**Module 4 Homework**

Chengbo Gu

**Problem 1. (20 points)**

are independent random samples from a distribution with mean 5 and standard deviation 3. Complete the following:

**(a)** For the sample mean, find its mean  and standard deviation **.

****

****

****

**(b)** Can you find the  approximately using CLT? If yes, what is your estimate for ? If no, why not?

No, we can’t

**Problem 2. (20 points)**

Suppose that for certain microRNA of size 20 the probability of a purine is binomially distributed with probability 0.7. Say there are 100 such microRNAs, each independent of the other. Let Y denote the average number of purine in these microRNAs. Find the probability that Y is great than 15. Please give a theoretical calculation, do NOT use Monte Carlo simulation to approximate. Show all the steps and formulas in your calculation.

Let  denote the number of purine in the 20 microRNAs.

Then we have .

Thus,





According to CLT, 



meanX **<-** 20\*0.7

VarX **<-** 20\*0.7\*0.3

print(1-pnorm(15, mean=meanX, sd=sqrt(VarX)/10))

[1] 5.317746e-07

**Problem 3. (20 points)**

monte carlo

**Problem 4. (20 points)**

Assume there are three independent random variables,, with m=3 degrees of freedom.

Define a new random variable Y as .  

Use Monte Carlo simulation to find the mean of Y. Submit your R script for the Monte Carlo simulation, and a brief summary of the actual simulation results.

x1 **<-** rchisq(10000, 10)

x2 **<-** rgamma(10000, 1, 2)

x3 **<-** rt(10000, 3)

y **<-** sqrt(x1)\*x2+4\*(x3^2)

print(mean(y))

[1] 13.5784

Summary

**Problem 5. (20 points)**

Complete exercise 10 in Chapter 3 of *Applied Statistics for Bioinformatics using R* (page 45-46). Submit the plot, and a brief explanation of your observation.

n **<-** 1000

a **<-** **function**(n) {sqrt(2\*log(n)) - 0.5\*(log(log(n)) + log(4\*pi)) \* (2\*log(n))^(-1/2)}

b **<-** **function**(n) {(2\*log(n))^(-1/2)}

an **<-** a(n)

bn **<-** b(n)

res **<-** rep(NA, n)

**for** (i **in** 1:n){

res[i] = (max(rnorm(n, 0, 1))-an)/bn

}

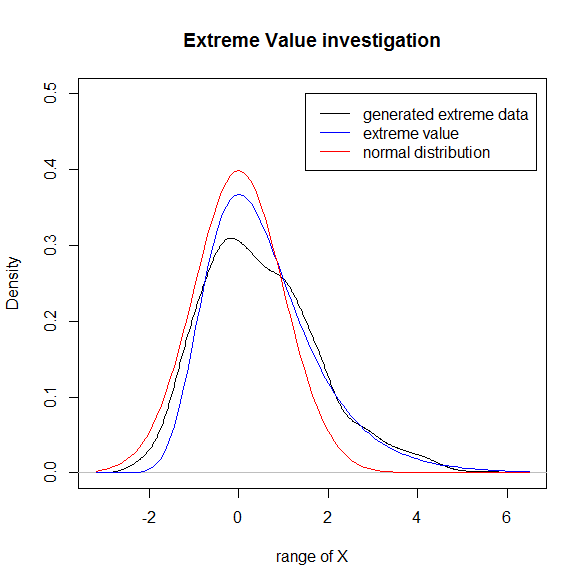
plot(density(res), ylim=c(0,0.5), xlab="range of X", main="Extreme Value investigation")

f **<-** **function**(x) {exp(-x)\*exp(-exp(-x))}

curve(f, range(density(res)$x), add=TRUE, col = "blue")

curve(dnorm, add=TRUE, col = "red")

legend(1.5,0.5, c("generated extreme data", "extreme value", "normal distribution"), lty = "solid", col = c("black","blue","red"))



The extreme value fits to the density of generated extreme data much better than

the red line.