Lab 1 Probability Distributions

Divyansh Tiwari

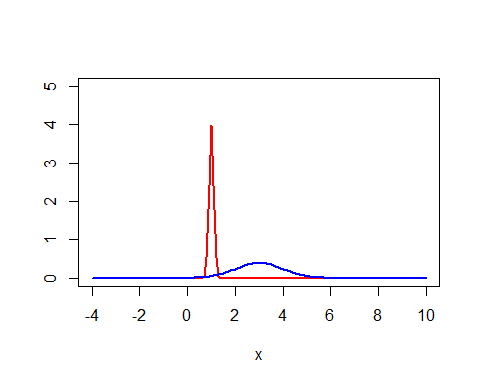
16/1/2023

Table of Contents

# Lab 1 Lab Manual Exercise

copy and paste your work by following each example from the lab manual for this exercise

rm(list = setdiff(ls(), lsf.str()))  
  
# Plot Normal Distributions with   
#-----------------------------------------  
# Same standard deviation, different mean  
#-----------------------------------------  
# Mean 1, sd 1  
# Grid of X-axis values  
x <- seq(-4, 10, 0.1)  
  
plot(x, dnorm(x, mean = 1, sd = 0.1), type = "l",  
 ylim = c(0, 5), ylab = "", lwd = 2, col = "red")  
# Mean 3, sd 1  
lines(x, dnorm(x, mean = 3, sd = 1), col = "blue", lty = 1, lwd = 2)



# # Function Syntax  
#   
# function\_name <- function(arg\_1, arg\_2, ...) {  
# Function body   
# }

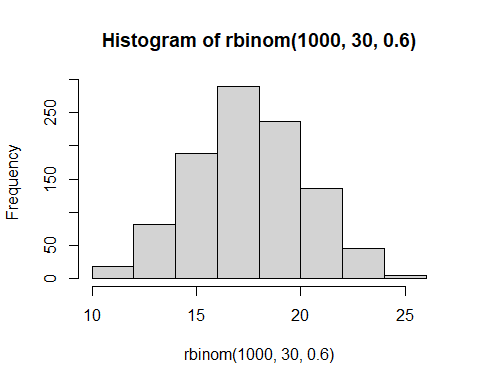
# Calculate the 60th %ile of the standard normal.  
qnorm(0.6,0,1)

## [1] 0.2533471

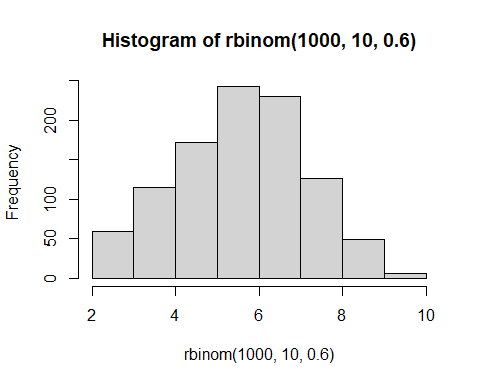
# Calculate the probability that a value lies below 0.8 in the standard normal distribution  
pnorm(0.8,0,1)

## [1] 0.7881446

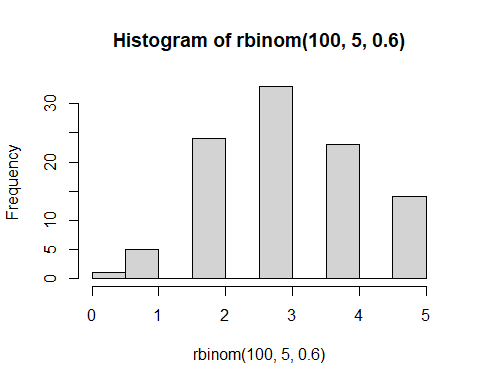
# Draw 1000 samples of 30 coin tosses with p(heads) = 0.6 # and plot the distribution  
# Syntax: rbinom (# observations, # trials per observation, probability of success )  
hist(rbinom(1000,30,0.6))



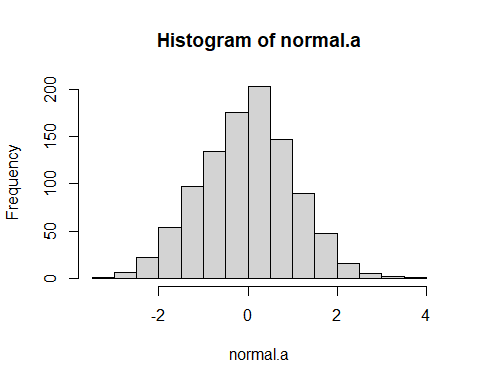
# Do the above with only 10 trials per observation  
hist(rbinom(1000,10,0.6))



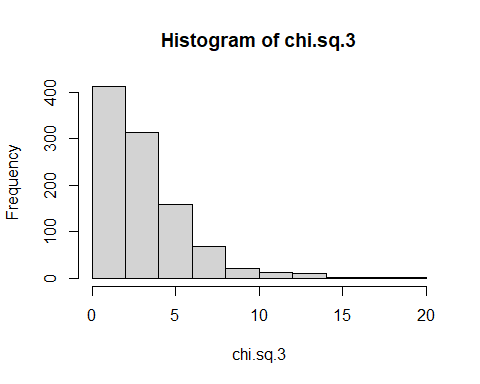
# Do the above with 100 observations and 5 trials per observation  
hist(rbinom(100,5,0.6))



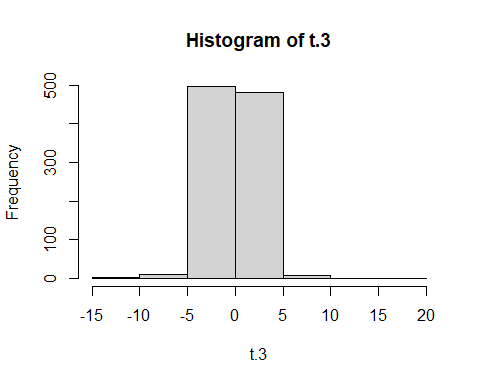
# Transformations between probability distributions  
  
# generate 1000 trials from a normal distribution  
normal.a <- rnorm( n=1000, mean=0, sd=1 )   
hist( normal.a )



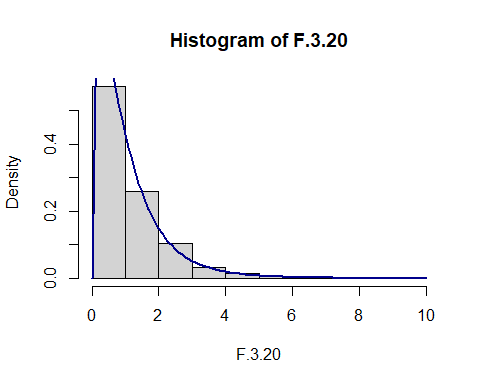
#next, we generate a chi-square distribution with 3 #degrees of freedom:  
  
normal.b <- rnorm( n=1000 ) # another set of normally distributed data  
normal.c <- rnorm( n=1000 ) # and another!  
  
# Take the SUM of SQUARES of the above 3 normally distributed variables a, b, and c  
chi.sq.3 <- (normal.a)^2 + (normal.b)^2 + (normal.c)^2   
  
# and the resulting chi.sq.3 variable should contain 1000 observations that follow a chi-square distribution with 3 degrees of freedom. You can use the hist() function to have a look at these observations yourself  
  
hist(chi.sq.3)



## Now how do we get to a t-distribution from Normal and chi-sq distributions?  
# First, take a scaled chi-sq by dividing it by the degrees of freedom  
scaled.chi.sq.3 <- chi.sq.3 / 3  
# Then take a normally distributed variable and divide them by the square root of the scaled chi-sq variable to get a t-distribution with the same degrees of freedom  
  
normal.d <- rnorm( n=1000 ) # yet another #set of normally distributed data  
t.3 <- normal.d / sqrt( scaled.chi.sq.3 ) # divide by #square root of scaled chi-square to get t  
hist (t.3)



## To get to an F distribution, take the ratio between two scaled chi-sq distributions.  
# F distribution with 3 and 20 degrees of freedom:  
# first take two chi-sq variables, with 3 dof and 20 dof respectively, and take the ratio:  
  
chi.sq.20 <- rchisq( 1000, 20) # generate chi square data with df = 20...  
scaled.chi.sq.20 <- chi.sq.20 / 20 # scale #the chi square variable...  
F.3.20 <- scaled.chi.sq.3 / scaled.chi.sq.20 # take the ratio of the two chi squares...  
hist( F.3.20, freq = FALSE) # ... and draw a picture  
curve(df(x, 3, 20),   
 col="darkblue", lwd=2, add=TRUE, yaxt="n")



## The curve above confirms this looks similar if you use the R built-in function df (just like dnorm, but for the F distribution)

# Lab 1 Generalization exercises

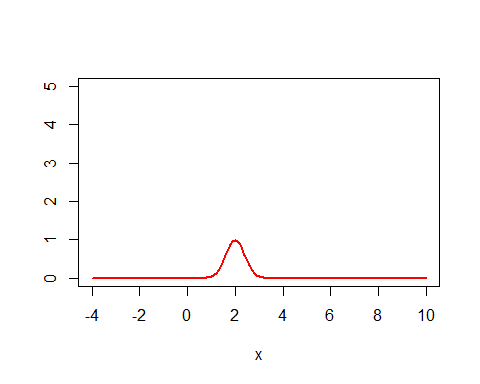
use the code from above to attempt to solve the extra things we ask you do for this assignment

# Q1 Plot a normal distribution with mean = 2, s.d. = 0.4  
  
  
# Q2 Calculate the 85th %ile of the above distribution.  
  
  
# Q3 Calculate the probability that a value lies in between 1 and 2 given the above distribution  
  
# Q4 Plot a simulated t-distribution with 5 degrees of freedom.  
  
# Q5 Plot a chi-sq distribution with 5 degrees of freedom.

# Lab 1 Written answer question

The following is the code for Q1

x <- seq(-4, 10, 0.1)  
plot(x, dnorm(x, mean = 2, sd = 0.4), type = "l",  
 ylim = c(0, 5), ylab = "", lwd = 2, col = "red")



The following is the code for Q2

qnorm(0.85, mean=2, sd=0.4)

## [1] 2.414573

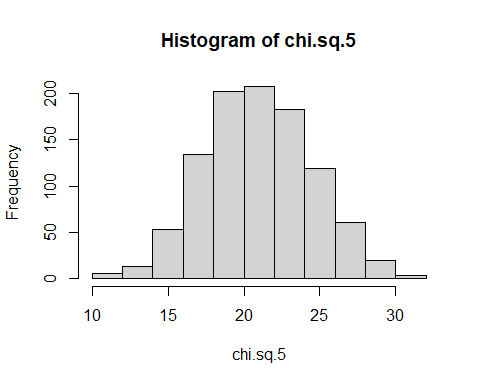
The following is the code for Q3

a <- pnorm(1, 2, 0.4)  
b <- pnorm(2, 2, 0.4)  
prob.between.1.and.2 <- b - a  
print(prob.between.1.and.2)

## [1] 0.4937903

The following is the code for Q4

normal.a <- rnorm( n=1000, mean=2, sd=0.4 )   
normal.b <- rnorm( n=1000 , mean=2, sd=0.4 )   
normal.c <- rnorm( n=1000 , mean=2, sd=0.4 )   
normal.d <- rnorm(n =1000 , mean=2, sd=0.4)  
normal.e <- rnorm (n = 1000 , mean=2, sd=0.4)  
chi.sq.5 <- (normal.a)^2 + (normal.b)^2 + (normal.c)^2 + (normal.d)^2 + (normal.e)^2  
hist(chi.sq.5)



scaled.chi.sq.5 <- chi.sq.5 / 5  
normal.f <- rnorm(n = 1000 , mean=2, sd=0.4)  
t.5 <- normal.f / sqrt( scaled.chi.sq.5 )   
hist (t.5)

