Tutorial 8 - Simulation Methods (Solutions).

Christopher A Oldnall

Welcome to the eigth tutorial of the Regression and Simulation methods module. This is the next script in developing your skills in R, whilst learning about simulation methods. Throughout this notebook we will start to consider how to implement different procedures necessary to be able to simulate data effectively. This week all exercises are taken from Week 8 of the SMSTC resources.

Exercise 0

Throughout remember we will need tidyverse, go ahead and do this as your first task.

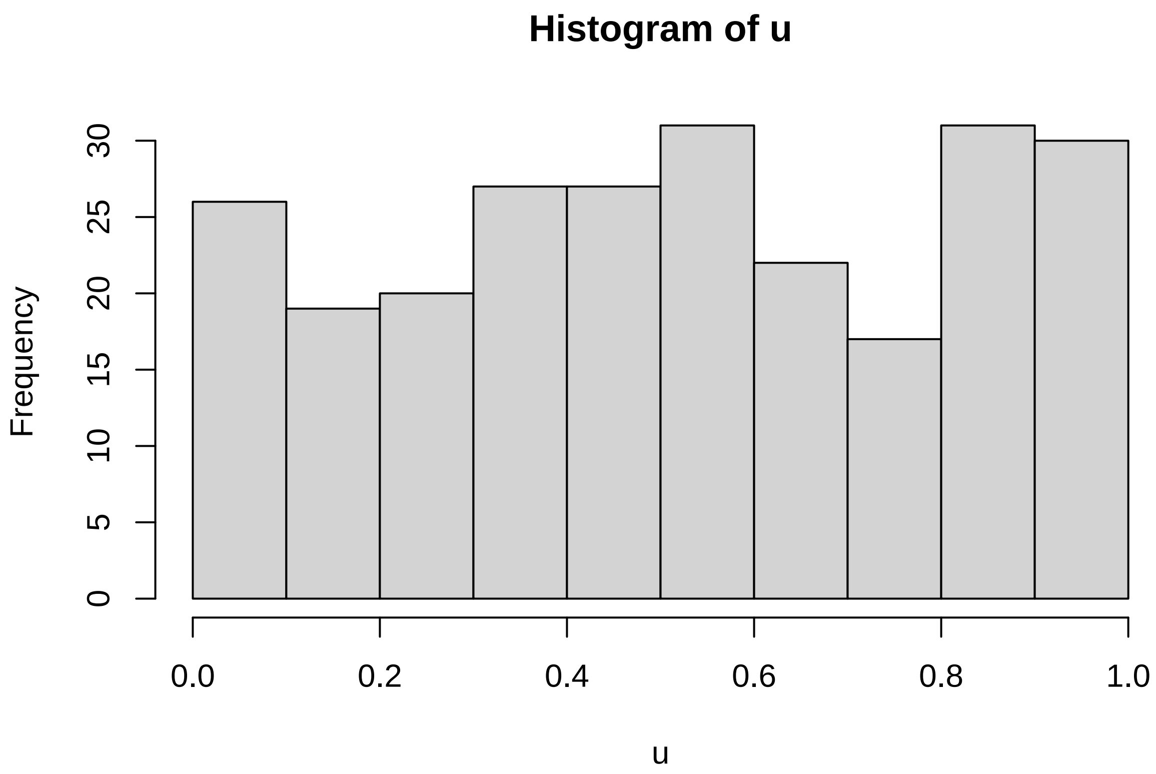
Your Answer:

```
library(tidyverse)
```

Exercise 1

In whatever way you like generate 100 random deviates of U(0, 1).

```
# BY LINEAR CONGRUENTIAL GENERATOR
set.seed(123)
m <- 4096 # 2^11
a <- 4*7+1
b <- 89
x <- rep(NA, 250)
x[1] <- 353
for (k in 2:250){
    x[k]<-(a*x[k-1]+b)%%m
  }
u <- x/m
hist(u)
```



```r
# OR BY REJECTION METHOD
u <- runif(1000, -2, 2)
h <- dnorm(0, 0, 1)
v <- runif(1000, 0, h)
accept <- rep(0, 1000)
for (i in 1:1000) {
  if (v[i] < dnorm(u[i],0,1)) accept[i] <- 1
  }
u[accept==1]
```

```
##   [1] -0.849689920 -0.364092313  0.112421952  0.205740058 -0.173541059
##   [6] -0.186663375  0.710282542  0.290533608 -1.588301269 -1.015649063
##  [11] -0.688317123  0.562027255  0.622823196  0.834121873  0.176264099
##  [16]  0.376568082 -0.843361051  0.762821114  1.181869671 -0.088816116
##  [21]  1.033838150 -0.727275969 -1.073496859 -0.341814657 -0.345102695
##  [26] -0.524618196 -1.390221009 -1.444775746 -1.067863602 -0.136150199
##  [31] -0.231199703  0.243791935 -1.173874442  1.013231457 -0.502148896
##  [36]  0.660460779 -1.620637356 -0.464121449 -0.902465422  1.258560156
##  [41] -0.205934634  1.249558038 -0.240673250  0.516884526 -0.098733704
##  [46] -0.480733849  0.451084013 -0.592808363 -1.025522109  0.672222350
##  [51] -0.329412881  1.152783336 -0.260429034  1.939827920  1.572204458
##  [56]  0.612407700 -0.625934111  0.627032512 -0.718507030 -0.132883834
##  [61]  0.046021840  0.399955837 -0.668705839 -0.045547865 -0.068390411
##  [66]  1.741199213  0.882385094 -1.430822818  0.197138624  0.341933412
##  [71] -0.381958873  0.591573917 -0.720717532 -0.769119957 -1.120929475
##  [76] -0.522044537  1.936876814 -1.432372369  0.760028406  0.477025934
##  [81]  0.691996370  0.948310952  0.084542903  0.639353799  1.287221841
##  [86]  1.145126207 -0.242273855 -0.362100190 -1.958131553  1.370917276
##  [91] -1.075352872 -1.017105288  0.928540822 -0.009890932 -0.448363881
##  [96] -0.440022259  0.287741256 -0.220927992 -1.128037325  0.009198253
## [101] -0.584381713  0.599940635 -0.501144174 -0.578218477  0.134751782
## [106] -1.115588249 -0.349015526  0.519892214 -1.264686037  0.673138599
## [111]  0.472071493 -0.511047759  0.119342743  1.498729371  0.327000399
## [116]  1.359071059 -0.750207340  0.833161289  0.377372776 -0.074840798
## [121] -0.939869074  0.258361739 -0.903333514 -0.714068974  0.479973241
## [126]  1.749256357 -0.133869191 -0.372669627  0.636921297  0.291468233
## [131]  0.405462904  0.060118909 -0.389706631 -0.543632541 -0.847042877
## [136] -0.071829578 -1.134980842  0.697505552 -1.809345490 -0.592445446
## [141] -0.364224008  1.283805296 -0.869886680  0.913577713 -0.419119461
## [146] -0.088618481  0.241013055  0.473404909 -0.286313965  0.168321469
## [151] -0.956572572 -0.411392187 -1.209021053  1.213674168  0.187304626
## [156]  0.649270568  0.532221439  0.898217386 -0.404240701  1.877425645
## [161] -1.112848261  0.372182607 -0.929914271  0.124281595 -1.327756758
## [166] -0.382403276 -0.113694889  0.696747370  1.800667917  0.065779577
## [171]  0.306076085 -0.654675176 -1.919902798  0.011252183  1.484173655
## [176]  0.940737223 -0.134110493  0.595272497  1.034372678 -0.413661622
## [181] -0.416429247  0.681128158 -1.712383611  1.016989608  1.266423550
## [186]  1.138301066  0.520527411 -0.076356680 -1.373452595 -1.967137921
## [191] -0.190166423 -0.030826685 -0.441651554 -0.141336232 -0.580867610
## [196]  1.211249092 -1.049002380 -0.584055590  1.427541681 -0.816418178
## [201]  0.815968245 -0.646434865  0.468941079 -0.854858605  0.951189612
## [206] -0.742916878 -0.029733809  0.789495065  0.565849416  0.575691660
## [211] -0.341058662  0.104118641 -1.099706597 -0.054352944 -0.446723539
## [216] -1.083020639  0.493190185 -1.453839212  1.869877977  0.060287232
## [221] -1.347718683  0.487609181  0.675086070 -0.324336412 -0.706620027
## [226] -1.228736212 -0.767521783 -0.546797826  1.135785915 -1.226485280
## [231] -0.373568535 -0.067329323 -0.312620199 -0.628764793 -0.179567796
## [236]  0.135059495  1.098366168 -0.764852669  0.339600374  1.043294501
## [241]  1.076775645  0.150708732  1.655981799 -1.258814232 -1.620150347
## [246] -1.158051684 -0.814791298  0.903932108  1.142751337 -1.528344635
## [251] -1.451730114  1.621238327  0.305207350 -0.418204564 -0.200790063
## [256]  0.826007605 -0.642749679 -0.732203008 -0.008204255 -0.895801309
## [261] -0.713097849 -0.086174465  0.189837872  0.576960882  0.385054179
## [266]  0.505027790 -0.788380339  0.254578734 -1.060721297  0.450687861
## [271] -0.051869720 -0.868028517  0.335489344  0.926830635  1.041598145
## [276] -0.567772146  0.693329926  0.095290407 -0.600792829 -1.767232815
## [281]  1.247309664 -1.501031646 -0.255480016 -0.143933493 -1.338807675
## [286]  0.339746229  0.764831326 -0.868590401  1.288120285 -0.290286877
## [291]  1.023549039  0.649542026 -0.221890412  0.508584738  1.255734885
## [296] -1.536881341  0.712895430 -0.284205616  1.885752267 -1.718044089
## [301]  0.745299787  1.147709411 -0.851599477 -1.680108351 -0.538182922
## [306]  0.144214883  0.015794848 -0.634714864 -0.141144903 -0.417357440
## [311]  0.943597401 -1.313026376 -0.180953475  1.080819038  1.260325980
## [316] -0.795429902 -0.541315888 -0.751548832  0.075219694  0.716053663
## [321]  1.612934241 -0.788449502  0.750430327 -0.211852021  1.265912555
## [326]  0.956126742 -0.605122025  1.317003294  0.142165567 -0.901818546
## [331]  1.203792848  1.328431097 -0.892579710 -1.674133760 -1.180245188
## [336]  0.277530651  0.118856366  0.347834627  0.662940570  0.119569834
## [341]  0.039371598 -1.195677536  0.601046002  0.615066729 -0.418984954
## [346]  0.188093671  0.213255950  1.624192446  0.349845468 -0.306145396
## [351]  1.798341292 -0.346778368  0.266936331 -0.039746188  1.251410752
## [356]  1.416399910 -0.528416209 -0.872753321  0.666820631  0.330958947
## [361]  0.106360318  1.523056460  0.033482519 -0.650020011  0.745961404
## [366] -1.096726307  1.205718335  1.290869567 -0.676008685 -0.503322449
## [371]  0.518981695  1.127292261  1.019461814  1.613603539  1.462003256
## [376]  1.101630365 -0.492734333 -0.542355673 -0.904994921  1.401869923
## [381] -0.550393150 -0.168313756  0.918526658  1.815804991  1.008150165
## [386]  1.275821533 -0.328891593  0.375353972 -0.797929555  0.458248629
## [391]  1.011728232 -0.096997977  0.268477865  0.946481384  1.636586018
## [396]  0.011633216 -0.597820908  1.225740655 -1.530675091  0.850746220
## [401] -1.371328699  0.588229400 -1.305929015  0.085255385 -1.654888427
## [406] -0.318260396  0.349112497 -0.162301313 -0.207427545  0.934991033
## [411]  0.859019242  0.202466887  0.089343064 -0.082913760  0.539202039
## [416]  0.158243656  0.332122381 -0.285547635 -0.576953737 -0.959470105
## [421]  0.527155487  0.185705035 -0.494222027 -0.284237635  0.523094394
## [426]  0.083369405  0.638485349 -0.052708507 -1.568452479 -0.325323367
## [431]  0.871538186  0.970156529  1.487995295  0.431471844  1.388965192
## [436]  0.451118596  1.172896671  0.589388062 -0.736748843  1.217025117
## [441]  1.968682477  0.389413930 -0.714686226  0.111716001  0.229750332
## [446] -0.040651794 -0.022830552  0.055124557 -1.054640444  0.301742679
## [451] -0.070968354  0.277410755 -0.350867548  0.732041130  0.703560005
## [456] -1.487742322 -0.996885864 -0.673812034 -0.370922840  0.541470308
## [461] -0.964795195  1.277868560  0.617603066  1.246878337 -0.154524293
## [466] -0.867469211  0.170126455 -0.042288784  1.204594945 -0.369477369
## [471] -0.554135118 -0.963020636 -0.117272666 -0.536618107 -1.514911783
## [476] -0.948814783  1.874564672 -0.046018071 -0.088711881  0.670560925
## [481]  0.780420978 -0.546953733  1.101188919 -0.819962915  0.359606495
## [486]  0.246702435 -0.754916764  0.422347366  0.972819661 -0.195324374
## [491] -0.641777944  1.087637929 -0.148099155  0.883361400  0.666022869
## [496]  0.288294883  0.815251800  0.628884242  0.945336116  0.450893874
## [501]  0.201036186 -0.948974889 -1.479821619  0.860534111  0.017759244
## [506] -0.255809313 -0.302191620  0.589768575 -0.125523350  0.471704493
## [511] -0.916738270  0.030731453  0.192129150 -1.437415242  1.047941293
## [516]  0.109579762  1.443957430  0.772795605  0.527400701  1.684260815
## [521]  0.701449525 -1.405625150  0.981367369 -0.316866498 -0.809107687
## [526] -0.962293319  0.262617374  0.678414184  0.186091177  1.036667208
## [531] -0.476385199 -1.796479546  0.170393482  1.409458401  0.334251453
## [536]  0.673294574  0.045258386 -1.984414628  0.304980676 -0.744629712
## [541]  1.837863135  0.364775028  0.125637343 -0.464253314 -0.721787082
## [546]  0.791786314  0.737945892 -0.607939789  0.218727318 -1.183616467
## [551]  1.082491846  0.385451843  1.830678970 -1.365246410  0.103897097
## [556]  1.478824281 -0.039103207 -0.443318708 -0.329780160 -1.628296718
## [561] -0.378334019 -0.632742223 -0.338970187 -0.783790124  0.241122150
## [566] -0.511369029  0.581710063 -0.360216286 -0.296379464  0.032632339
## [571] -0.201600357  0.493045534 -1.440088276  0.277773099  0.193122296
## [576] -1.532689510 -0.086522213  1.127875659 -0.922366210  0.572886238
## [581]  1.792475351 -0.323817100 -0.168492812  0.846770317  1.679392231
## [586]  0.508442802  1.029316637 -0.267259438 -1.104879996  0.288594718
## [591] -0.399323286  0.261861523  0.568455278 -0.434004993  0.838319419
```

```
1-sum(accept)/1000
```

```
## [1] 0.405
```

Exercise 2

Use the numbers from exercise 1 to generate 250 poisson deviates.

```
xpois <- rep(NA,250)
for (j in 1:250){
  xpois[j] <- max(c(0,which(ppois(0:15,2)<u[j])))
  }
```

Exercise 3

Generate $y_1, ..., y_{60}$ from the binary GLM: $\text{logit}(\mathbb{E}(Y_i)) = -3 + 0.1x_i$, where $x_i = i$ for $i = 1, ..., 60$. Calculate the maximum likelihood estimate of the intercept and slope parameter. Repeat these two steps 1000 times. Use the 1000 resulting maximum likelihood estimates to check whether maximum likelihood estimation in the given scenario is unbiased.

```
x <- 1:60
runs <- 10000
MLEs1 <- rep(NA,runs)
MLEs2 <- rep(NA,runs)
for (k in 1:runs){
  y <- rbinom(60,size=1,plogis(-3+0.1*x))
  mod <- glm(y~x,family=binomial)
  MLEs1[k] <- mod$coeff[1]
  MLEs2[k] <- mod$coeff[2]
}
mean(MLEs1)
```

```
## [1] -3.224592
```

```
mean(MLEs2)
```

```
## [1] 0.1076154
```
