MATH 556: MATHEMATICAL STATISTICS I

PROBABILITY FUNCTIONS IN R

Several functions are available in R to perform calculations for probability distributions. The main function we use to specify probability distributions for a random variable X defined on the probability space $\mathcal{P} = (\Omega, \mathcal{F}, P)$ is the *cumulative distribution function* (cdf), $F_X(.)$, defined for any $x \in \mathbb{R}$ by

$$F_X(x) = P_X[X \leq x] \equiv P_X((-\infty, x]) = P(\{\omega \in \Omega : X(\omega) \leq x\})$$

provided the set $\{\omega \in \Omega : X(\omega) \leq x\} \in \mathcal{F}$; recall that the probability space for X is now $(\mathbb{R}, \mathcal{B}, P_X)$, where $\mathcal{B}$ is the (Borel) sigma algebra generated on $\mathbb{R}$ by – for example – the half-open sets $(-\infty, x]$ for $x \in \mathbb{R}$.

For convenience, we may also use representations of F_X via *mass* or *density* functions. Consider the (minimal) *support* $\mathbb{X}$ defined as the smallest (measurable) set in $\mathbb{R}$ such that

$$P_X(\mathbb{X}) = 1.$$

- If $\mathbb{X}$ is a *countable* set, say

  $$\mathbb{X} = \{t_1, t_2, \ldots\} \qquad \text{for } t_1 < t_2 < \cdots$$

  then X is a discrete random variable, and we may specify the *probability mass function* (pmf), $f_X(.)$ as the function such that

  $$P_X(B) = \sum_{t_j \in B} f_X(t_j)$$

  and where

  $$f_X(x) = P_X[X = x] \equiv P(\{\omega \in \Omega : X(\omega) = x\}) \qquad x \in \mathbb{R}.$$

  Specifically,

  $$F_X(x) = \sum_{t \in \mathbb{X} \,:\, t \leq x} f_X(t) \qquad x \in \mathbb{R}$$

  In this case, $F_X(x)$ is *non-decreasing* in x.

- If $F_X(x)$ can be represented (using the standard notion of integration)

  $$F_X(x) = \int_{-\infty}^{x} f_X(t)\, dt \qquad x \in \mathbb{R}$$

  then X is a continuous random variable (that is, $F_X(x)$ is absolutely continuous with respect to x), and $f_X(x)$ is the *probability density function* (pdf). By standard calculus results, we have that

  $$f_X(x) = \left.\frac{dF_X(t)}{dt}\right|_{t=x}$$

  wherever $F_X(x)$ is differentiable. In this case, $F_X(x)$ is *monotonically increasing* in x on support $\mathbb{X}$, and we have that

  $$f_X(x) > 0 \quad x \in \mathbb{X}$$

  and we may take $f_X(x) = 0$ for $x \notin \mathbb{X}$. We have

  $$\int_{-\infty}^{\infty} f_X(x)\, dx = \int_{\mathbb{X}} f_X(x)\, dx = 1.$$

We also have the notion of an inverse function for F_X. The *quantile function*, $Q_X(.)$, is defined for $0 < p < 1$ by

$$Q_X(p) = \inf\{x \in \mathbb{R} : p \leq F_X(x)\}.$$

That is, for any fixed p, $0 < p < 1$, $Q_X(p)$ is the pth *quantile*, the smallest x value for which the inequality $p \leq F_X(x)$ holds: if we track the value of $F_X(x)$ as x increases, there must exist a point where the $F_X(x)$ first passes p – this point coincides with $Q_X(p)$.

Example: Continuous case
Suppose

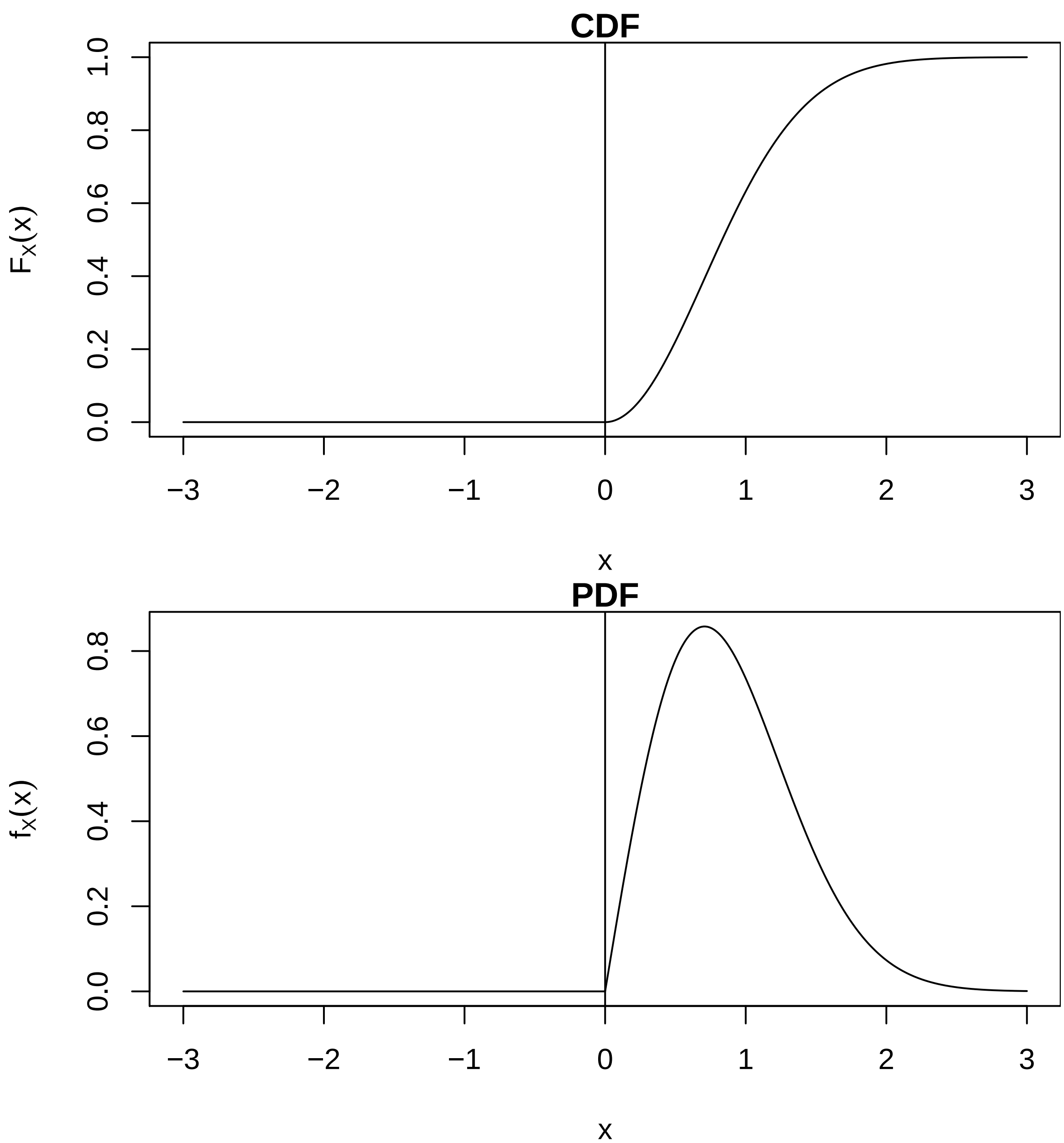
$$F_X(x) = \begin{cases} 0 & x < 0 \\ 1 - e^{-x^2} & x \geq 0 \end{cases}.$$

Then if follows that

$$f_X(x) = \begin{cases} 0 & x < 0 \\ 2xe^{-x^2} & x > 0 \end{cases}.$$

and we may define $f_X(0) = 0$ for completeness.

```
x<-seq(-3,3,by=0.01)
Fx<-0*(x<0) + (1-exp(-x^2))*(x >= 0)
fx<-0*(x<0) + 2*x*exp(-x^2)*(x >= 0)
par(mfrow=c(2,1),mar=c(4,4,1,0))
plot(x,Fx,type='l',main='CDF',ylab=expression(F[X](x)));abline(v=0)
plot(x,fx,type='l',main='PDF',ylab=expression(f[X](x)));abline(v=0)
```



We have for $0 < p < 1$ that

$$Q_X(p) = \sqrt{-\log(1-p)}$$

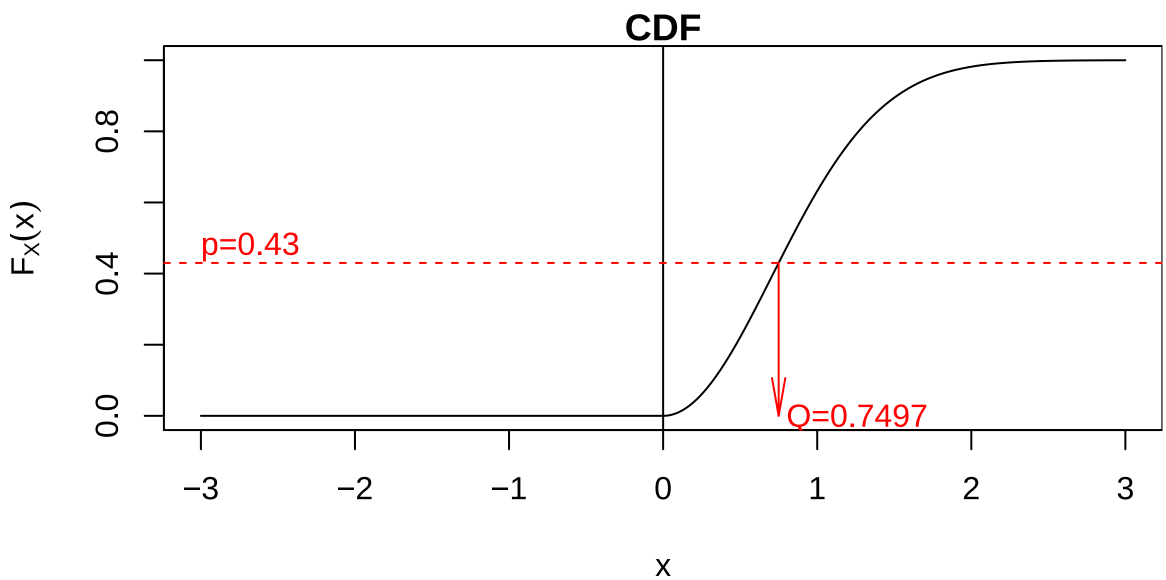
For example, if $p = 0.43$, we have that

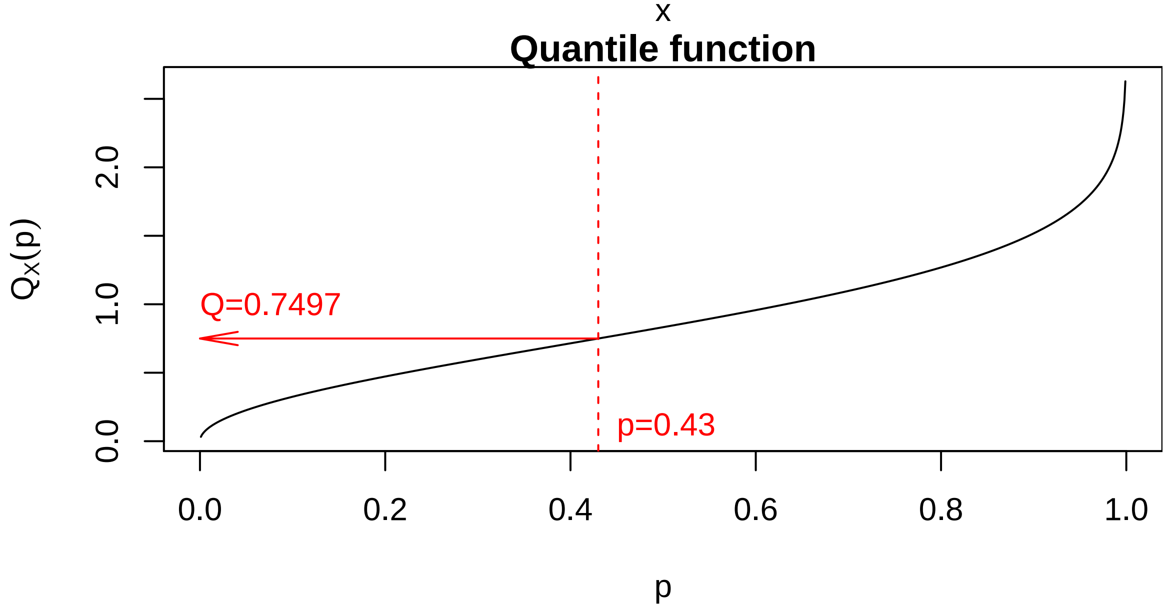
$$Q_X(0.43) = \sqrt{-\log(1-0.43)} = 0.749746.$$

```
p<-seq(0.001,0.999,by=0.001);Qp<-sqrt(-log(1-p))
par(mfrow=c(2,1),mar=c(4,4,1,0))
plot(x,Fx,type='l',main='CDF',ylab=expression(F[X](x)));abline(v=0)
p0<-0.43;(Q0<-sqrt(-log(1-p0)))

+ [1] 0.7497459

abline(h=0.43,lty=2,col='red');arrows(Q0,p0,Q0,0,col='red',angle=10,length=0.2)
text(-3,0.475,"p=0.43",adj=0,col='red');text(0.80,0,"Q=0.7497",adj=0,col='red')
plot(p,Qp,type='l',main='Quantile function',ylab=expression(Q[X](p)))
abline(v=0.43,lty=2,col='red');arrows(p0,Q0,0,Q0,col='red',angle=10,length=0.2)
text(0,1,"Q=0.7497",adj=0,col='red');text(0.45,0.1,"p=0.43",adj=0,col='red')
```



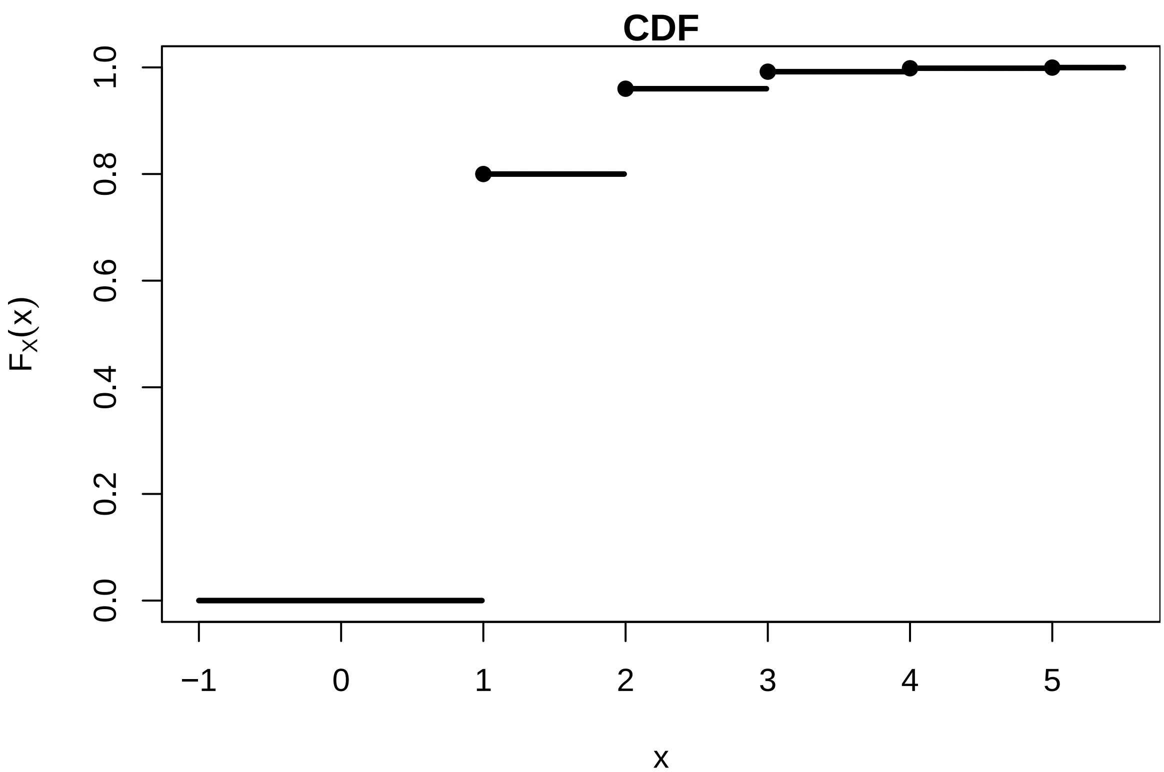


Example: Discrete case
Suppose

$$F_X(x) = \begin{cases} 0 & x < 1 \\ 1 - (1-\theta)^{\lfloor x \rfloor} & x \geq 1 \end{cases}.$$

where $0 < \theta < 1$ is a fixed constant (parameter) and where $\lfloor x \rfloor$ denotes the integer part of x (that is, the largest integer no greater than x). This function is a step-function, with steps at the positive integers $\{1, 2, \ldots\}$. For $\theta = 0.8$ we have the following plot:

```
x<-seq(-1,5.5,by=0.01)
th<-0.8
Fx<-0*(x<1) + ((1-(1-th)^floor(x)))*(x >= 1)
xv<-1:5
Fxv<-1-(1-th)^xv
par(mar=c(4,4,1,0))
plot(x,Fx,pch=19,cex=0.25,main='CDF',ylab=expression(F[X](x)))
points(xv,Fxv,pch=19)
```

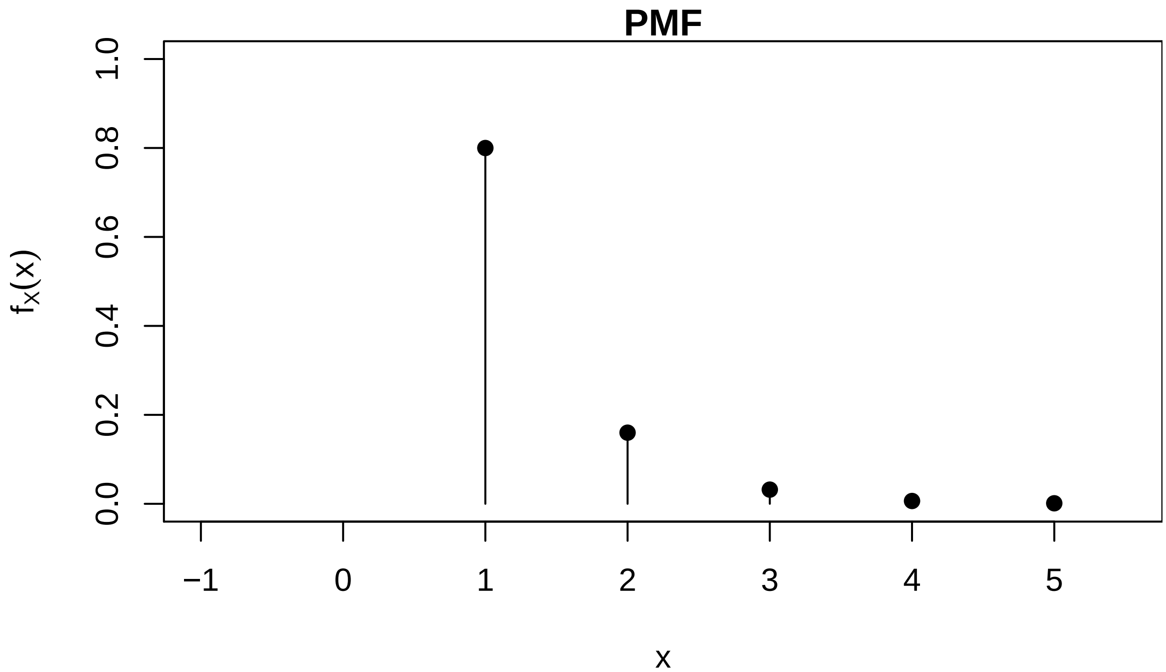


We may deduce that X is discrete, and that for the pmf, $f_X(1) = 1 - (1-\theta) = \theta$, and for $x = 2, 3, \ldots$

$$f_X(x) = F_X(x) - F_X(x-1) = (1-\theta)^{x-1} - (1-\theta)^x = (1-\theta)^{x-1}\theta$$

with $f_X(x) = 0$ for all other x.

```
x<-c(1:5)
th<-0.8
fx<-(1-th)^(xv-1)*th
par(mar=c(4,4,1,0))
plot(x,fx,pch=19,main='PMF',ylab=expression(f[X](x)),xlim=range(-1,5.5),ylim=range(0,1))
for(i in 1:length(x)){lines(c(x[i],x[i]),c(0,fx[i]))}
```



For the quantile function, for $0 < p < 1$

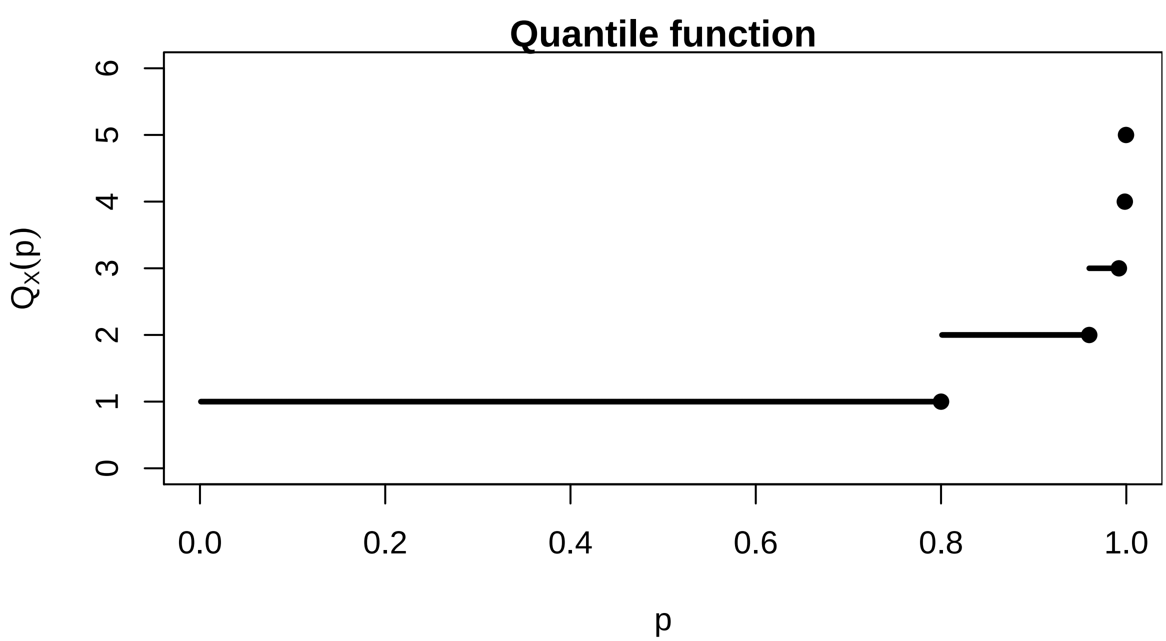
$$Q_X(p) = \inf\{x : p \leq 1 - (1-\theta)^{\lfloor x \rfloor}\} = \inf\{x : \log(1-p) \geq \lfloor x \rfloor \log(1-\theta)\} = \inf\left\{x : \frac{\log(1-p)}{\log(1-\theta)} \leq \lfloor x \rfloor\right\}$$

as $\log(1-\theta) < 0$. It is evident, therefore, that

$$Q_X(p) = \left\lceil \frac{\log(1-p)}{\log(1-\theta)} \right\rceil$$

where $\lceil x \rceil$ denotes the smallest integer greater than or equal to x. Note that this function is *left-continuous* as a function of p, as $F_X(x)$ is *right-continuous* as a function of x.

```
p<-seq(0.001,0.999,by=0.001); Qp<-ceiling(log(1-p)/log(1-th))
par(mar=c(4,4,1,0))
plot(p,Qp,pch=19,cex=0.25,main='Quantile function',ylab=expression(Q[X](p)),ylim=range(0,6))
points(Fxv,xv,pch=19)
```

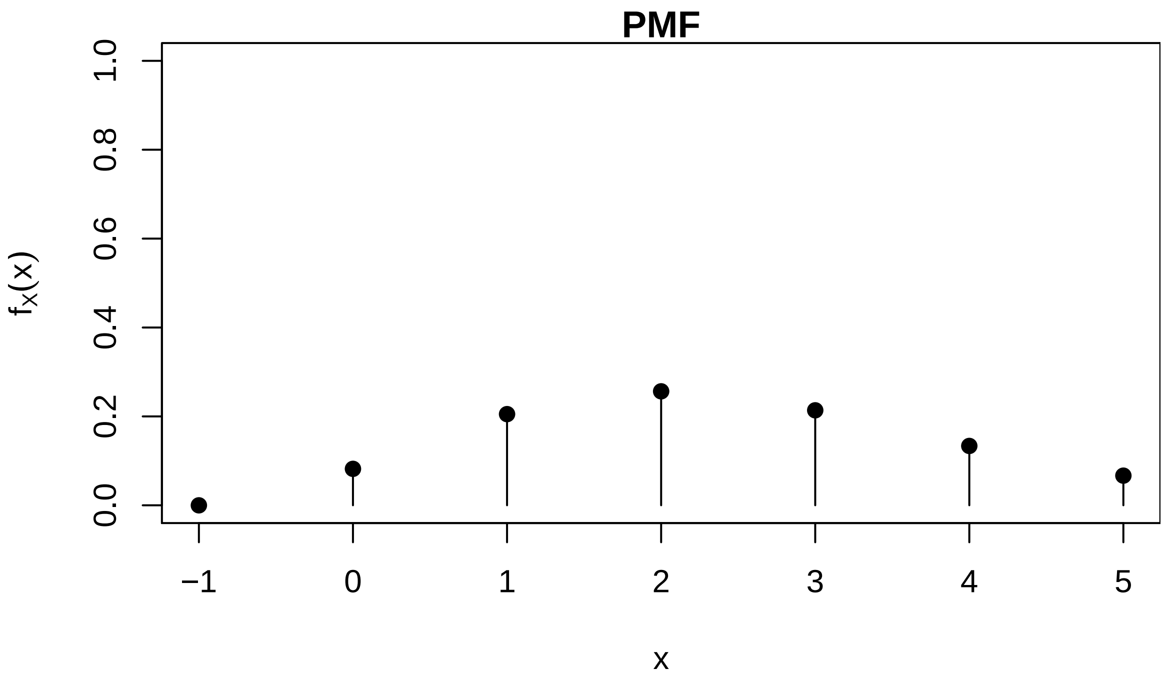


Example: Discrete case
Suppose now that

$$f_X(x) = e^{-\lambda}\frac{\lambda^x}{x!} \qquad x = 0, 1, 2, \ldots$$

and $f_X(x) = 0$ for all other values of x, where $\lambda > 0$ is a fixed constant (parameter). This is the $Poisson(\lambda)$ distribution, and for $\lambda = 2.5$ we have the following plot:

```
x<-seq(-1,5,by=1);fx<-x
lambda<-2.5
fx[x<0]<-0
fx[x>=0]<-exp(-lambda)*lambda^x[x>=0]/factorial(x[x>=0])
par(mar=c(4,4,1,0))
plot(x,fx,pch=19,main='PMF',ylab=expression(f[X](x)),ylim=range(0,1))
for(i in 1:length(x)){lines(c(x[i],x[i]),c(0,fx[i]))}
```



In R, the Poisson pmf is computed by the dpois function:

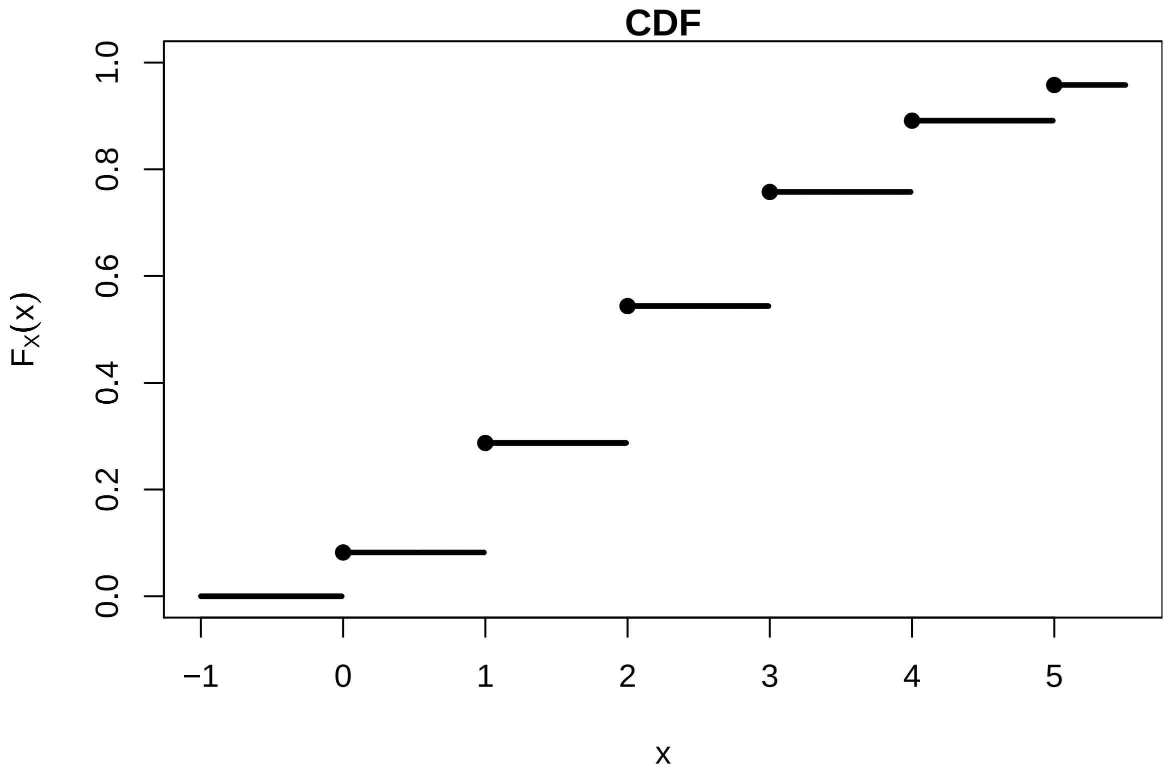
```
rbind(x,fx,dpois(x,lambda))

+    [,1]     [,2]      [,3]      [,4]     [,5]      [,6]       [,7]
+ x    -1 0.000000 1.0000000 2.0000000 3.000000 4.0000000 5.00000000
+ fx    0 0.082085 0.2052125 0.2565156 0.213763 0.1336019 0.06680094
+       0 0.082085 0.2052125 0.2565156 0.213763 0.1336019 0.06680094
```

For the cdf, there is no simple closed form, we may merely write that

$$F_X(x) = \begin{cases} 0 & x < 0 \\ \sum_{t=0}^{\lfloor x \rfloor} e^{-\lambda}\frac{\lambda^t}{t!} & x \geq 0 \end{cases}.$$

```
x<-seq(-1,5.5,by=0.01);fx<-Fx<-x*0                    #Form a continuum of x values
xsub<-x >= 0 & x == floor(x)                           #Identify the integers
fx[xsub]<-exp(-lambda)*lambda^x[xsub]/factorial(x[xsub]) #Compute the pmf at the integers
Fx<-cumsum(fx)                                         #Compute the cdf
par(mar=c(4,4,1,0))
plot(x,Fx,pch=19,cex=0.25,ylim=range(0,1),main='CDF',ylab=expression(F[X](x)))
points(x[xsub],Fx[xsub],pch=19)
```



In R, the Poisson cdf is computed by the ppois function:

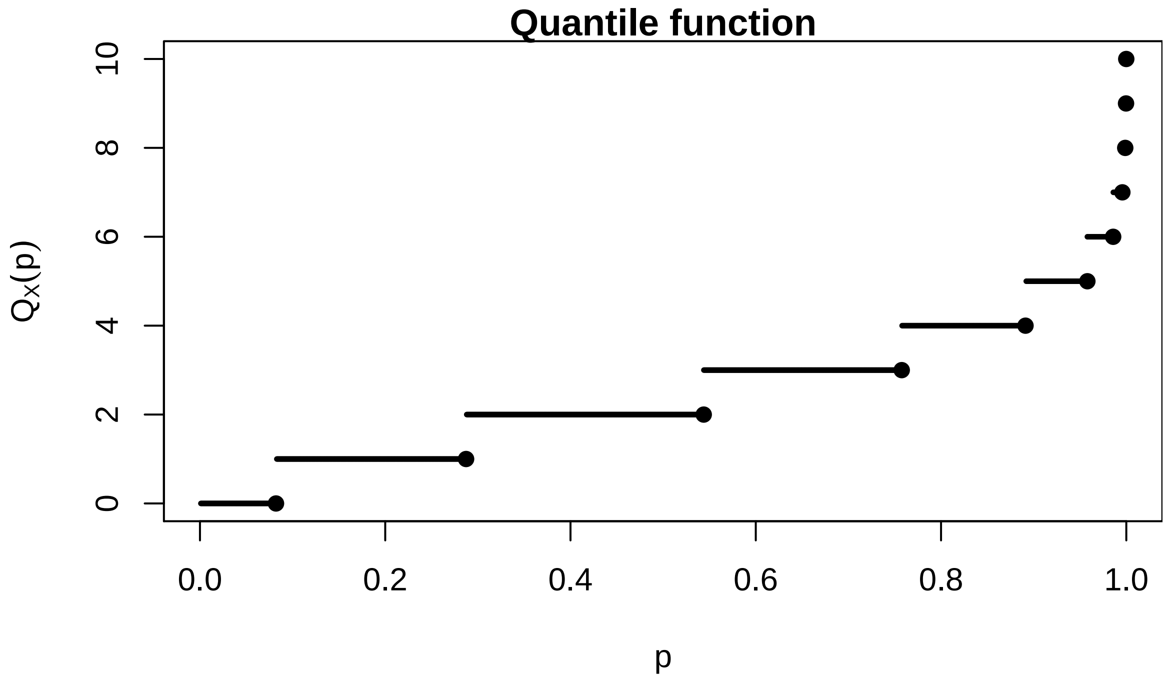
```
rbind(x=x[xsub],Fx=Fx[xsub],ppois_cdf=ppois(0:5,lambda))

+               [,1]      [,2]      [,3]      [,4]     [,5]     [,6]
+ x         0.000000 1.0000000 2.0000000 3.0000000 4.000000 5.000000
+ Fx        0.082085 0.2872975 0.5438131 0.7575761 0.891178 0.957979
+ ppois_cdf 0.082085 0.2872975 0.5438131 0.7575761 0.891178 0.957979
```

For the quantile function, we must again compute numerically: that is, for $0 < p < 1$, we find the smallest (integer) x such that

$$p \leq \sum_{t=0}^{\lfloor x \rfloor} e^{-\lambda} \frac{\lambda^t}{t!}$$

```
p<-seq(0.001,0.999,by=0.001);
Qp<-rep(0,length(p))
x<-seq(-1,15.5,by=0.01);fx<-Fx<-x*0                   #Form a continuum of x values
xsub<-x >= 0 & x == floor(x)                          #Identify the integers
fx[xsub]<-exp(-lambda)*lambda^x[xsub]/factorial(x[xsub])  #Compute the pmf at the integers
Fx<-cumsum(fx)                                        #Compute the cdf
for(i in 1:length(p)){
    if(length(x[p[i]<=Fx]) == 0){
        Qp[i]<-NA
    }else{
        Qp[i]<-min(x[p[i]<=Fx])
    }
}
par(mar=c(4,4,1,0))
plot(p,Qp,pch=19,cex=0.25,main='Quantile function',ylab=expression(Q[X](p)),ylim=range(0,10))
points(Fx[xsub],x[xsub],pch=19)
```



In R, the Poisson quantile function is computed by the qpois function:

```
par(mar=c(4,4,1,0))
plot(p,qpois(p,lambda),pch=19,cex=0.25,ylab=expression(Q[X](p)),ylim=range(0,10))
title('Quantile function computed via qpois')
points(Fx[xsub],x[xsub],pch=19)
```