HW4

Sebastian Kirkpatrick

2024-10-28

Question 1

```
samp <- function(x, k) {
  n <- length(x)
  y <- c()

  for (i in 1:k) {
    j <- as.integer(runif(1,min=1,max=n-i+2))
    y[i] <- x[j]
    x[j] <- x[n + 1 - i]
  }

  return(y)

}
```

```
x <- c(0,4,2,10)
samp(x,3)
```

```
## [1]  4  0 10
```

Question 2

a)

N $\sim Poisson(\lambda = 10)$

b)

$X_i \sim Geometric(p = 0.8)$

c)

Y $= \sum_{i=1}^{N} X_i$

```r
n <- function(lam,n) {
  u <- runif(n)
  x <- qpois(u,lam)
  return(x)
}

xi <- function(p,n) {
  u <- runif(n)
  x <- qgeom(u,p)
  return(x)
}

yt <- function(p,lam,nsim) {
  Y <- c()
  N <- n(lam,nsim)

  for (i in 1:nsim) {
    XI <- xi(p, N[i])
    Y[i] <- sum(XI)
  }

  return(Y)
}


ez <- function(p, lam, nsim) {
  shoes <- c()
  c <- rpois(nsim, lam)
  for (i in 1:nsim) {
    shoe <- rgeom(c[i], p)
    shoes[i] <- sum(shoe)
  }
  return(shoes)
}

mean(yt(0.8, 10, 100000))
```

```
## [1] 2.5051
```

```r
mean(ez(0.8, 10, 100000))
```

```
## [1] 2.5045
```

Just checking with the built in functions to make sure my runif() + cdf version works properly

d)

The probability of Y = 0 is the combined probability that no one comes in or that everyone who does come in buys the only pair they try on, which is 0.8^n, where n is the number of customers that day. $E[0.8^n] =$

```r
exp(10 * (0.8 -1))
```

```
## [1] 0.1353353
```

```r
set.seed(740)
mean(yt(0.8, 10, 1000000)==0)
```

```
## [1] 0.135622
```

```r
mean(ez(0.8, 10, 1000000)==0)
```

```
## [1] 0.135373
```

$P(Y=0) \approx .135$

Question 3

a)

```r
g <- function(x) {
  -(x^4) + 16
}

q3 <- function(n) {
  x <- runif(n,-2,2)
  y <- g(x)
  c <- 4 * (1/n) * sum(y)
  return(c)
}

set.seed(740)
q3(100000)
```

```
## [1] 51.18061
```

$C \approx 51.18$

b)

```r
pdf <- function(x) {
  out <- (16 - x^4) / 51.18
  return(out)
}

cdf <- function(x) {
  ((16 * x) / 51.18) - ((x^5) / (51.18 * 5)) + 0.5
}
```

```
icdf <- function(p) {
  f <- function(x, p) {
    abs(cdf(x) - p)
  }

  optimize(f, lower=-2, upper=2, p=p)$minimum
}

icdf <- Vectorize(icdf)


set.seed(740)
u <- runif(100)
icdf(u)
```

```
##   [1]  0.04997858 -0.43730005  1.41108533  0.73280249 -0.19118418  1.96564167
##   [7] -0.39448144  0.90551454  0.18850769  0.56017195  1.13744044  0.22803197
##  [13]  1.46345654  0.34554349 -0.18304689 -0.88231000  0.93052172  0.04374609
##  [19] -0.41530765  0.99593052 -1.71634194 -0.98273171  1.10088938 -0.11979915
##  [25]  0.77689664  0.39289558 -0.14725946 -1.26564147 -0.56118708  0.94163043
##  [31] -1.79620552  1.59030911 -0.55336414 -0.35932316  0.24064344  1.85886246
##  [37]  1.57271814  0.86010671 -0.76633489 -0.48343664  0.26917772  1.12328112
##  [43] -0.99554825  1.47763446  0.88375324 -0.53745055 -0.32408159 -0.39724291
##  [49]  0.13386456 -0.53854514  0.69669114  0.49627018  1.01556354  0.39667227
##  [55]  0.84568542  0.46525189  1.46713372 -1.65144710  0.06487891 -0.52580086
##  [61]  1.89129008 -0.78766645 -1.74434192 -0.09315115  1.05946281 -0.24727928
##  [67] -1.04487327 -0.23983031 -0.53376415  0.98619680 -0.63794595  0.56253725
##  [73] -0.21018280 -1.30670324 -0.50209258  0.24580028  0.80105924  0.55106862
##  [79] -0.59350750 -1.08686777  0.31632353  0.02434288 -1.39603073  0.02007432
##  [85] -0.46319585  0.41377825 -1.22889965 -1.41556700  1.47791878 -1.46722746
##  [91] -0.82146611 -0.91856760 -1.05510632  0.47157605  0.82671275  1.01145207
##  [97] -1.19417897  1.20457084  0.59598836  0.45864055
```

```
min(icdf(u))
```

```
## [1] -1.796206
```

```
max(icdf(u))
```

```
## [1] 1.965642
```

c)

```
pdf(0)
```

```
## [1] 0.3126221
```

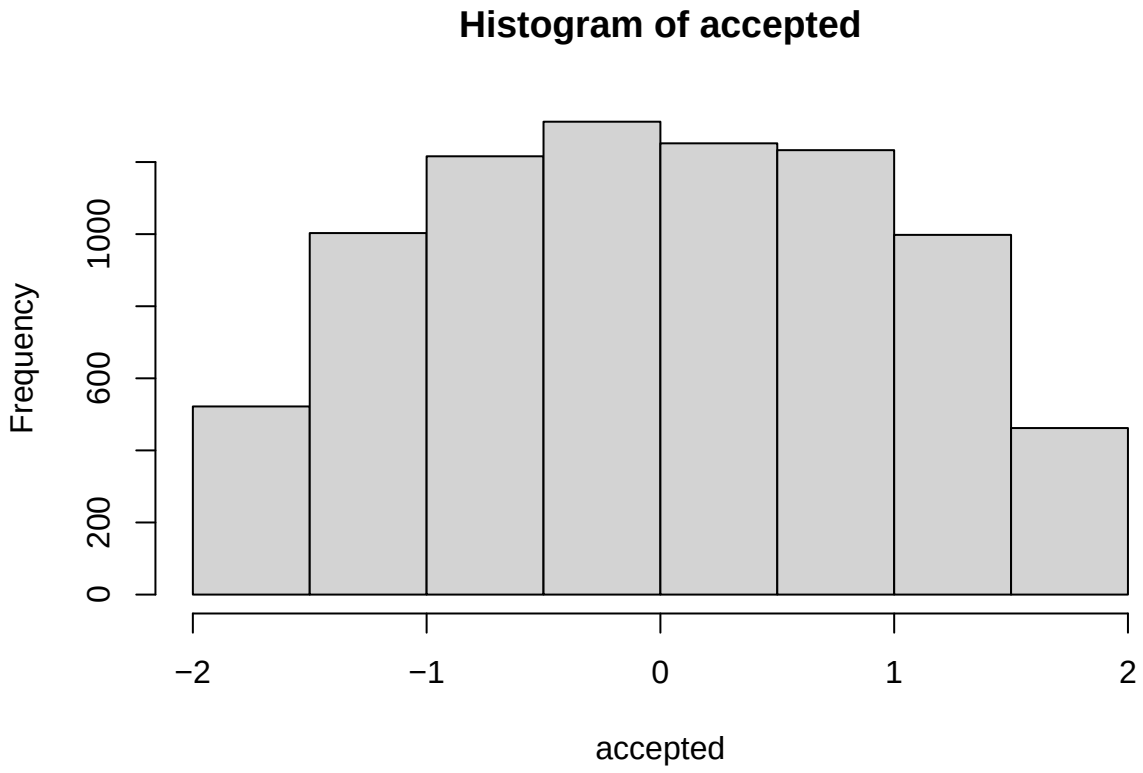
```
region <- function(n) {
  val <- c()
  x <- runif(n, -2, 2)
```

```
  y <- runif(n, 0, pdf(0))

  for (i in 1:n) {
    if (y[i] < pdf(x[i])) {
      val <- c(val, x[i])
    }
  }

  return(val)
  }

accepted <- region(10000)
hist(accepted)
```



Question 4

a)

```
stupid_grad_question <- function(n, alpha) {
  u <- runif(n)
  x <- alpha * tan((u - 0.5) * pi)
  return(x)
}
```

```
stupid_grad_question(10, 1)
```

```
##  [1] 166.59665384  -0.06179180  -0.03805564  -0.03180655  -0.01345288
##  [6]   0.07410464   0.27132688  -0.37382717  20.25270915  -0.40858393
```

b)

```
a <- dnorm(0)
b <- dcauchy(0)

a/b
```

```
## [1] 1.253314
```

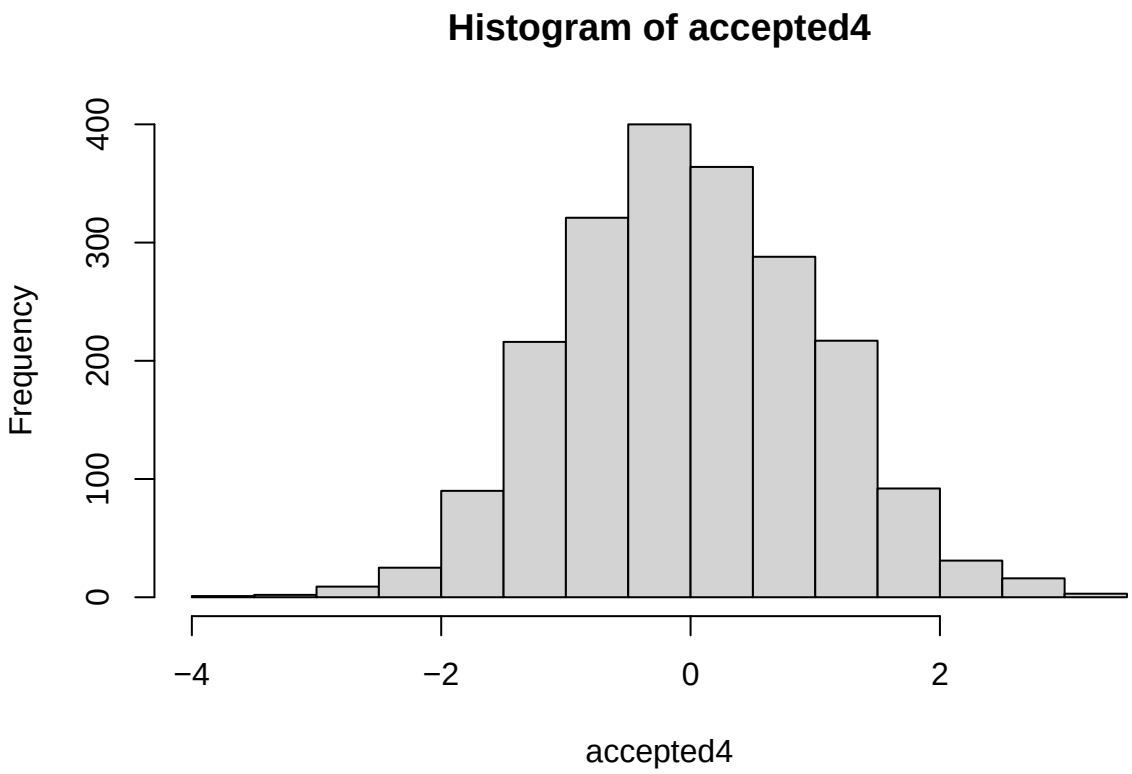
```
region4 <- function(n, alpha) {
  val <- c()
  x <- runif(n,-5,5)
  y <- runif(n,0, 1.5 * dcauchy(0))

  c_pdf <- function(x,alpha){
    return(alpha / ((alpha^2 + x^2) * pi))
  }

  for (i in 1:n) {
    if (y[i] < dnorm(x[i])) {
      val <- c(val, x[i])
    }
  }

  return(val)
  }

accepted4 <- region4(10000, 1)
hist(accepted4)
```



Not a fan of this hw. We can talk on the plane for better ideas