Statistics 147 LAB #4
10 pts; Summer 2020

NAME: Wesley Chang **ID: (last 4 #s only)** 0996

This lab is designed to give the student practice generating confidence intervals and performing tests of hypothesis for a single population mean.

REMINDER: You reject $\boldsymbol{H_0}$ if p-value $<$ specified of $\boldsymbol{\alpha}$. We'll use $\boldsymbol{\alpha = 0.05}$.

SAS

Data File: You will need the data file **plant.dat** from Blackboard (under Data Files). You should have downloaded this file for Lab #3.

Note: The data starts on Line 2.

Four chemical plants, producing the same product and owned by the same company, discharge effluent into streams in the vicinity of their locations. To check the extent of the pollution created by the effluents and to determine whether the amount of polluting effluents varies from plant to plant, the company collected random samples of liquid waste from each of the four plants. The data, in pounds per gallon of waste, is given in the table below.

PlantA	PlantB	PlantC	PlantD
1.65	1.70	1.40	1.58
1.72	1.85	1.75	1.77
1.50	1.36	1.58	1.48
1.37	2.05	1.65	1.69
1.60	1.80	1.55	1.65
1.40	2.10	1.45	1.65
1.75	1.95	1.66	1.79
1.38	1.65	1.70	1.58
1.65	1.80	1.85	1.77
1.55	2.00	1.24	1.60

Let Plant A be sample 1, Plant B be sample 2, Plant C be sample 3, and Plant D be sample 4.

1. In SAS, one can use **proc means** to generate the information for generating confidence intervals and performing a test of hypothesis for a single population mean. **Proc means** tests whether a population mean is 0. Thus, one must first transform the data so that it is centered about 0. This can be accomplished by creating a new variable that is the old data minus that hypothesized mean. One can also use **proc univariate** to test a single mean by specifying the value in the **proc univariate** statement.

   (i) Invoke SAS and use **proc means** to generate the appropriate information to test whether the mean discharge effluent for **Plant B** is 1.75 pounds/gallon. Recall, in Lab #3, you read in the data. So open your SAS program file (lab3su20.sas) and make the following changes/additions. (Add titles and then look for 2 rows of * for the new code.)

   NOTE: DO NOT TYPE THE PROGRAM IN AGAIN, JUST MODIFY YOUR EXISTING CODE AND SAVE IT AS lab4su20s.sas.

   NOTE: Delete the code associated with *data looptry*.

```
/* Set up format of the output */
options nocenter ps = 55 nocenter ls = 78 nodate nonumber formdlim='*';
 /* ls = linesize,  ps = pagesize
           nocenter      justifies the output so it is not centered on the page
           nodate        suppresses printing of today's date on each page of output
           nonumber      suppresses printing of page number on each page of output
           formdlim      overrides the internal page breaks and replaces them
                            with the designated symbol*/
/* Use DM to clear all windows except the editor window */
DM log "odsresults; clear; out; clear; log; clear;";
ods graphics off;

title1 'Statistics 147 Lab #4, Summer 2020';
title2 'Your name goes here';
title3 'Question 1';

/* Create temporary SAS dataset named lab3su20 */

data lab3su20;
    /* Open data file plant.dat. Be sure to specify the path indicating where
       you have saved the data file.  The actual data starts on Line 2.*/
    infile 'c:\Luke\summer2020\su20147\datafiles\plant.dat' firstobs = 2;

    /* Create nested do loops to read in the data
       NOTE: There are 10 rows and four columns of data
    First, Do loop for the rows*/
    do row = 1 to 10;
        /* Do Loop for the columns */
        do plant = 1 to 4;
            /* Use If-Then-Else Structure to name the plants */
            if      plant = 1 then name = 'Plant A';
            else if plant = 2 then name = 'Plant B';
            else if plant = 3 then name = 'Plant C';
            else                   name = 'Plant D';

            /* Input response (data values) */
            input dischrg @@;
            /* Output the data */
            output;
        /* Close the Do loop for columns */
        end;
    /* Close the Do loop for rows */
    end;
run;

/* Print the results */
proc print noobs data = lab3su20;
    title4 'Part (i) Read in and Print data';
run;

/* First sort the data according to the variable plant */
proc sort data = lab3su20;
   by plant;
```

```
run;

/* Print the results */
proc print noobs data = lab3su20;
    title4 'Print to check sorted';
run;

/* Use proc means to generate the mean and variance for each plant
    n               number of observations
    mean            sample mean
    var             sample variance
    by plant        group the data according to the plant from which the observation
                    was selected
    var discharge   generate mean and variance of the variable dischrg (for each plant) */

proc means n mean var data = lab3su20;
   title4 'Part (ii): Descriptive Statistics';
   by plant;
   var dischrg;
run;


/* Create new temporary SAS dataset which only contains the
   observations from Plant B */
data onlyB;
    /* Use set command to bring in all the data */
    set lab3su20;

    /* Use if statement to restrict the data to Plant B, i.e., Plant 2*/

    if plant = 2;    /* Could also use: if name = 'Plant B'; */
run;

/* *************************************************** */
/* *************************************************** */
/*               NEW CODE BEGINS HERE                  */

    /*  Create new variable representing the transformed data
        new_data = (original variable) - (mean value to test) */
    new_data = dischrg - 1.75;

/* Use proc means to generate information for testing mean
   of new_data = 0
      n       number of observations
      mean    sample mean
      t       t-test statistic
      probt   p-value for two-sided test */

proc means n mean t probt data = onlyB;
   title4 'Part (i) Testing new_data mean = 0 using proc means';
   var new_data;
run;
/*          END OF NEW CODE                  */
```

```
/* ************************************************** */
/* ************************************************** */
proc print noobs data = onlyB;
proc means n mean stddev;
    var dischrg;
run;

/* Create new SAS dataset which only contains the
   observations from Plants A and  B */
data bothAB;
    /* Use set command to bring in all the data */
    set lab3su20;

    /* Modify title4 */
    title4 'Part (iv): Plant A and Plant B with Descriptive Statistics ';

    /* Use if statement to restrict the data to Plants A and  B, i.e., Plants 1 and 2 */

    if plant = 1 or plant = 2;
    /* Could also use: if name = 'Plant A' or name = 'Plant B'; */


proc print data = bothAB;
    title4 'Print bothAB.  Make sure sorted by the variable PLANT';
run;

/* Use proc means to generate some descriptive statistics.
   Use the by statement to generate statistics for each plant. */
proc means n mean stddev data = bothAB;
   by plant;
   var dischrg;
run;

quit;
```

Reminder: Save your SAS file as **lab4su20.sas**. Execute the program and complete the following:

```
Statistics 147 Lab #4, Summer 2020
Your name goes here
Question 1
Part (i) Testing new-data mean = 0 using proc means

The MEANS Procedure

      Analysis Variable : new_data

 N             Mean     t Value      Pr > |t|
-----------------------------------------------

 10    0.0760000     1.09       0.3037
-----------------------------------------------
```

♠ H_0 : $\mu_B = 1.75$

♠ H_a : $\mu_B \neq 1.75$

♠ The p-value is 0.3037.

♠ RR: Reject H_0 if p-value $< \alpha = 0.05$

♠ What is your conclusion? (Be sure to justify your answer!)

Conclusion: Since the p-value = greater than is (less than, greater than) [circle your choice] $\alpha = 0.05$, (reject, do not reject) [circle your choice] $H_0 \rightarrow$ it (is, is not) [circle your choice] reasonable to assume the true mean discharge effluent for Plant B is significantly different from 1.75 pounds/gallon.

(ii) One can also use **proc univariate**: We need to include a value of μ_0 to test in the **proc univariate** statement. We can do this using **mu0** as follows:

```
/* Use proc univariate to test mu0 value
   Use ods select TestsForLocation to suppress printing of
   all output except the tests for location */
proc univariate mu0 = 1.75 data = onlyB;
     ods select TestsForLocation;
     title4 'Part (ii) Using proc univariate to test mean = 1.75 vs mean not= 1.75';
     var dischrg;
```

Add the above lines of code, **right after** the following block of code

```
proc means n mean t probt data = onlyB;
     title4 'Part (ii) Testing new-data mean = 0 using proc means';
     var new_data;
```

in your program.

Save and execute your program. Complete the following:

```
Statistics 147 Lab #4, Summer 2020
Your name goes here
Question 1
Part (ii) Using proc univariate to test mean = 1.75 vs mean not= 1.75

            Tests for Location: Mu0=1.75
Test              -Statistic-     -----p Value------

Student's t       t  1.090872     Pr > |t|     0.3037
Sign              M         2     Pr >= |M|    0.3438
Signed Rank       S        11     Pr >= |S|    0.2852
```

(iii) One can also use **proc ttest**: We need to include a value of H_0 to test in the **proc ttest** statement. We can do this using **h0** as follows:

```
/* Use proc ttest to test single mean; specify value to test: h0 = value_to_test */
proc ttest h0 = 1.75 data = onlyB;
   title4 'Part (iii) Using proc ttest to test mean = 1.75 vs mean not= 1.75';
   var dischrg;
run;
```

Add the above lines of code, **right after** the following block of code

```
/* Use proc univariate to test mu0 value
   Use ods select TestsForLocation to suppress printing of
   all output except the tests for location */

proc univariate mu0 = 1.75 data = onlyB;
    /*Be sure to include the 'S' in TestSForLocation */
    ods select TestsForLocation;
    title4 'Part (ii) Using proc univariate to test mean = 1.75 vs mean not= 1.75';
    var dischrg;
run;
```

in your program.

Save and execute your program. Complete the following:

```
Statistics 147 Lab #4, Summer 2020
Your name goes here
Question 1
Part (iii) Using proc ttest to test mean = 1.75 vs mean not= 1.75

The TTEST Procedure
Variable:  dischrg

  N        Mean      Std Dev      Std Err      Minimum      Maximum
 10      1.8260       0.2203       0.0697       1.3600       2.1000

    Mean        95% CL Mean         Std Dev      95% CL Std Dev
  1.8260      1.6684    1.9836       0.2203      0.1515    0.4022

    DF     t Value     Pr > |t|

     9        1.09      0.3037
                        --------
```

(iv) When using **proc ttest**, one can specify a one-sided alternative by using the **sides** option. Use **proc ttest** and the **sides** option to generate the appropriate information to test whether the mean discharge effluent for **Plant B** is greater than 1.75 pounds/gallon.

We can do this as follows:

```
/* Use proc ttest to test single mean; specify value to test: h0 = value_to_test
   Use sides = upper to generate test for mu_B > 1.75 */
proc ttest h0 = 1.75 sides = upper data = onlyB;
   title4 'Part (iv) Using proc ttest to test Ho: mean = 1.75 vs Ha: mean > 1.75';
   var dischrg;
run;
```

Add the above lines of code, **right after** the following block of code

```
/* Use proc ttest to test single mean; specify value to test: h0 = value_to_test */
```

```
proc ttest h0 = 1.75 data = onlyB;
   title4 'Part (iii) Using proc ttest to test mean = 1.75 vs mean not= 1.75';
   var dischrg;
run;
```

Complete the following.

```
Statistics 147 Lab #4, Summer 2020
Your name goes here
Question 1
Part (iv) Using proc ttest to test mean = 1.75 vs mean > 1.75

The TTEST Procedure
Variable:  dischrg
  N        Mean     Std Dev     Std Err     Minimum     Maximum
 10      1.8260      0.2203      0.0697      1.3600      2.1000

    Mean       95% CL Mean         Std Dev      95% CL Std Dev
  1.8260      1.6983   Infty        0.2203      0.1515   0.4022

    DF    t Value    Pr > t

     9       1.09    0.1518
                    ---------
```

- ♠ H_0 : $\mu_B = 1.75$
- ♠ H_a : $\mu_B > 1.75$
- ♠ The p-value is ______________.
- ♠ RR: Reject H_0 if p-value $< \alpha = 0.05$
- ♠ What is your conclusion? (Be sure to justify your answer!)

Conclusion: Since the p-value = 0.1518 is (less than, greater than) [circle your choice] $\alpha = 0.05$, (reject, do not reject) [circle your choice] H_0 → it (is, is not) [circle your choice] reasonable to assume the true mean discharge effluent for Plant B is significantly larger than 1.75 pounds/gallon.

(v) Using **proc means**, generate a **99%** confidence interval for the true mean discharge for Plant B (i.e.,plant 2).

We can do this as follows:

```
/* Use proc means to generate confidence interval
   Specify value of alpha to use: 99% -> alpha = 0.01 */
proc means n mean stddev clm alpha = 0.01 data = onlyB;
   title4 'Part (v) 99% CI using proc means';
   var dischrg;
run
```

Add the above lines of code, **right after** the following block of code:

```
/* Use proc ttest to test single mean; specify value to test: h0 = value_to_test
   Use sides = upper to generate test for mu_B > 1.75 */
```

```
proc ttest h0 = 1.75 sides = upper data = onlyB;
   title4 'Part (iv) Using proc ttest to test mean = 1.75 vs mean > 1.75';
   var dischrg;
run;
```

Complete the following:

```
Statistics 147 Lab #4, Summer 2020
Your name goes here
Question 1
Part (v) 99% CI using proc means

The MEANS Procedure

                      Analysis Variable : dischrg

                                                 Lower 99%      Upper 99%
 N            Mean            Std Dev          CL for Mean     CL for Mean
```

N	Mean	Std Dev	Lower 99% CL for Mean	Upper 99% CL for Mean
10	1.8260000	0.2203129	1.5995870	2.0524130

Confidence Intervals *Limits*: 1.5995870, 2.0524130

Interpretation:

Based on the data given, we can have 99% confidence that the true mean discharge for Plant B lies between 1.5995870 and 2.0524130

(vi) Using **proc ttest**, generate a **99%** confidence interval for the true mean discharge for Plant B (i.e.,plant2).

We can do this as follows:

```
/* Use proc ttest to generate confidence interval
Specify the value of alpha to use: -> alpha = 0.01 */
proc ttest alpha = 0.01 data = onlyB;
    title4 'Part (vi) 99% CI using proc ttest';
var dischrg;
run;
```

Add the above lines of code, **right after** the following block of code:

```
/* Use proc means to generate confidence interval
   Specify value of alpha to use: 99% -> alpha = 0.01 */
proc means n mean stddev clm alpha = 0.01 data = onlyB;
   title4 'Part (v) 99% CI using proc means';
   var dischrg;
run;
```

Complete the following:

```
Statistics 147 Lab #4, Summer 2020
Your name goes here
Question 1
Part (vi) 99% CI using proc ttest
The TTEST Procedure

Variable:  dischrg
  N        Mean      Std Dev     Std Err     Minimum     Maximum
 10      1.8260       0.2203      0.0697      1.3600      2.1000

    Mean          99% CL Mean         Std Dev       99% CL Std Dev

  1.8260       1.5996   2.0524        0.2203       0.1361    0.5018
              --------  --------

    DF     t Value    Pr > |t|
     9       26.21      <.0001
```

Exit SAS

R

Invoke R and complete the following.

Reading in the Data File

1. Consider the data file, **plant.dat**. Note that the data file includes headings in Line 1. Read in and print the data.

   REMINDER: Always be sure to change the path to the data file to the location where you have saved the file!

   Let's use an *R script* to enter our commands. Open **R**. From the main menu select **File** → **New script**. The **R Editor** window will open. (It will say *untitled* until you save the script.)

   ★ Move the cursor to the **R Editor** window and type in the following:

   ```
   # Statistics 147 Lab #4 Summer 2020
   # Your name goes here
   #
   # R Question 1
   # Use the read.table command to read in the data
   # format: read.table(file = "filename including path",header = TRUE)
   # Be sure to change the path to your data file.
   plant_data = read.table(file = "[REPLACE/WITH/PATH/TO/YOUR/FILE]/plant.dat",header= TRUE)
   # Alternatively, you could have changed the current working directory with setwd(...)
   # and simply used the file name e.g.) read.table(file = "plant.dat",header= TRUE)
   ```

```
# Print the data as a check
plant_data
```

Make sure your cursor is in the **R Editor** window.

▲ To **save** your script,from the main menu, select **File** → **Save As**. Select the location where you would like to save your script and type **lab4_su20_XX**, where XX = initials of your name.

▲ To execute your script, from the main menu, select **Edit** → **All**.

You should see everything you typed, plus the data, in the **R Console** window. When you see the data, have Ruihan, Luke or your labmate check it. Then place your initials here. ____________________

★ Use the attach() function to make each column individually accessible. Use the **names()** function to obtain column names. To accomplish these tasks, type the following in the **R Editor** window.

```
# Use attach command to get access to individual columns
attach(plant_data)
# Use the names() function to obtain column names
names(plant_data)
# Print the data
PlA
PlB
PlC
PlD
```

NOTE: That is an **"L"** in PlA, PlB, PlC, and PlD. That is *NOT* a one (1).

Make sure your cursor is in the **R Editor** window. Save your script and then

▲ highlight the new text you just typed.

▲ From the main menu, select **Edit** → **Run line or selection**, or hit **Ctrl + R** (Windows) or **Cmnd + Return** (Mac).

You should see everything you just typed, plus the output, in the **R Console** window. When you see the data, have someone check it. Then, initial here. ____________________

2. Refer to R Question 1. Generate the mean, median, variance, and standard deviation for the **Plant A**.

   In the **R Editor** window, type the following:

```
# R Question 2
# Sample Mean: Use mean() function
mean_PlA = mean(PlA)
# Print the value
mean_PlA
#
# Sample Median: Use median() function
median_PlA = median(PlA)
# Print the value
median_PlA
#
# Sample Variance: Use var() function
variance_PlA = var(PlA)
# Print the value
variance_PlA
#
# Sample Standard Deviation: Use sd() function
sd_PlA = sd(PlA)
# Print the value
sd_PlA
```

Make sure your cursor is in the **R Editor** window. Save your script and then

▲ highlight the new text you just typed.

▲ From the main menu, select **Edit** → **Run line or selection**, or hit **Ctrl** + **R** (Windows) or **Cmnd** + **Return** (Mac).

Complete the following from the R Console window.

```
> # Sample Mean: Use mean() function
> mean_PlA = mean(PlA)
> # Print the value
> mean_PlA
```

[1] ____1.557____

```
> # Sample Median: Use median() function
> median_PlA = median(PlA)
> # Print the value
> median_PlA
```

[1] ____1.575____

```
> # Sample Variance: Use var() function
> variance_PlA = var(PlA)
> # Print the value
> variance_PlA
```

[1] ____0.01969____

```
> # Sample Standard Deviation: Use sd() function
> sd_PlA = sd(PlA)
> # Print the value
> sd_PlA
```

[1] ____0.1403211____

3. Refer to R Question 1. Use the **summary** command to generate the default descriptive statistics in **R** for **Plant B**.

   In the **R Editor** window, type the following:

```
# R Question 3
# Generate default descriptive statistics for Plant B (PlB)
summary_PlB = summary(PlB)
# Print the results
summary_PlB
```

Make sure your cursor is in the **R Editor** window. Save your script and then

▲ highlight the new text you just typed.

▲ From the main menu, select **Edit** → **Run line or selection**.

Complete the following from the R Console window.

```
> # R Question 3
> # Generate default descriptive statistics for Plant B (PlB)
> summary_PlB = summary(PlB)
> # Print the results
```

```
> summary_PlB
   Min.      1st Qu.     Median      Mean     3rd Qu.     Max.
```

1.360 1.725 1.825 1.826 1.988 2.100

4. (Your Turn) Using **R** and your script, complete the following table for the **Plant C** data.

	Mean	Median	Variance	Standard Deviation
Plant C	1.583	1.615	0.03257889	0.1804962

NOTE: Before proceeding, be sure to install and load the **TeachingDemos** package. This allows you to use the `t.test()` function to generate confidence intervals and test of hypotheses for a single mean or the difference of two means and the `var.test()` function to test equality of variances.

The general format is

```
t.test(x, y,alternative = "what", mu = diff, var.equal = FALSE,paired = FALSE
       conf.level = level)
var.test(x, y, ratio = 1,alternative = "what",conf.level = level)
```

where

what	greater, less, two.sided
diff	hypothesized difference between two means
var.equal	TRUE or FALSE (May be omitted for non-independent samples)
paired	TRUE for non-independent sample, FALSE for independent samples (default is FALSE, so this can be omitted for independent samples)
level	confidence level (0.90, 0.95, etc.)

You can always enter `?t.test` and `?var.test` in the **R Console** to read more.

5. Practicing confidence intervals and tests of hypothesis.

   ♦ Using **R**, find and interpret a **98%** confidence interval for the true mean discharge for **Plant A**. Add the following lines of code to your script.

   ```
   # R Question 4
   # Generate 98% CI for Plant A
   # Use t.test
   # Format: t.test(name_of_variable,alternative = appropriate option,
   # conf.level = confidence-level-in-decimal-format)
   t.test(PlA,alternative="two.sided",conf.level= 0.98)
   ```

   Make sure your cursor is in the **R Editor** window. Save your script and then
   - ▲ highlight the new text you just typed.
   - ▲ From the main menu, select **Edit** → **Run line or selection**.

   Complete the following from the R Console window.

   ```
   > t.test(PlA, alternative = "two.sided", conf.level = 0.98)

           One Sample t-test
   ```

```
data:  PlA
t = 35.0886, df = 9, p-value = 6.13e-11
alternative hypothesis: true mean is not equal to 0
98 percent confidence interval:
```

1.431803 ------------------ 1.682197 -----------------

```
sample estimates:
mean of x
```

1.557 ------------------

Interpretation:

For Plant A, we can say with statistical confidence that 98% of values will fall between 1.431803 and 1.682197

- (Your turn!) Using **R**, find and interpret a **96%** confidence interval for the true mean discharge for **Plant B**. Be sure you write the command you used to obtain your output.

  Command:

  t.test(PlB,alternative="two-sided",conf.level=0.96

  Interval Limits:

  ```
  96 percent confidence interval:
  ```

  1.658903 ------------------ 1.993097 -----------------

  Interpretation:

  For Plant B, we can say with statistical confidence that 96% of values will fall between 1.658903 and 1.993097

◆ Using **R** to complete the calculations, test the hypothesis that the true mean discharge effluent (call it μ_A) for Plant A is significantly **less than** 1.50 pounds/gallon.

Add the following lines of code to your script.

```
# Test mu(PlA) < 1.50
# Use t.test
# Format: t.test(name_of_variable,alternative = appropriate option,
# conf.level = confidence-level-in-decimal-format)
t.test(PlA,alternative="less",mu = 1.5, conf.level= 0.95)
```

Make sure your cursor is in the **R Editor** window. Save your script and then

▲ highlight the new text you just typed.

▲ From the main menu, select **Edit** → **Run line or selection**.

Complete the following from the R Console window.

```
        One Sample t-test
data:  PlA
t = 1.2846, df = 9, p-value = 0.8845
alternative hypothesis: true mean is less than 1.5
```

♠ H_0 : $\mu_A = 1.50$

♠ H_a : $\mu_A < 1.50$

♠ The p-value is 0.8845.

♠ RR: Reject H_0 if p-value $< \alpha = 0.05$

♠ What is your conclusion? (Be sure to justify your answer!)

Conclusion: Since the p-value = 0.8845 is (less than, greater than) [circle your choice] $\alpha = 0.05$, (reject, do not reject) [circle your choice] H_0 → it (is, is not) [circle your choice] reasonable to assume the true mean discharge effluent for Plant A is significantly less than 1.50 pounds/gallon.

◆ (Your turn!) Using **R** to complete the calculations, test the hypothesis that the true mean discharge effluent (call it μ_B) for Plant B is significantly **different from** 1.75 pounds/gallon. Be sure to include the command you used to generate your output.
Command:

t.test(PlB,alternative=“two.sided”,conf.level=0.95)

Output from R:

```
        One Sample t-test

data:  P1B
t = 26.21, df = 9, p-value = 8.272e-10
alternative hypothesis: true mean is not equal to 1.75
95 percent confidence interval:
 1.668398   1.983602
sample estimates:
mean of x
 1.826
```

- ♠ H_0 : $\mu_B = 1.75$
- ♠ H_a : $\mu_B \neq 1.75$

- ♠ The p-value is 8.272e-10.
- ♠ RR: Reject H_0 if p-value $< \alpha = 0.05$
- ♠ What is your conclusion? (Be sure to justify your answer!)

Conclusion: Since the p-value = 8.272e-10 is (less than, greater than) [circle your choice] $\alpha = 0.05$, (reject, do not reject) [circle your choice] $H_0 \rightarrow$ it (is, is not) [circle your choice] reasonable to assume the true mean discharge effluent for Plant B is significantly different from 1.75 pounds/gallon.

NOTE: Be sure to save your script! Remember the script is ordinary text, so can be copied into LaTeX , Word and/or Notepad.

You have now successfully completed Lab #4! Please submit your completed lab worksheet to iLearn. Don't forget to save your R and SAS scripts! Have a good week!!!

Luke & Ruihan