PSYC2020L Assignment 4

[name]

2025-09-12

Goals for Review:

- Probability Distributions Review
- *Z*-Scores in R
- Generating Data
- R Packages
- Reading Files

For each question, please make sure you include code output. If the question asks for comparison or further explanation, please make sure to include that as well.

Setting Up

1) Rename this assignment as “Lab 4 Assignment [Last, First].Rmd”

2) Check and set (if needed) your working directory. If you’d like to load packages with `library()`, please do so in this block.

```
here::here()
```

```
## [1] "C:/Users/jessi/OneDrive - Georgia Institute of Technology/Courses/GTA/PSYC 2020/PSY
C 2020L Site"
```

3) Read in the `tour_de_france.csv` file and assign it to a variable (e.g., `tdf_data`). Display the first six rows of this dataset.

```
tdf <- rio::import(here::here("slides", "Lab 4 - Probability Distributions", "tour_de_franc
e.csv"))
head(tdf)
```

```
##   year winner_avg_speed total_distance             winner winner_nationality
## 1 1903            25.68           2428      Maurice Garin             France
## 2 1904            25.27           2420       Henri Cornet             France
## 3 1905            27.11           2994  Louis Trousselier             France
## 4 1906            24.46           4545       Rene Pottier             France
## 5 1907            28.47           4488 Lucien Petit-Breton             France
## 6 1908            28.74           4488 Lucien Petit-Breton             France
##   starting_city
## 1         Paris
## 2         Paris
## 3         Paris
## 4         Paris
## 5         Paris
## 6         Paris
```

Measures of Variability

4) What is the variance and SD of `winner_avg_speed`?

```
var(tdf$winner_avg_speed)
```

```
## [1] 26.94034
```

```
sd(tdf$winner_avg_speed)
```

```
## [1] 5.190409
```

If they do some square or square root transformation to get one from the other, that works too.

5) What is the variance and SD of `total_distance`?

```
var(tdf$total_distance)
```

```
## [1] 485801.2
```

```
sd(tdf$total_distance)
```

```
## [1] 696.9944
```

If they do some square or square root transformation to get one from the other, that works too.

Z-Scores

6) Create a new variable within the Tour de France data that contains the `winner_avg_speed` variable standardized. No output needed for this question.

```
tdf$winner_avg_speed_z <- (tdf$winner_avg_speed - mean(tdf$winner_avg_speed)) / sd(tdf$winner_avg_speed)
```

7) Create a new variable within the Tour de France data that contains the `total_distance` variable standardized. No output needed for this question.

```
tdf$total_distance_z <- (tdf$total_distance - mean(tdf$total_distance)) / sd(tdf$total_distance)
```

8) Calculate the variance and SD of the new *z*-scored version `winner_avg_speed`, and compare it to the original, non-*z*-scored version. What differences do you observe?

```
var(tdf$winner_avg_speed_z)
```

```
## [1] 1
```

```
sd(tdf$winner_avg_speed_z)
```

```
## [1] 1
```

The variance and SD for the z-scored version are both one. This is because the z-score transformation sets the variance
[or SD]
to one. This obviously much less than the original variance of 26.94 and SD of 5.19.

Generation from Normal Distribution

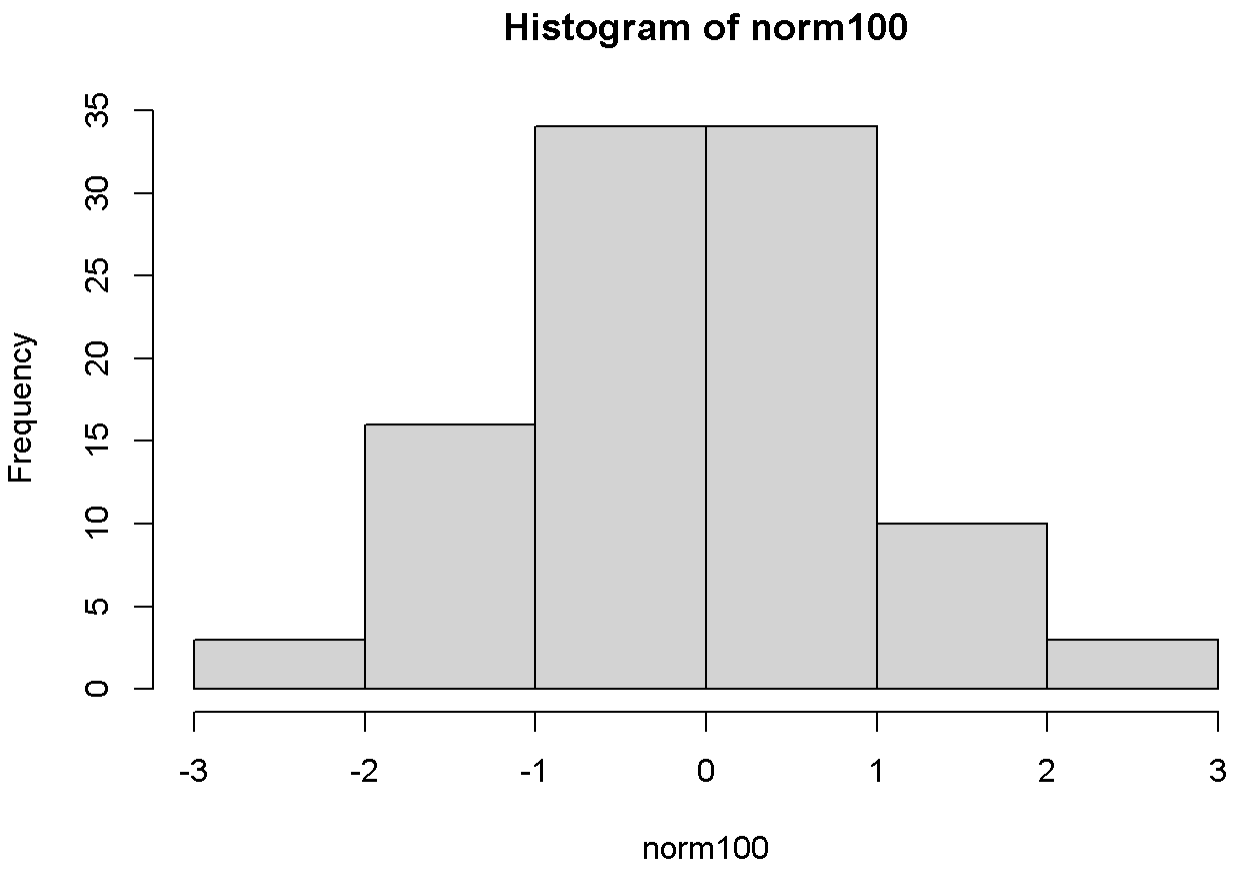
9) Generate three vectors and assign them to variables: one with 10 draws from a standard normal distribution, one with 50 draws from a standard normal distribution, and one with 100 draws from a standard normal distribution. Display just the vector with 10 draws.

```
norm10 <- rnorm(10, 0, 1)
norm50 <- rnorm(50, 0, 1)
norm100 <- rnorm(100, 0, 1)
norm10
```

```
##  [1]  0.018773332 -0.759642623 -0.008700049 -0.068767451  0.707531276
##  [6]  0.845746245 -0.401554479 -1.434600751  1.107168300 -1.004408729
```

10) Plot a histogram of the vector out of the three that will appear the most representative of the underlying standard normal distribution.

```
hist(norm100)
```

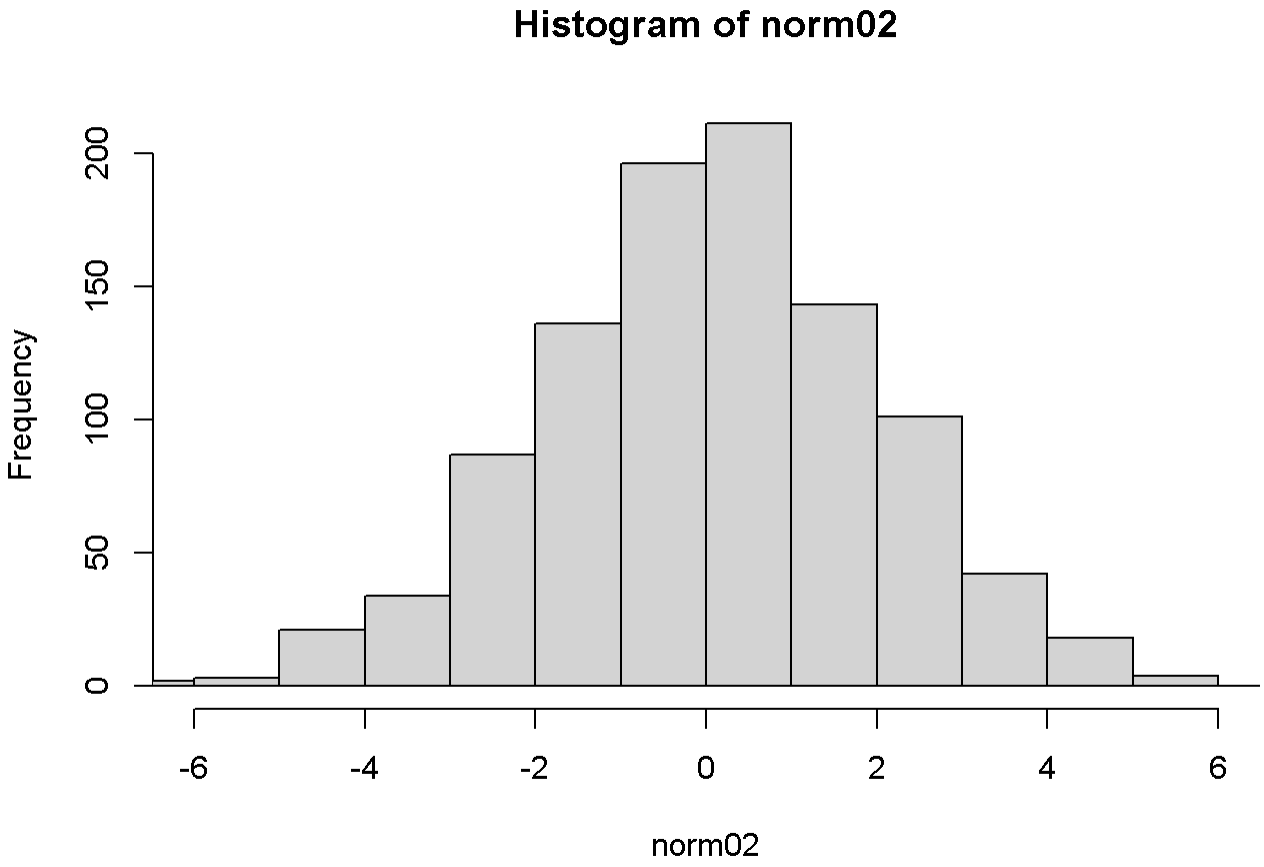


11) Create two more vectors and assign them to variables: one with 1000 draws from a normal distribution with mean of zero and SD of 2 and one with 1000 draws from a normal distribution with mean of 2 and variance of 4. No output is needed for this question.

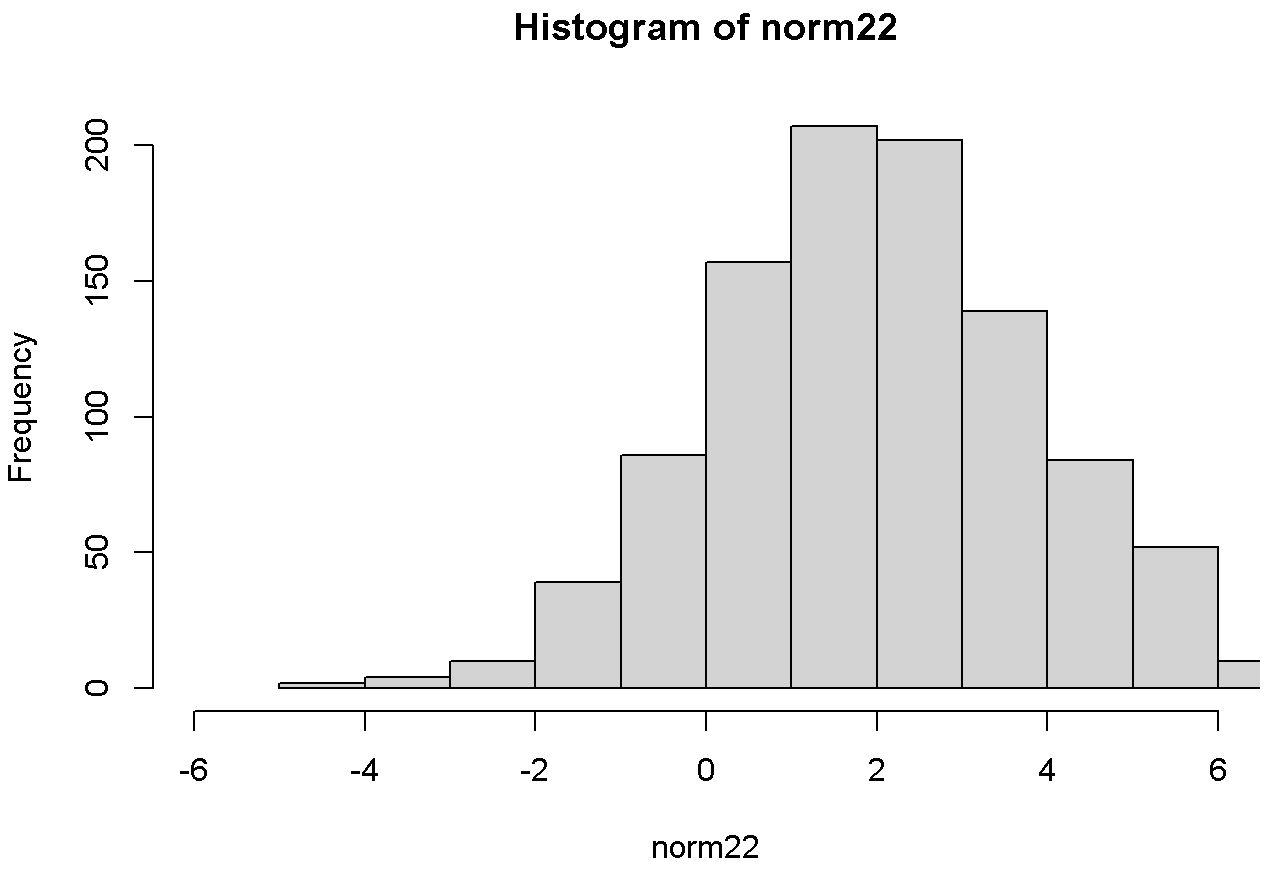
```
norm02 <- rnorm(1000, 0, 2)
norm22 <- rnorm(1000, 2, sqrt(4))
```

12) Plot each of the two vectors from (11) in a histogram. Set the x-axis limits to be the same for both (hint: use the ? tool if you don't remember how to set axis limits!), and compare the two plots.

```
hist(norm02, xlim = c(-6, 6))
```




```
hist(norm22, xlim = c(-6, 6))
```



The histograms have the same variance, but the one with mean of 2 is shifted to the right.

Matrices

13) Create a vector of sequential integers from 1 to 10 and assign it to a variable. No output is needed for this question.

```
myvec <- 1:10
```

14) Turn this vector in a 2 rows x 5 columns matrix that fills in the rows first. Assign this matrix to a variable and display the matrix as output.

```
mymat <- matrix(myvec, nrow = 2, ncol = 5, byrow = T)
mymat
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    2    3    4    5
## [2,]    6    7    8    9   10
```

Indexing

15) From that matrix, please index:

15a) The second row.

```
mymat[2, ]
```

```
## [1]  6  7  8  9 10
```

15b) The third column of the first row.

```
mymat[1, 3]
```

```
## [1] 3
```

15c) The second through fourth columns of the first and second rows.

```
mymat[1:2, 2:4]
```

```
##      [,1] [,2] [,3]
## [1,]    2    3    4
## [2,]    7    8    9
```