STA511 Final Homework

Abbas Rizvi

December 14, 2015

1 Solutions

1. Random normal variables were generated using the Laplace distribution given by:

$$g(x) = \frac{\theta}{2}e^{-\theta|x|}, \text{ for } \theta > 0$$

(a) The optimal rejection constant c is `1.315` and the optimal θ is `1.0001` when $c = sup\frac{f(x)}{g(x)}$ where $f(x)$ is the standard normal pdf $\sim N(0,1)$.

$$\begin{aligned}
c =& sup\frac{f(x)}{g(x)} \\
c =& sup\left(\frac{\frac{1}{\sqrt{2}\pi}e^{-x^2/2}}{\frac{\theta}{2}e^{-\theta|x|}}\right) \\
c =& \frac{\sqrt{\frac{2}{\pi}}e^{\frac{-x^2}{2}-\theta|x|}}{\theta} \\
c =& \begin{cases} \frac{\sqrt{\frac{2}{\pi}}e^{\frac{-x^2}{2}+\theta x}}{\theta}, & \text{x } \geq 0 \\ \frac{\sqrt{\frac{2}{\pi}}e^{\frac{-x^2}{2}-\theta x}}{\theta}, & \text{x } < 0 \end{cases}
\end{aligned}$$

Now we take the derivative of the upper piecewise function with respect to x. Afterwards we set the derivative to zero and solve for θ.

$$\begin{aligned}
c =& \frac{\sqrt{\frac{2}{\pi}}}{\theta}\left[e^{\frac{-x^2}{2}+\theta x}\frac{d}{dx}\right] \\
=& \frac{\sqrt{\frac{2}{\pi}}e^{\frac{-x^2}{2}+\theta x}\cdot(-x+\theta)}{\theta} = 0 \\
\theta =& x, \text{ for } x \geq 0
\end{aligned}$$

Now we take the derivative of the lower piecewise function with respect to x. Afterwards we set the derivative to zero and solve for θ.

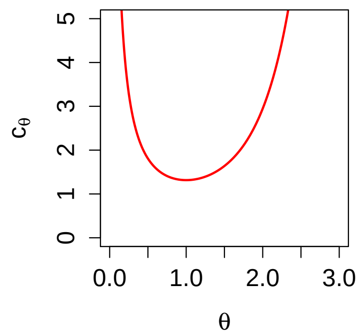
$$\begin{aligned}
c =& \frac{\sqrt{\frac{2}{\pi}}}{\theta}\left[e^{\frac{-x^2}{2}-\theta x}\frac{d}{dx}\right] \\
=& \frac{\sqrt{\frac{2}{\pi}}e^{\frac{-x^2}{2}-\theta x}\cdot(-x-\theta)}{\theta} = 0 \\
\theta =& -x, \text{ for } x < 0
\end{aligned}$$

Each critical point was subsequently plugged into c_θ and both yielded the following:

$$c = \frac{\sqrt{\frac{2}{\pi}}e^{\frac{\theta^2}{2}}}{\theta}, \text{ for } -\infty < 0 < \infty \text{ and } \theta > 0$$

The optimal θ was determined by minimzing the $c \geq \frac{f(x)}{g(x)}$ curve with respect to θ. The function was computed with random variables and a corresponding plot was produced in `R` (Figure 1). The minimum c (`1.315489`) and optimal θ (`1.0001`) were computed using `R`'s `nlminb` function.

Figure 1



(b) 1000 observations were generated from $N(0,1)$ using a generalized rejection method. In order to generate these observations, we have to sample from the Laplace distribution pdf g(x). Since we are sampling from a continuous distribution, in order to sample from the pdf, we need to first use the inversion method. The inversion method takes the quantile function of Laplace CDF G(x) to generate random variables of the Laplace pdf. The random variables then were sampled using the generalized rejection method, with `0.752` or 75.2% of the observed random variables being accepted.

$$G(x) = \begin{cases} \frac{1}{2}e^{x} & x < 0 \\ 1 - \frac{1}{2}e^{-x} & x \geq 0 \end{cases}$$

$$G^{-1}(p) = \begin{cases} ln(2p) & p \in [0, \frac{1}{2}] \\ -ln[2(1-p)] & p \in [\frac{1}{2}, 1] \end{cases}$$

(c) A goodness of fit (Kolmogorov-Smirnov) test was conducted on the 1000 generated samples were consistent with a standard Normal Distribution. The null hypothesis for the K-S test is that the accepted (observed) Laplace random variables from the generalized rejection method were drawn from a normal distribution. The K-S test has a p-value of `0.43`, suggesting that we failed to reject the null hypothesis, colloquially meaning, that the Laplace random variables are likely to have been drawn from a standard normal distribution. This can be seen visually in Figure 2b.

2. I elected not to do problem 2.

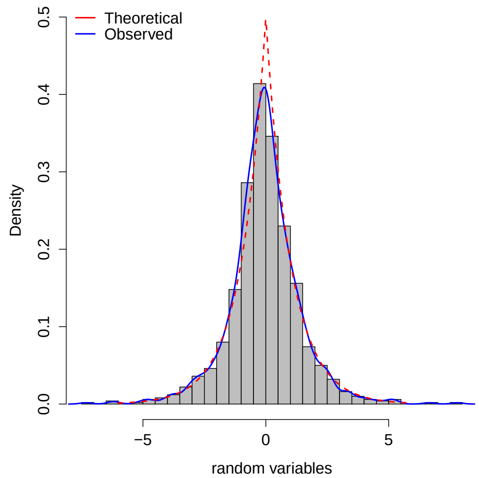
3. The distribution described by the following density function was considered:

$$f(x) = \begin{cases} x+1 & x \in [-1, 0] \\ -x+1 & x \in [0, 1] \\ 0 & \text{elsewhere} \end{cases}$$

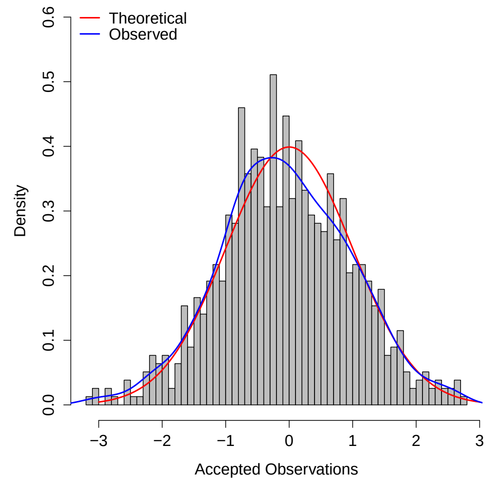
(a) The rejection method requires that pdf f be bounded and nonzero only on some finite interval $[a, b]$. The rejection constant (c) was determined by $c = max\{f(x); a \leq x \leq b\}$. The following algorithm was implemented via `R`:

1. Generating Xuniform on (a, b).
2. Generating Yuniform on $(0, c)$.
3. If Y $\leq f(x)$, then output$X$, otherwise go to 1.

Figure 2



(a) Inversion Method



(b) Generalized Rejection Method

100 random observations were generated from f using simple rejection sampling. The accepted observations were `0.59` normalized to total number of generated observations.

(b) The inversion method involves sampling the PDF using the quantile of its CDF. We determined the CDF of the density function and then subsequently computed its corresponding quantile function. Since this given density follows a triangular distribution, the CDF and quantile function are shown as:

$$F(x) = \begin{cases} \frac{(x+1)^2}{2} & \text{for } -1 < x \leq 0 \\ 1 - \frac{(-x+1)^2}{2} & \text{for } 0 < x < 1 \end{cases}$$

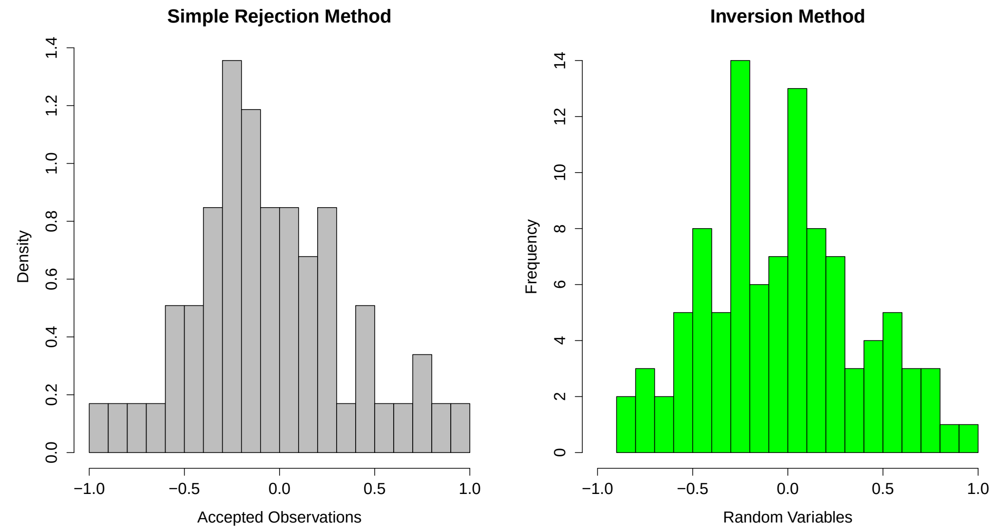
$$F^{-1}(u) = \begin{cases} \sqrt{2u} - 1 & u \in [0, \frac{1}{2}] \\ 1 - \sqrt{2(-u+1)} & u \in [\frac{1}{2}, 1] \end{cases}$$

(c) The histograms for part (a) and part (b) can be visualized in Figure 3a and Figure 3b, respectively.

4. Simulations were conducted such that a coin was flipped until a pattern emerged. Pattern 1 was given as head (H), tail (T), tail (T), and pattern 2 was head (H), tail (T), head (T). The first simulation was the average number of tosses needed to obtain pattern 1 or pattern 2. The second simulation simulated 100,000 flips and counted the number occurrences of pattern 1 and 2.

   (a) 1,000 iterations were conducted for the first simulation. The average number of flips needed until pattern 1 (HTT) emerged was `7.138`. The average number of flips needed until pattern 2 (HTH) emerged was `9.491`. This answer does not surprise me. Intuitively, pattern 1 should occur in fewer flips because after 2 flips if the pattern is H-T, if a tail emerges as the subsequent flip in the sequence, the desired pattern (pattern 1) has been reached, but if heads emerges, the pattern is not penalized as much as pattern 2 (HTH) because H is the first flip (position 1) needed for pattern 1. However, on the contrary, if the sequence of flips was already H-T, pattern 2 would be achieved with a 3rd flip of H, but if it is T, the flips continue until a H is achieved to start the desire pattern again.

Figure 3



(a) Simple Rejection Method   (b) Inversion Method

(b) A simulation with 10 iterations was conducted in order to count how many times pattern 1 and pattern 2 emerged in 100,000 flips. Pattern 1 emerged 12540/100000 flips (12.54%) and pattern 2 emerged 12472/100000 flips (12.47%). These values are quite close, with pattern 1 still occuring more often. I thnk that these results more or less agree with part (a) because the probability of each outcome per flip is 0.5, so with such a large sample population, I would expect the number of occurrences to be close to one another, with a slight edge towards the more likely pattern (pattern 1).

5. Let $X_1, ..., X_n \sim$ Uniform(a,5) where a is an unknown parameter.

(a) The methods of moments estimator for a was found to be:

$$\begin{aligned} E[X] =& \bar{X} \\ \frac{a-5}{2} =& \sum_{i=1}^{n} \frac{X_i}{n} \\ \tilde{a}_{MOM} =& \frac{2\sum_{i=1}^{n} X_i}{n} - 5 \end{aligned}$$

(b) The MLE for α was found to be:

$$\begin{aligned} MLE =& (\hat{a}, \hat{b}) = (\min\left\{X_{[1]}\right\}, \max\left\{X_{[n]}\right\}) \\ L(X_i|a) =& \prod_{i=1}^{n} \left[\frac{1}{5-\alpha}\right]^n, \text{ for } \alpha \leq x \leq 5 \\ =& \left[\frac{1}{5-a}\right]^n, \text{ for } \alpha \leq \min\left\{X_{[i]}\right\} \leq \max\left\{X_{[n]}\right\} \leq 5 \\ \hat{a}_{MLE} =& X_{[1]} \end{aligned}$$

(c) $\tau = E(X) = \int_a^5 x \cdot f(x)dx$ where $f(x) = \frac{1}{5-a}$. The MLE for τ was found to be:

$$
\begin{aligned}
\tau = E(X) &= \int_a^5 x \cdot f(x) dx \\
&= \int_a^5 x \cdot \frac{1}{5-a} dx \\
&= \frac{1}{5-a} \int_a^5 x dx \\
&= \frac{1}{5-a} \left[\frac{x^2}{2} \Bigg|_a^5 \right] \\
&= \frac{1}{5-a} \left(\frac{5^2 - a^2}{2} \right) \\
&= \frac{1}{5-a} \left(\frac{(5-a)(5+a)}{2} \right) \\
\hat{\tau}_{MLE} &= \frac{5 + \hat{a}_{MLE}}{2}
\end{aligned}
$$

(d) Let $\hat{\tau}$ be the MLE of τ. Let $\tilde{\tau}$ be the MOM based estimator of $\tau = E(X)$.

$$
\begin{aligned}
\hat{\tau}_{MLE} &= \frac{5 + X_{[1]}}{2} \\
\tilde{\tau}_{MOM} &= \frac{5 + 2\sum_{i=1}^{n} X_i}{2n}
\end{aligned}
$$

We suppose that $a = 1$ and $n = 10$.

$$
\bar{X} = \frac{\sum_{i=1}^{n=10} X_i}{n}
$$

The coverage probability was computed for a 95% confidence interval for τ bootstrap method with $\hat{\tau}$. The confidence intervals were:

$$
\begin{aligned}
\text{Normal CI: } & (2.83, 3.24) \\
\text{Pivotal CI: } & (2.47, 3.04) \\
\text{Percentile CI: } & (3.04, 3.61)
\end{aligned}
$$

This was compared to the coverage prbability for a 95% interval for τ using the same bootstrap method but for $\tilde{\tau}$ instead of $\hat{\tau}$. The confidence intervals were:

$$
\begin{aligned}
\text{Normal CI: } & (1.90, 3.39) \\
\text{Pivotal CI: } & (1.90, 3.36) \\
\text{Percentile CI: } & (1.94, 3.39)
\end{aligned}
$$

2 Appendix

1. ```
#part a
foverg <- function(theta){sqrt(2/pi)*exp(theta^2/2)/theta}

theta <- seq(0.000001, 10, length=1000)

opt.theta <-nlminb(0.00001,function(theta){foverg(theta)})$par
c <- nlminb(0.00001,function(theta){foverg(theta)})$objective

plot(theta,(sqrt(2/pi)*exp(theta^2/2))/theta,type="l", col="red",lwd=2,ylim=c(0,5),xlim=c(0,3),
     ylab=expression(c[theta]),xlab=expression(theta))

#part b
#inversion method
laplace.inv <- function(p){
       if(p < 1/2){
               log(2*p, base=exp(1))
       } else {
               -log(2*(1-p), base=exp(1))
       }
}

p <- runif(1000)
laplace.rv <- sapply(p,laplace.inv) #random variables from laplace
hist(laplace.rv)
laplace.dist <- function(x){ #theta as 1
        if(x<0){
                (1/2)*exp(x)
        }else {
                (1/2)*exp(-x)
        }
}

rvs <- seq(-6,6,length=1000)
laplace.pdf <- sapply(rvs,laplace.dist)
hist(laplace.rv, ylim=c(0,0.5), col="gray", xlab="random variables", prob=T, breaks=50, main="")
lines(density(laplace.rv), col="blue", lwd=2)
lines(rvs, laplace.pdf, col="red", lwd=2, lty=2)
legend("topright",legend=c("Theoretical PDF","Observed PDF"),
       col=c("red", "blue"),lwd=c(2,2), bty="n")

#generalized rejection method
goverf <- function(x,theta=1){y = sqrt(pi/2)*theta*exp((x^2/2)-theta*abs(x))}
xc <- laplace.rv
uc <- runif(1000,0,1)
tc <- c*sapply(xc,goverf)
ut <- uc*tc
accept.obs <- xc[ut <= 1]
length(accept.obs)/1000

nseq <- seq(-3,3,length=1000)
norm <- dnorm(nseq)
```
```

```
hist(accept.obs, col="gray", prob=T, breaks=50, ylim=c(0,0.6),
xlab = "Accepted Observations", main="")
lines(nseq, norm, col="red", lwd=2)
lines(density(accept.obs), col="blue", lwd=2)
legend("topright",legend=c("Theoretical PDF","Observed PDF"),
       col=c("red", "blue"),lwd=c(2,2), bty="n")


#part c
ks.test(accept.obs, "pnorm", 0,1)

        One-sample Kolmogorov-Smirnov test

data:  accept.obs
D = 0.0319, p-value = 0.43
alternative hypothesis: two-sided
```

2. Did not do question 2.

3. ```
####question 3####
#part a
rm(list=ls())
set.seed(333)
f.pdf <- function(x){
        if(-1 <= x & x <= 0){
                x + 1
        }else if(0 <= x & x <= 1){
                -x + 1
        }
        else{
                0
        }
}
c.max <- optimize(f.pdf, interval=c(-1,1), maximum=TRUE)$objective
#simple rejection
xc <- runif(100,-1,1) #generate x uniform[a,b]
uc <- runif(100,0,1) #generate uniform [0,1]
tc <- c.max/sapply(xc,f.pdf) #t = c/fx
ut <- uc*tc
accept.obs <- xc[ut <= 1]
length(accept.obs)/100 #accept 50.6%

#part b
#inversion method
icdf <- function(u){
        if(0 < u & u < 0.5){
                sqrt(2*u)-1
        }else if(0.5 <= u & u < 1){
                1 - sqrt(2*(-u+1))
        }else{
                u=0}
}
rand.unif <- runif(100,0,1)
rvs <- sapply(rand.unif, icdf)
```
```

```
hist(rvs, col="green", breaks=20, main="Inversion Method", xlab="Random Variables", xlim=c(-1,1))

#part c is in a and b
```

4.
```
#part a
rm(list=ls())
require(zoo)
set.seed(333)
coinflip <- function(seq){
        flip <- c()
        ht <- c()
        match <- FALSE
        while (any(match) == FALSE){
                flip <- sample(c("H", "T"), 1, replace = TRUE, prob=c(0.5,0.5))
                ht <- c(ht,flip)
                if (length(ht) > 3){
                        match <- rollapply(ht,3,identical,seq)
                }
        }
        return(length(match)+1)
}

num.trials <- 1000
trials <- data.frame("hth"=rep(NA,num.trials),"htt"=rep(NA,num.trials))
for (i in 1:num.trials){
        trials[i,] <- c(coinflip(c("H","T","H")), coinflip(c("H","T","T")))
}
summary(trials)

      hth              htt
 Min.   : 3.000   Min.   : 3.000
 1st Qu.: 4.000   1st Qu.: 3.000
 Median : 7.000   Median : 6.000
 Mean   : 9.491   Mean   : 7.138
 3rd Qu.:13.000   3rd Qu.: 9.000
 Max.   :59.000   Max.   :47.000

#part b
rm(list=ls())
set.seed(333)
require(zoo)
num.trials <- 10
trials <- data.frame("hth"=rep(NA,num.trials),"htt"=rep(NA,num.trials))
for (j in 1:num.trials){
        coin <- c('h','t')
        s <- sample(x=coin, size=100000, replace=T, prob=c(0.5,0.5))
        hth <- length(which(rollapply(s,3,identical,c("h","t","h"))))
        htt <- length(which(rollapply(s,3,identical,c("h","t","t"))))
        trials[j,] <- c(hth,htt)
}
>summary(trials)
      hth             htt
 Min.   :12292   Min.   :12447
 1st Qu.:12436   1st Qu.:12509
```

```
 Median :12472   Median :12548
 Mean   :12478   Mean   :12540
 3rd Qu.:12553   3rd Qu.:12570
 Max.   :12587   Max.   :12625
```

5.
```
#########################
####### question 5D ######
#########################
rm(list=ls())
set.seed(333)
x.i <- runif(10,1,5)
tau.mle <- (5+min(x.i))/2
tau.mom <- mean(x.i)
t.mle <- c()
t.mom <- c()
for(i in 1:10000){
        boot.obs <- sample(x.i, 10, replace = T)
        t.mle[i] <- (5+min(boot.obs))/2
        t.mom[i] <- mean(boot.obs)
}
se.mle = sqrt(var(t.mle))
se.mom <- sqrt(var(t.mom))

Normal.mle = c(tau.mle-2*se.mle, tau.mle+2*se.mle)
Normal.mom = c(tau.mom-2*se.mom, tau.mom+2*se.mom)

pivotal.mle = c(2*tau.mle-quantile(t.mle,.975),2*tau.mle-quantile(t.mle,.025))
pivotal.mom = c(2*tau.mom-quantile(t.mom,.975),2*tau.mom-quantile(t.mom,.025))

percentile.mle <- c(quantile(t.mle,0.025),quantile(t.mle,0.975))
percentile.mom <- c(quantile(t.mom,0.025),quantile(t.mom,0.975))
```