

MTH 3220 Lab 2

Due Thu., Sept. 26

1 Part A: F Test for Two Standard Deviations

1.1 Infant Head Circumferences Data Set

The data below are from a study comparing **head circumferences** (cm) of infants treated for pulmonary hypertension (**PH**) to those of infants not so treated (**Control**).

HeadCirc	Group
35.2	Control
35.6	Control
33.9	PH
36.7	Control
35.1	Control
36.0	Control
35.1	PH
33.4	PH
33.4	Control
31.3	Control
33.5	Control
35.8	Control
36.3	Control
34.5	PH
34.3	Control
37.9	PH
33.1	Control
31.3	PH
32.4	Control
35.1	Control
33.6	Control
32.5	PH
31.8	Control
34.1	Control
32.9	PH
35.2	Control
34.8	Control
34.5	Control
31.6	Control
36.3	PH
31.9	Control
31.9	Control
34.2	PH
32.8	Control
34.0	Control

Here are the data in a more convenient format:

Control	35.2,	35.6,	36.7,	35.1,	36.0,	33.4,	31.3,	33.5,	35.8,	36.3,
	34.3,	33.1,	32.4,	35.1,	33.6,	31.8,	34.1,	35.2,	34.8,	34.5,
	31.6,	31.9,	31.9,	32.8,	34.0					
PH	33.9,	35.1,	33.4,	34.5,	37.9,	31.3,	32.5,	32.9,	36.3,	34.2

1. Use `c()` and `<-` to create two vectors, one containing the **Control** head circumferences and the other the **PH** ones.

- Use `sd()` and `var()` to compute the sample **standard deviations** and **variances** of the **Control** and **PH** head circumferences.
- We want to test the hypotheses

$$H_0 : \sigma_1 = \sigma_2$$

$$H_a : \sigma_1 \neq \sigma_2$$

The function `var.test()` will carry out an **F test for two standard deviations**. Among its arguments are:

`x, y` numeric vectors of data values
`alternative` a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less".

Use `var.test()` with the **Control** and **PH** vectors to carry out the **F test**.

2 Part B: One-Factor ANOVA

2.1 Blood Coagulation Times Data Set

The table below shows **coagulation times** (seconds) for samples of blood drawn from 24 animals receiving four different **diets, A, B, C, and D**. The animals were randomly assigned to the diets, and the blood samples were taken and tested in random order.

Diet (Treatment)			
A	B	C	D
62	63	68	56
60	67	66	62
63	71	71	60
59	64	67	61
59	65	68	63
63	66	68	64
62	64	66	63
61	67	68	59

The data are in the file **blood.txt** *stacked* in the order in which the blood samples were drawn.

- The function `read.table()` will read data from a text (.txt) file into a data frame in R. Among its arguments are:

`file` the name of the file which the data are to be read from. Each row of the table appears as one line of the file.
`header` a "logical" value indicating whether the file contains the names of the variables as its first line.

For example, suppose you save the **blood.txt** file as **C:\Users\Myname\Downloads\blood.txt**. You would read it into a data frame named **my.data** by typing:

```
my.data <- read.table("C:/Users/Myname/Downloads:/blood.txt", header = TRUE)
```

(Note the **direction** of the **slash**, '/', in the file path name.)

Save **blood.txt** and use `read.table()` (with `header = TRUE`) to read it into a data frame called, say, `my.data`. **Hint**: Here is a little trick you can use:

```
# Select the file:
my.file <- file.choose()
# Look at the path name for the selected file:
my.file
# Read the data from the file:
my.data <- read.table(my.file, header = TRUE)
```

2. We want to test the hypotheses

$$H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4$$

$$H_a : \text{At least two of the } \mu_i \text{'s are different.}$$

where μ_1 , μ_2 , μ_3 , and μ_4 are the true (unknown) **mean responses** to **diets A, B, C, and D**.

The function `aov()` will carry out **a one-factor ANOVA**. Among its arguments are:

<code>formula</code>	a formula specifying the ANOVA model
<code>data</code>	a data frame in which the variables specified in the formula will be found.

Carry out the **one-factor ANOVA** (using `my.data` from Step 1) and save the results as `my.anova` by typing:

```
my.anova <- aov(CoagTime ~ Diet, data = my.data)
```

Above, the *formula* `CoagTime ~ Diet` indicates `CoagTime` is the **response variable** and `Diet` is the **explanatory variable** (or **factor**).

3. The function `summary()` will produce an **ANOVA table** from a so-called *aov* object. Type:

```
summary(my.anova)
```

to see the **ANOVA table**.

4. We can pass an R *formula* as the main argument to `boxplot()` along with a data frame. Make **side-by-side boxplots** of the **coagulation times** for the four **diets**, for example by typing:

```
boxplot(CoagTime ~ Diet, data = my.data, col = "lightblue",
        main = "Boxplots of 4 Diets", ylab = "Time (Seconds)")
```

5. The function `aggregate()` will compute summary statistics separately for each treatment group. Among its arguments are:

<code>formula</code>	a formula, such as $y \sim x$, where the y variables are numeric data to be split into groups according to the grouping x variables (usually factors).
<code>data</code>	A data frame in which the variables specified in the formula should be taken.
<code>FUN</code>	a function to compute the summary statistics which can be applied to all data subsets.

Use `aggregate()` to compute the mean coagulation time for each **diet** by typing:

```
aggregate(formula = CoagTime ~ Diet, data = my.data, FUN = mean)
```

6. Now use `aggregate()` to compute the standard deviation (`FUN = sd`) of the coagulation times separately for each **diet**.