Annand\_assignment4

Joseph Annand

2023-06-02

# Normal Distrubution area function —————————–

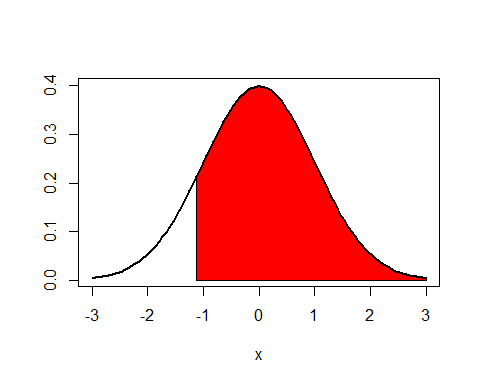
# https://r-coder.com/normal-distribution-r/#Plot\_Normal\_distribution\_in\_R  
normal\_area <- function(mean = 0, sd = 1, lb, ub, acolor = "lightgray", ...) {  
 # Create sequence of 100 equally spaced numbers between mean - 3 and mean + 3   
 x <- seq(mean - 3 \* sd, mean + 3 \* sd, length = 100)   
   
 # If no lower bound is defined, set lower bound to lower limit of the distribution   
 if (missing(lb)) {  
 lb <- min(x)  
 }  
 # If no upper bound is defined, set upper bound to upper limit of distribution  
 if (missing(ub)) {  
 ub <- max(x)  
 }  
   
 # Create empty plot   
 plot(x, dnorm(x, mean, sd), type = "n", ylab = "")  
   
 # Initialize sequence between the bounds  
 x2 <- seq(lb, ub, length = 100)  
 # Initialize set of probability density function values of x2  
 y <- dnorm(x2, mean, sd)  
   
 # Draw the area of interest under the curve   
 polygon(c(lb, x2, ub), c(0, y, 0), col = acolor)  
 # Plot the normal distribution curve  
 lines(x, dnorm(x, mean, sd), type = "l", ...)  
}

# Question 4.2 ————————————————–

a <- pnorm(-1.13, lower.tail = FALSE)  
a\*100

## [1] 87.07619

normal\_area(mean = 0, sd = 1, lb = -1.13, acolor = "red", lwd = 2)

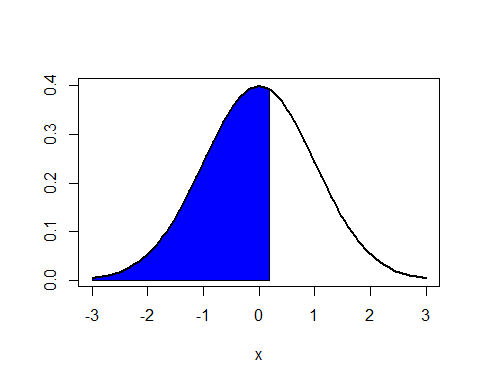


87.08% of a standard normal distribution is found in Z > -1.13.

b <- pnorm(0.18)  
b\*100

## [1] 57.14237

normal\_area(mean = 0, sd = 1, ub = 0.18, acolor = "blue", lwd = 2)

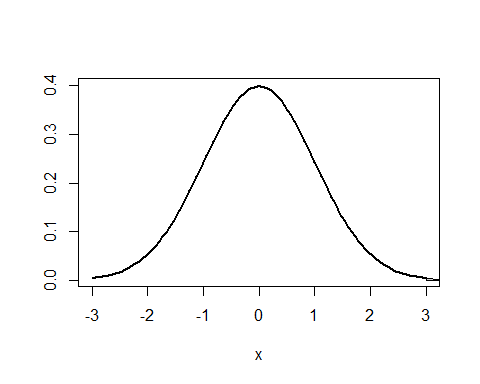


57.14% of a standard normal distribution is found in Z > 1.48.

c <- pnorm(8, lower.tail = FALSE)  
c\*100

## [1] 6.220961e-14

normal\_area(mean = 0, sd = 1, lb = 8, lwd = 2)

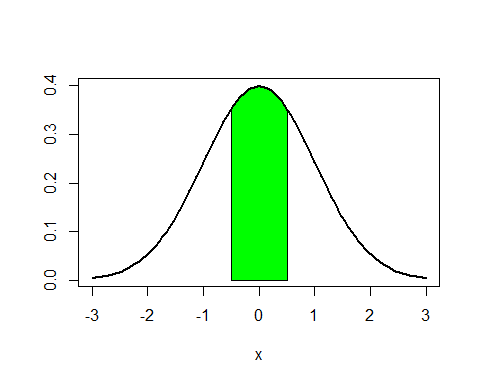


6.22 x 10^-14 % of a standard normal distribution is found in Z > 8. Note that because such a small fraction of the distribution is in this region, the graph does not have visible area under the curve representing the region.

d <- pnorm(0.5) - pnorm(-0.5)  
d\*100

## [1] 38.29249

normal\_area(mean = 0, sd = 1, lb = -0.5, ub = 0.5, acolor = "green", lwd = 2)



38.29% of a standard normal distribution is found in abs(Z) < 0.5.

# Question 4.4 ————————————————–

leo\_time <- 4948  
mary\_time <- 5513  
  
mens\_mean <- 4313  
mens\_sd <- 583  
  
womens\_mean <- 5261  
womens\_sd <- 807

Part a:

paste("N(mu = ", as.character(mens\_mean), ", sigma = ", as.character(mens\_sd), ")")

## [1] "N(mu = 4313 , sigma = 583 )"

paste("N(mu = ", as.character(womens\_mean), ", sigma = ", as.character(womens\_sd), ")")

## [1] "N(mu = 5261 , sigma = 807 )"

Part b:

z\_score\_leo <- (leo\_time - mens\_mean) / mens\_sd  
z\_score\_leo

## [1] 1.089194

z\_score\_mary <- (mary\_time - womens\_mean) / womens\_sd  
z\_score\_mary

## [1] 0.3122677

The Z-score for Leo’s time is 1.089, and the Z-score for Mary’s time was 0.312. The Z-scores represent their times as a number of standard deviations above or below the mean. Z-scores tell us where their times fall in the distribution of completion times for their respective group.

Part c:

Mary ranked better in her group based on her Z-score. Mary’s time was 0.312 standard deviations above, or slower than, the mean of her group, while Leo’s time was 1.089 standard deviations above, or slower than, the mean in his group. Leo’s time is slower for his group compared to Mary’s

Part d:

pnorm(z\_score\_leo, lower.tail = FALSE) \* 100

## [1] 13.80342

Leo finished faster than 13.8% of his group.

Part e:

pnorm(z\_score\_mary, lower.tail = FALSE) \* 100

## [1] 37.74186

Mary finished faster than 37.7% of her group.

Part f:

If the distribution was not nearly normal, the Z-scores for Mary’s and Leo’s times would remain the same given that the means and standard deviations are the same. However, the answers for parts c through e may not be valid. If the distributions of each group are both not nearly normal, the shapes of the curves for the two groups could be different and Mary’s and Leo’s times cannot be compared using Z-scores; this means the answer in part c using the Z-scores to compare Mary’s and Leo’s rankings would not be accurate. If the shapes of the curves are vastly different from a normal distribution, the pnorm() function will not accurately calculate the tail areas under the curve and our answers for part d and e will be incorrect.

# Question 4.8 ————————————————–

Part a:

capm\_mean <- 14.7  
capm\_sd <- 33  
  
pnorm(0, mean = capm\_mean, sd = capm\_sd)

## [1] 0.3279957

The portfolio has a return of less than 0% in 32.80% of years.

Part b:

# qnorm converts a percentile to a Z-score for a given normal distribution  
qnorm(0.85, mean = capm\_mean, sd = capm\_sd)

## [1] 48.9023

The cutoff for the highest 15% of annual returns wiht this portfolio is 48.90%.

Lab 4: Normal Distribution

Joseph Annand

2023-06-04

library(tidyverse)  
library(openintro)  
head(fastfood)

## # A tibble: 6 × 17  
## restaurant item calories cal\_fat total\_fat sat\_fat trans\_fat cholesterol  
## <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Mcdonalds Artisan G… 380 60 7 2 0 95  
## 2 Mcdonalds Single Ba… 840 410 45 17 1.5 130  
## 3 Mcdonalds Double Ba… 1130 600 67 27 3 220  
## 4 Mcdonalds Grilled B… 750 280 31 10 0.5 155  
## 5 Mcdonalds Crispy Ba… 920 410 45 12 0.5 120  
## 6 Mcdonalds Big Mac 540 250 28 10 1 80  
## # ℹ 9 more variables: sodium <dbl>, total\_carb <dbl>, fiber <dbl>, sugar <dbl>,  
## # protein <dbl>, vit\_a <dbl>, vit\_c <dbl>, calcium <dbl>, salad <chr>

mcdonalds <- fastfood %>%  
 filter(restaurant == "Mcdonalds")  
dairy\_queen <- fastfood %>%  
 filter(restaurant == "Dairy Queen")

normal\_plot <- function(food\_data, mean = 0, sd = 1, …) { # X grid for non-standard normal distribution x <- food\_data

# Density function

f <- dnorm(x, mean, sd)

plot(x, f, type = “l”, lwd = 2, col = “blue”, ylab = ““, xlab =”cal\_fat”, …) abline(v = mean) # Vertical line on the mean }

### Exercise 1

The Dairy Queen distribution centers around 200, while the McDonald’s distribution centers around 250. The spread for the McDonald’s distribution is much larger, with a longer right tail that goes to 1250. Both plot show a bell shape in their distributions.

ggplot(dairy\_queen,  
 mapping = aes(x = cal\_fat),  
) + geom\_histogram(binwidth = 20.0, colour = 'white',  
 fill = 'tomato')

A picture containing text, screenshot, diagram, plot

Description automatically generated

ggplot(mcdonalds,  
 mapping = aes(x = cal\_fat),  
) + geom\_histogram(binwidth = 20.0, colour = 'white',  
 fill = 'limegreen')

A picture containing diagram, screenshot, text, plot

Description automatically generated

### Exercise 2

Both curves are bell-shaped and unimodal. The Dairy Queen curve is fairly symmetric and appears close to normal. The McDonald’s curve appears to be right-skewed and not very normal.

mcmean <- mean(mcdonalds$cal\_fat)  
mcsd <- sd(mcdonalds$cal\_fat)  
  
dqmean <- mean(dairy\_queen$cal\_fat)  
dqsd <- sd(dairy\_queen$cal\_fat)  
  
ggplot(data = dairy\_queen, aes(x = cal\_fat)) +  
 geom\_blank() +  
 geom\_histogram(aes(y = after\_stat(density))) +  
 stat\_function(fun = dnorm, args = c(mean = dqmean, sd = dqsd), col = "tomato")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

A picture containing screenshot, text, diagram, line

Description automatically generated

ggplot(data = mcdonalds, aes(x = cal\_fat)) +  
 geom\_blank() +  
 geom\_histogram(aes(y = after\_stat(density))) +  
 stat\_function(fun = dnorm, args = c(mean = mcmean, sd = mcsd), col = "limegreen")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

A graph with a green line

Description automatically generated with low confidence

### Exercise 3

All the points in the Q-Q plot of the simulated data do not fall on the line. Compared to probability plot for the real data, the simulated data probability plot has a similar range of points that are on or close to the line around the center. At the higher end of the graph, the points drift above the line. While the points on the real data plot remain near the line on the lower end of the graph, the points on the simulated data plot drift below the line in the same area.

ggplot(data = dairy\_queen, aes(sample = cal\_fat)) +  
 # used stat\_qq instead of geom\_line(stat = "qq")  
 stat\_qq(colour = "tomato") +  
 stat\_qq\_line()

A graph with red dots

Description automatically generated with low confidence

sim\_norm <- rnorm(n = nrow(dairy\_queen), mean = dqmean, sd = dqsd)  
  
ggplot(mapping = aes(sample = sim\_norm)) +  
 stat\_qq() +  
 stat\_qq\_line()

A picture containing diagram, line, plot, screenshot

Description automatically generated

### Exercise 4

The real data plot appears very similar to all of the simulated data plots. The comparisons provide evidence that the real data for Dairy Queen’s calories form fat are nearly normal.

qqnormsim(sample = cal\_fat, data = dairy\_queen)

A picture containing text, screenshot, diagram, plot

Description automatically generated

### Exercise 5

The Q-Q plot for the real data differs from that of the simulated data. The skewness of the real data can be seen in the higher end of the graph where several points deviate above the line. This skewness is not seen in the Q-Q plot for the simulated data, as the points in the same area are on or close to the line. The simulated data plots do not show evidence that the real data is nearly normal due to the difference at the higher end of the graph.

ggplot(data = mcdonalds, aes(sample = cal\_fat)) +  
 stat\_qq(colour = "limegreen") +  
 stat\_qq\_line()

A picture containing diagram, screenshot, line, plot

Description automatically generated

mc\_sim\_norm <- rnorm(n = nrow(mcdonalds), mean = mcmean, sd = mcsd)  
  
ggplot(mapping = aes(sample = mc\_sim\_norm)) +  
 stat\_qq() +  
 stat\_qq\_line()

A picture containing diagram, line, plot, screenshot

Description automatically generated

qqnormsim(sample = cal\_fat, data = mcdonalds)

A picture containing text, screenshot, diagram, line

Description automatically generated

### Exercise 6

What is the probability of a random Taco Bell product has more than 10 grams of protein?

pnorm method : 0.8507232 empirical method: 0.826087

What is the probability of a random Sonic product has less than 10 grams of sugar?

pnorm method: 0.8105874 empirical method: 0.7924528

The probabilities for a random Sonic product having less than 10 grams of sugar are closer than those for a Taco Bell product having more than 10 grams of protein.

1 - pnorm(q = 600, mean = dqmean, sd = dqsd)

## [1] 0.01501523

dairy\_queen %>%  
 filter(cal\_fat > 600) %>%  
 summarise(percent = n() / nrow(dairy\_queen))

## # A tibble: 1 × 1  
## percent  
## <dbl>  
## 1 0.0476

# What is the probability of a random Taco Bell product has more than 10 grams of protein?  
taco\_bell <- fastfood %>%  
 filter(restaurant == "Taco Bell")  
  
1 - pnorm(q = 10, mean = mean(taco\_bell$protein), sd = sd(taco\_bell$protein))

## [1] 0.8507232

taco\_bell %>%  
 filter(protein > 10) %>%  
 summarise(percent = n() / nrow(taco\_bell))

## # A tibble: 1 × 1  
## percent  
## <dbl>  
## 1 0.826

# What is the probability of a random Sonic product has less than 10 grams of sugar?  
sonic <- fastfood %>%  
 filter(restaurant == "Sonic")  
  
pnorm(q = 10, mean = mean(sonic$sugar), sd = sd(sonic$sugar))

## [1] 0.8105874

sonic %>%  
 filter (sugar < 10) %>%  
 summarise(percent = n() / nrow(sonic))

## # A tibble: 1 × 1  
## percent  
## <dbl>  
## 1 0.792