

# STP Assignment 2

Deepak Karunakaran

October 1, 2015

## Question 1

Refer to the air-conditioning data set `aircondit` provided in the `boot` package. The 12 observations are the times in hours between failures of air-conditioning equipment. Assume that the times between failures follow an exponential model  $\text{Exp}(\lambda)$ . Obtain the MLE of the hazard rate  $\lambda$  and use bootstrap to estimate the bias and standard error of the estimate

## Answer:

All the failure times are independent. The exponential distribution can be simulated in R with `rexp(n, lambda)` where `lambda` is the rate parameter. The mean of exponential distribution is  $1/\text{mean}$  and the standard deviation is also  $1/\text{mean}$

We take the derivative of the log likelihood function of the exponential distribution with respect to `lambda` to find the maximum value of the `lambda`. Second derivative can be taken to ensure that it is the maximum and not the minimum value.

The maximum likelihood estimator for the `lambda` of an exponential distribution is:  **$1/\text{mean}$**

## Different approaches attempted for this question

1. Using `sample()` function to
2. Using `boot()` function.
3. Using `optimize()` function to find the `lambda` maximum value of the function

```
###Q1 Maximum likelihood Estimate of lamda and calculation of standard error and bias
###Using boot()
library(boot)
mle=1/mean(aircondit$hours)
my.mean=function(x,indices){return(1/mean(x[indices]))}
lambda.boot=boot(aircondit$hours,my.mean,1000)
print(mle)
```

```
## [1] 0.00925212
```

```
print(lambda.boot)
```

```
##  
## ORDINARY NONPARAMETRIC BOOTSTRAP  
##  
##  
## Call:  
## boot(data = aircondit$hours, statistic = my.mean, R = 1000)  
##  
##  
## Bootstrap Statistics :  
##      original      bias    std. error  
## t1* 0.00925212 0.001357439 0.004316773
```

```
#Using sample()  
b=1000 #number of repitions  
xbar=rep(0,b)  
obs=aircondit$hours;  
n=length(obs)  
for(i in 1:b){x=sample(obs,n,replace=TRUE);xbar[i]=mean(x)}  
lambda.bt=1/xbar;  
  
#results  
summary(lambda.bt)
```

```
##      Min.  1st Qu.  Median    Mean 3rd Qu.    Max.  
## 0.003840 0.007634 0.009685 0.010750 0.012870 0.037500
```

```
sd(lambda.bt)
```

```
## [1] 0.004413971
```

```
lambda.bt.sub=lambda.bt-mean(lambda.bt)  
quantile(lambda.bt.sub,c(0.05,.95))
```

```
##           5%           95%  
## -0.004973701 0.008635258
```

**mle=0.00925212**

**Bias = 0.001277963**

Standard Error= 0.004132903

## Question 2

Implement the bivariate Spearman rank correlation test for independence as a permutation test. Use the `cor` function with `method=spearman`. Compare the achieved significance level of the permutation test with the p-value reported by `cor.test` on the same samples.

## Answer:

Spearman's rank coefficient is a non-parametric measure of association of monotonic relation between two numeric variables. It is different from Pearson that it uses ranks.

We try to see if there's a monotonic relationship between two variables. We use a Spearman rank correlation test with and without permutations and then compare the p-values obtained for rho. We permute one of the variables independently of the other in the example the response is permuted

**Null Hypotheses: There is no correlation between variable 1 and variable 2.**

## Methodology:

1. Calculate Spearman correlation coefficient ( $\hat{\theta}$ ) using `cor.test`.
2. Do the permutation 1000 times. Higher number of permutations destroys existing dependencies between variables.
3. Calculate Spearman correlation coefficient value ( $\hat{\theta}^*$ ) for these permutations and create empirical distribution, where  $i=1, 2, \dots, M$ .
4. Estimate p-value as :  
ASL:  $ASL = P(\hat{\theta}^* \geq \hat{\theta} \mid H_0)$

If the ASL value is less than alpha, null hypothesis is rejected.

ASL small implies:  $\Rightarrow$  Strong evidence against  $H_0 \Rightarrow H_0$  not true

ASL : 0 means perfect correlation.

To reject the null hypothesis we can compare the significance level and the p-value. The significance level is the probability that rho will fall in the critical region under the null hypothesis. If the P value is small, you can reject the idea that the correlation is due to random sampling. If the p\_value is large, we cannot conclude that the correlation is real. **We resample under the null hypothesis and counting the outcomes at least as extreme as the outcome coming from the original data** `cor.test` was used to calculate the Spearman coefficient. The `cor` function with `method="Spearman"` was used to calculate the correlation parameters

**\*\* cor.test\*\*** will give other details: t = test statistic df = degree of freedom, p-value, alternative hypothesis: true correlation is not equal to 0.95 percent confidence interval, sample estimates:

**spear1-** Spearman's rank correlation rho using `cor.test` before permutation.

**asl**- Achieved Significance Level calculated as the ratio of number of times the rho obtained after permutation exceeds the rho before to the total number of permutations. Comparing the ASL and the p-value

```
###Spearman rank correlation test for independenceas a permutation test.
# by Deepak Karunakaran
var1=runif(1000)
var2=runif(1000)
test <- function(var2) suppressWarnings(cor(var1, var2, method="spearman"))
spear1 <- suppressWarnings(cor.test(var1, var2, method="spearman"))
p <- replicate(10000, test(sample(var2, length(var2)))) # Simulated permutati
on distribution
p.out <- sum(abs(p) >= spear1$estimate) #Count of rho_permuted exceeding rh
o_original
asl=p.out / length(p) # Proportion of exceeding the the p-value.
spear1$p.value
```

```
## [1] 0.4884209
```

```
asl
```

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

```
spear1
```

```
##
## Spearman's rank correlation rho
##
## data: var1 and var2
## S = 170320000, p-value = 0.4884
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
## rho
## -0.02193045
```

**In this case achieved significance level= 0.8395 is greater than p\_value=0.8346456. Very Less Correlation. We can accept the NULL**

```

###Spearman rank correlation test for independence as a permutation test.
# by Deepak Karunakaran
var1=1:1000
var2=1:1000
test <- function(var2) suppressWarnings(cor(var1, var2, method="spearman"))
spear1 <- suppressWarnings(cor.test(var1, var2, method="spearman"))
p <- replicate(10000, test(sample(var2, length(var2)))) # Simulated permutation distribution
p.out <- sum(abs(p) >= spear1$estimate) #Count of rho_permuted exceeding rho_original
asl=p.out / length(p) # Proportion of exceeding the p-value.
spear1$p.value

```

```
## [1] 0
```

```
asl
```

```
## [1] 0
```

```
spear1
```

```

##
## Spearman's rank correlation rho
##
## data: var1 and var2
## S = 0, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
## rho
## 1

```

In this case achieved significance level and than p\_value are 0. Very High Correlation. We accept the NULL

```

###Spearman rank correlation test for independence as a permutation test.
# by Deepak Karunakaran
var1=1:1000
var2=1000:1
test <- function(var2) suppressWarnings(cor(var1, var2, method="spearman"))
spear1 <- suppressWarnings(cor.test(var1, var2, method="spearman"))
p <- replicate(10000, test(sample(var2, length(var2)))) # Simulated permutation distribution
p.out <- sum(abs(p) >= spear1$estimate) #Count of rho_permuted exceeding rho_original
asl=p.out / length(p) # Proportion of exceeding the p-value.
spear1$p.value

```

```
## [1] 0
```

```
asl
```

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

```
spear1
```

```

##
## Spearman's rank correlation rho
##
## data: var1 and var2
## S = 333330000, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
## rho
## -1

```

**In this case there is a conflict. The achieved significance level= 1 and p\_value=0. We reject the NULL since ASL=0.**

Other Observations

**Spearman cannot handle tied ranks properly.** Warning message: In `cor.test.default(var1, var2, method = "spearman")`: Cannot compute exact p-value with ties suppressWarnings can be used to avert this situation It suppresses warnings when there are ties. `suppressWarnings (cor.test(x,y) OR **Kendall tau rank correlation**` is also a non-parametric test for statistical rank-transformed) variables—like Spearman's, but unlike Spearman's, can handle ties. tau-b is specifically adapted to handle ties.

## Question 3

Write a function to generate a random sample of size n from the Beta( $\alpha$ ,  $\beta$ ) distribution by the acceptance-rejection method. Generate a random sample of size 1000 from the Beta(3, 2) distribution. Graph the histogram of the sample with the theoretical Beta(3, 2) density superimposed.

## Answer

**Probability Density Function:**  $12 \times (x^2 \times (1 - x))$

**Maximum value of y =16/9 at x=2/3**

```
###Beta(3, 2) distribution
### by Deepak Karunakaran
# we can simulate X from Unif(0,1) and Y from Unif(0,16/9) and accept X if Y<f(X)
rzl<-function ()
{
  repeat {
    x <- runif(1, 0, 1)
    y <- runif(1, 0, 16/9)
    fx <- 12 * x* x* (1 - x)
    if (y < fx)
      return(x)
  }
}
```

```
##we repeat the simulation block indefinitely, until the acceptance criterion i  
s satisfied, at which point we break the loop by returning the accepted value.
```

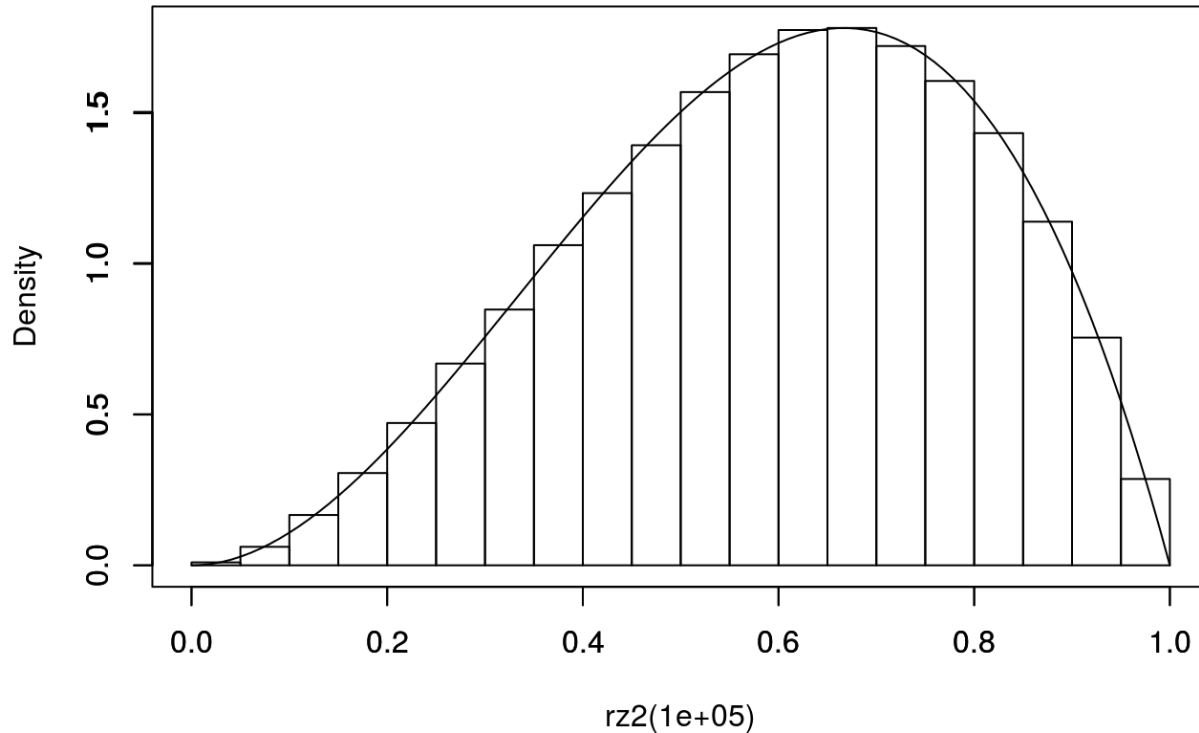
```
#We can now use this to simulate a vector of n such quantities as follows.
```

```
rz2<-function (n)  
{  
  zvector <- vector("numeric", n)  
  for (i in 1:n) {  
    zvector[i] <- rz1()  
  }  
  zvector  
}
```

```
# We can now plot histogram of the sample with the theoretical Beta(3, 2) densi  
ty superimposed.
```

```
curve(12 * x* x* (1 - x),0,1,xlab="",ylab="")  
par(new=TRUE);  
hist(rz2(100000),xlim = c(0,1),prob=TRUE)
```

**Histogram of rz2(1e+05)**





## Question 4

The Pareto( $\alpha$ ,  $\beta$ ) distribution has cdf  $F(x)$  with  $x \geq \beta > 0$  and  $\alpha > 0$ . Derive the probability inverse transformation  $F^{-1}(U)$  and use the inverse transform method to simulate a random sample from the Pareto(2, 2) distribution. Graph the density histogram of the sample with the Pareto(2, 2) density superimposed for comparison.

## Answer

Probability distribution function:  $8/x^3$  ( $x > 2$ )

Cumulative distribution function:  $1-4/x^2$  ( $x > 2$ )

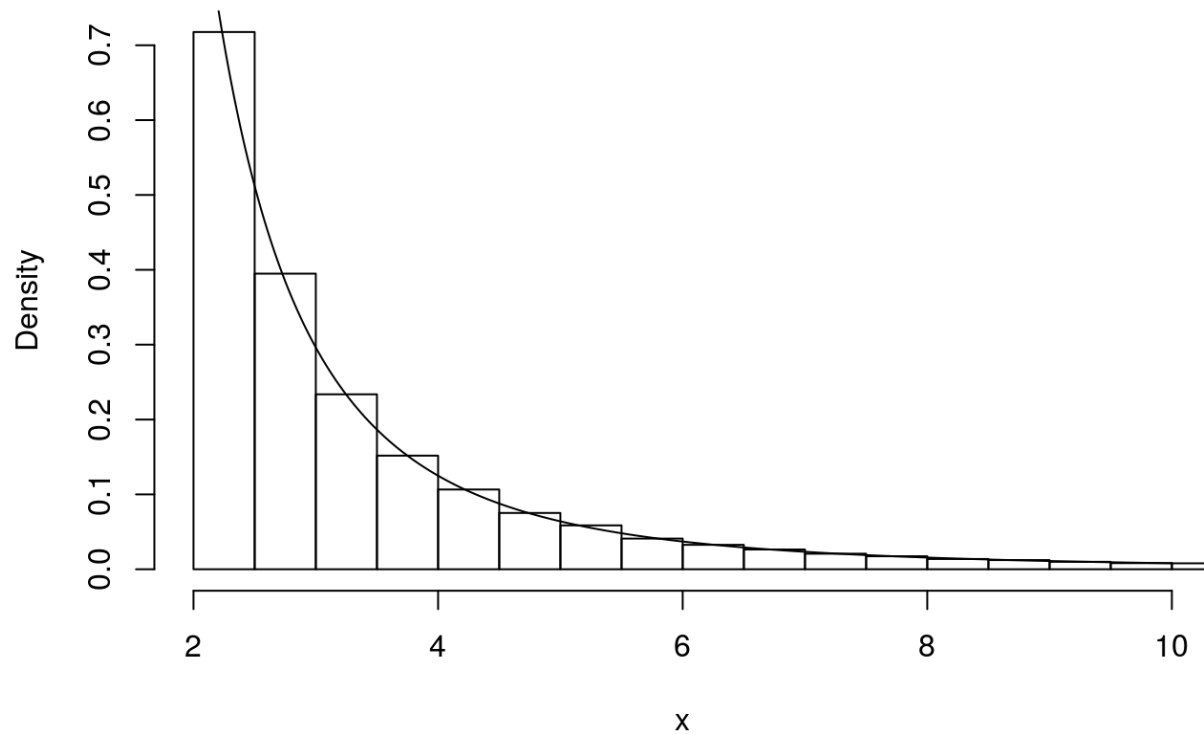
Inverse Function:  $F^{-1}(u) = 2/\sqrt{1-u}$

Pareto function is:  $1-4/x^2$  ( $x > 2$ )

```
### (Inverse transform method, continuous case for pareto(2,2))
### by Deepak Karunakaran
n <- 100000
u <- runif(n)
x <- 2/(sqrt(1-u));
hist(x, prob = TRUE, xlim=c(2,10),breaks=2000,main="Histogram of pareto(2,2) PD
F f(x)=8/x^3") #density histogram of sample
y <- seq(2, 10, .001)

#lines(y, 8/y^3)      #density curve f(x)
par(new=TRUE);
lines(y,8/y^3,xlim=c(2,10),xlab="",ylab="")      #density curve f(x)
```

## Histogram of $\text{pareto}(2,2)$ PDF $f(x)=8/x^3$



## Steps

1. Draw 1000 random values from the  $\text{Uniform}(0,1)$  distribution. Call these values  $u$
2.  $u \leftarrow \text{runif}(1000)$
3. Compute  $F^{-1}(u)$  to get values of  $x$ . The values in  $x$  are drawn from  $f(x)$
4. We then compare the density of our draws to the target density  $f(x)$