HW4 Computing in Stats

Elias Washor

2024-09-26

Question 1

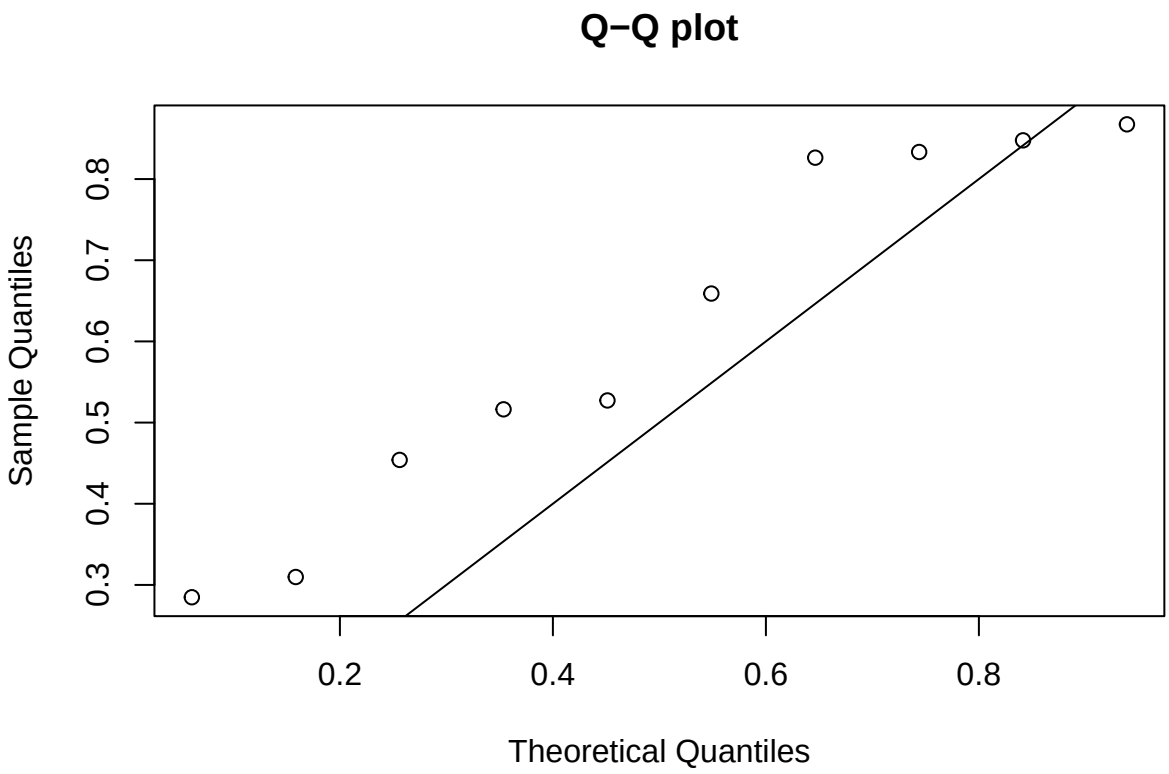
```
#set.seed(5400)
LCG <- function(n, a, c, m, seed) {
  state <- seed
  random_numbers = rep(NA, n)
  for (i in 1:n) {
    state <- (a * state + c) %% m
    random_numbers[i] <- (state/m)
    }
  return (random_numbers)
}

LCG(10, 1103515245, 12345, 2^31, 5400)
```

```
##  [1] 0.8673853 0.8263298 0.8333236 0.3097559 0.4540139 0.2848278 0.8476917
##  [8] 0.5272413 0.6589490 0.5163056
```

```
#runif(10)
```

Question 2 - Q-Q Plot

```
#library(MKpower)
ten_unifs <- LCG(10, 1103515245, 12345, 2^31, 5400)

plot(qunif(ppoints(10)), sort(ten_unifs),
     xlab = "Theoretical Quantiles",
     ylab = "Sample Quantiles", main ="Q-Q plot")

abline(0, 1)
```



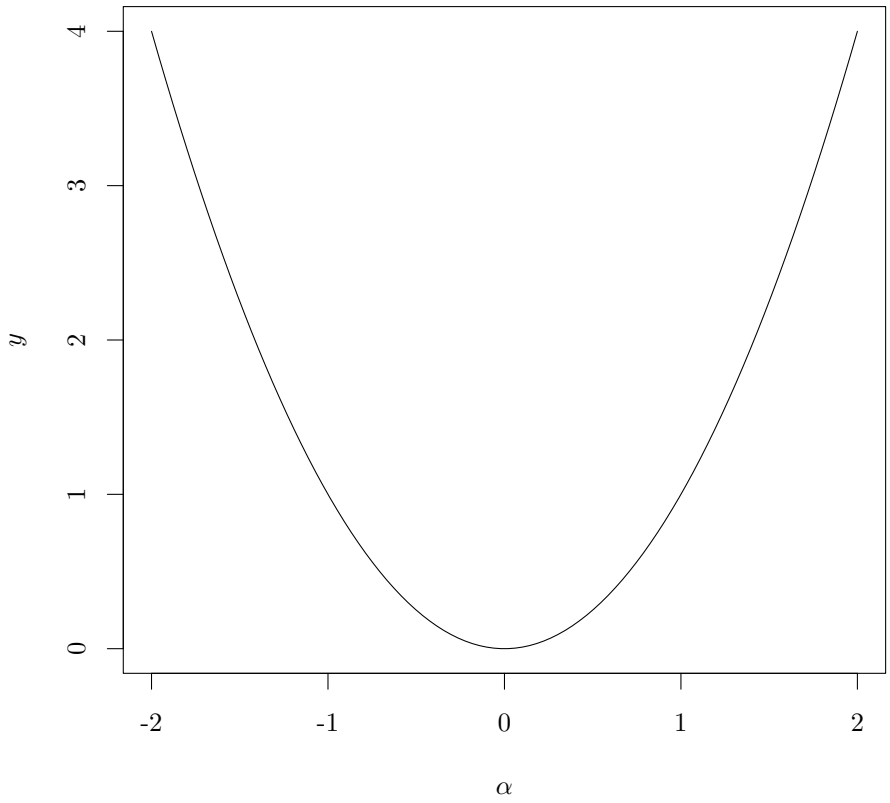
The numbers indeed follow the Uniform (0, 1) distribution as the points form a roughly straight line and fit close to the reference uniform line.

Question 3

```
library(tikzDevice)

alpha = seq(-2, 2, len=100)
y = alpha^2
tikz('myplot.tex', width=5, height=5)
plot(y~alpha, xlab="$\\alpha$", ylab="$y$", type='l')
dev.off()
```

```
'''
\begin{figure}[ht]
\centering
\input{myplot.tex}
\end{figure}

\begin{table}[ht]
\centering
\begin{tabular}{lc}
  \hline
  Name & PDF \\
```

```
  \hline
  Gamma & $p(y \mid \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} y^{\alpha - 1} \exp(- \beta y)$
  \hline
\end{tabular}
\caption{A univariate continuous density}
\end{table}

'''
```



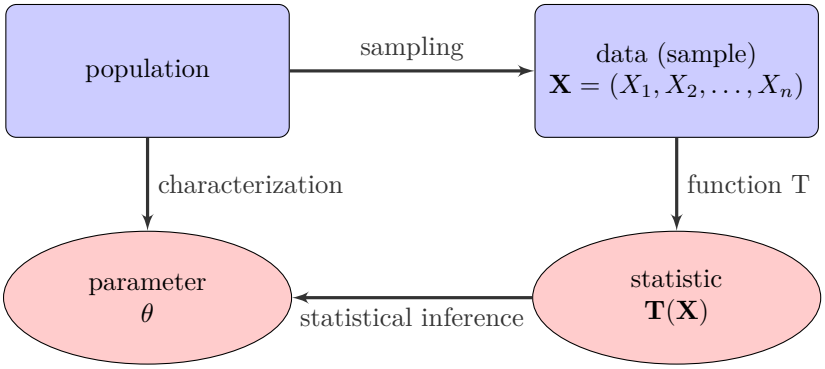
Name	PDF
Gamma	$p(y \mid \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} y^{\alpha-1} exp(-\beta y)$

Table 1: A univariate continuous density

Question 4)

```
"""
\tikzstyle{block} = [rectangle, draw, fill = blue!20,
    text width = 10em, text centered, rounded corners, minimum height=5em]
```

```
\tikzstyle{line} = [draw, very thick, color=black!80, -latex']
\tikzstyle{cloud} = [draw, shape=ellipse, fill=red!20, node distance=2.5cm,
    minimum height=5em, text centered, text width=7em]

\begin{tikzpicture}[scale=0.5, auto]
  \node [block] (pop) {population};
  \node [block, right of=pop, node distance=7cm] (rs) {data (sample) \\ $\mathbf{X} = (X_1, X_2, \ldots
  \node [cloud, below of=rs, node distance=3cm] (re) {statistic \\ $\mathbf{T(X)}$};
  \node [cloud, left of=re, node distance=7cm] (le) {parameter \\ $\theta$};

  \path [line] (pop) -- (rs) node[pos=0.5, above, align=left] {sampling};
  \path [line] (rs) -- (re) node[pos=0.5, right, align=left] {function T};
  \path [line] (re) -- (le) node[pos=0.5, below, align=left] {statistical inference};
  \path [line] (pop) -- (le) node[pos=0.5, right, align=left] {characterization};
\end{tikzpicture}

"""
```



Question 5)

```
oldgroup <- split(1:30, rep(1:6, rep(5, 6)))


regroup_helper <- function() {
  new_grps <- matrix(nrow= 6, ncol=5)
  units <- 1:30
  for (r in 1:6){
    exc <- 1:5 + (5 * r - 1)
    candidates <- setdiff(units, exc)
    i <- 1
    while (i < 6) {
      selected <- sample(candidates, 1)
      row <- (selected - 1) %/% 5
      neighbors <- 1:5 + (5 * row)
      new_grps[r,i] <- selected
      candidates <- setdiff(candidates, c(selected, neighbors))
      units <- setdiff(units, selected)

      i <- i + 1
```

```
    }
  }
  return(new_grps)
}

regroup <- function(grp) {
  valid <- FALSE
  while (!valid) {
    M <- regroup_helper()
    valid <- (length(unique(unlist(as.list(M)))) == 30)
    }
  return(M)
}

(final <- regroup(oldgroup))
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]   28   15   19    1   22
## [2,]   30    5   20   25    7
## [3,]   14   21    3   10   27
## [4,]    9   11   26   18    2
## [5,]   16    4   13   24    6
## [6,]   17    8   29   23   12
```

```
(length(unique(unlist(as.list(final)))))
```

```
## [1] 30
```