Probability Distributions in R

Probability Distributions in R

Working with probability distributions

When we have a probability distribution there are several operations that we can do conditioning on certain parameters values:

- generate random x values
- calculate the density of a certain x value
- calculate the cumulative probability of a certain x value
- calculate the x value associated to a certain cumulative probability

Probability Distributions in R

In R there are several probability distributions (PD) implemented as functions. Basically the corresponding equation of the PD is converted into R code. For example, the Gaussian distribution Probability Density Function (PDF) is represented in Equation 1.

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{1}$$

Gaussian distribution example

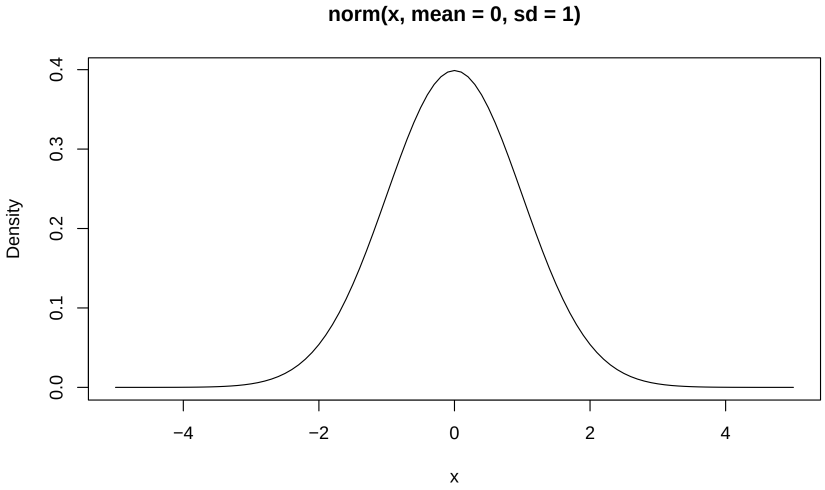
Let's convert the Equation 1 into R code. Our variable is x then we have μ and σ that are the mean and standard deviation of the Gaussian distribution.

```
norm <- function(x, mean = 0, sd = 1){
    1 / sqrt(2 * pi * sd^2) * exp(-((x - mean)^2)/(2 * sd^2
}

norm(0)
## [1] 0.3989423
norm(2)
## [1] 0.05399097
norm(-1)
## [1] 0.2419707
```

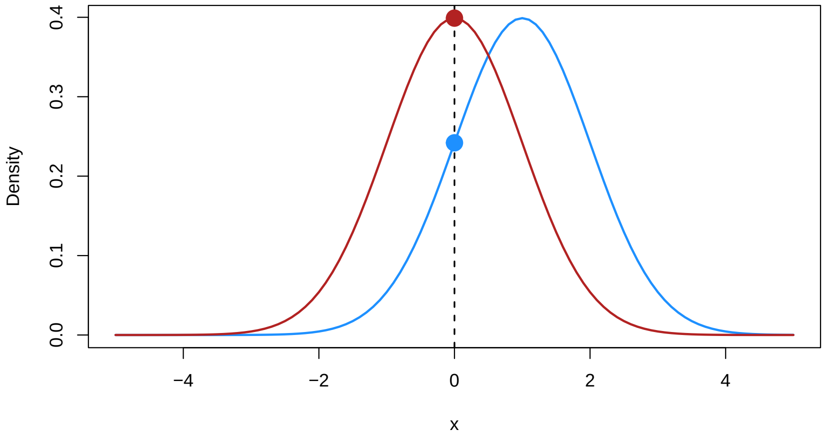
Gaussian distribution example

With the previous code we are calculating the probability density of a certain value given the parameters. Let's use `norm()` for a sequence of values and plot the results.



Gaussian distribution example

Clearly, if we change the parameters, the calculated densities will be different. For example:



Gaussian distribution in R

Fortunately we do not need to write the probabilities distribution manually but a lot of them are already included in R. For example, the `norm()` function can be replaced by `dnorm()`.

```r
norm(0, 1, 2)
```

```
[1] 0.1760327
```

```r
dnorm(0, 1, 2)
```

```
[1] 0.1760327
```

`d`, `q`, `r` and `p` functions

`d`, `q`, `r` and `p` functions

Actually in R there are already implemented a lot of probability distributions. This document https://cran.r-project.org/web/views/Distributions.html provides a very comprehensive overview.

The general idea is always the same, regardless the distribution:

- generate random x values **there is the `r` function**
- calculate the density of a certain x value **there is the `d` function**
- calculate the cumulative probability of a certain x value **there is the `p` function**
- calculate the x value associated to a certain cumulative probability **there is the `q` function**

`d`, `q`, `r` and `p` functions

The combination is `d`, `p`, `q` or `r` + the function contaning the equations of that specific distribution. Thus we can use `dnorm()`, `pnorm()`, `qnorm()` and `rnorm()`.

Maximum Likelihood

The `d` function provides the probability density (or likelihood) of a certain value(s) fixing the parameters. What about fixing the value(s) and changing the parameters?

Let's assume we have $n = 10$ values from a Normal distribution with unknown parameters:

```
 [1]  8.99 14.66 11.12 11.11 -1.10 15.23 14.33  4.63 11.26
```

We can calculate the mean and standard deviation:

```r
mean(x)
## [1] 10
sd(x)
## [1] 5
```

Maximum Likelihood

Now, we can calculate the likelihood of the 10 values. Which values should we used for the parameters? We can try different values for μ and σ:

```
dnorm(x, 0, 1)
##  [1] 1.092433e-18 8.572607e-48 5.579440e-28 6.565480e-28
##  [6] 1.822419e-51 1.074744e-45 8.977072e-06 1.127428e-28
dnorm(x, 10, 5)
##  [1] 0.078187324 0.051680597 0.077809381 0.077860138 0.0
##  [7] 0.054871529 0.044784961 0.077281059 0.079710824
dnorm(x, -5, 2)
##  [1] 4.676681e-12 2.076632e-22 1.556279e-15 1.650843e-15
##  [6] 1.236046e-23 1.054073e-21 1.858781e-06 8.727618e-16
```

Maximum Likelihood

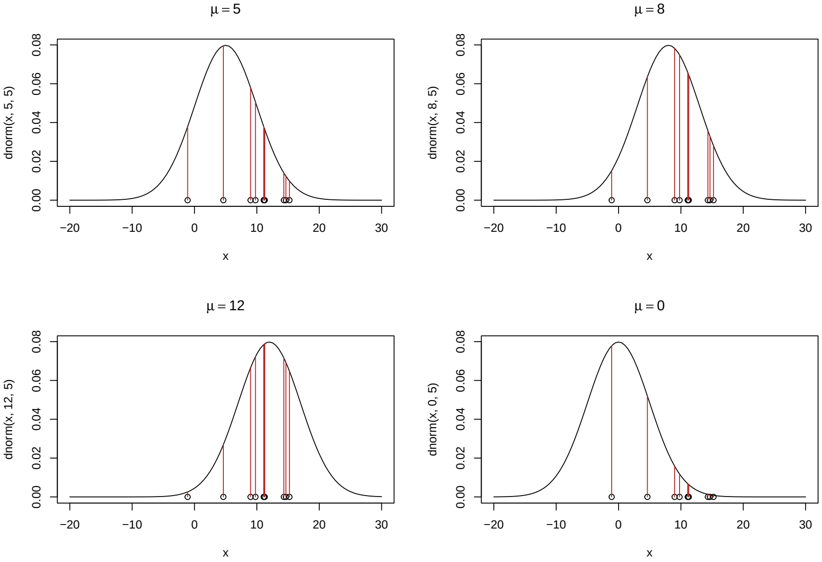
We can take the product (or the sum of the log-transformed values):

```
prod(dnorm(x, 0, 1))
## [1] 1.008627e-270
prod(dnorm(x, 10, 5))
## [1] 1.16165e-13
prod(dnorm(x, -5, 2))
## [1] 4.354571e-142
```

Maximum Likelihood

What about varying a parameter, e.g., μ? We can fix the σ to a certain value, for example 5.

Maximum Likelihood



Maximum Likelihood

There is a point where the likelihood is maximised. For example, for a linear model like in this case (just estimating the mean) the Maximum Likelihood Estimator (MLE) is just the sample mean.