([abs(tStatisticUnderH1(n)) > quantile(TDist(n-1),0.975) for _ in 1:10^6])/10^6
   )   for  n in 20:20:200]
```

Out[43]:

```
10-element Array{Tuple{Int64,Float64},1}:
 (20,0.077233)
 (40,0.108311)
 (60,0.140502)
 (80,0.172199)
 (100,0.204112)
 (120,0.236719)
 (140,0.268571)
 (160,0.300264)
 (180,0.33131)
 (200,0.3619)
```