Lab sheet 4: Probability in R

Prog. Workshop

10-09-2023

Sample space

How do you roll a dice or flip a coin in R?

```
sample(6, size = 10, replace = TRUE)
```

```
##  [1] 4 1 2 4 4 4 3 2 3 4
```

```
sample(7:10, size = 10, replace = TRUE)
```

```
##  [1]  9  9 10  9  8  7  8 10  7  9
```

```
sample(c("H","T"), size = 10, replace = TRUE)
```

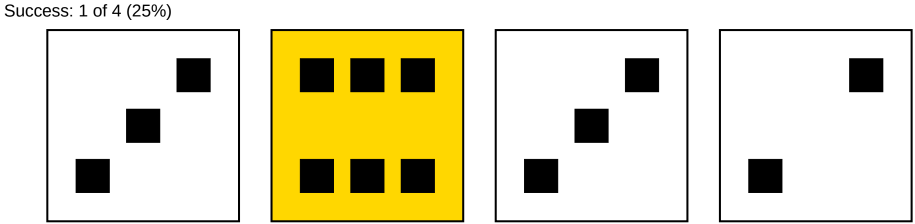
```
##  [1] "H" "T" "H" "H" "H" "T" "H" "H" "T" "T"
```

Other way

```r
library(tidydice)
library(dplyr)
library(explore)
```

continued

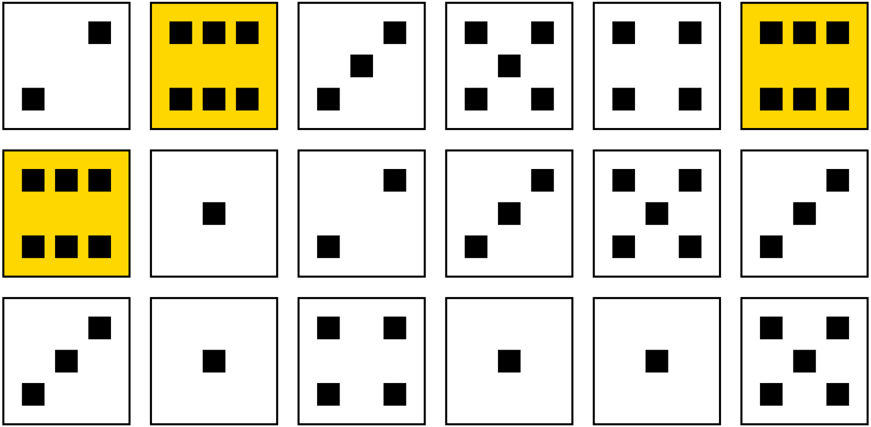
```r
set.seed(123)
roll_dice(times = 4) %>% plot_dice()
```



continued

```r
roll_dice(times = 6, rounds = 3) %>%
  plot_dice()
```

Success: 3 of 18 (16.7%)



Probability distributions

```r
library(distr)
X <- Binom(size = 3, prob = 1/2)
X
```

```
## Distribution Object of Class: Binom
##  size: 3
##  prob: 0.5
```

```r
d(X)(1)   # pmf of X evaluated at x = 1
```

```
## [1] 0.375
```

```r
p(X)(2)   # cdf of X evaluated at x = 2
```

```
## [1] 0.875
```

Mean, VAR, SD

```
library(distrEx)
E(X)
```

```
## [1] 1.5
```

```
var(X)
```

```
## [1] 0.75
```

```
sd(X)
```

```
## [1] 0.8660254
```

```
E(5*X+3)
```

```
## [1] 10.5
```

```
var(5*X+3)
```

```
## [1] 18.75
```

Functions of Random Variables

```
X <- Binom(size = 3, prob = 0.6)
Y <- 3*X+2
Y
```

```
## Distribution Object of Class: AffLinLatticeDistribution
```

```
X <- Norm(mean = 0, sd = 1)
Y <- 3*X+2
Y
```

```
## Distribution Object of Class: Norm
##  mean: 2
##  sd: 3
```

continued

```r
X <- Norm(mean = 0, sd = 1)
Y <- X^2
Y
```

```
## Distribution Object of Class: AbscontDistribution
```

continued

```r
X <- Norm(mean = 0, sd = 1)
Y <- X^2
Y
```

```
## Distribution Object of Class: AbscontDistribution
```

```r
p(Y)(0.5)
```

```
## [1] 0.5204999
```

```r
Z <- Chisq(df = 1)
p(Z)(0.5)
```

```
## [1] 0.5204999
```

The Normal (Gaussian) Distribution

- Probability density function is given by

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\Big(-\frac{(x-\mu)^2}{2\sigma^2}\Big)$$

  where μ is the mean and σ is the standard deviation of the distribution.
- We usually denote the distribution as $N(\mu, \sigma^2)$
- $N(0, 1)$ is the standard normal distribution.

Normal distribution in R

- pdf at Z: `dnorm(Z,mean,sd)`
- cdf at Z: `pnorm(Z,mean,sd)`

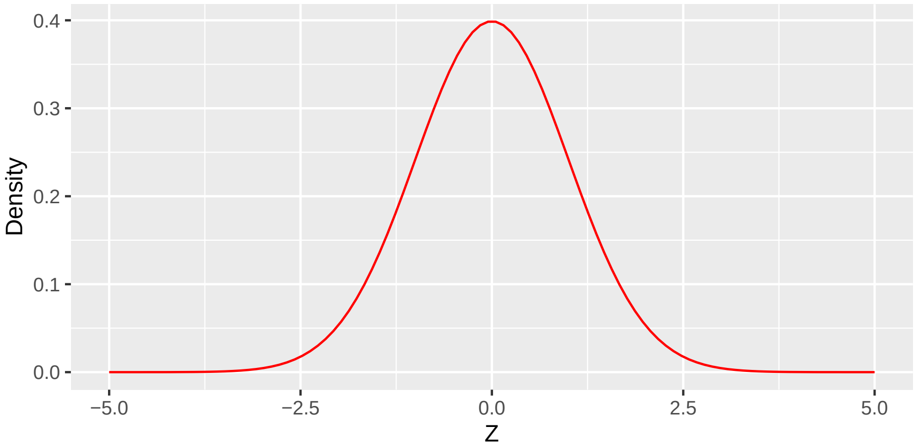
Let us see an example:

```
z <- seq(-5,5,length.out=100)
dstandard <- data.frame(Z=z,
               Density=dnorm(z,mean=0,sd=1)
               , Distribution=pnorm(z,mean=0,sd=1))
head(dstandard)
```

```
##           Z      Density Distribution
## 1 -5.000000 1.486720e-06 2.866516e-07
## 2 -4.898990 2.451061e-06 4.816530e-07
## 3 -4.797980 3.999890e-06 8.013697e-07
## 4 -4.696970 6.461166e-06 1.320248e-06
## 5 -4.595960 1.033101e-05 2.153811e-06
## 6 -4.494949 1.635096e-05 3.479323e-06
```

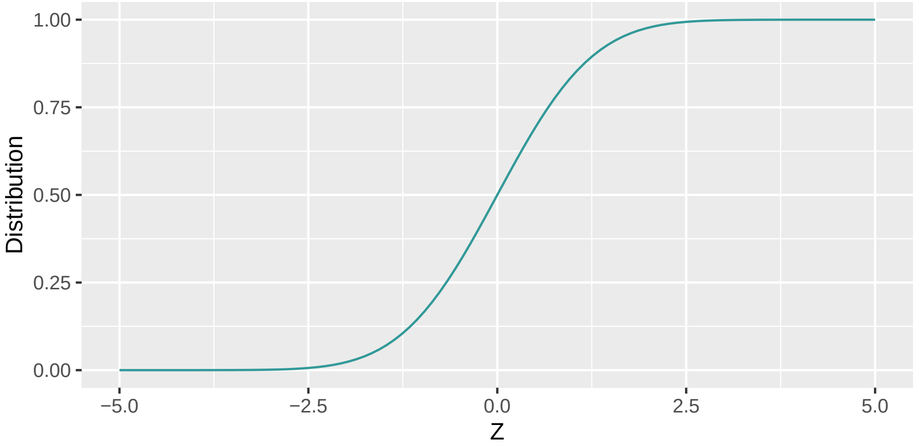
Plots of pdf

```
ggplot(data = dstandard,aes(Z))+
geom_line(aes(y = Density),color="red")
```



Plots of cdf

```
ggplot(data = dstandard,aes(Z))+
geom_line(aes(y = Distribution),color="#339999")
```



Chernoff and Chebychev bounds

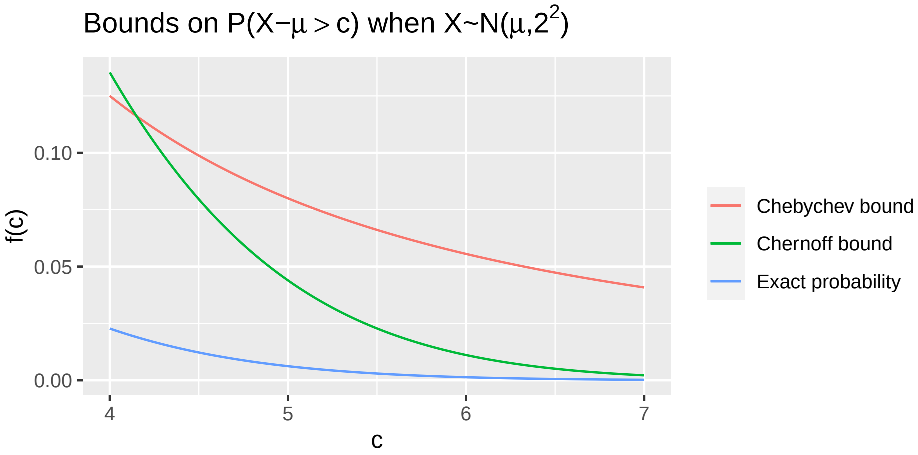
- Chebychev bound on the right tail

$$P(X - \mu \geq c) \; \leq \; \frac{\sigma^2}{2c^2}$$

- Chernoff bound on the right tail

$$P(X - \mu \geq c) \; \leq \; e^{-ct^* + \sigma^2 {t^*}^2/2} \; = \; e^{-c^2/2\sigma^2}$$

Plots



Bounds on $P(X-\mu > c)$ when $X \sim N(\mu, 2^2)$
f(c)
0.10
0.05
0.00
4
5
6
7
c
Chebychev bound
Chernoff bound
Exact probability

Poisson Distribution

Probability density function is given by:

$$P[X = k] = \frac{\lambda^k e^{-\lambda}}{k!},\ k = 0, 1, 2, \ldots$$

Command to use:

- pdf at N: `dpois(N,lambda)`
- cdf at N: `ppois(N,lambda)`

Example in R

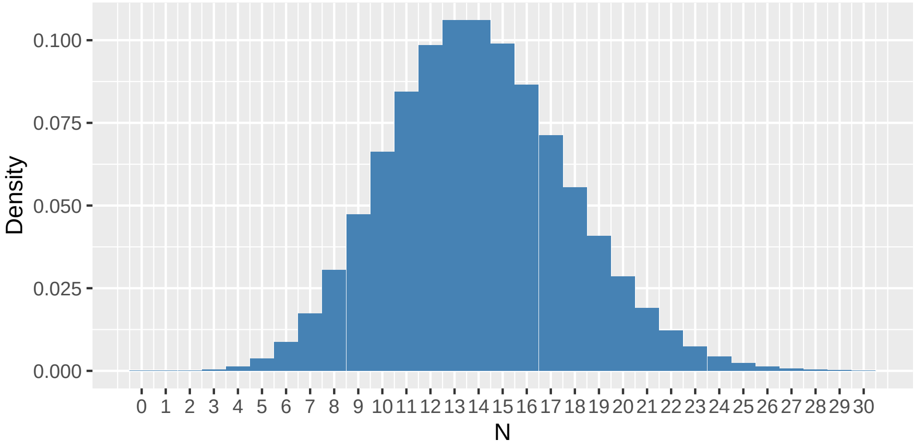
Example:

```
k=seq(from=0,to=30,by=1)
dpoisson <- data.frame(N=k,
            Density=dpois(k, lambda=14),
            Distribution=ppois(k, lambda=14))
head(dpoisson)
```

```
##   N      Density Distribution
## 1 0 8.315287e-07 8.315287e-07
## 2 1 1.164140e-05 1.247293e-05
## 3 2 8.148981e-05 9.396275e-05
## 4 3 3.802858e-04 4.742485e-04
## 5 4 1.331000e-03 1.805249e-03
## 6 5 3.726801e-03 5.532050e-03
```

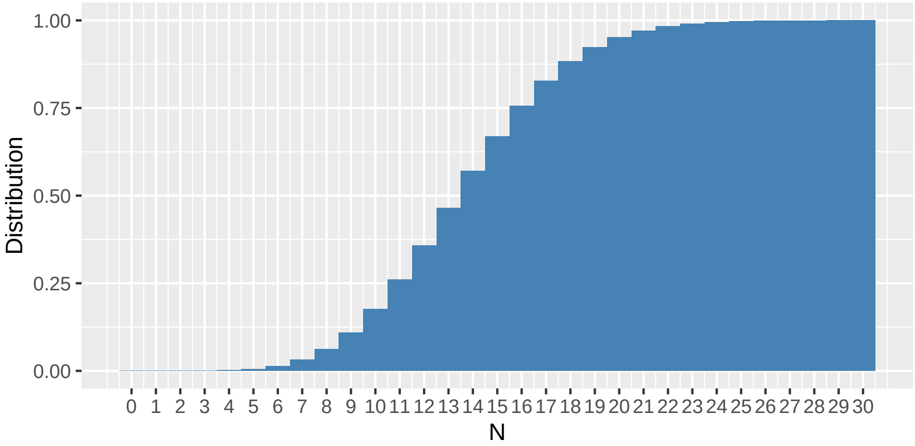
Plot of pdf

```
ggplot(data=dpoisson, aes(x=N, y=Density)) +
    geom_bar(stat="identity",width=0.99,fill="steelblue")+
    scale_x_continuous(breaks=k)
```



Plot of cdf

```
ggplot(data=dpoisson, aes(x=N, y=Distribution)) +
    geom_bar(stat="identity",width=0.99,fill="steelblue")+
    scale_x_continuous(breaks=k)
```



Binomial Distribution

The probability density function of the binomial distribution is given by:

$$P[X = k] = \binom{n}{k} p^k (1-p)^{n-k},\ \ k = 0, 1, 2, \ldots, n$$

Command to use:

- pdf at N: `dbinom(N,size,prob)`
- cdf at N: `pbinom(N,size,prob)`

Example in R

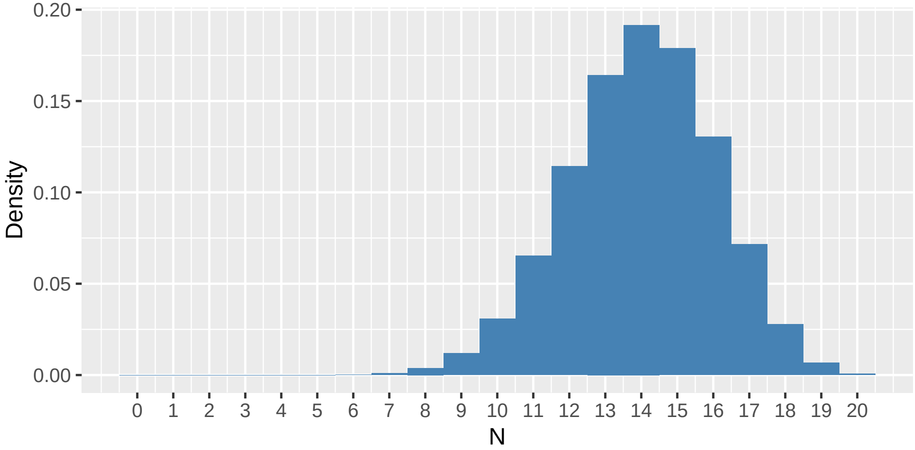
Example:

```
k=seq(from=0,to=20,by=1)
dbinomial <- data.frame(N=k,
            Density=dbinom(k, size = 20, p = 0.7),
            Distribution=pbinom(k, size = 20, p = 0.7))
head(dbinomial)
```

```
##   N      Density Distribution
## 1 0 3.486784e-11 3.486784e-11
## 2 1 1.627166e-09 1.662034e-09
## 3 2 3.606885e-08 3.773088e-08
## 4 3 5.049639e-07 5.426947e-07
## 5 4 5.007558e-06 5.550253e-06
## 6 5 3.738977e-05 4.294002e-05
```

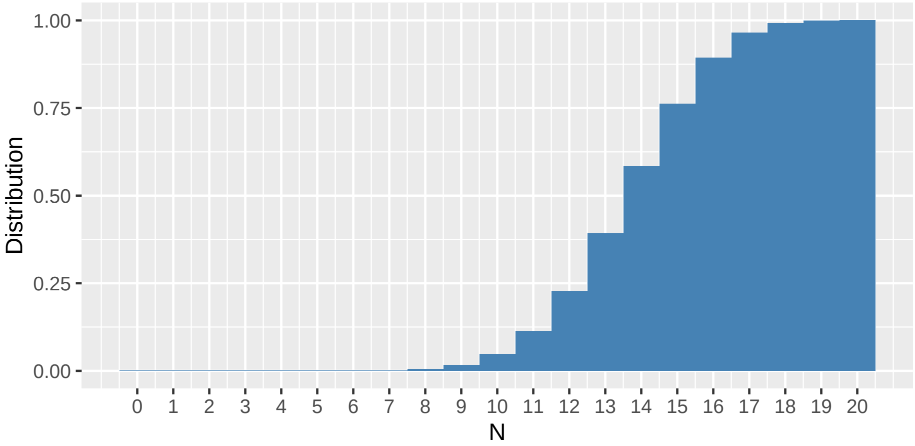
Plot of pmf

```
ggplot(data=dbinomial, aes(x=N, y=Density)) +
    geom_bar(stat="identity",width=0.99,fill="steelblue")+
    scale_x_continuous(breaks=k)
```



Plot of cdf

```
ggplot(data=dbinomial, aes(x=N, y=Distribution)) +
    geom_bar(stat="identity",width=0.99,fill="steelblue")+
    scale_x_continuous(breaks=k)
```



Exponential distribution

The probability density function of the exponential distribution is given by:

$$p(x) = \lambda e^{-\lambda x}, \; x \geq 0$$

Command to use:

- pdf at N: `dexp(N,rate)`
- cdf at N: `pexp(N,rate)`

Example in R

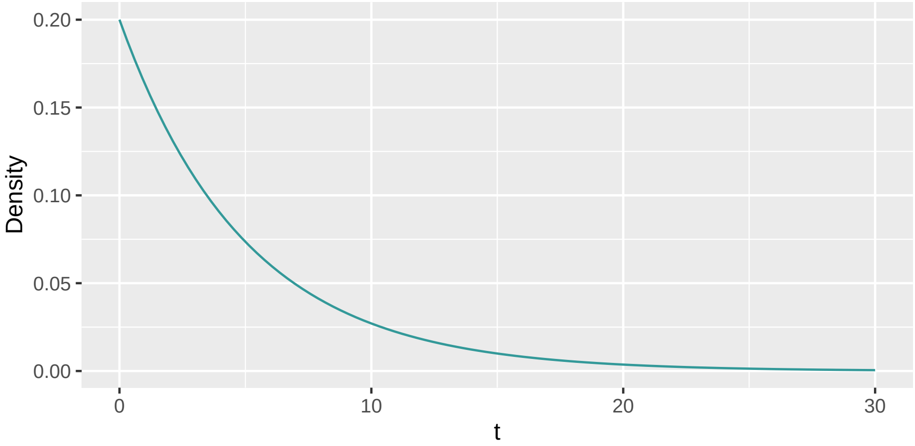
Example:

```
t=seq(from=0,to=30,length.out=100)
dexponential <- data.frame(t=t,
            Density=dexp(t, rate = 0.2),
            Distribution=pexp(t, rate = 0.2))
head(dexponential)
```

```
##           t   Density Distribution
## 1 0.0000000 0.2000000   0.00000000
## 2 0.3030303 0.1882388   0.05880606
## 3 0.6060606 0.1771692   0.11415397
## 4 0.9090909 0.1667506   0.16624708
## 5 1.2121212 0.1569446   0.21527681
## 6 1.5151515 0.1477153   0.26142329
```

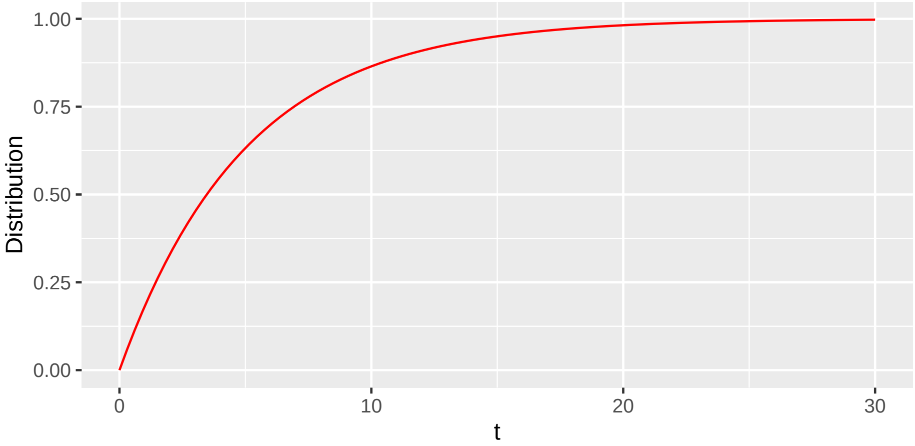
Plots of pdf

```
ggplot(data = dexponential,aes(t))+
geom_line(aes(y = Density),color="#339999")
```



Plots of cdf

```
ggplot(data = dexponential,aes(t))+
geom_line(aes(y = Distribution),color="red")
```



χ^2 Distribution

The probability density function of the χ^2 distribution is given by:

$$p(x) = \begin{cases} \dfrac{x^{\frac{k}{2}-1}e^{-\frac{x}{2}}}{2^{\frac{k}{2}}\Gamma\left(\frac{k}{2}\right)}, & x > 0; \\ 0, & \text{otherwise.} \end{cases}$$

Command to use:

- pdf at x: `dchisq(x,df)`
- cdf at x: `pchisq(x,df)`

Example in R

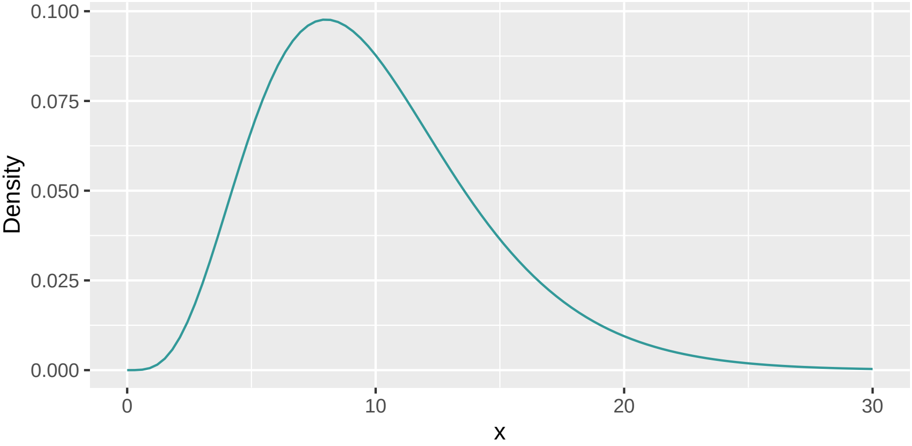
Example:

```
x=seq(from=0,to=30,length.out=100)
dchisquare <- data.frame(x=x,
            Density=dchisq(x, df = 10),
            Distribution=pchisq(x, df = 10))
head(dchisquare)
```

```
##           x      Density Distribution
## 1 0.0000000 0.000000e+00 0.000000e+00
## 2 0.3030303 9.435846e-06 5.866292e-07
## 3 0.6060606 1.297474e-04 1.655698e-05
## 4 0.9090909 5.644968e-04 1.109463e-04
## 5 1.2121212 1.533254e-03 4.127579e-04
## 6 1.5151515 3.217007e-03 1.112635e-03
```

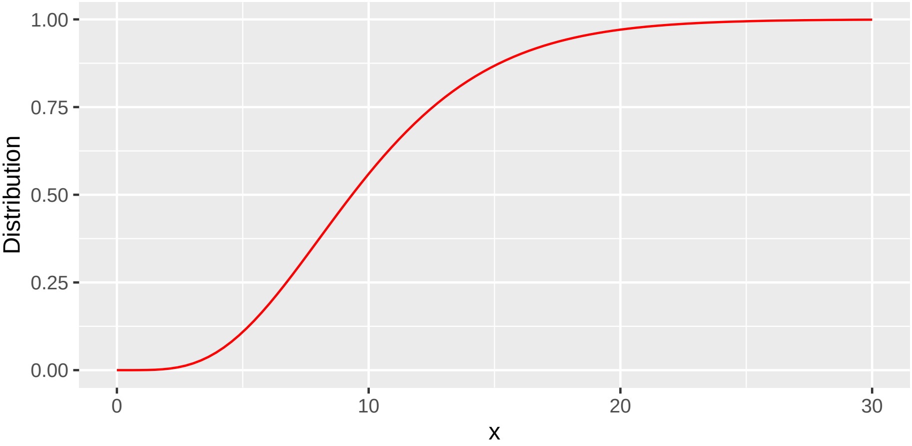
Plots of pdf

```
ggplot(data = dchisquare,aes(x))+
geom_line(aes(y = Density),color="#339999")
```



Plots of cdf

```
ggplot(data = dchisquare,aes(x))+
geom_line(aes(y = Distribution),color="red")
```



t-distrbution

The pdf of t-distribution is given by:

$$p(t) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi}\,\Gamma(\frac{\nu}{2})}\left(1+\frac{t^2}{\nu}\right)^{-\frac{\nu+1}{2}},$$

where ν is the number of degrees of freedom.