HW 3

Jared Yu

May 3, 2018

```r
# 3

set.seed(243)
u = runif(5000)
x = log(1/(1-u*(1-exp(-2))))
plot(density(x), main = "Sample of 5000 Observations")

# Q4(a)

q = function(x) exp(-x)/(1+x^2)
g1 = function(x) exp(-x)
g2 = function(x) 2/(pi*(1+x^2))

# using g1 to sample f(x)

set.seed(243)

f_g1 = rep(0, 5000)

alpha = 1
count = 0

ptm <- proc.time()
while (any(f_g1==0)) {
  x = rexp(1)
  u = runif(1)

  if (u <= q(x)/(alpha*g1(x))) {
    count = count + 1
    f_g1[count] = x
  }
}
t1 = proc.time() - ptm # time run for this section
# user  system elapsed
# 0.24    0.01    0.31

# plots

plot(density(f_g1), col="blue", lty=1, main = "Estimated f(x) using g1", xlim=c(0,5))
lines(density(rexp(5000)), col="red", lty=2)
legend("topright", c("estimate f(x)", "g1(x)"), col=c("blue", "red"), lty=c(1,2))

# using g2 to sample f(x)

set.seed(243)

f_g2 = rep(0, 5000)
```

```
alpha = pi/2
count = 0

ptm <- proc.time()
while (any(f_g2==0)) {
  x = abs(rcauchy(1))
  u = runif(1)

  if (u <= q(x)/(alpha*g2(x))) {
    count = count + 1
    f_g2[count] = x
  }
}
t2 = proc.time() - ptm # time run for this section
# user  system elapsed
# 0.44    0.06    0.62

# plots

plot(density(f_g2), col="blue", lty=1, main = "Estimated f(x) using g2", xlim=c(0,5))
lines(density(abs(rcauchy(5000))), col="red", lty=2)
legend("topright", c("estimate f(x)", "g2(x)"), col=c("blue", "red"), lty=c(1,2))

# plot both estimated f on the sample figure

plot(density(f_g1), col="blue", lty=1, main = "Estimated f(x)", xlim=c(0,5))
lines(density(f_g2), col="red", lty=2)
legend("topright", c("using g1", "using g2"), col=c("blue", "red"), lty=c(1,2))

# Q5(d)

# q-function

q = function(x, theta){
 return(sqrt(4+x)*(x^(theta-1))*exp(-x))
}

# envelope function, alpha*g

ag = function(x, theta){
  alpha = gamma(theta)+0.5*gamma(theta+0.5)
  return(alpha*(2*(x^(theta-1))+x^(theta-0.5))*exp(-x))
}

# function to sample a value from mixture gamma model (mgamma)

rmgamma = function(theta){
  u = runif(1)
  if (u < (2*gamma(theta))/(2*gamma(theta)+gamma(theta+0.5))) {
```

```
    x = rgamma(1, shape = theta, scale = 1)
  } else {
    x = rgamma(1, shape = theta+0.5, scale = 1)
  }
  return(x)
}

# function to simulate n values from f(x)

rf = function(n, theta){
  xvec = rep(0, n)
  counter = 0
  while (counter < n){
    u = runif(1)
    xg = rmgamma(theta) # a value from mixture gamma
    if (u < q(xg, theta)/ag(xg, theta)) {
      counter = counter + 1
      xvec[counter] = xg
    }
  }
  return(xvec)
}

# perform the simulations for all three thetas

set.seed(2018)

n = 1000
thetas = c(0.5, 1, 1.5)

sample_f = vector("list", length(thetas))

for (i in 1:length(thetas)){
  sample_f[[i]] = rf(n, thetas[i])
}

# graph the estimated densities


for (i in 1:length(thetas)){
  if (i==1) par(mfrow=c(1,length(thetas)))
  plot(density(sample_f[[i]]), ylim=c(0, 1.5), xlim=c(0,10), main = paste("theta =", t
hetas[i]))
  if (i==length(thetas)) par(mfrow=c(1,1))
}
```