Assignment 2 - Ram Yoogesh (20867060)

Question 1

(A)

From the question we can consider that

$$p = Pr(Y \leq Q_Y(p))$$

Upon substituting the values of Y we get,

$$p = Pr(aX + b \leq Q_Y(p))$$

$$p = Pr(X \leq \frac{Q_Y(p) - b}{a})$$

$$p = F_X(\frac{Q_Y(p) - b}{a})$$

From the above 2 equations, we can write this as

$$F_X^{-1}(p) = F_X^{-1}(F_X(\frac{Q_Y(p) - b}{a}))$$

$$Q_Y(p) = aQ_X(p) + b$$

Hence proved.

(b)

$$Pr(Q_X(U) \leq x) = Pr(F_X^{-1}(U) \leq x)$$
$$Pr(Q_X(U) \leq x) = Pr(F_X^{-1}(U) \leq x)$$

$$Pr(Q_X(U) \leq x) = Pr(U \leq F_X(x))$$
$$Pr(Q_X(U) \leq x) = G_U(F_X(x))$$

$$Pr(Q_X(U) \leq x) = F_X(x)$$

(c) (i)

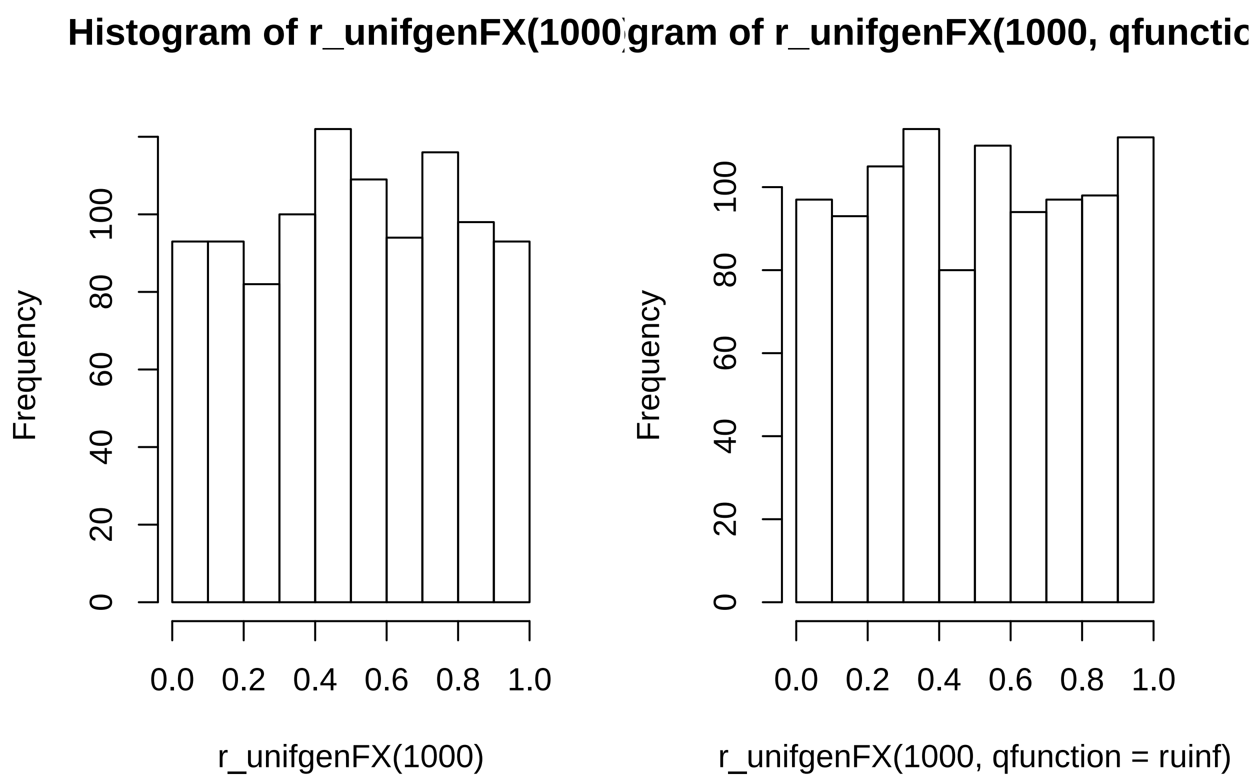
Writing the r_unifgenFX function to return random derivates

```
r_unifgenFX <- function(n, qfunction = qnorm)
{
  randomderivates <- runif(n, min =0, max = 1)
  return (randomderivates)
}
```

(c) (ii)

Executing the following snippets as instructued

```
set.seed(1234567)

oldpar <- par(mfrow = c(1, 2))
hist(r_unifgenFX(1000))

hist(r_unifgenFX(1000, qfunction = ruinf))
```



```r
par(oldpar)
```

(c)(iii)

Generating pseudo-random observations from a student t distribution

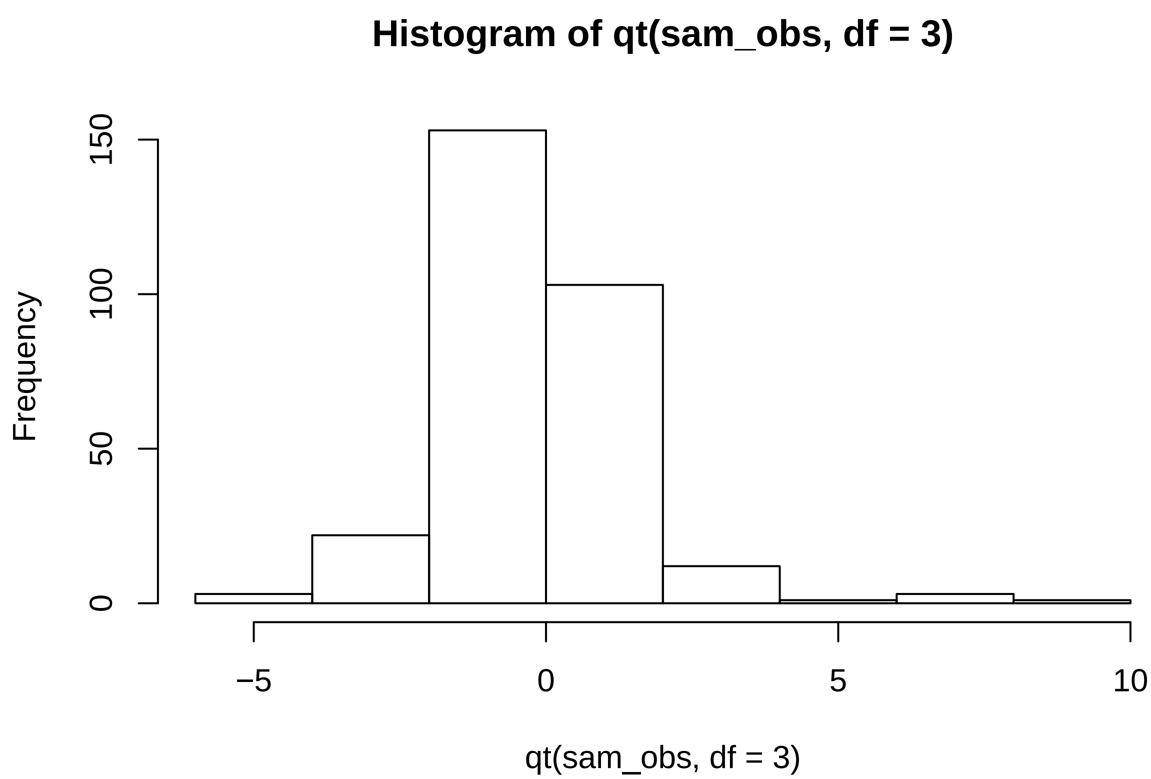
```r
r_unifgen <- function (n, dof)
{
  robs_student_t <- rt(n, df=dof)
  return (robs_student_t)
}
```

```r
sam_obs <- r_unifgen(1000, 3)

# Now passing the vector of random values to the qt fucntion to find the quantile values.

 x_val <- qt(sam_obs, df= 3, lower.tail = TRUE)
```

```
## Warning in qt(sam_obs, df = 3, lower.tail = TRUE): NaNs produced
```

```r
hist(qt(sam_obs, df = 3))
```

```
## Warning in qt(sam_obs, df = 3): NaNs produced
```



(d)

```r
# Just analyzing the dataset before interpreting the questions !

str(mtcars)
```

```
## 'data.frame':    32 obs. of  11 variables:
##  $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
##  $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
##  $ disp: num  160 160 108 258 360 ...
##  $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
##  $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
##  $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
##  $ qsec: num  16.5 17 18.6 19.4 17 ...
##  $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
##  $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
##  $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
##  $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

```r
# we can infer that it has 11 variables and 32 rows . Names of these are mpg, cyl, disp, hp, drat, wt,
head(mtcars)
```

```
##                    mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

(d)(i)

In order to answer this question ,lets use the summary function to get some insights. We can infer from summary of qsec (min value- 14.50, 1st quartile - 16.89 and so on).

```r
summary(mtcars)
```

```
##       mpg             cyl             disp             hp       
##  Min.   :10.40   Min.   :4.000   Min.   : 71.1   Min.   : 52.0  
##  1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8   1st Qu.: 96.5  
##  Median :19.20   Median :6.000   Median :196.3   Median :123.0  
##  Mean   :20.09   Mean   :6.188   Mean   :230.7   Mean   :146.7  
##  3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0   3rd Qu.:180.0  
##  Max.   :33.90   Max.   :8.000   Max.   :472.0   Max.   :335.0  
##       drat             wt             qsec             vs        
##  Min.   :2.760   Min.   :1.513   Min.   :14.50   Min.   :0.0000  
##  1st Qu.:3.080   1st Qu.:2.581   1st Qu.:16.89   1st Qu.:0.0000  
##  Median :3.695   Median :3.325   Median :17.71   Median :0.0000  
```

```
##  Mean   :3.597   Mean   :3.217   Mean   :17.85   Mean   :0.4375  
##  3rd Qu.:3.920   3rd Qu.:3.610   3rd Qu.:18.90   3rd Qu.:1.0000  
##  Max.   :4.930   Max.   :5.424   Max.   :22.90   Max.   :1.0000  
##        am              gear            carb      
##  Min.   :0.0000   Min.   :3.000   Min.   :1.000  
##  1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:2.000  
##  Median :0.0000   Median :4.000   Median :2.000  
##  Mean   :0.4062   Mean   :3.688   Mean   :2.812  
##  3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:4.000  
##  Max.   :1.0000   Max.   :5.000   Max.   :8.000
```

```
# Now lets use the Quantile function to answer the question
quantile(mtcars$qsec)
```

```
##      0%     25%     50%     75%    100% 
## 14.5000 16.8925 17.7100 18.9000 22.9000
```

Qsec - The variable denotes the 1/4 mile time took by cars.

Qunatile function basically prodcues the values at the respective quartiles. Say 0, 25, 50, 75 and 100. What it means for Qsec is that at 25th pecent quartile the value of qsec is 16.89 (It tooks a car 16.89s to reach 1/4 mile). At 50th percent, also called as the median of the dataset, the value at 50th percent quartile is 17.71s, likewise for the remaining quartiles.

d(ii)

we can pass mtcars$qsec in the X argument and for the probability we can use runif to generate 1000 observations

```
samobs_fromqsec <- quantile(mtcars$qsec, prob = runif(1000))
samobs_fromqsec
```

```
## 82.56272019% 55.43537533% 25.26772677% 28.88000016% 56.21380128% 48.23564820% 
##     19.45783     17.98370     16.89499     17.01434     17.98853     17.59155 
## 27.80648707% 93.64463692% 16.56456550%  1.18075679%  2.22223180% 48.60856393% 
##     16.97440     20.01627     16.49240     14.53660     14.56889     17.61510 
## 39.00647471% 17.17150714% 96.79571246% 23.35539414% 60.25798912%  2.62663255% 
##     17.30920     16.53756     20.23788     16.87721     18.20399     14.58143 
## 15.67640505% 31.63931684% 11.19827144% 60.73381805%  4.63054057% 97.24547171% 
##     16.37301     17.02000     15.66030     18.24825     14.95273     20.61154 
## 68.23912810% 68.41987553% 35.84659728% 52.15265874% 86.24651006% 41.70240827% 
##     18.60154     18.60210     17.07811     17.84677     19.78666     17.39277 
## 23.88116326% 83.79194455%  8.69364745% 76.68455190%  4.22513322% 79.65354961% 
##     16.88209     19.46927     15.47255     18.90000     14.85093     19.27400 
## 37.64747565%  8.97748205% 47.33024519% 84.31135849% 67.42749871% 38.16132473% 
##     17.21768     15.48047     17.54103     19.52870     18.59220     17.25750 
## 14.64794646%  4.94745784% 33.35952470% 12.02421337% 91.91094527% 25.07700184% 
##     16.17534     15.03231     17.03024     15.74735     20.00492     16.89322
```

```
## 59.71529905% 29.30435641% 73.51423618% 33.78774493% 26.55488406% 97.56579010%
##     18.15352     17.02000     18.83893     17.03423     16.92784     20.87766
## 25.48536612% 29.33372608% 53.45269199% 65.57387246% 69.89205494% 34.77731368%
##     16.89701     17.02000     17.91125     18.54623     18.60667     17.04343
## 43.90856060% 17.13485958% 19.73887279% 25.05627696%  8.32081330% 66.50878717%
##     17.41223     16.53483     16.72024     16.89302     15.46215     18.56942
## 55.26927833% 26.58797766% 91.64741621%  5.32587986% 19.63268633% 61.39929765%
##     17.98267     16.92907     20.00411     15.12733     16.71464     18.30743
## 18.96279245% 82.42500224%  0.58414079% 86.01700889% 29.71542988% 57.77933504%
##     16.67083     19.45655     14.51811     19.75607     17.02000     17.99823
## 49.98949510% 36.16389406% 23.97085091% 63.16237934% 39.99208040% 97.30362794%
##     17.70928     17.10270     16.88293     18.42767     17.33975     20.65985
## 76.90825551% 88.04478203%  9.35689837% 47.29932379% 61.35180183% 61.18349247%
##     18.90000     19.92939     15.49106     17.53930     18.30419     18.29006
## 81.59333244% 97.56507545% 41.85280132% 76.74217701% 99.85544763% 99.14391234%
##     19.44882     20.87706     17.39744     18.90000     22.77991     22.18876
## 35.03837977%  4.08151275% 96.18970093% 13.72014924% 54.62887629%  9.24995951%
##     17.04586     14.81487     20.18195     15.99701     17.96959     15.48807
##  9.57933494% 86.29760696% 52.09488226% 81.45562124% 58.72106701% 12.85156454%
##     15.49726     19.79347     17.84391     19.44754     18.06106     15.83455
## 50.87438170% 72.53598233% 13.05597590%  4.89486768% 44.62784743% 97.92759221%
##     17.76963     18.75098     15.86936     15.01910     17.41669     21.17824
## 74.42114935% 89.96940241% 54.45640450% 90.44165486% 59.44494465% 42.64210395%
##     18.90000     19.98905     17.96104     20.00037     18.12838     17.40438
## 89.97218476% 63.65564424% 98.26391332%  3.76559685% 15.85332071% 59.63822310%
##     19.98914     18.46131     21.45766     14.73554     16.40701     18.14635
## 70.40746557% 19.02837143% 30.85463264% 12.50515957% 29.92917513% 11.11923300%
##     18.60826     16.67571     17.02000     15.79804     17.02000     15.65197
## 16.12096806% 13.10673954% 60.93945040% 18.51580143%  3.11191664% 42.97619576%
##     16.45845     15.87912     18.26737     16.63758     14.59647     17.40645
## 90.81213993% 55.85000983% 36.22468703% 55.41786146% 28.97401412% 83.11288627%
##     20.00152     17.98627     17.10741     17.98359     17.01783     19.46295
## 34.28008170% 45.95526094% 18.82636116% 27.78894154% 10.51862428% 50.46582369%
##     17.03880     17.46430     16.66068     16.97375     15.58866     17.74177
## 40.00436899% 46.61056800% 61.94783424% 52.03986771% 91.47919260% 79.96346029%
##     17.34014     17.50087     18.34484     17.84118     20.00359     19.32588
## 41.47342306% 32.03756956% 93.84348404% 30.03271334% 59.69637299% 23.96383011%
##     17.38568     17.02000     20.02921     17.02000     18.15176     16.88286
## 10.13508830% 44.45318650% 60.48490664% 15.83801790% 42.91884473% 58.31603471%
##     15.54824     17.41561     18.22510     16.40407     17.40610     18.02339
## 21.68378509% 19.76913752% 97.30679605% 60.18004464% 21.48169677% 54.16405422%
##     16.82274     16.72183     20.66249     18.19674     16.81209     17.94654
## 30.52252387% 52.81632212%  7.18180772% 32.22636990% 68.67917217% 12.07703731%
##     17.02000     17.87969     15.43037     17.02000     18.60291     15.75292
## 21.96024337% 56.93879572% 39.68072014% 77.84693227%  7.78091645% 96.77311194%
##     16.83730     17.99302     17.33010     18.97158     15.44709     20.21993
## 62.56218953%  5.35220518% 76.08212270% 72.01643141% 28.49725410%  3.43481961%
##     18.38674     15.13394     18.90000     18.70428     17.00010     14.65248
##  4.25333907% 85.87827445% 55.56464666% 56.16303030% 54.82415550% 89.18526161%
##     14.85801     19.73757     17.98450     17.98821     17.97928     19.96474
## 39.17641195%  6.61698075%  0.52501378% 60.79356549% 52.81056943% 30.50776182%
##     17.31447     15.41461     14.51628     18.25380     17.87940     17.02000
## 10.21651097% 31.64146643% 70.74450289% 79.02869396% 57.32117717% 17.50590673%
##     15.55682     17.02000     18.60931     19.16940     17.99539     16.56244
```

```
## 80.33520530% 27.67639011% 78.53511046% 84.19519907% 60.64314751% 97.29950440%
##     19.38811     16.96956     19.08678     19.51322     18.23981     20.65643
## 14.43649335% 44.71281979% 42.56872064% 48.42840463% 12.77212040% 59.03215043%
##     16.13469     17.41722     17.40393     17.60282     15.82618     18.08999
## 99.94138249% 35.24499438% 46.80802277% 56.60681063% 29.94399627% 25.77130969%
##     22.85130     17.04778     17.51189     17.99096     17.02000     16.89967
## 16.32712027% 99.45530235% 22.34047703% 96.98357540%  1.75254641% 63.25369675%
##     16.47474     22.44747     16.85734     20.39395     14.55433     18.43390
##  3.89308827%  3.83684309% 65.16432751% 73.16015372% 26.20007875% 40.79077656%
##     14.76755     14.75343     18.53608     18.80710     16.91464     17.36451
## 98.29375474% 73.41952804% 63.97085178% 88.11367962%  4.58539762% 47.65324581%
##     21.48245     18.83042     18.48281     19.93152     14.94139     17.55905
## 62.49401374% 30.27512734% 71.14703904% 29.84693730% 42.12123554%  6.29242084%
##     18.38209     17.02000     18.62612     17.02000     17.40115     15.37003
## 95.52195999%  2.50572101% 86.83456127% 59.68835352% 38.12123768%  8.11396807%
##     20.13848     14.57768     19.86505     18.15102     17.25440     15.45638
##  9.58540177% 95.15380803%  6.21218453% 78.64528128% 64.12392831% 43.46727473%
##     15.49743     20.11451     15.34988     19.10522     18.49325     17.40950
## 78.74434425% 95.78504299% 67.11769404% 11.66728989% 71.27704201% 13.61519306%
##     19.12180     20.15561     18.58452     15.70973     18.63781     15.97684
##  3.32151200% 57.99432388% 35.22213146% 21.98217737% 71.34997507%  3.56033260%
##     14.62403     17.99956     17.04757     16.83846     18.64436     14.68400
## 72.32125557% 27.39335718% 91.71770946% 33.38422896% 99.33601078% 44.12672729%
##     18.73168     16.95903     20.00432     17.03047     22.34836     17.41359
## 32.38993089% 82.13553496%  0.13211018% 35.75784068% 15.85842727% 69.85904539%
##     17.02123     19.45386     14.50410     17.07123     16.40799     18.60656
## 28.75272119% 92.32198375% 75.85187803% 52.15039302% 96.91810017% 32.01746123%
##     17.00960     20.00620     18.90000     17.84666     20.33956     17.02000
## 36.59960355% 60.72604805% 55.74902296% 60.73641607% 52.42969303% 67.94086182%
##     17.13647     18.24752     17.98564     18.24849     17.86051     18.60062
## 94.38108227% 97.19274095% 56.16172699% 76.81455724% 33.90080368% 20.82112569%
##     20.06421     20.56773     17.98820     18.90000     17.03528     16.77727
## 72.53334927% 36.64303324% 22.17044423% 60.30403329% 69.60928398% 53.59754032%
##     18.75075     17.13984     16.84838     18.20828     18.60579     17.91844
## 81.12893058%  9.77212514% 41.87956522% 47.28574241% 33.18166195% 67.44142794%
##     19.44450     15.50998     17.39827     17.53854     17.02859     18.59255
## 82.93805579% 48.08529613%  1.30763687% 76.81563040%  6.92166009% 33.77703801%
##     19.46132     17.58316     14.54054     18.90000     15.42311     17.03413
## 14.39823096%  1.50932721% 99.53401121%  3.05180331% 79.75445329% 31.66873844%
##     16.12734     14.54679     22.51286     14.59461     19.29090     17.02000
## 34.37210552% 72.88307224%  8.10276193%  9.35531305% 78.51319511% 10.54438695%
##     17.03966     18.78219     15.45607     15.49101     19.08311     15.59138
## 72.84079292% 86.44529721%  1.40377271% 34.13213578% 11.44323149%  7.16045424%
##     18.77839     19.81316     14.54352     17.03743     15.68612     15.42978
## 19.58475069% 76.13811439% 90.71499419% 52.21936542% 11.03495799% 26.99951492%
##     16.71212     18.90000     20.00122     17.85008     15.64308     16.94438
## 26.58121542% 76.58006530% 60.09711921%  4.26129107% 45.40479179% 90.48830299%
##     16.92882     18.90000     18.18903     14.86001     17.43359     20.00051
## 18.30044126% 27.45125513% 83.46215112%  5.39561377% 34.88374229% 95.29818657%
##     16.62155     16.96119     19.46620     15.14484     17.04442     20.12391
## 28.02474026% 77.93569483% 44.25319396% 91.97108389% 95.98682306% 37.29054451%
##     16.98252     18.98644     17.41437     20.00511     20.16874     17.19002
## 75.06954707% 69.45112473% 14.14768214% 87.51211669% 73.04591804% 68.16971598%
##     18.90000     18.60530     16.07918     19.91288     18.79683     18.60133
```

```
## 96.62130100% 46.81982549% 22.41111279% 37.31333553% 13.41029117% 72.31173571%
##     20.21005     17.51255     16.86107     17.19178     15.93746     18.73083
## 24.01478505%  6.60919521% 93.22497279%  4.72624635% 26.77474464%  0.79783136%
##     16.88334     15.41440     20.00900     14.97676     16.93602     14.52473
##  2.78722225% 77.10185470% 60.30889172% 20.73760142% 56.35227684% 53.29456567%
##     14.58640     18.90000     18.20873     16.77287     17.98938     17.90341
## 95.04546374%  3.24914595% 70.16517220% 13.15136352% 72.33748881%  0.16035712%
##     20.10746     14.60586     18.60751     15.88769     18.73314     14.50497
## 28.34046830% 47.39938923% 87.03422311% 34.36332927% 45.03041860% 21.73480587%
##     16.99427     17.54489     19.89166     17.03958     17.41919     16.82542
## 16.61894140% 14.54678690% 53.45341756%  6.95348750% 12.78197691% 69.79420648%
##     16.49645     16.15589     17.91129     15.42400     15.82722     18.60636
## 72.62953056% 22.03578202% 10.66227849% 96.46102663% 39.27409495% 38.01164555%
##     18.75939     16.84129     15.60380     20.19961     17.31750     17.24590
## 58.38360724% 34.80120264% 73.31172081% 22.90898398% 53.42700039%  1.84633748%
##     18.02968     17.04365     18.82072     16.87305     17.90998     14.55724
## 15.08750771% 63.29574878% 17.24004459% 33.91181941% 30.51505988% 93.33004639%
##     16.25982     18.43677     16.54266     17.03538     17.02000     20.00932
## 46.75896422% 70.92740706% 95.40609394% 84.90456820% 27.15169690% 87.29395294%
##     17.50915     18.60987     20.13094     19.60778     16.95004     19.90611
## 38.25742987% 13.65208260% 65.83302207%  9.65588801% 80.08583849%  2.32546174%
##     17.26495     15.98393     18.55266     15.49940     19.34637     14.57209
## 58.55699237% 55.30167094% 89.49665471% 76.27238836% 82.44216554% 79.31343431%
##     18.04580     17.98287     19.97440     18.90000     19.45671     19.21707
## 18.22255831% 60.08912972% 93.48142892% 17.35302303% 41.98244275% 12.19641664%
##     16.61576     18.18829     20.00979     16.55106     17.40029     15.76550
## 26.47810658% 42.74377776% 80.79843645% 95.90880473%  4.35019971% 97.45289234%
##     16.92499     17.40501     19.44143     20.16366     14.88234     20.78386
## 73.61760666% 69.47557076% 25.35871693% 25.29182946% 12.13420972% 67.97527664%
##     18.84822     18.60537     16.89584     16.89521     15.75895     18.60072
##  7.65236567% 56.63355999% 35.78151937% 45.26173177% 53.13574385% 61.09007385%
##     15.44350     17.99113     17.07307     17.42560     17.89553     18.28138
## 39.66444773% 36.98221291% 79.51229669% 93.28805828% 73.40476594% 88.85469981%
##     17.32960     17.16612     19.25036     20.00919     18.82909     19.95450
## 93.58648648% 17.18177774% 10.95236172% 76.61375646% 47.08925670% 74.05835572%
##     20.01248     16.53832     15.63438     18.90000     17.52758     18.88785
## 74.75191192% 42.31789222% 26.81746774% 87.59383108% 45.83986071% 22.99319741%
##     18.90000     17.40237     16.93761     19.91541     17.45786     16.87384
##  1.41022040% 67.78565834% 98.73533929% 73.29129106% 37.71403006% 29.70401703%
##     14.54372     18.60014     21.84932     18.81889     17.22284     17.02000
## 30.47169046%  9.74121834% 95.07730026% 42.72531895% 47.90643668% 76.87365452%
##     17.02000     15.50672     20.10953     17.40490     17.57318     18.90000
## 47.22851661% 75.55451011% 75.90153133% 55.41112577% 54.00159298% 34.52233209%
##     17.53535     18.90000     18.90000     17.98355     17.93848     17.04106
## 47.08693470% 12.83334901% 27.03833464% 15.93411532%  9.90233915% 63.29211826%
##     17.52745     15.83263     16.94583     16.42254     15.52371     18.43652
## 16.77585389% 31.74803695% 70.65067308% 29.44693493% 37.09916805% 42.24277923%
##     16.50812     17.02000     18.60902     17.02000     17.17519     17.40191
##  8.54225080% 34.23246755% 19.08565883% 36.18063692% 93.47018781% 82.50684077%
##     15.46833     17.03836     16.67997     17.10400     20.00976     19.45731
## 63.21110416% 72.14571137% 62.12100883% 66.16487014% 67.66258357% 57.21479284%
##     18.43100     18.71590     18.35665     18.56089     18.59803     17.99473
##  5.36231538% 38.48180657% 53.67963444% 10.90712373% 98.32699744% 45.25751041%
##     15.13648     17.28234     17.92251     15.62961     21.51007     17.42537
```

```
## 22.78159168% 52.73905243% 16.94112117% 31.96576117% 81.51818509% 66.50885914%
##     16.87187     17.87586     16.52042     17.02000     19.44812     18.56942
## 13.57316636% 72.02767946% 95.75497692% 85.96036911% 41.74553351% 75.75423541%
##     15.96876     18.70529     20.15365     19.74852     17.39411     18.90000
## 12.89697641% 84.40036611% 48.22059954% 73.86768851% 73.69824101%  5.75277749%
##     15.83934     19.54057     17.59071     18.87071     18.85547     15.23452
## 25.82413440% 61.11394193% 95.29512636% 84.01542867% 46.99528466% 96.75052785%
##     16.90066     18.28360     20.12371     19.48926     17.52234     20.21846
## 46.84289142% 23.36761556% 84.24755870% 43.43918408% 24.22170751%  1.71236598%
##     17.51383     16.87732     19.52020     17.40932     16.88526     14.55308
## 85.24440827% 58.27056135% 44.43893421% 97.71033176% 97.17268494% 43.72285171%
##     19.65308     18.01916     17.41552     20.99774     20.55107     17.41108
## 18.71776013% 76.24595363% 57.17659069% 66.35393803% 63.68160241% 84.55621938%
##     16.65260     18.90000     17.99449     18.56558     18.46309     19.56134
## 60.06125291% 65.00372610% 74.43937077%  6.23836399% 32.96195886% 42.14920430%
##     18.18570     18.53209     18.90000     15.35645     17.02655     17.40133
## 12.64910898%  1.81188858% 32.50640330% 85.90682147% 27.74250342% 22.88595950%
##     15.81322     14.55617     17.02231     19.74138     16.97202     16.87284
## 64.14323167% 16.91371752% 24.95216082% 28.92098962%  9.79609087% 36.06261581%
##     18.49457     16.51838     16.89206     17.01586     15.51251     17.09485
## 75.45577143% 39.29699399% 66.22044703%  1.77657036% 95.96122208% 79.98571936%
##     18.90000     17.31821     18.56227     14.55507     20.16708     19.32961
## 66.72027924% 41.24587236% 60.62650210%  9.99697151% 18.15322030% 16.98545185%
##     18.57466     17.37862     18.23826     15.53368     16.61060     16.52372
## 41.12361702% 84.56324122% 47.37622682% 75.14393383% 65.30329937% 15.62882415%
##     17.37483     19.56228     17.54359     18.90000     18.53952     16.36386
## 15.09863986% 44.41795330% 31.24085737% 12.40868827% 80.10751610% 77.32013552%
##     16.26196     17.41539     17.02000     15.78788     19.35000     18.90000
## 50.06167986% 67.83466404% 48.18816858% 84.23408743% 72.55344826% 53.59719626%
##     17.71421     18.60029     17.58890     19.51840     18.75255     17.91842
## 16.33490126% 50.49642548% 51.64629957% 59.75686992% 43.05594647% 92.04971453%
##     16.47532     17.74386     17.82166     18.15739     17.40695     20.00535
## 85.31953988% 31.67108058% 34.41605130% 96.39332630%  2.97899190% 24.32897761%
##     19.66309     17.02000     17.04007     20.19521     14.59235     16.88626
## 25.48749412% 24.09915086% 94.47427723% 96.59550174% 81.15423890% 99.85084964%
##     16.89703     16.88412     20.07028     20.20837     19.44473     22.77609
## 97.60253797%  3.14412797% 79.93268368% 55.07816458% 59.18669226% 47.88220010%
##     20.90819     14.59747     19.32073     17.98148     18.10436     17.57183
## 24.91296099% 46.38534256% 35.99187839% 45.90941428% 98.89915055% 64.06740695%
##     16.89169     17.48830     17.08937     17.46175     21.98541     18.48940
## 70.36883677% 50.94557947% 22.72563661% 16.62752677% 41.22941352% 74.28119148%
##     18.60814     17.77449     16.87135     16.49709     17.37811     18.90000
## 87.53741237% 88.58820109% 67.99261391% 74.94103750% 63.08594604% 24.34056166%
##     19.91366     19.94623     18.60078     18.90000     18.42246     16.88637
## 80.14161089% 85.96292431% 92.31216845% 46.04137866%  7.87999530% 91.09166777%
##     19.35571     19.74886     20.00617     17.46911     15.44985     20.00238
##  9.49554904% 28.32591338% 79.40266200% 54.38385801% 31.61785780% 44.82378322%
##     15.49493     16.99372     19.23201     17.95744     17.02000     17.41791
##  4.26726425% 30.74625696% 85.35248607% 38.83596610% 28.42529260% 30.33586959%
##     14.86151     17.02000     19.66749     17.30391     16.99742     17.02000
## 62.00674351% 86.75340994% 20.40956866% 26.56454812% 87.85976169% 31.21055092%
##     18.34886     19.85423     16.75558     16.92820     19.92365     17.02000
## 67.31209850% 57.36703717% 21.98351745% 71.48725458% 13.06888685% 18.91626215%
##     18.58934     17.99568     16.83853     18.65670     15.87184     16.66737
```

```
## 80.24957601% 41.20583872% 22.39940511% 93.57218065% 98.73882155% 45.31057237%
##     19.37378     17.37738     16.86045     20.01155     21.85221     17.42833
## 12.81808245% 18.01249799% 44.47380307% 58.58124807% 63.32538957% 63.98238875%
##     15.83103     16.60013     17.41574     18.04806     18.43879     18.48360
## 12.16778138% 29.05457439% 12.31505962% 40.91488745% 99.67018734% 76.10014169%
##     15.76248     17.02000     15.77801     17.36836     22.62599     18.90000
## 37.00816529% 24.56743084% 67.42958466% 26.44265243% 83.73999661%  8.90877424%
##     17.16813     16.88848     18.59225     16.92367     19.46878     15.47855
## 46.72322555% 66.06355556% 53.30952778% 91.19596991% 51.07847746% 10.50131891%
##     17.50716     18.55838     17.90415     20.00271     17.78355     15.58684
##  4.79479257% 54.75345890% 71.30119833% 74.11295131% 83.15522433% 49.14441174%
##     14.99397     17.97577     18.63998     18.89275     19.46334     17.65165
## 76.08220354% 80.09538068% 77.08752465% 17.06873046% 14.92410402% 85.47089314%
##     18.90000     19.34797     18.90000     16.52991     16.22841     19.68327
## 24.67310040% 39.86026088% 63.77517125% 82.30975408% 83.46085041%  2.02628975%
##     16.88946     17.33567     18.46947     19.45548     19.46619     14.56281
##  2.96028783% 53.83577924% 19.33191991% 20.53826631% 38.83300663%  3.57876588%
##     14.59177     17.93025     16.69829     16.76237     17.30382     14.68863
## 65.28212179% 80.93495762%  7.22998099% 77.01999543% 39.99942837% 79.26487084%
##     18.53900     19.44270     15.43172     18.90000     17.33998     19.20894
## 53.40622498% 37.87850908% 63.99103454% 16.44384186% 80.80983381% 28.09660786%
##     17.90895     17.23558     18.48419     16.48342     19.44153     16.98519
## 87.32806256% 93.31556375% 65.04365678% 11.40901975% 86.95976383% 65.43348215%
##     19.90717     20.00928     18.53308     15.68251     19.88174     18.54275
## 95.39866608% 49.40109081% 98.39737427%  4.04826868% 56.32452504%  5.57796340%
##     20.13045     17.66915     21.56854     14.80652     17.98921     15.19063
## 21.52089642% 74.90079550% 43.62576238% 70.41066410% 35.86152999% 39.38994452%
##     16.81415     18.90000     17.41048     18.60827     17.07927     17.32109
## 31.86560806% 15.69787287% 91.38069924% 72.28258590% 46.89795701% 85.03222708%
##     17.02000     16.37713     20.00328     18.72820     17.51691     19.62480
## 33.02136210% 37.55417708%  3.94666980% 69.79467750% 72.05057438% 64.72545194%
##     17.02710     17.21045     14.78101     18.60636     18.70735     18.52519
## 85.84579923% 27.52170900% 34.16188948% 90.09796020% 65.11145337% 46.34797394%
##     19.73325     16.96381     17.03771     19.99304     18.53476     17.48622
## 44.88396195% 73.40279100% 87.35191254% 54.45923128% 28.59221858% 23.11727786%
##     17.41828     18.82891     19.90791     17.96118     17.00363     16.87499
## 21.28031352% 18.48676102% 35.14927663% 12.66192277% 58.92328313% 61.42355395%
##     16.80147     16.63542     17.04689     15.81457     18.07987     18.30909
## 70.32375119% 46.27673177% 61.57636507% 83.19388423% 20.68892031% 74.24138442%
##     18.60800     17.48224     18.31951     19.46370     16.77031     18.90000
## 47.12421673% 53.25396592% 20.11453786% 74.80059259% 95.31693645% 35.32434523%
##     17.52953     17.90140     16.74004     18.90000     20.12513     17.04852
## 87.33771513% 13.70111536% 33.15433143% 71.66129923% 74.41812137% 18.53249948%
##     19.90747     15.99335     17.02834     18.67235     18.90000     16.63882
## 17.73185916% 86.99647773% 93.63507691% 56.21887785% 25.97823131% 83.69040547%
##     16.57925     19.88663     20.01564     17.98856     16.90639     19.46832
## 22.02233742% 83.39316403% 64.81756696% 27.13593268% 15.06997556% 56.01573207%
##     16.84058     19.46556     18.52748     16.94946     16.25645     17.98730
## 59.87513752% 85.82230527%  7.91459973% 71.09776745% 27.67979351% 58.26617382%
##     18.16839     19.73011     15.45082     18.62169     16.96969     18.01875
## 98.70134722% 69.24041132%  4.92303916% 85.29697452% 11.07348304% 56.63433259%
##     21.82108     18.60465     15.02618     19.66009     15.64715     17.99113
## 37.92338141% 59.32995910% 79.39228364% 81.30657717% 10.98658799% 68.42494446%
##     17.23906     18.11769     19.23027     19.44615     15.63799     18.60212
```

```
## 37.41018709% 65.33071480% 27.19118700% 62.65377004% 23.32095359% 80.02774257%
##     17.19929     18.54020     16.95151     18.39299     16.87688     19.33664
## 19.95808426% 25.72483297% 99.51649935% 33.78074255% 36.43465349% 63.07213523%
##     16.73179     16.89924     22.49831     17.03416     17.12369     18.42152
## 44.43080260% 97.41456355% 64.61671668% 72.28058111%  1.73730173% 54.55520772%
##     17.41547     20.75202     18.52249     18.72802     14.55386     17.96594
## 61.41715152% 75.15509601% 94.66101383% 22.54937261% 14.52931121% 56.81358271%
##     18.30865     18.90000     20.08243     16.86835     16.15253     17.99224
## 70.93339919% 40.63691117% 91.75832681% 72.55729849% 78.08536687% 46.20462921%
##     18.60989     17.35974     20.00445     18.75290     19.01149     17.47822
## 22.50396991% 76.93846370% 66.98903928% 22.16803334% 29.07169119% 75.58811470%
##     16.86596     18.90000     18.58133     16.84826     17.02000     18.90000
## 73.63500544%  2.14465510% 39.42447419% 73.48798802% 90.55221451% 80.10338463%
##     18.84979     14.56648     17.32216     18.83657     20.00071     19.34931
##  3.41605686%  5.69554651% 82.49853631% 54.33428572% 69.03089781% 94.94171189%
##     14.64777     15.22015     19.45724     17.95498     18.60400     20.10071
##  0.06568639% 72.88971250% 77.78107417% 83.54128832%  7.20302784% 43.21466640%
##     14.50204     18.78279     18.96055     19.46693     15.43096     17.40793
##  3.64648220% 20.12152942% 70.23718113% 50.54231337% 45.92351597% 11.44708442%
##     14.70563     16.74040     18.60774     17.74699     17.46253     15.68652
## 54.81209792% 94.53195620% 97.71798649% 91.69078285% 73.80243479% 55.56996779%
##     17.97868     20.07403     21.00410     20.00424     18.86484     17.98453
## 97.57585758%  8.70030054% 31.77857948% 11.53389849% 38.38203405% 53.95998186%
##     20.88602     15.47274     17.02000     15.69567     17.27461     17.93642
## 25.71577791%  8.40294303% 49.98907296% 93.10448200% 28.68647466% 49.54541053%
##     16.89916     15.46444     17.70925     20.00862     17.00714     17.67900
## 20.35663442% 64.99704269%  5.45324318% 65.19135209% 85.10193222% 73.31093068%
##     16.75279     18.53193     15.15931     18.53675     19.63409     18.82065
##  8.17208826% 15.19526031% 91.54207751% 37.29624164% 31.14801680% 87.10475930%
##     15.45800     16.28053     20.00378     17.19046     17.02000     19.90025
## 45.04128115% 63.69198360% 54.01141618% 45.99868061%  5.10073281% 97.10975897%
##     17.41926     18.46379     17.93897     17.46673     15.07079     20.49879
##  6.05379744% 90.78179481% 76.71940716% 24.72775439% 69.98805224% 56.39949203%
##     15.31011     20.00142     18.90000     16.88997     18.60696     17.98968
## 47.78090077% 53.10578591% 77.94339205% 25.25776459% 47.38254880% 44.61515930%
##     17.56617     17.89405     18.98772     16.89490     17.54395     17.41661
## 86.15671732% 31.45588804% 35.46103423% 24.78235832% 54.04915076% 33.35702941%
##     19.77469     17.02000     17.04979     16.89048     17.94084     17.03022
## 62.74363173% 18.14192550% 75.55908652% 85.25939037% 84.09904470% 89.40786717%
##     18.39912     16.60976     18.90000     19.65508     19.50040     19.97164
## 13.47049223% 60.51653360% 23.12048471% 41.96974782% 85.32504595% 87.69730539%
##     15.94903     18.22804     16.87502     17.40021     19.66383     19.91862
## 31.87065898%  7.27591109% 76.11394958% 34.80541727% 58.23198906% 55.26489369%
##     17.02000     15.43300     18.90000     17.04369     18.01557     17.98264
## 39.07513544% 35.03422923% 11.03386143% 46.58671003% 19.76789541% 57.03613772%
##     17.31133     17.04582     15.64297     17.49954     16.72177     17.99362
## 82.14986406% 94.39426269% 79.65792359% 54.57027170%
##     19.45399     20.06507     19.27474     17.96669
```

(d)(iii)

Before answering the question, let us first analyse the cylinders data from mtcars dataset.

```
# Printing all 32 rows of data with respect to number of cylinders
print(mtcars$cyl)
```

```
##  [1] 6 6 4 6 8 6 8 4 4 6 6 8 8 8 8 8 8 4 4 4 4 8 8 8 8 4 4 4 8 6 8 4
```

```
# Running the summary to get more info
summary(mtcars$cyl)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   4.000   4.000   6.000   6.188   8.000   8.000
```

```
#calculating the mode of cyl
y <- mtcars$cyl
y <- table(y)
names(y)[which.max(y==max(y))]
```

```
## [1] "8"
```

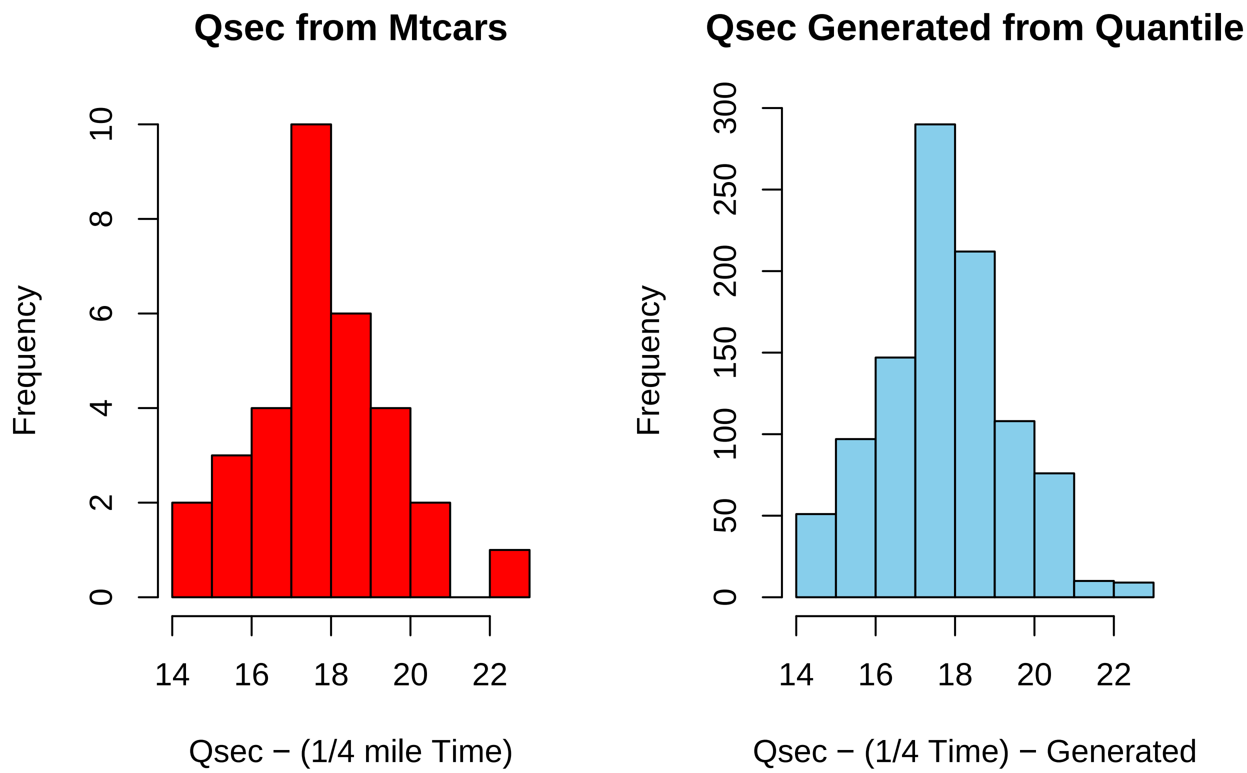
```
## Using the quantile function for cyl
quantile(mtcars$cyl, prob=runif(10))
```

```
##  38.4802% 78.30924% 36.98379% 39.03124% 34.94357% 5.158628% 87.79199% 7.831603%
##  6.000000  8.000000  6.000000  6.000000  5.665013  4.000000  8.000000  4.000000
##  3.97302%  80.8397%
##  4.000000  8.000000
```

Yes, it does work for cyl. From the actual data cyl has only 3 varieties (namely 4, 6 and 8). So when we compute quantile, we will get these 3 values irrespective of the probabilities we choose.

d(iv)

```
dat <- par(mfrow = c(1,2))
hist(mtcars$qsec, main = "Qsec from Mtcars", xlab = "Qsec - (1/4 mile Time)", col = 'red')
hist(samobs_fromqsec, main = "Qsec Generated from Quantile", xlab = "Qsec - (1/4 Time) - Generated", col
```

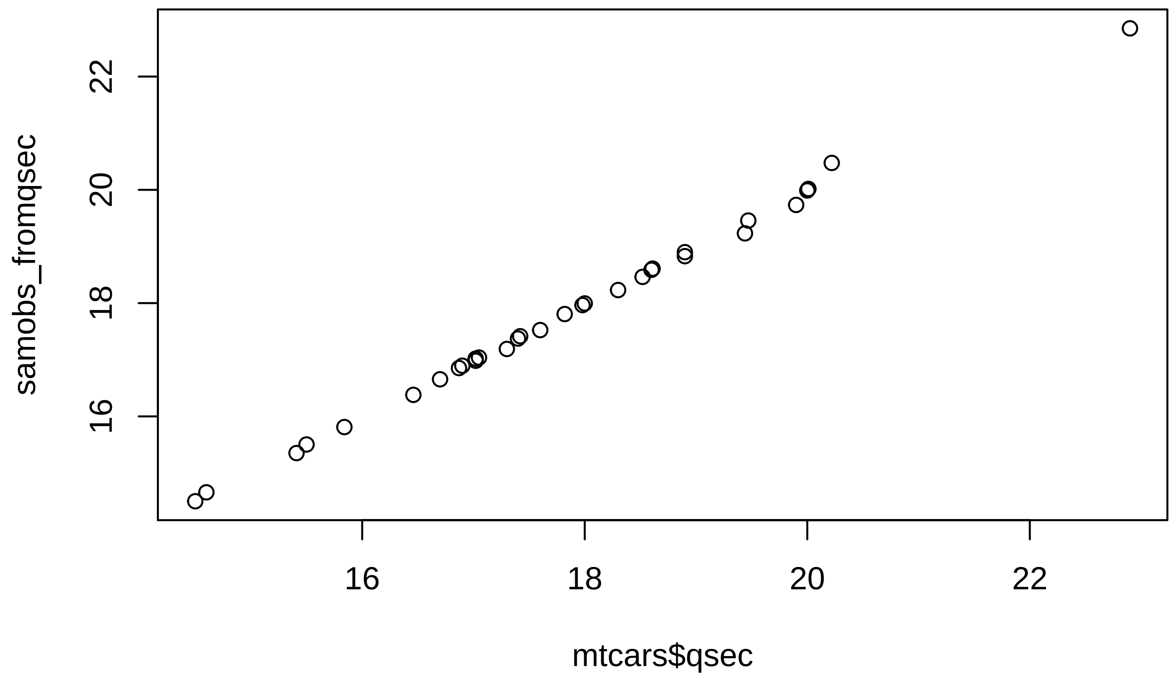



```
par(dat)
```

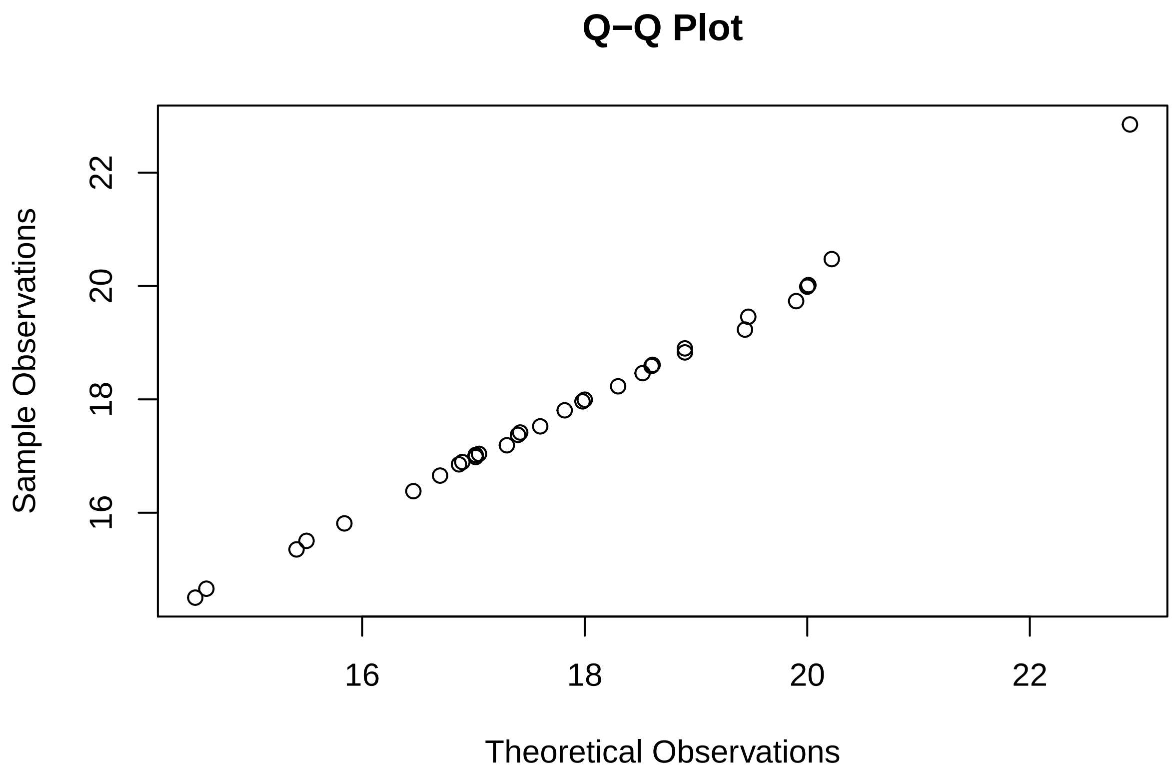
From the histograms we can infer that both the graphs follow the same distribution. Although in the left (Original Qsec data from mtcars) we can see that there is no bin at 21-22 seconds - That is, there was no car that took 21-22 seconds to attain 1/4 a mile. On the contrary on the right (Qsec generated using the quantile function) we can see that there is a bin at 21-22 since the data was geneated with prob 0 and 1. On the Y-Axis (Left side Graph) we can see that the frequency goes from 0 to 10 with a step size of 2. Whereas on the Y-axis (Right side graph) we can see that the frequency goes from 0 to 250 with a step size of 50.

d(v)

```
qqplot(mtcars$qsec, samobs_fromqsec)
```




```
qqplot(mtcars$qsec, samobs_fromqsec, main = "Q-Q Plot",xlab = "Theoretical Observations", ylab = "Sampl
```



Generally QQ plots a scatterplot using two quantiles against each other.From the graph we can infer that the points form a line that's roughly straight. This states that both the quantiles come from the same distribution.

d(vi)

Step 1 - Inorder to produce a bootstrap distribution for some estimator theta, Let us use quantile to produce some data. I am using cylinders from mtcars dataset. I used the quantile fucntion with number of bootstrap samples drawn as 100 with a two sided confidence interval(95%) argument.

```
quantile(mtcars$cyl, probs = runif(50), CI.type = "two.sided", nbboot=100)
```

```
##  56.00587%  4.199123%   76.1902%  83.41437%  62.53828%  83.00134%  39.83738%
##   6.723641   4.000000   8.000000   8.000000   8.000000   8.000000   6.000000
##  56.17437%  12.53718%  24.38958%  16.61068%  16.29795%  30.58102%  69.53286%
##   6.828107   4.000000   4.000000   4.000000   4.000000   4.000000   8.000000
##  49.74612%  64.31155%  12.35054%  65.31711%  91.66203%  28.39503%  79.67062%
##   6.000000   8.000000   4.000000   8.000000   8.000000   4.000000   8.000000
##  49.13785%  10.84139%  24.14521%  9.745784% 0.6232897%  61.11649%  13.30628%
##   6.000000   4.000000   4.000000   4.000000   4.000000   8.000000   4.000000
##  60.64681%   33.4013%  63.99745%  63.43665%  38.81281%  58.90254%  79.40596%
```

```
##   8.000000   4.708803   8.000000   8.000000   6.000000   8.000000   8.000000
##  90.96417%  94.81757%  27.15874%  23.83174%  20.02258%  71.44294%  88.76813%
##   8.000000   8.000000   4.000000   4.000000   4.000000   8.000000   8.000000
##  8.288973%  36.00035%  11.23074%  41.27613%  64.51539%  56.33739%  12.87077%
##   4.000000   6.000000   4.000000   6.000000   8.000000   6.929183   4.000000
##  1.997268%
##   4.000000
```

I would recommend using quantiles over bootsrtap. The main reason behind is that quantile function is computationally less expensive when compared to bootstrapping.