

## lesson 03

曾舸舵

2022/3/1

### 3.3

由于给出了 cdf

$$1 - \left(\frac{b}{x}\right)^a$$

, 可知 pdf

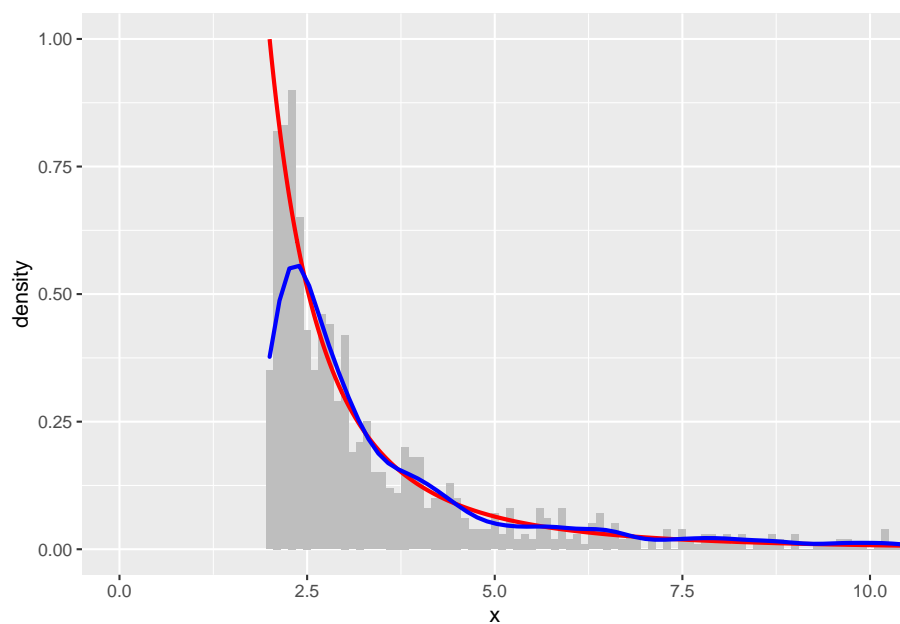
$$\frac{ab^a}{x^{a+1}}$$

则易于用逆变换法得出

$$x \sim \frac{b}{1 - U(0, 1)^{\frac{1}{a}}}$$

```
rpareto <- function(n=1, a=2, b=2){  
  U <- runif(n)  
  return(b / (1-U)^(1/a))  
}  
  
a <- 2  
b <- 2  
x <- rpareto(1000, a, b)  
as.data.frame(x) %>%  
  ggplot(aes(x, ..density..)) +  
  geom_histogram(binwidth = 0.1, fill='grey') +  
  geom_line(aes(y=a * b**a / x**(a + 1)), color='red', size=1) +
```

```
geom_density(color = 'blue', size = 1) +
coord_cartesian(xlim = c(0, 10), ylim = c(0, 1))
```



### 3.4

给出 pdf，此题易于得出 cdf

$$\int_{-\infty}^x \frac{x}{\sigma^2} e^{-x^2/(2\sigma^2)} dx = 1 - e^{-x^2/(2\sigma^2)}$$

可用逆变换得出

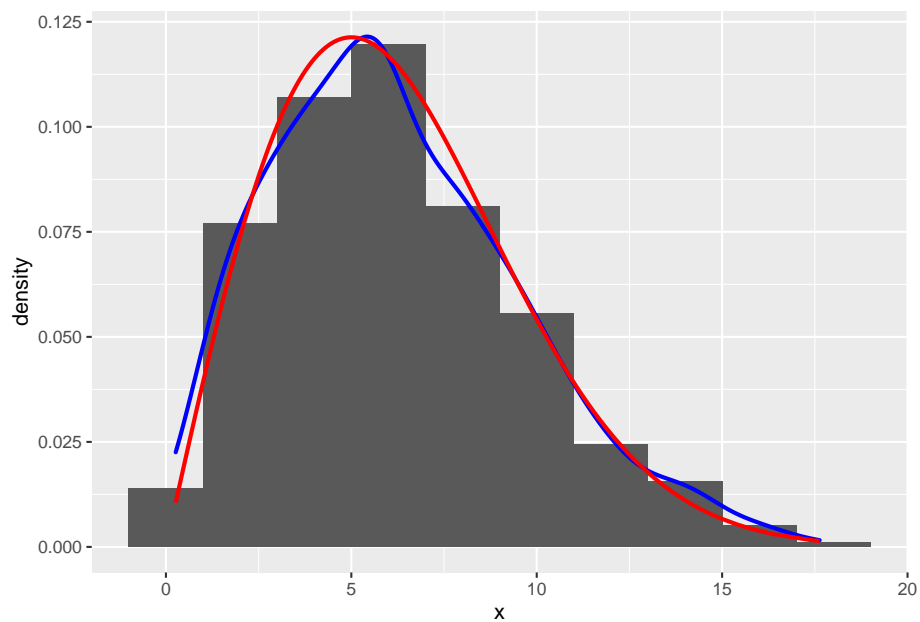
$$x \sim \sqrt{-2\sigma^2 \ln(1 - U(0, 1))}$$

```
rarrayleigh <- function(n=1, sigma){
  return(sqrt(-2 * sigma^2 * log(1 - runif(n))))
}
```

```

sigma <- 5
x <- rrayleigh(1000, sigma)
as.data.frame(x) %>%
  ggplot(aes(x, ..density..)) +
  geom_histogram(binwidth = 2) +
  geom_density(color = 'blue', size = 1) +
  geom_line(aes(y=x / sigma^2 * exp(-0.5 * x**2 / sigma**2)), color='red', size=1)

```



### 3.9

- 按照题目描述构造

```

rREK <- function(n){
  xvec <- numeric(n)

```

```

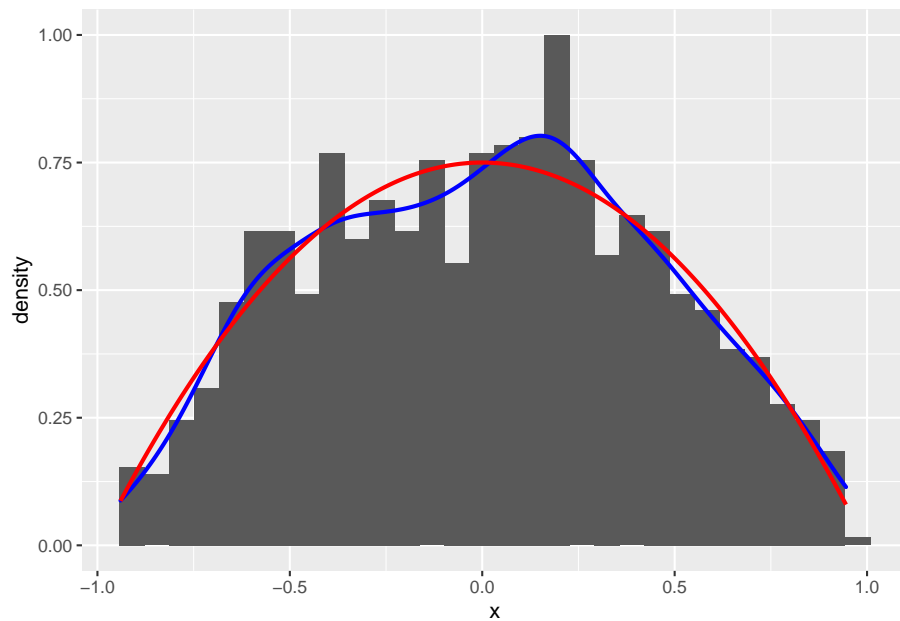
for (i in 1:n){
  u1 <- runif(n,-1,1)
  u2 <- runif(n,-1,1)
  u3 <- runif(n,-1,1)
  if(abs(u3[i]) >= abs(u2[i]) && abs(u3[i]) >= abs(u1[i])){
    ans <- u2[i]
  }else{
    ans <- u3[i]
  }
  xvec[i] <- ans
}
return(xvec)
}

x <- rREK(1000)

as.data.frame(x) %>%
  ggplot(aes(x,y=..density..), binwidth = 0.05) +
  geom_histogram() +
  geom_density(color = 'blue', size = 1) +
  geom_line(aes(x=x,y = 3 / 4 *(1-x**2)), color = 'red', size = 1)

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

```



### 3.11

Generate a random sample of size 1000 from a normal location mixture. The components of the mixture have  $N(0, 1)$  and  $N(3, 1)$  distributions with mixing probabilities  $p_1$  and  $p_2 = 1 - p_1$ . Graph the histogram of the sample with density superimposed, for  $p_1 = 0.75$ . Repeat with different values for  $p_1$  and observe whether the empirical distribution of the mixture appears to be bimodal. Make a conjecture about the values of  $p_1$  that produce bimodal mixtures.

solution: 按照题目意思来的

```
rMix <- function(n=1000,p=0.75) {
  xvec <- numeric(n)
  for (i in seq_along(xvec)){
    U <- runif(1)

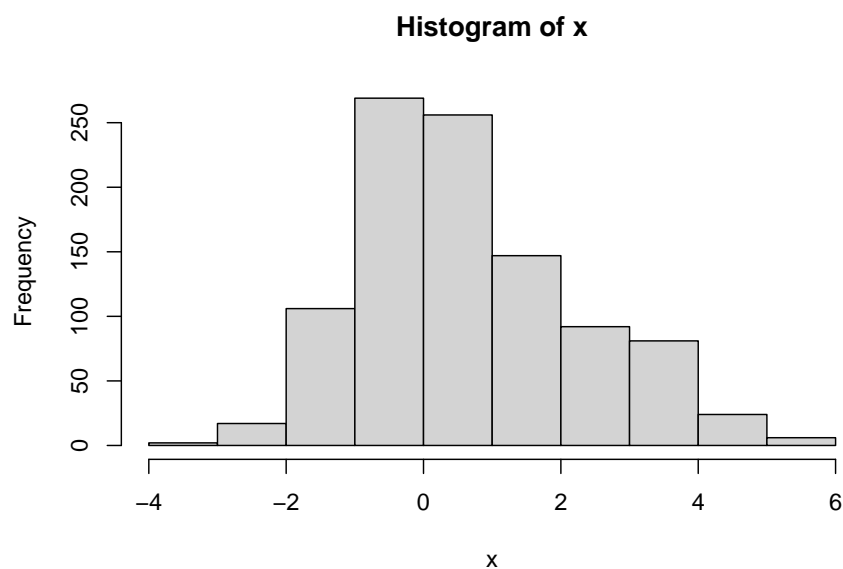
    if (U < p){
```

```

      xvec[i] <- rnorm(1)
    }else{
      xvec[i] <- rnorm(1,3,1)
    }
  }
  return(xvec)
}

x <- rMix()
hist(x)

```



### 3.13

由 3.12 可知构造，且 cdf 为

$$\frac{r\beta^r}{(\beta + y)^{r+1}}$$

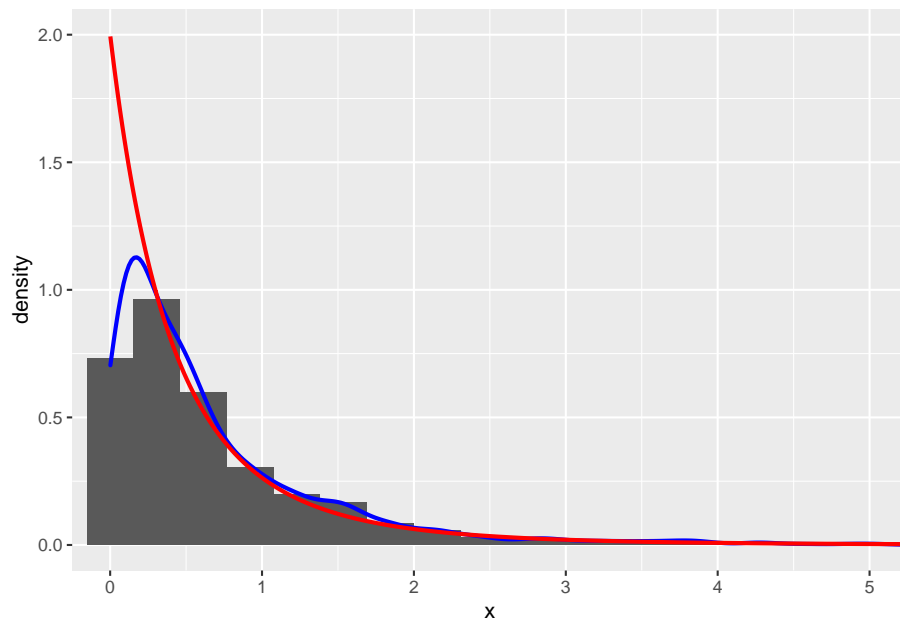
```

rPareto <- function(n,r,beta){
  args <- list(rate=rgamma(n,r,beta), n=1)
  x <- args %>%
    pmap(rexp)
  return(as_vector(x))
}

r <- 4
beta <- 2
x<-rPareto(1000, r, beta)
as.data.frame(x) %>%
  ggplot(aes(x, ..density..), binwidth = 0.05) +
  geom_histogram() +
  geom_density(color = 'blue', size = 1) +
  geom_line(aes(y=r * beta**r / (beta + x)^(r + 1)), color = 'red', size = 1) +
  coord_cartesian(xlim = c(0, 5), y = c(0, 2))

```

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

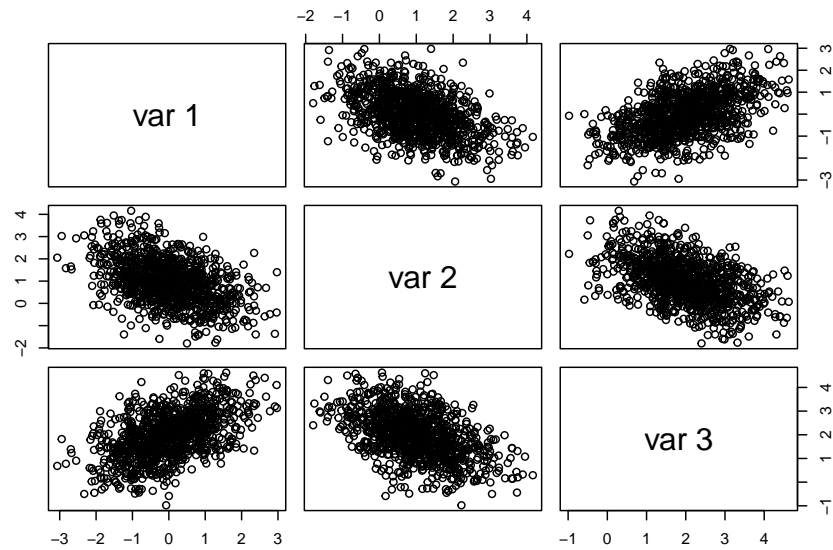


---

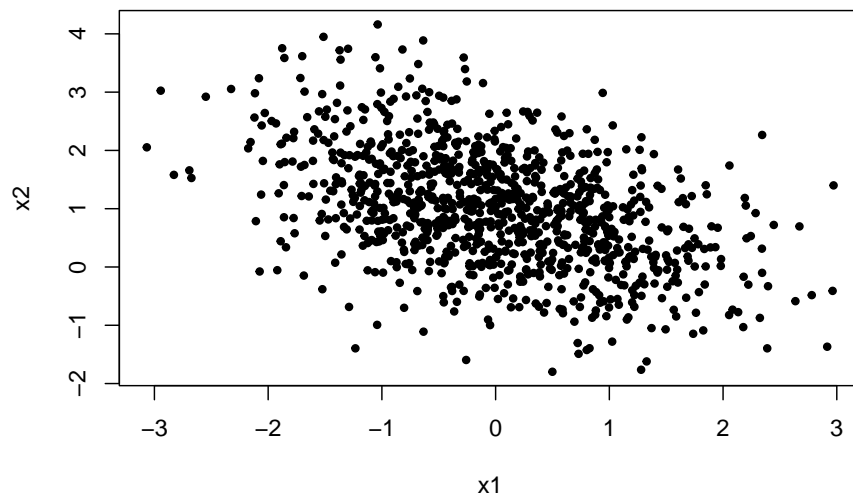
## 3.14

```
rng.mnorm <- function(n, mu, Sigma){  
  m <- length(mu)  
  M <- chol(Sigma) # Sigma = M' M  
  y <- matrix(rnorm(n*m), n, m) %*% M  
  for(j in seq(along=mu)){  
    y[,j] <- y[,j] + mu[j]  
  }  
  return(y)  
}  
  
mu <- c(0,1,2)  
sigma <- c(1,-0.5,0.5,-0.5,1,-0.5,0.5,-0.5,1) %>% matrix(3,3)  
  
x <- rng.mnorm(1000,mu,sigma)  
pairs(x)
```

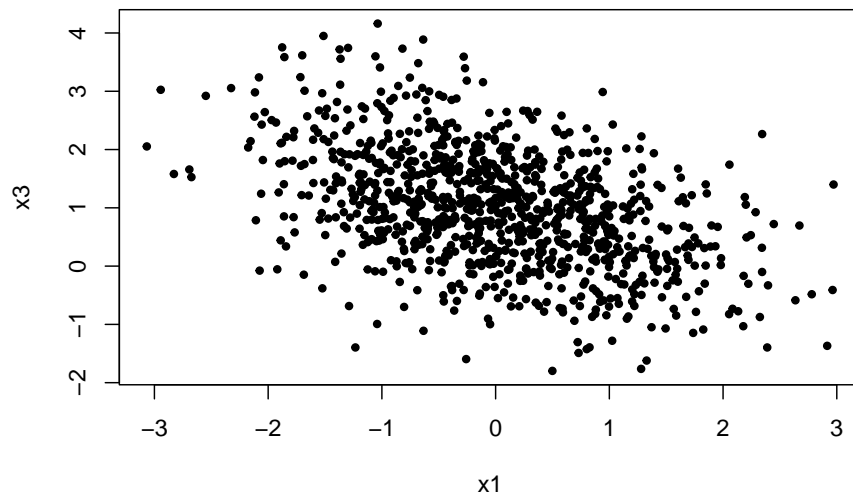




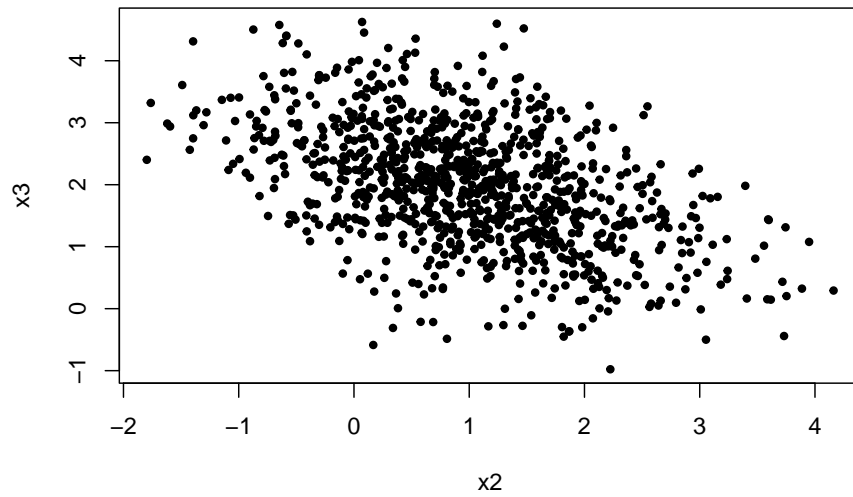
```
plot(x[,1:2], xlab = "x1", ylab = "x2", pch = 20)
```



```
plot(x[,1:3], xlab = "x1", ylab = "x3", pch = 20)
```



```
plot(x[,2:3], xlab = "x2", ylab = "x3", pch = 20)
```



```
#pairs(MASS::mvrnorm(1000,mu,sigma))
```

---

## 3.16

由题意易知

```
cov(scale(bootstrap::scor))
```

```
##          mec          vec          alg          ana          sta
## mec 1.0000000 0.5534052 0.5467511 0.4093920 0.3890993
## vec 0.5534052 1.0000000 0.6096447 0.4850813 0.4364487
## alg 0.5467511 0.6096447 1.0000000 0.7108059 0.6647357
## ana 0.4093920 0.4850813 0.7108059 1.0000000 0.6071743
## sta 0.3890993 0.4364487 0.6647357 0.6071743 1.0000000
```

---

## 3.20

```
N_poisproc <- function(t, lambda=1) {
  Tn <- rexp(100, lambda)
  Sn <- cumsum(Tn)
  return(min(which(Sn > t)) - 1)
}

cmd_poisporc <- function(t,shape,rate=1,lambda=1) {
  ftmp <- function(t0) {
    return(sum(rgamma(N_poisproc(t0,lambda),shape,rate)))
  }
}
```

```
    return(ftmp(t))  
  }
```

```
#verify  
shape <- 1  
rate <- 2  
t<-10  
lambda <- 1  
N <- 1e5  
  
mean_cmd <- mean(replicate(N, cmd_poisporc(t, shape, rate, lambda)))  
var_cmd <- var(replicate(N, cmd_poisporc(t, shape, rate, lambda)))  
  
exp_gamma <- shape / rate  
var_gamma <- shape / rate**2  
moment_2_gamma <- var_gamma + exp_gamma**2  
  
near(mean_cmd, lambda * t * exp_gamma, tol = 1)
```

```
## [1] TRUE
```

```
near(var_cmd, lambda * t * moment_2_gamma, tol = 1)
```

```
## [1] TRUE
```