# STAT 420: Homework 2

*Summer 2016, Dalpiaz*

*Due: Tuesday, September 6 by 11:59 PM CDT*

## Contents



> "The fool wonders, the wise man asks."
>
> — **Benjamin Disraeli**

## Directions

Students are encouraged to work together on homework using the discussion boards. However, sharing, copying, or providing any part of a homework solution or code is an infraction of the University's rules on academic integrity. Any violation will be punished as severely as possible.

- Your assignment must be submitted through Coursera. You are required to upload one `.zip` file, named `hw02_yourNetID.zip`, which contains:
  - Your RMarkdown file which should be saved as `hw02_yourNetID.Rmd`. For example `hw02_dalpiaz2.Rmd`.
  - The result of knitting your RMarkdown file as `hw02_yourNetID.html`. For example `hw02_dalpiaz2.html`.
  - Any outside data provided as a `.csv` file used for the homework.
  - This will roughly match the `.zip` provided.
- Your resulting `.html` file will be considered a "report" which is the material that will determine the majority of your grade. Be sure to visibly include all `R` code and output that is relevant to answering the exercises. (You do not need to include irrelevant code you tried that resulted in error or did not answer the question correctly.)
- You are granted an unlimited number of submissions, but only the last submission *before* the deadline will be viewed and graded.
- If you use this `.Rmd` file as a template, be sure to remove the quotation, directions section, and consider removing `eval = FALSE` from any code chunks provided (if you would like to run that code as part of your assignment).
- Your `.Rmd` file should be written such that, when stored in a folder with any data you are asked to import, it will knit properly without modification. If your `.zip` file is organized properly, this should not be an issue.

- Unless otherwise stated, you may use `R` for each of the exercises.
- Be sure to read each exercise carefully!
- Include your name and NetID in the final document, not only in your filenames.

# Assignment

## Exercise 1 (Writing Simple Functions)

For each of the following parts, use the following vectors:

```
a = 1:10
b = 10:1
c = rep(1, times = 10)
d = 2 ^ (1:10)
```

**(a)** Write a function called `sum_of_squares`.

- Arguments:
  - A vector of numeric data `x`.
- Output:
  - The sum of the squares of the elements of the vector. $\sum_{i=1}^{n} x_i^2$

Provide your function, as well as the result of running the following code:

```
sum_of_squares(x = a)
sum_of_squares(x = c(c, d))
```

**(b)** Write a function called `sum_of_power`.

- Arguments:
  - A vector of numeric data `x`.
  - `p` which should have the default value of `2`.
- Output:
  - $\sum_{i=1}^{n} x_i^p$

Provide your function, as well as the result of running the following code:

```
sum_of_power(x = a)
sum_of_power(x = a, p = 3)
sum_of_power(x = a, p = a)
sum_of_power(x = a, p = c(1, 2))
```

**(c)** Write a function called `rms_diff`.

- Arguments:
  - A vector of numeric data `x`.

  - A vector of numeric data `y`.
- Output:
  - $\sum_{i=1}^{n}(x_i - y_i)^2$

Provide your function, as well as the result of running the following code:

```
rms_diff(x = a, y = b)
rms_diff(x = d, y = c)
rms_diff(x = d, y = 1)
rms_diff(x = a, y = 0) ^ 2 * length(a)
```

## Exercise 2 (Plotting, Testing)

For this exercise we will use the data that is stored in `intelligence.csv` which records IQs of a random sample of residents of Pawnee and Eagleton, Indiana.

**(a)** Load the data from `intelligence.csv` into a variable in `R` called `intelligence`. Show the code used to do this.

**(b)** Create a side-by-side boxplot that compares the IQs across the two towns. Be sure to give the plot a title and label the axes appropriately.

**(c)** Are people from Eagleton smarter than people from Pawnee? Perform an appropriate statistical test using the given sample data. That is, test $H_0 : \mu_E = \mu_P$ vs $H_1 : \mu_E > \mu_P$, where

- $\mu_E$ is the mean IQ of a resident of Eagleton.
- $\mu_P$ is the mean IQ of a resident of Pawnee.

Explicitly state the p-value of the test and the resulting statistical decision at a significance level $\alpha = 0.10$. Interpret the results in the context of the problem.

**(d)** Repeat **(c)** using a two-sided alternative hypothesis. What changes?

## Exercise 3 (Writing More Functions)

In this exercise we will write our own functions related to performing a one-sample $t$ test. That is $H_0 : \mu = \mu_0$ versus $H_1 : \mu \neq \mu_0$, where $\mu_0$ is the hypothesized value of $\mu$.

Throughout this exercise you may **not** use the `t.test()` function inside your functions. You may use it to check your work separately, but no such double-checks should appear in your final report.

Some built in `R` functions that may be useful to you when writing your functions include: `c()`, `ifelse()`, `mean()`, `sd()`, `abs()`, `length()`, `sqrt()`, and `pt()`.

**(a)** Write a function called `do_t_test` which takes two inputs:

- `x`: A vector which stores observations.
- `mu`: The hypothesized value of $\mu$ which defaults to `0`.

The function should output:

- The value of the test statistic, $t$.
- The p-value of the test. The function only needs to be able to handle a two-sided alternative.

In order to output both, consider using `c(t, pval)` as the last line of your function, and store those two values elsewhere in the body of your function.

**(b)** Write a function called `make_decision` which takes two inputs:

- `pval`: The p-value of a test.
- `alpha`: The significance level of a test. Set a default value of `0.05`.

The function should output `"Reject!"` or `"Fail to Reject."` based on the comparison of `pval` to `alpha`.

**(c)** Now we will test the quality of your functions from parts **(a)** and **(b)**. Run the following code:

```
set.seed(42)
y = rnorm(25, 1.4, 1)
pval = do_t_test(y, mu = 2)[2]
pval
make_decision(pval, alpha = 0.10)
```

If your `do_t_test()` and `make_decision()` functions are correct, you should obtain a decision of `"Fail to Reject."` You will also be evaluated on whether the numeric p-value you obtain is correct.

## Exercise 4 (CLT Simulation)

For this exercise we will simulate from the exponential distribution. If a random variable $X$ has an exponential distribution with rate parameter $\lambda$, the pdf of $X$ can be written

$$f(x; \lambda) = \lambda e^{-\lambda x}$$

for $x \geq 0$.

Also recall,

$$\mu = E[X] = \frac{1}{\lambda}$$
$$\sigma^2 = Var[X] = \frac{1}{\lambda^2}$$

**(a)** This exercise relies heavily on generating random observations. To make this reproducible we will set a seed for the randomization. Alter the following code to make `birthday` store your birthday in the format: `yyyymmdd`. For example, William Gosset, better known as *Student*, was born on June 13, 1876, so he would use:

```
birthday = 18760613
set.seed(birthday)
```

**(b)** Simulate 10000 samples of size **5** from an exponential distribution with $\lambda = 2$. Store the mean of each sample in a vector. Plot a histogram of these sample means. (Be sure to give it a title, and label the axes appropriately.) Based on the histogram, do you think the central limit theorem applies here?

**(c)** Simulate 10000 samples of size **100** from an exponential distribution with $\lambda = 2$. Store the mean of each sample in a vector. Plot a histogram of these sample means. (Be sure to give it a title, and label the axes appropriately.) Based on the histogram, do you think the central limit theorem applies here?

**(d)** We just repeated ourselves, so that means we probably should be writing a function. Write a function called `sim_xbars_exp` which takes three inputs:

- The number of samples to simulate.
- The sample size.
- The rate parameter of an exponential distribution.

The function should output a vector of sample means which are the result of sampling from an exponential distribution as specified by the inputs.

Use your function to simulate 25000 samples of size **50** from an exponential distribution with $\lambda = 3$. Store the mean of each sample in a vector. Plot a histogram of these sample means. (Be sure to give it a title, and label the axes appropriately.)

## Exercise 5 (More Simulation)

Let $X$ follow an exponential distribution with rate parameter $\lambda_X = 2$. Let $Y$ follow a Poisson distribution with rate parameter $\lambda_Y = 3$.

We write $sd(X)$ for the true standard deviation of $X$ and $m(Y)$ for the true median of $Y$.

Let $s_x$ be the sample standard deviation of $X$ which is an estimate of $sd(X)$. Also let $m_y$ be the sample median which is an estimate of $m(Y)$.

Suppose we take samples of size $n_x = 10$ from $X$ and take samples of size $n_y = 5$. Consider the statistic

$$\frac{s_x}{m_y}.$$

What is the (sampling) distribution of $\frac{s_x}{m_y}$? Who knows? Ask a statistician interested in theory. Instead of using mathematics, simulate $\frac{s_x}{m_y}$ 5000 times and store the results. Plot a histogram of the observed values of $\frac{s_x}{m_y}$. Comment on the shape of the histogram and empirical distribution of $\frac{s_x}{m_y}$. Before running your code, set the same seed used for the previous exercise. For full credit, do **not** use a `for` loop.