R Lab. - Exercise 3

Michele Guadagnini - Mt. 1230663

May 4, 2020

Exercise 1 - TOLC-I University orientation and evaluation test

A) Using the integrate() R function, determine the constant c (and verify it analytically)

```
# definition of pdf with c = 1
const <- 1
dtolc <- function(x){
     z <- ifelse((x>=1) & (x<=2),
                 const*(x-1)*(2-x),
                 0)
    return (z)
}
# computing the constant c by setting the integral of pdf to 1
integral <- integrate(dtolc, 1., 2.)
const <- 1./integral$value
```

The appropriate value for the integration constant is: 6.
Analitically:

$$\int_1^2 \frac{f(x)}{const} dx = \int_1^2 (x-1)(2-x)dx = \left[-\frac{x^3}{3} + \frac{3}{2}x^2 - 2x \right]_1^2 = \frac{1}{6} \tag{1}$$

So, the constant needed to correctly normalize the pdf is:

$$\frac{1}{6} * const = 1 \implies const = 6 \tag{2}$$

B) Write the set of four R functions and plot them

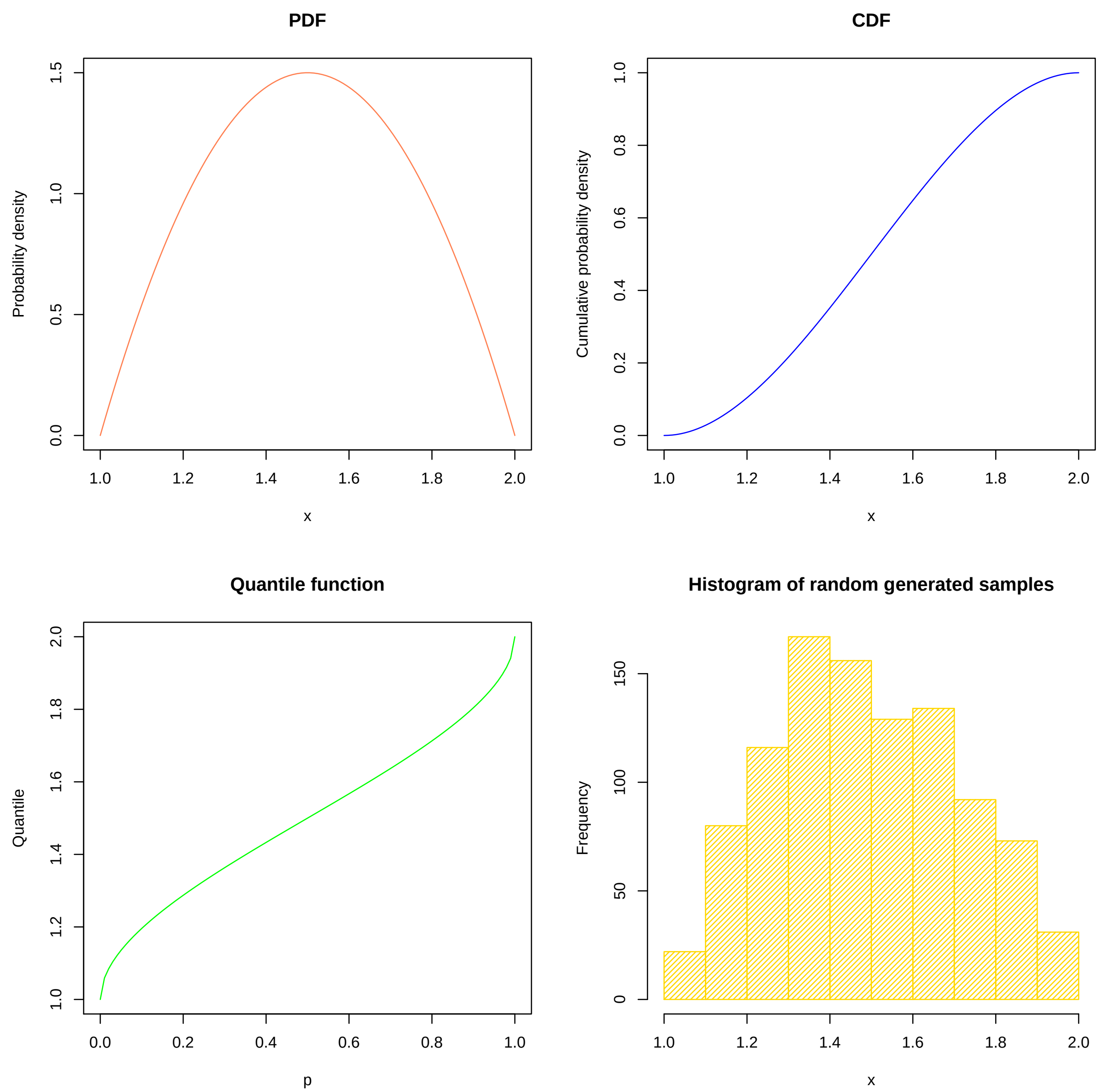
```
# defining the 3 remaining functions
ptolc <- Vectorize(
     function(x){
        if(x<=1) { return (0) }
        if(x>=2) { return (1) }
        else { return (integrate(dtolc,1,x)$value) }
     }
)

qtolc <- Vectorize( inverse(ptolc,lower=1,upper=2) )

rtolc <- function(n) {
     us <- runif(n)
     return (qtolc(us))
}

par(mfrow=(c(2,2)))
x <- seq(1., 2., length=100)
plot(x, dtolc(x), type='l', col='coral', main="PDF",
     ylab="Probability density", xlab="x")
```

```r
plot(x, ptolc(x), type='l', col='blue', main="CDF",
     ylab="Cumulative probability density", xlab="x")
p <- seq(0,1,length=100)
plot(p, qtolc(p), type='l', col='green', main="Quantile function",
     ylab="Quantile", xlab="p")
hist(rtolc(1000), col='gold', density=30, main="Histogram of random generated samples",
     ylab="Frequency", xlab="x")
```



C) Evaluate the probability that the student will finish the aptitude test in more than 75 minutes and that it will take between 90 and 120 minutes.

```r
p75    <- integrate(dtolc, 1.25, 2)
p90120 <- integrate(dtolc, 1.5, 2)
```

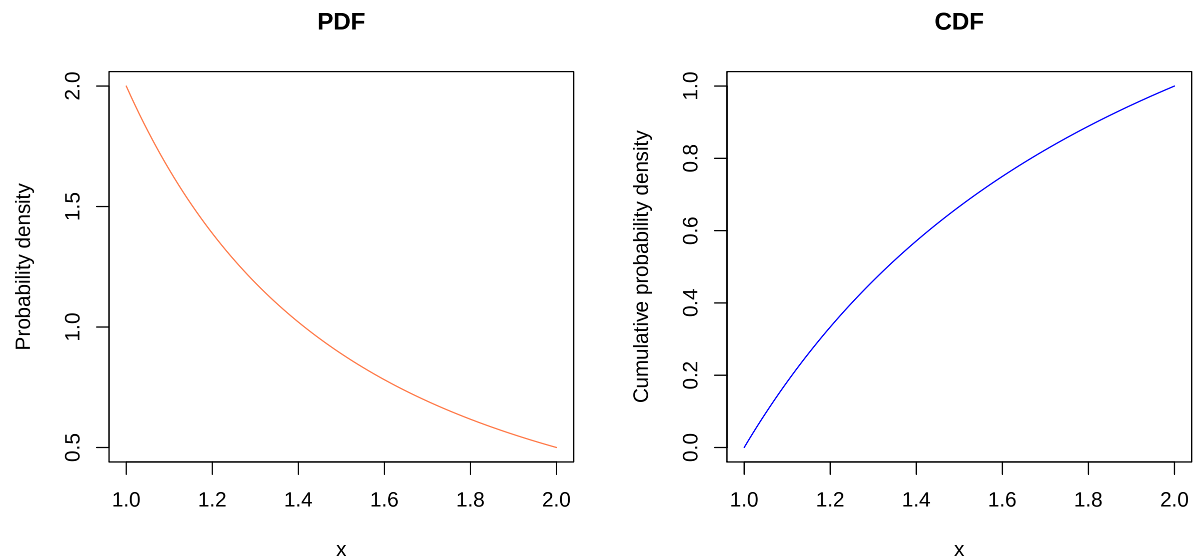
The probability for the test to last more than 75 minutes is: 0.84375.
The probability that the test lasts between 90 and 120 minutes is: 0.5.

Exercise 2 - Lifetime of tyres

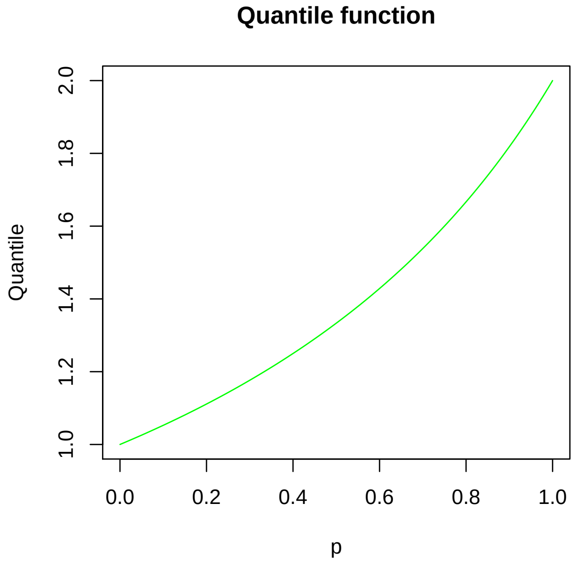
A) Write the set of four R functions and plot them

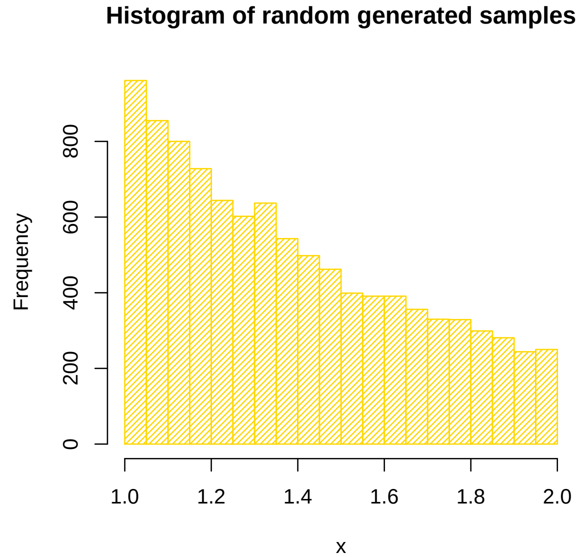
```
dtyres <- function(x){
    z <- ifelse((x>=1)&(x<=2), 2/x^2, 0)
    return (z)
}
ptyres <- function(x){
    z <- ifelse((x>=1)&(x<=2), 2*(1 - 1/x), 0)
    return (z)
}
qtyres <- function (p) {
     z <- ifelse((p>=0)&(p<=1), 2/(2-p), -100)
     return (z)
}
rtyres <- function (n) {
     us <- runif(n)
     return (qtyres(us))
}

par(mfrow=c(1,2))
x <- seq(1., 2., length=100)
plot(x, dtyres(x), type='l', col='coral', main="PDF",
     ylab="Probability density", xlab="x")
plot(x, ptyres(x), type='l', col='blue', main="CDF",
     ylab="Cumulative probability density", xlab="x")
```




```
par(mfrow=c(1,2))
p <- seq(0, 1, length=100)
plot(p, qtyres(p), type='l', col='green', main="Quantile function",
     ylab="Quantile", xlab="p")
hist(rtyres(10000), col='gold', density=30,  main="Histogram of random generated samples",
     ylab="Frequency", xlab="x")
```





B) Determine the probability that tires will last less than 15000 km

```
p15k <- integrate(dtyres, 1, 1.5)
```

The probability that the tyres last less than 15000km is: 0.6666667.

C) Sample 3000 random variables from the distribution and determine the mean value and the variance

```
nsamples <- 3000
samples <- rtyres(nsamples)

mean  <- sum(samples)/nsamples
mean2 <- sum(samples*samples)/nsamples
var   <- mean2 - mean*mean
```

Mean and variance of the random generated data are:

- mean: 1.3880386
- variance: 0.0762904

Exercise 3 - Markov's inequality

Markov's inequality represents an upper bound to probability distributions. Having defined a function:

$$G(k) = 1 - F(k) \equiv P(X \geq k) \tag{3}$$

Plot G(k) and the Markov's upper bound for the following distributions:

A) exponential distribution ($\lambda = 1$)

B) uniform distribution between 3 and 5

C) binomial distribution with n=1 and p=0.5

D) poisson distribution with $\lambda = 0.5$

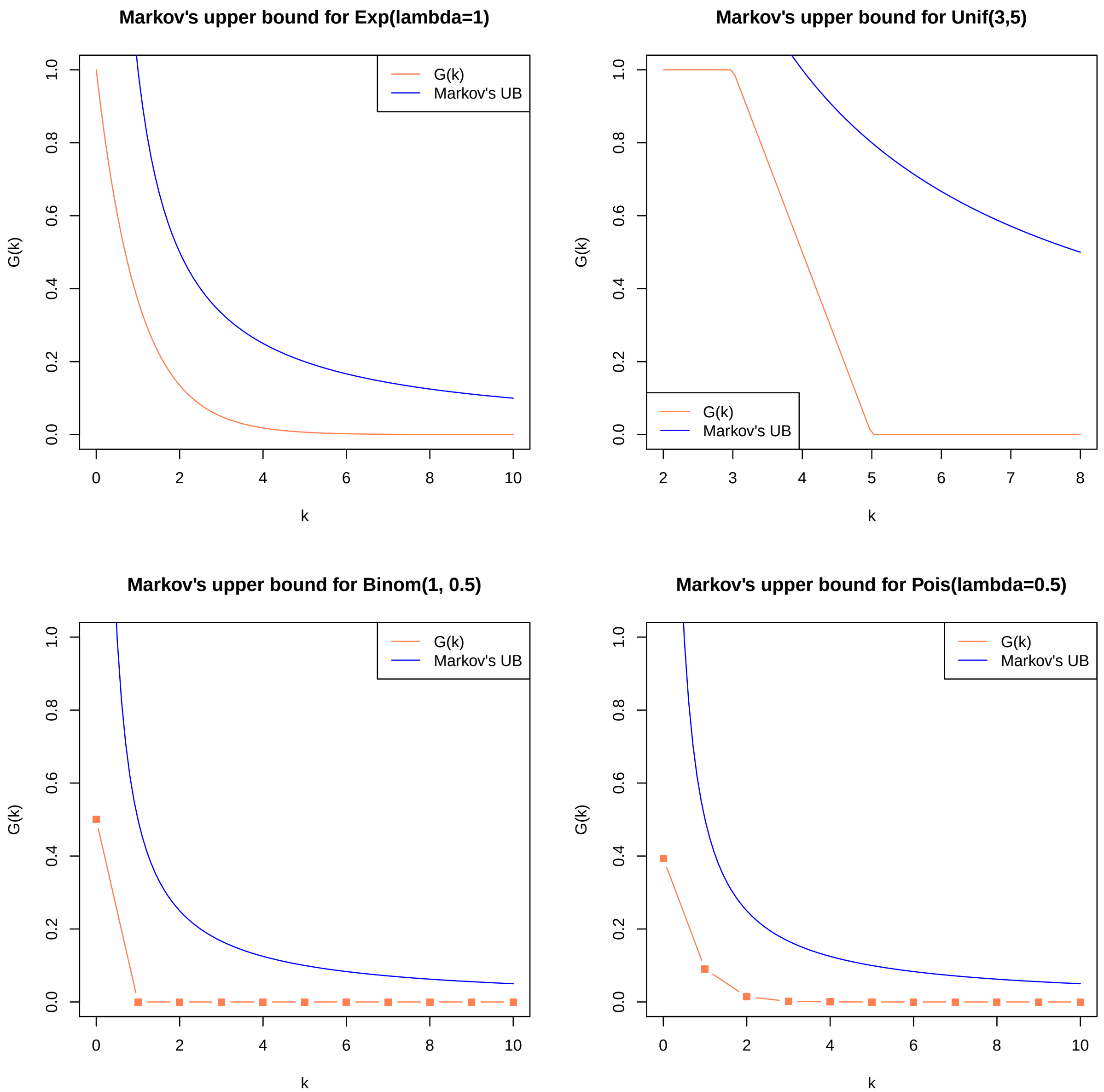
```r
par(mfrow=c(2,2))

## A) exponential distribution
lambda <- 1
Eexp <- 1/lambda
Gkexp <- function(k){ return(1-pexp(k, rate=lambda))}
MBexp <- function(k){ return(Eexp/k) }
ks <- seq(0,10,length=100)
plot(ks, Gkexp(ks), type='l', col='coral', main="Markov's upper bound for Exp(lambda=1)",
     xlab="k", ylab="G(k)", ylim=c(0,1))
lines(ks, MBexp(ks), type='l', col='blue')
legend("topright", legend=c("G(k)","Markov's UB"), col=c("coral", "blue"), lty=1:1)

## B) uniform distribution
x1 <- 3
x2 <- 5
Eunif <- (x1+x2)/2
Gkunif <- function(k){ return(1-punif(k, x1, x2)) }
MBunif <- function(k){ return(Eunif/k) }
ks <- seq(x1-1,x2+3,length=100)
plot(ks, Gkunif(ks), type='l', col='coral', main="Markov's upper bound for Unif(3,5)",
     xlab="k", ylab="G(k)", xlim=c(x1-1,x2+3), ylim=c(0,1))
lines(ks, MBunif(ks), type='l', col='blue')
legend("bottomleft", legend=c("G(k)","Markov's UB"), col=c("coral", "blue"), lty=1:1)

## C) binomial distribution
n <- 1
p <- 0.5
Ebinom <- n*p
Gkbinom <- function(k){ return(1-pbinom(k, n, p)) }
MBbinom <- function(k){ return(Ebinom/k) }
ks <- seq(0,10,length=11)
kc <- seq(0,10,length=100)
plot(ks, Gkbinom(ks), type='b', col='coral', main="Markov's upper bound for Binom(1, 0.5)",
     xlab="k", ylab="G(k)", ylim=c(0,1), pch=15)
lines(kc, MBbinom(kc), type='l', col='blue')
legend("topright", legend=c("G(k)","Markov's UB"), col=c("coral", "blue"), lty=1:1)
```

```r
## D) poisson distribution
lambda <- 0.5
Epois <- lambda
Gkpois <- function(k){ return(1-ppois(k, lambda)) }
MBpois <- function(k){ return(Epois/k) }
ks <- seq(0,10,length=11)
kc <- seq(0,10,length=100)
plot(ks, Gkpois(ks), main="Markov's upper bound for Pois(lambda=0.5)",
     col='coral', type='b', xlab="k", ylab="G(k)", ylim=c(0,1), pch=15)
lines(kc, MBpois(kc), type='l', col='blue')
legend("topright", legend=c("G(k)","Markov's UB"), col=c("coral", "blue"), lty=1:1)
```



Exercise 4 - Chebyshev's inequality

Use R to show, with a plot, that Chebyshev's inequality is is an upper bound to the following distributions:

A) normal distribution, N(μ = 3, σ = 5)

B) exponential distribution, Exp(λ = 1)

C) uniform distribution, U (1 - $\sqrt{2}$, 1 + $\sqrt{2}$)

D) Poisson distribution, Pois(λ = 1/3)

```
par(mfrow=c(2,2))

## A) normal distribution
mu <- 3
sigma <- 5
Gknorm <- function(k){ return(1 -pnorm(k*sigma+mu,mu,sigma)
                              +pnorm(mu-k*sigma, mu, sigma)) }
CBnorm <- function(k){ return(1/(k**2)) }
ks <- seq(0,10,length=100)
plot(ks, Gknorm(ks), main="Chebyshev's upper bound for Normal distribution",
    xlab="k", ylab="P(|x-mu|>=k*sigma)", ylim=c(0,1), type='l', col='coral')
lines(ks, CBnorm(ks), type='l', col='blue')
legend("topright", legend=c("P(|x-mu|>=k*sigma)","Chebyshev's UB"),
       col=c("coral", "blue"), lty=1:1)

## B) exponential distribution
lambda <- 1
Eexp <- 1/lambda
Sexp <- 1/lambda
Gkexp <- function(k){ return(1 -pexp(k*Sexp+Eexp,rate=lambda)
                             +pexp(Eexp-k*Sexp,rate=lambda)) }
MBexp <- function(k){ return(1/k^2) }
ks <- seq(0,10,length=100)
plot(ks, Gkexp(ks), main="Chebyshev's upper bound for Exponential distribution",
     xlab="k", ylab="P(|x-mu|>=k*sigma)", ylim=c(0,1), type='l', col='coral')
lines(ks, MBexp(ks), type='l', col='blue')
legend("topright", legend=c("P(|x-mu|>=k*sigma)","Chebyshev's UB"),
       col=c("coral", "blue"), lty=1:1)

## C) uniform distribution
x1 <- 1-2^0.5
x2 <- 1+2^0.5
Eunif <- (x1+x2)/2
Sunif <- sqrt(((x2-x1)^2)/12)
Gkunif <- function(k){ return(1 -punif(Eunif+k*Sunif,x1,x2)
                              +punif(Eunif-k*Sunif,x1,x2)) }
CBunif <- function(k){ return(1/k^2) }
ks <- seq(0,10,length=100)
plot(ks, Gkunif(ks), main="Chebyshev's upper bound for Uniform distribution",
     xlab="k", ylab="P(|x-mu|>=k*sigma)", ylim=c(0,1), type='l', col='coral')
lines(ks, CBunif(ks), type='l', col='blue')
legend("topright", legend=c("P(|x-mu|>=k*sigma)","Chebyshev's UB"),
       col=c("coral", "blue"), lty=1:1)
```

```r
## D) poisson distribution
lambda <- 1/3
Epois <- lambda
Spois <- lambda
Gkpois <- function(k){ return(1 -ppois(Epois+k*Spois,lambda)
                              +ppois(Epois-k*Spois,lambda)) }
CBpois <- function(k){ return(1/k^2) }
ks <- seq(0,10,length=11)
kc <- seq(0,10,length=100)
plot(ks, Gkpois(ks), main="Chebyshev's upper bound for Poisson distribution",
     xlab="k", ylab="P(|x-mu|>=k*sigma)", ylim=c(0,1), type='b', col='coral', pch=15)
lines(kc, CBpois(kc), type='l', col='blue')
legend("topright", legend=c("P(|x-mu|>=k*sigma)","Chebyshev's UB"),
       col=c("coral", "blue"), lty=1:1)
```