Data analysis and visualization using R

Distributions, Sampling and Testing

Michiel Noback

november 2015

Distribution functions

statistic tests

Contents

- distributions and their functions
- sampling from a distribution
- sampling from your own data
- statistical tests

Distribution functions

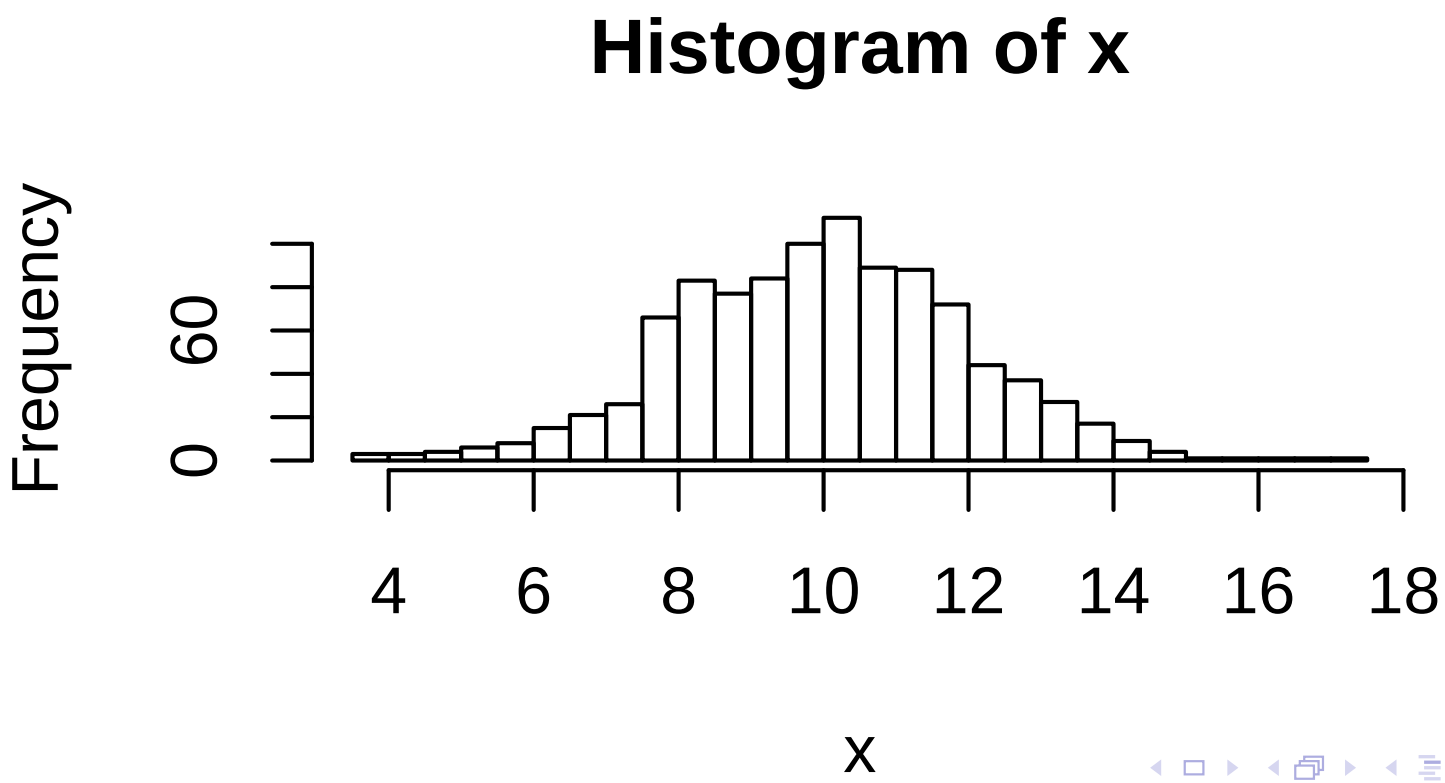
Distribution associated functions

- R provides related functions for several distributions that can be used for
  - sampling: **r**xxxx
  - Probability Density Function (PDF): **d**xxxx
  - Cumulative Distribution Function (CDF): **p**xxxx
  - Quantile Function (inverse of `pxxxx`): **q**xxxx

rnorm()

- Random numbers from a normal distribution with parameters:
  - `n` number of observations
  - `mean` the mean of the distribution
  - `sd` the standard deviation of the distribution

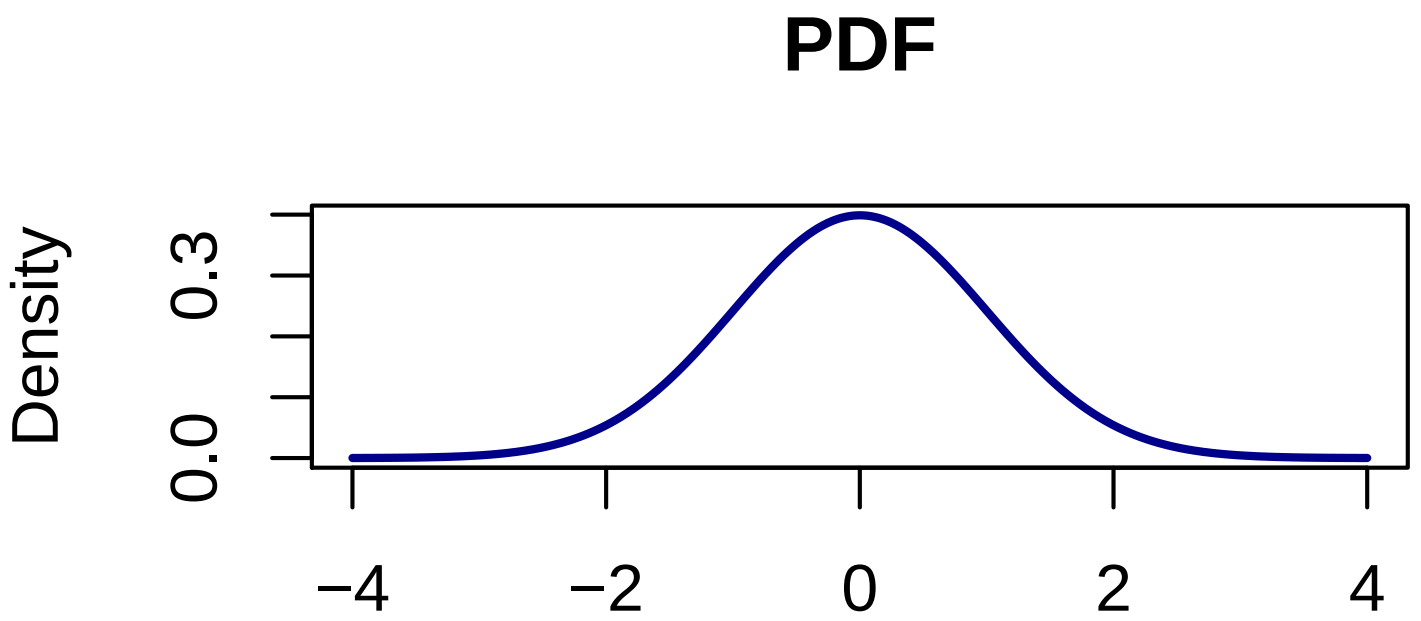
```
x <- rnorm(n=1000, mean=10, sd=2)
hist(x, breaks=20)
```



dnorm()

`dnorm(x, mean, sd)` gives the density (height) of x on the normal ditribution with the given mean and sd.

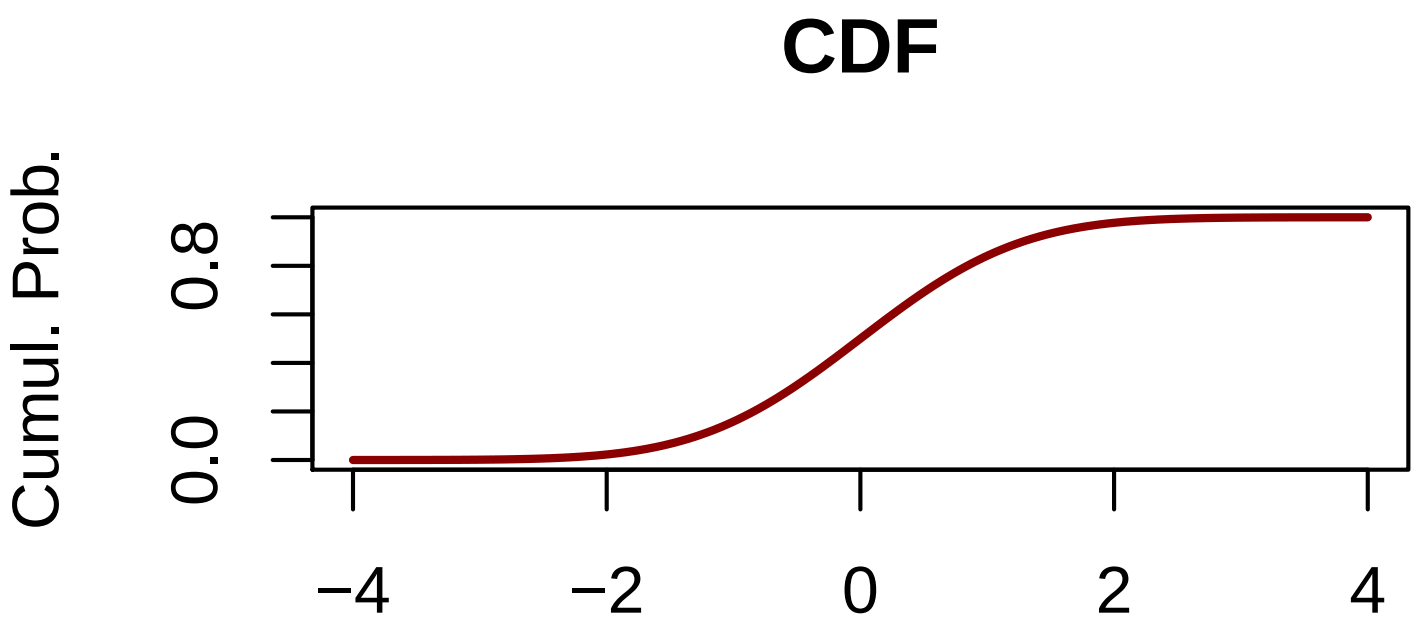
```
xseq <- seq(-4, 4, 0.01)
densities <- dnorm(xseq, 0, 1)
plot(xseq, densities, col="darkblue", xlab="", ylab="Densit
     type="l", lwd=2, main="PDF")
```



pnorm()

`pnorm(q, mean, sd)` gives the area under the standard normal curve to the left of q

```
xseq <- seq(-4, 4, 0.01)
cumulative <- pnorm(xseq, 0, 1)
plot(xseq, cumulative, col="darkred", xlab="", ylab="Cumul
     type="l", lwd=2, main="CDF")
```



other distributions

Similar to the `xnorm()` functions, there are corresponding functions for

- the binomial distribution `xbinom()`
- the poisson distribution `xpois()`
- Chi square `chisq`
- and many others – see `http://www.statmethods.net/advgraphs/probability.html`

Sampling from a set of values: `sample()`

- For some research aspects, shuffling the data (permutation) or taking random samples from a larger set is required.
- The `sample()` function can be used for both.

permutations without replacement

```
x <- 1:10
sample(x)
```

```
##  [1] 10  7  6  9  8  1  4  2  5  3
```

```
sample(x)
```

```
##  [1]  2  8  5  1 10  9  7  4  6  3
```

permutations with replacement

```
sample(x, replace = T)
```

```
##  [1] 1 9 7 4 6 1 7 8 6 3
```

```
sample(x, replace = T)
```

```
##  [1]  1  3  6  3  4  7  7  4 10  2
```

sampling integers from range 1 to X

```
sample.int(1e3, 5, replace = F)
```

```
## [1] 205 407  59 239 604
```

```
sample.int(2, 10, replace = T)
```

```
##  [1] 1 1 1 2 2 2 2 2 1 2
```

sampling a fixed set

```
sample(x, size = 2, replace = T)
```

```
## [1] 4 5
```

```
sample(x, size = 2, replace = T)
```

```
## [1] 6 9
```

```
sample(x, size = 2, replace = T)
```

```
## [1] 4 4
```

sampling with probabilities

```
sample(1:3, size = 10, replace = T, prob = c(0.25, 0.5, 0.2
```

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

```
sample(1:2, size = 10, replace = T, prob = c(0.2, 0.8))
```

```
##  [1] 2 2 2 2 2 1 2 2 2 2
```

reproducible sampling using `set.seed()`

```
set.seed(1234)
sample(x)
```

```
##  [1]  2  6  5  8  9  4  1  7 10  3
```

```
set.seed(1234)
sample(x)
```

```
##  [1]  2  6  5  8  9  4  1  7 10  3
```

statistic tests

overview

test	application	R function
t-toets	diff between 2 means	t.test( )
F-toets	diff between 2 variances	var.test( )
1-way ANOVA	diff between >=2 means	aov( )
chi2-toets	relation between 2 nominal vars	chisq.test( )
z-toets	standaard normaal verdeeld	z.test( ) *

*[in package:TeachingDemos]