Stichprobenverteilug einer Normalverteilung und Intervalschätzung für den Mittelwert

Grundgesamtheit generieren und darstellen

Eine Grundgesamtheit N(μ, σ^2) (Universum) als size_universe Zufallszahlen generieren. Der Mittelwert ist mu, die Standardabeichung ist sigma.

```
size_universe=10000
mu <- 10
sigma <- 2
universe = rnorm(size_universe, mean=mu, sd=sigma)
cat("First 5 random numbers", head(universe, 5), "\n" )
```

```
## First 5 random numbers 9.559195 13.34651 9.138365 11.62766 10.59408
```

Durchschnitt und Standardabweichung aus den Daten "schätzen". Verteilung graphisch darstellen:

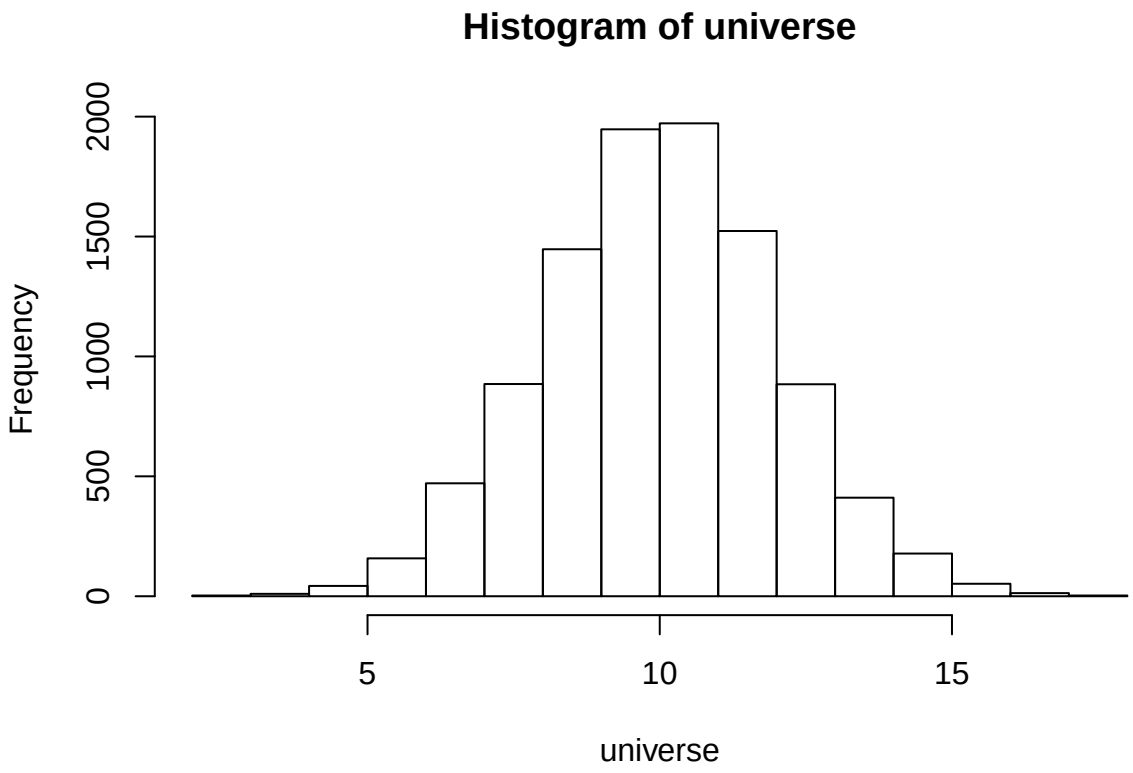
```
cat("Mean of the universe:", mean(universe), "\n")
```

```
## Mean of the universe: 10.00901
```

```
cat("Standard Deviation of the universe:", sd(universe), "\n")
```

```
## Standard Deviation of the universe: 1.996457
```

```
hist(universe)
```



Eine Stichprobe ziehen

Jetzt ziehen wir eine kleine Stichprobe:

```
sample_size <- 4 # Sample size
my_sample <- sample(universe, sample_size)
head(my_sample) # Print the first elements of the sample on the screen
```

```
## [1]  8.928247 10.599980  8.126041 10.602863
```

Der Mittelwert und Standardabweichung der Stichprobe sind "nah" zu mu und sigma aber nicht genau gleich:

```
cat("Mean of the sample:", mean(my_sample), "\n")
```

```
## Mean of the sample: 9.564283
```

```
cat("Standard Deviation of the sample:", sd(my_sample), "\n")
```

```
## Standard Deviation of the sample: 1.241558
```

Stichprobeverteilung

Wir können jetzt die Stichprobeziehung mehrmals durchführen. Der Mittelwert der sample_size Zufallsvariabeln ist auch eine Zufallsvariabel. Schauen wir mal, wie diese verteilt ist.:

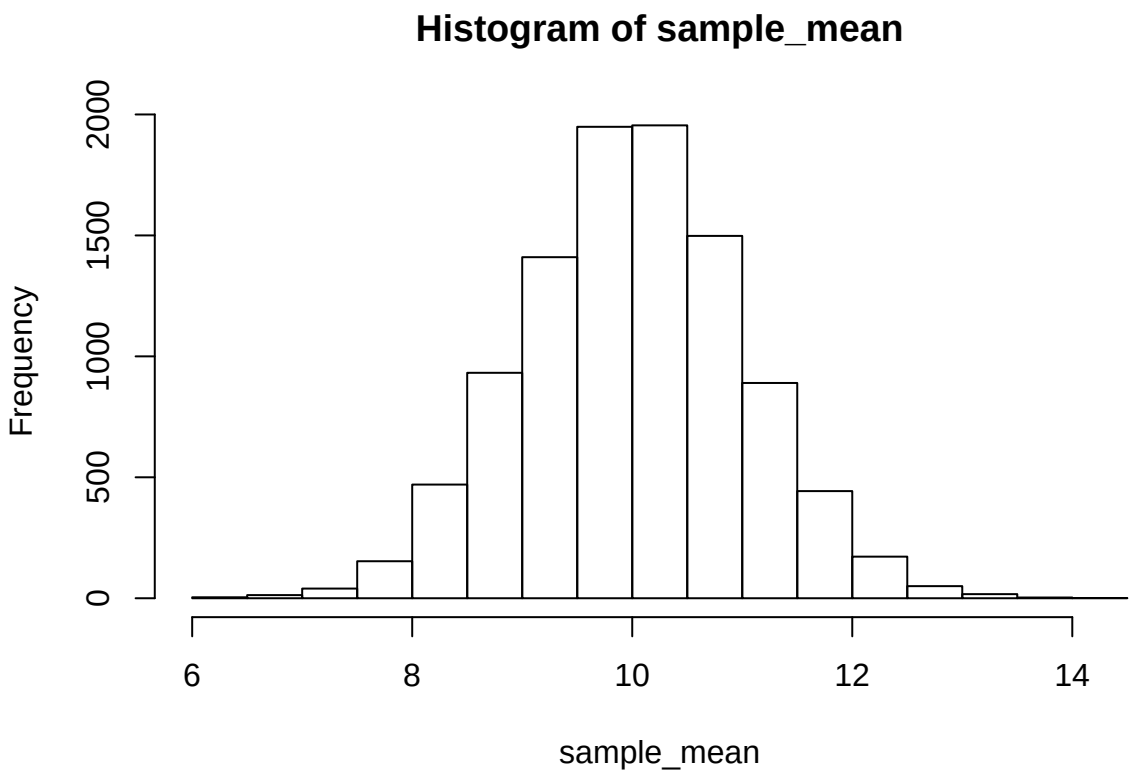
```
n_iterations <-  10000 # Number of times that we repeat
sample_mean <-  NULL
for (i in 1:n_iterations) {
    my_sample <- sample(universe, sample_size)
    x_bar <- mean(my_sample)
    sample_mean <- c(sample_mean, x_bar)
}
cat("Mean of the sampling distribution:", mean(sample_mean), "\n")
```

```
## Mean of the sampling distribution: 10.00276
```

```
cat("Standard Deviation of the sampling distribution:", sd(sample_mean), "\n")
```

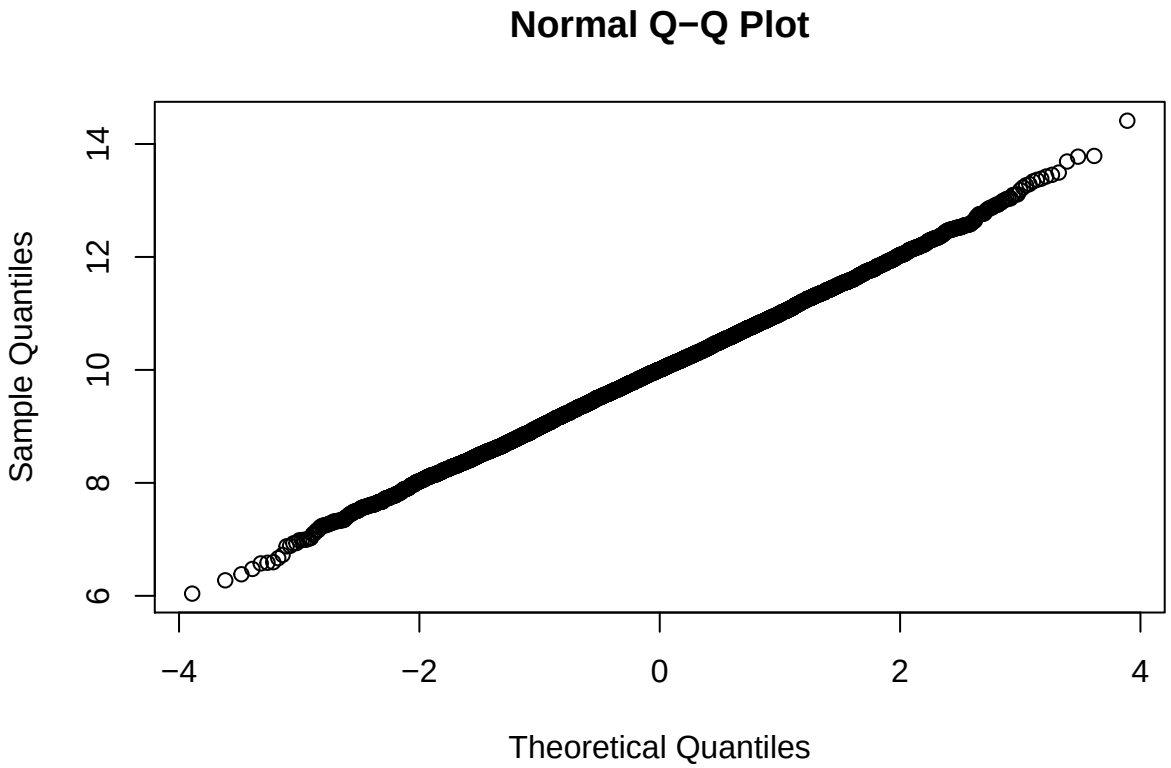
```
## Standard Deviation of the sampling distribution: 1.006511
```

```
hist(sample_mean)
```



Um zu schauen, ob dieser normalverteilt ist, können wir die qqnorm Funktion verwenden. Fallen die dargestellent Punkte entlang einer Gerade, so ist die Verteilung normal.

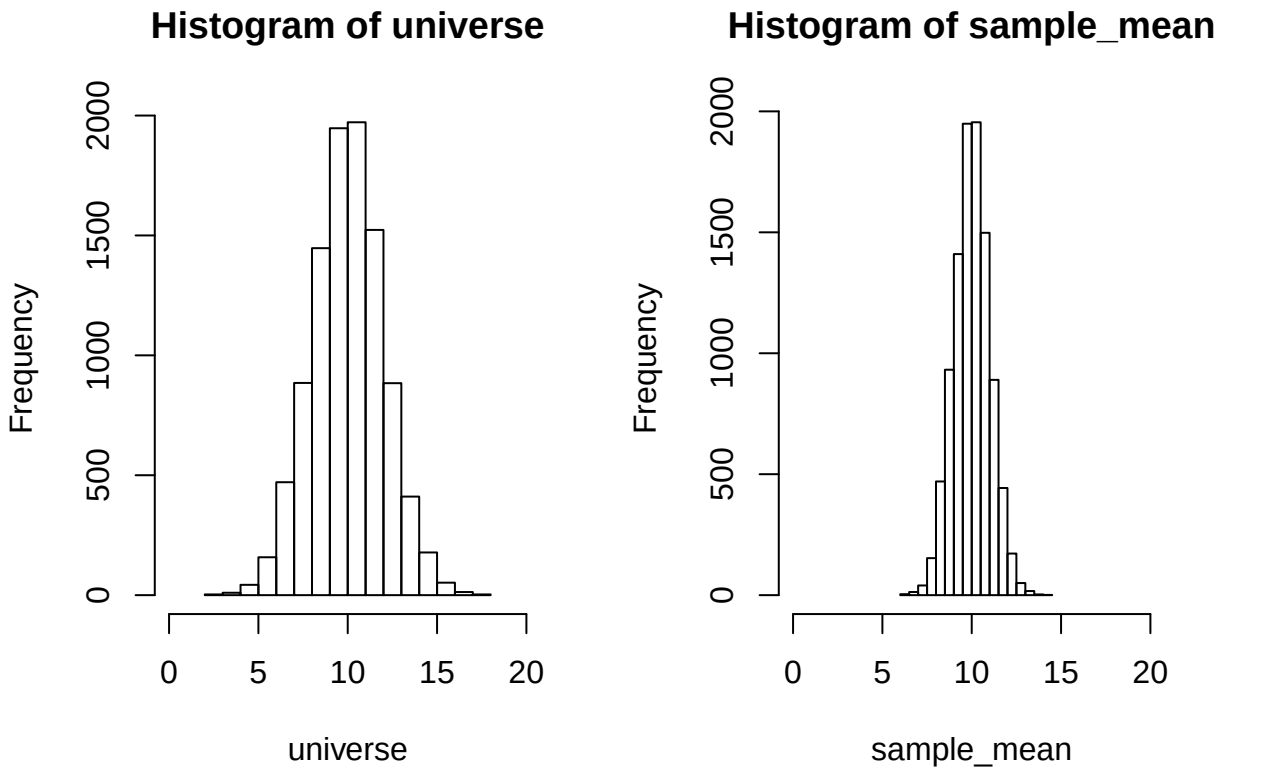
```
qqnorm(sample_mean)
```



Schlussfolgerung (was nutzt uns das Ganze)

Stellen wir das Universum und die Stichprobenverteilung nebeneinander dar:

```
par(mfrow = c(1, 2))
hist(universe, freq = TRUE, xlim = c(mu - 5 * sigma, mu + 5 * sigma))
hist(sample_mean, freq = TRUE, xlim = c(mu - 5 * sigma, mu + 5 * sigma))
```




```
par(mfrow = c(1, 1))
```

Gemäss CLT (Zentralergrenzwertsatz) erwarten wir, dass der (Mittelwert der Stichprobe) $\bar{X}$ wie $\mathrm{N}(\mu, \frac{\sigma^2}{\sqrt{n}})$ verteilt ist. Ist es so?

```
cat("Mean of the universe:", mean(universe), "\n")
```

```
## Mean of the universe: 10.00901
```

```
cat("Mean of the sampling distribution:", mean(sample_mean), "\n")
```

```
## Mean of the sampling distribution: 10.00276
```

Diese Werte sind praktisch gleich. Das ist gut.

```
cat("Standard Deviation of the sampling distribution:", sd(sample_mean), "\n")
```

```
## Standard Deviation of the sampling distribution: 1.006511
```

```r
cat("Theoretical value of its standard deviation:", sigma / sqrt(sample_size), "\n")
```

```
## Theoretical value of its standard deviation: 1
```

Auch diese Werte sind praktisch gleich. Das ist gut. Die Simulation steht im Einklang zur Theorie.

Anders gezeigt. Die Verteilung $\frac{\bar{X}-\mu}{\frac{\sigma}{\sqrt{n}}}$ ist N(mu, 1) verteilt.

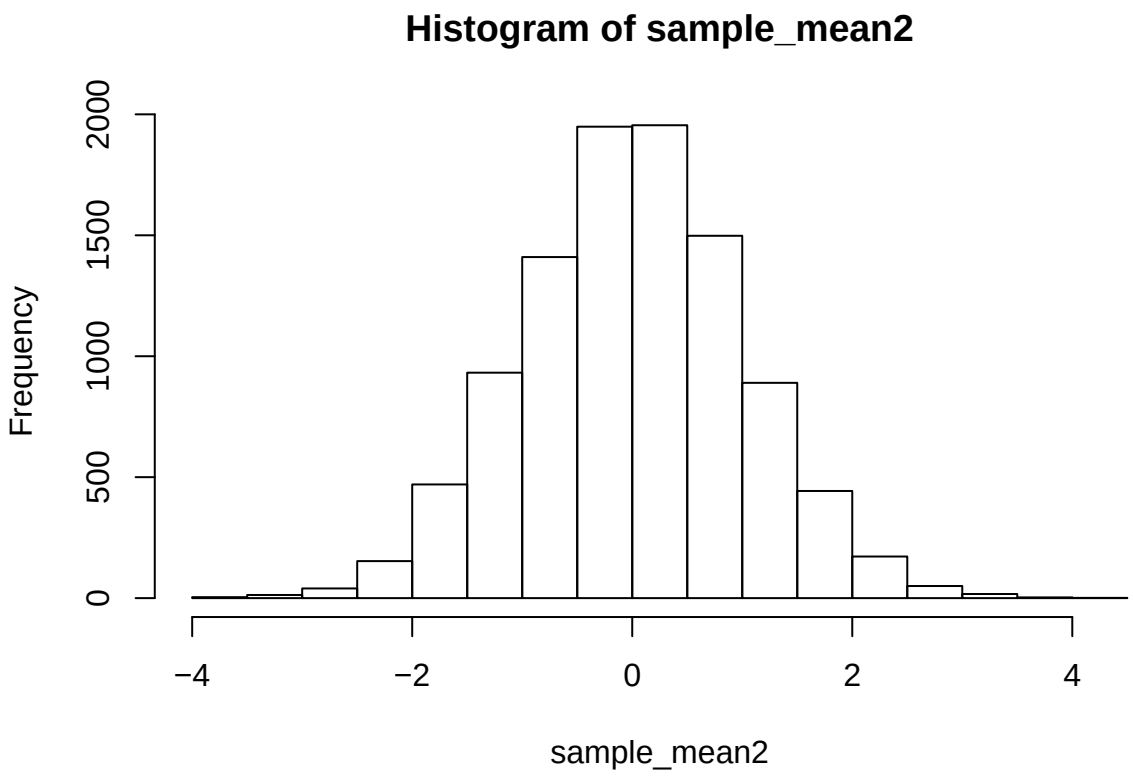
```r
sample_mean2 <- (sample_mean - mu) / (sigma / sqrt(sample_size) )
cat("Mean of sample_mean2:", mean(sample_mean2), "\n")
```

```
## Mean of sample_mean2: 0.002760773
```

```r
cat("SD of sample_mean2:", sd(sample_mean2), "\n")
```

```
## SD of sample_mean2: 1.006511
```

```r
hist(sample_mean2)
```



wie aus der Formel erwartet.

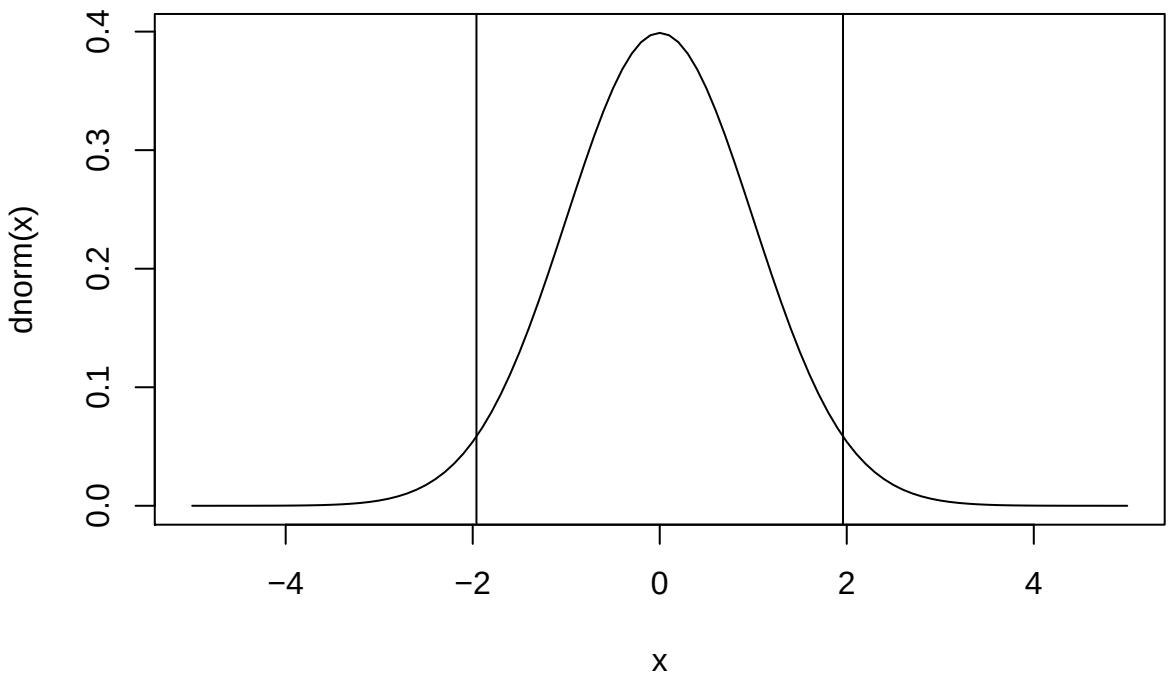
Konfidenzinterval wenn σ bekannt ist

Wir sind jetzt in der Lage, die Schätzung vom μ zu quantifizieren. Setzen wir ein Konfidenzniveau 1 - α für die Schätzung. α ist die Irrtumswahrscheinlichkeit. Angenommen die Standardabweichung der Grundgesamtheit σ ist bekannt, verwenden wir folgende Formel: Das Konfidenzinterval berechnen wir dann als:

$$\mu = \bar{X} \pm Z_{\frac{\alpha}{2}} \frac{\sigma}{\sqrt{n}}$$

Der Wert Z_α berechnet sich als qnorm(1 - alpha / 2) mir R.

```
alpha = 0.05
curve(dnorm(x), from = -5, to = 5)
abline(v = qnorm(alpha / 2)); abline(v = qnorm(1 -alpha / 2))
```



Statt immer wieder Zahlen in die Formel einzustecken (fehleranfällig), können wir folgende Funktion verwenden:

```
# Define a function for the confidence interval for estimating the mean of a normal distribution
# when we have a sample of n data points and we know the mean sigma of the universe
conf.int.z <- function(xbar, sigma, n, alpha) {

    d <- qnorm(1 - alpha / 2) * sigma / sqrt(n)

    cat("With", 1-alpha, "certainty the mean mu lies between", xbar, "+/-", d,  "\n")
    cat("With", 1-alpha, "certainty the mean mu lies between", xbar -d, "and", xbar + d, "\n")

    return(d)

}


# Test the function. Example 10.4, Dürr, Meyer.
conf.int.z(xbar = 80.50, sigma = 2.2, n = 40, alpha = 0.05)
```

```
## With 0.95 certainty the mean mu lies between 80.5 +/- 0.6817745
## With 0.95 certainty the mean mu lies between 79.81823 and 81.18177

## [1] 0.6817745
```

Realistischer Fall: Konfidenzinterval wenn σ unbekannt ist

Betrachten wir nochmals:

$$\frac{\bar{X} - \mu}{\frac{\sigma}{\sqrt{n}}}$$

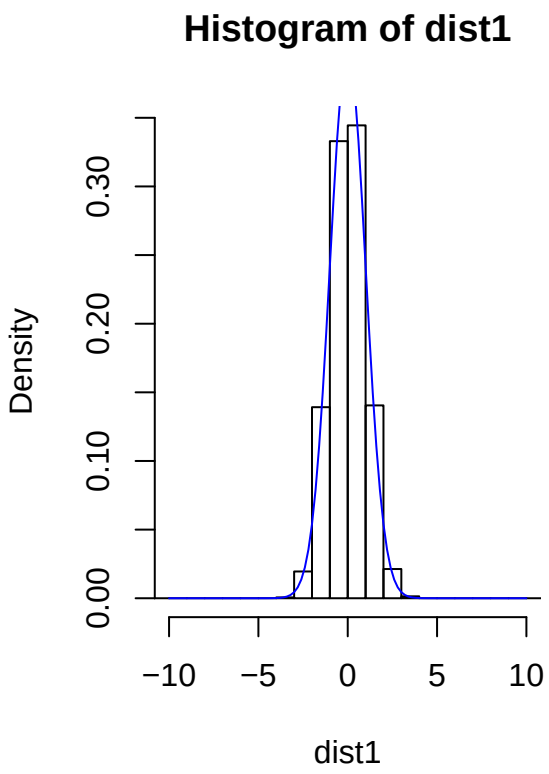
Wie schon erwähnt diese Verteilung ist N(mu, 1) verteilt. Das ist gut, vorausgesetzt wir haben σ. Wenn nicht, können wir die berechnete Standardabweichung S aus den Daten verwenden. Wir hoffen, dass

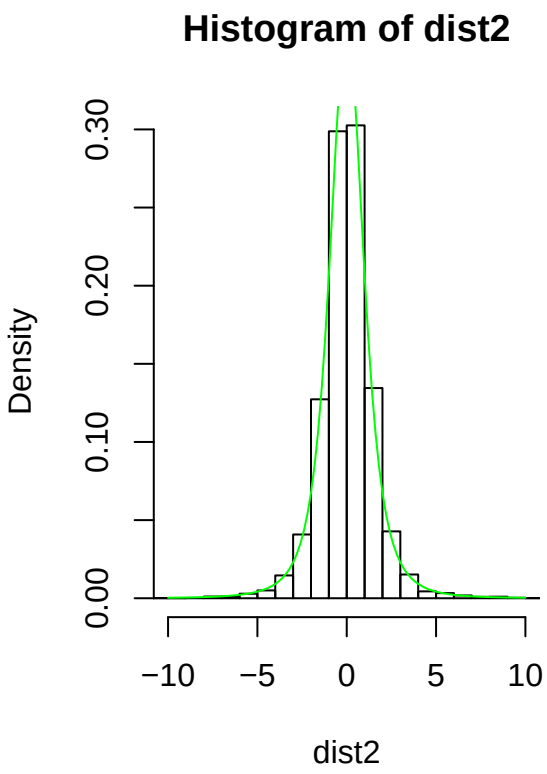
$$\frac{\bar{X} - \mu}{\frac{S}{\sqrt{n}}}$$

irgendwelche bekannte Verteilung folgt.Zuerst überlegen wir uns. Weil S nicht ganz gleich σ ist, sondern eine Schätzung mit Fehler, erwarten wir, dass unsere Schätzung von μ jetzt unpräziser wird.

Zuerst die Simulation:

```
n_iterations <-  10000 # Number of times that we repeat
dist1 <-  NULL
dist2 <- NULL
for (i in 1:n_iterations) {
    my_sample <- sample(universe, sample_size)
    x_bar <- mean(my_sample)
    S <- sd(my_sample)
    d1 <- (x_bar - mu) / (sigma / sqrt(sample_size) )
    dist1 <- c(dist1, d1)
    d2 <- (x_bar - mu) / (S / sqrt(sample_size) )
    dist2 <- c(dist2, d2)
}
#cat("Mean of the sampling distribution:", mean(sample_mean), "\n")
#cat("Standard Deviation of the sampling distribution:", sd(sample_mean), "\n")
par(mfrow = c(1, 2))
hist(dist1, freq = F, xlim = c(-10, 10), breaks = seq(-100, 100))
curve(dnorm, add = T, col = "blue")
hist(dist2, freq = F, xlim = c(-10, 10), breaks = seq(-100, 100))
curve(dt(x, df = sample_size -1), col = "green", add = T)
```






```
par(mfrow = c(1, 1))
sd(dist1)
```

```
## [1] 0.9961402
```

```
sd(dist2)
```

```
## [1] 1.605731
```

Die rechte Verteilung ist die t-Verteilung (auch bekannt als Studentverteilung). Für eine niedrige Anzahl Freiheitsgraden ist die Studentverteilung breiter als die Normalverteilung. Sobald die Anzahl Freiheitsgraden (sprich die Stichprobegrösse) steigt, nähert sich die Studentverteilung die Standardnormalverteilung an.

Hier ist eine schöne Grafik, die ich heruntergeladen habe:

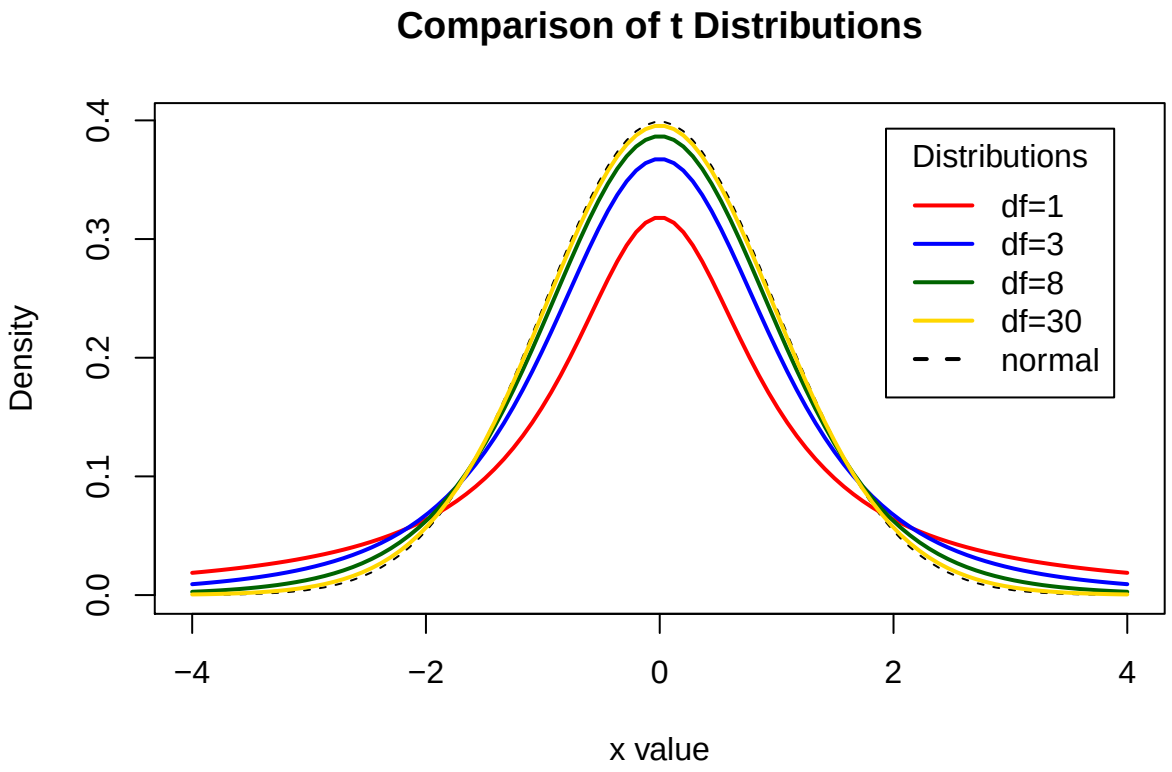
```
x <- seq(-4, 4, length=100)
hx <- dnorm(x)

degf <- c(1, 3, 8, 30)
colors <- c("red", "blue", "darkgreen", "gold", "black")
labels <- c("df=1", "df=3", "df=8", "df=30", "normal")

plot(x, hx, type="l", lty=2, xlab="x value",
     ylab="Density", main="Comparison of t Distributions")

for (i in 1:4){
  lines(x, dt(x,degf[i]), lwd=2, col=colors[i])
}
```

```r
legend("topright", inset=.05, title="Distributions",
       labels, lwd=2, lty=c(1, 1, 1, 1, 2), col=colors)
```



und das Konfidenzinterval wird folgendermassen berechnet:

$$\mu = \bar{X} \pm t_{\frac{\alpha}{2},df} \frac{S}{\sqrt{n}}$$

wobei df = n -1

Und hier die Funktion, die es berechnet.

```r
conf.int.t <- function(xbar, S, n, alpha) {


    d <- qt(1 - alpha / 2, n - 1) * S / sqrt(n)

    cat("With", 1-alpha, "certainty the mean mu lies between", xbar, "+/-", d,  "\n")
    cat("With", 1-alpha, "certainty the mean mu lies between", xbar -d, "and", xbar + d, "\n")

    return(d)

}

# Test the function
conf.int.t(xbar = 10, S = 2, n = 4, alpha = 0.05)
```

```
## With 0.95 certainty the mean mu lies between 10 +/- 3.182446
## With 0.95 certainty the mean mu lies between 6.817554 and 13.18245
```

```
## [1] 3.182446
```

Vergleichen wir zum Konfidenzinterval, wenn wir statt S = 2, einen bekannten $\sigma = 2$ haben.

```
conf.int.z(xbar = 10, sigma = 2, n = 4, alpha = 0.05)
```

```
## With 0.95 certainty the mean mu lies between 10 +/- 1.959964
## With 0.95 certainty the mean mu lies between 8.040036 and 11.95996
```

```
## [1] 1.959964
```