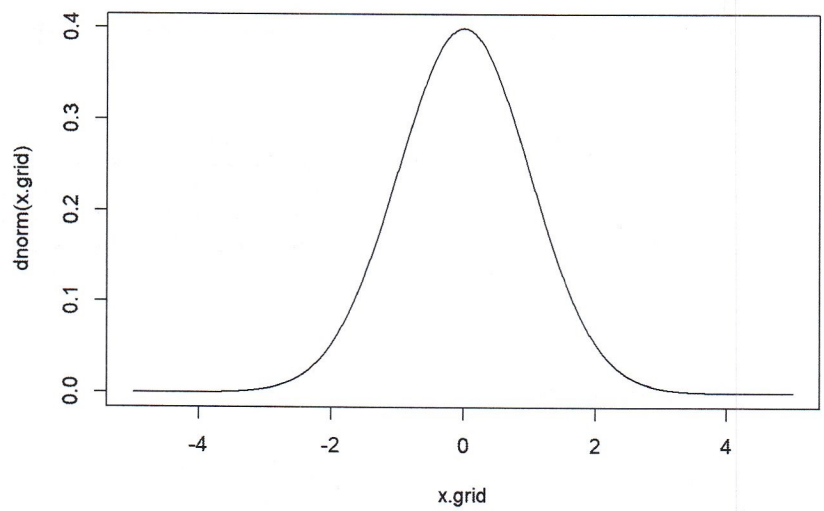
01 - T-test failur

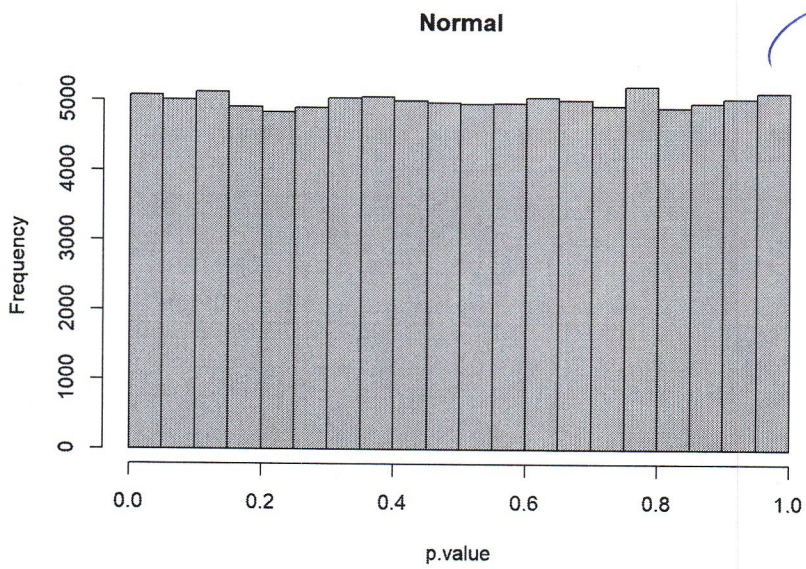
```
### ---------------------------------------------------------------------------
### ---------------------------------------------------------------------------
### Empirical sgnificance level of 1-pop t-test
### ---------------------------------------------------------------------------
### ---------------------------------------------------------------------------
n <- 5
B <- 100000
alpha <- 0.05
set.seed(240279)

### ---------------------------------------------------------------------------
### Normal data
### ---------------------------------------------------------------------------
x.grid <- seq(-5, 5, by=0.01)
plot(x.grid, dnorm(x.grid), type='l')
```

⇒ the assumptions for the t test are valid (we expect here everything to work properly)



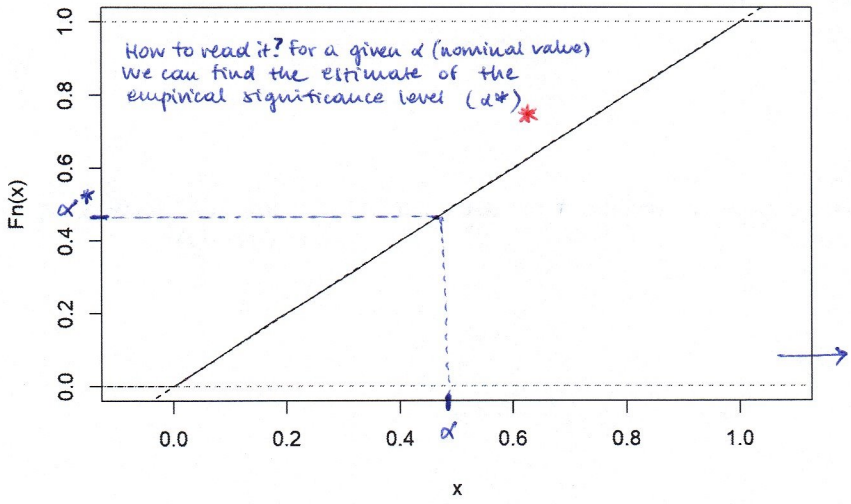

```
p.value <- numeric(B)
for(j in 1:B)
{
  x.sample <- rnorm(n)
  p.value[j] <- t.test(x.sample)$p.value
}

hist(p.value, main = 'Normal')
```

For each dataset we're collecting the p-value



Montecarlo estimate of the distribution of the p-values under H_0
(it resamble very much a normal distribution)

```
plot(ecdf(p.value), main = 'Normal')
abline(0,1, lty=2, col='red')
```



How to read it? For a given α (nominal value) we can find the estimate of the empirical significance level (α^*) *

Empirical cumulative distribution function (= exactly the integral from 0 to α of the previous density)

How is it approximated with Monte Carlo?
For a given value α it counts how many p-values have a value smaller than α and it divides this sum by B (= 100.000 = # p-values)

It's a strict line, this is consistent with the fact that the previous distribution is uniform

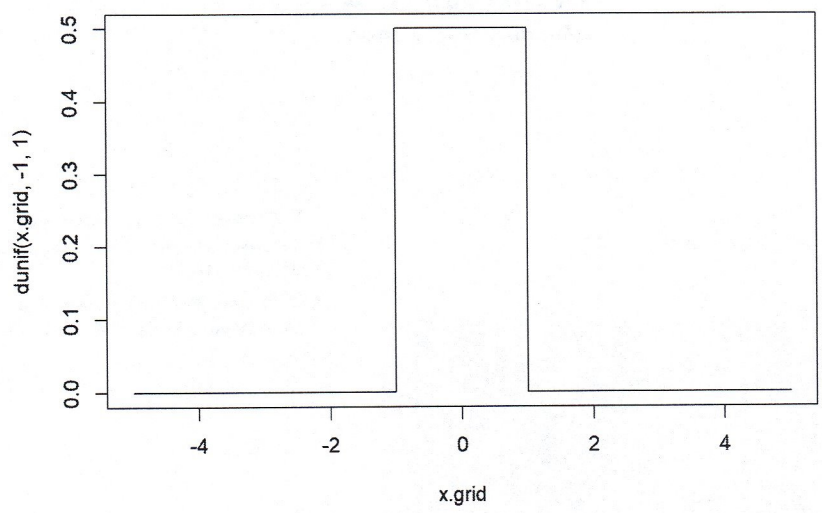
```
estimated.alpha <- sum(p.value < alpha)/B
c(estimated.alpha - sqrt(estimated.alpha*(1-estimated.alpha)/B)*qnorm(0.975),
  estimated.alpha,
  estimated.alpha + sqrt(estimated.alpha*(1-estimated.alpha)/B)*qnorm(0.975))
```

```
## [1] 0.04935014 0.05071000 0.05206986
```

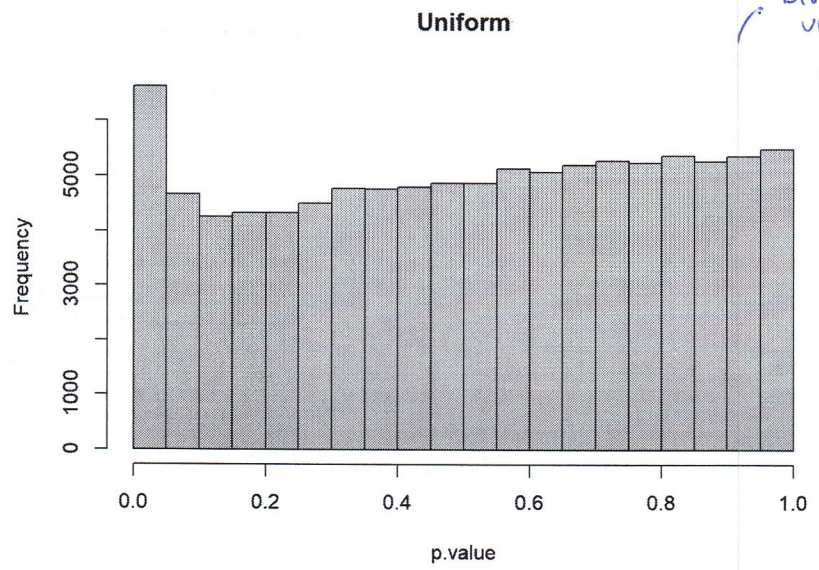
* for instance if we select a nominal value $\alpha = 0.05$ we obtain an empirical value $\alpha^* = 0.05071$ (with confidence interval :)

```
### ----------------------------------------------------------------------
### Leptocurtic data (e.g. Uniform)
### ----------------------------------------------------------------------
x.grid <- seq(-5, 5, by=0.01)
plot(x.grid, dunif(x.grid, -1, 1), type='l')
```

(short tail distribution)

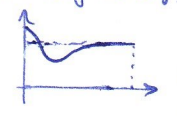



```
p.value <- numeric(B)
for(j in 1:B)
{
  x.sample <- runif(n, -1, 1)
  p.value[j] <- t.test(x.sample)$p.value
}

hist(p.value, main = 'Uniform')
```

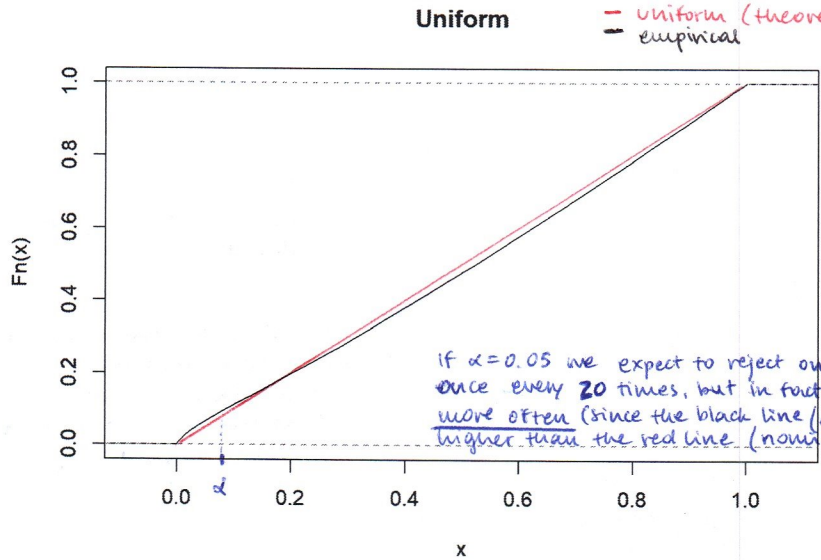


Distribution of the p-values under H_0

This time the distribution is clearly not uniform.
(We already expect the cumulative distribution to be a little bit "wiggly" at the beginning / center)



```
plot(ecdf(p.value), main = 'Uniform')
```



= uniform (theoretical)
= empirical

The violation is not that huge.

If $\alpha = 0.05$ we expect to reject on average once every 20 times, but in fact we reject more often (since the black line (empirical α) is higher than the red line (nominal α))

For larger values of α happens the opposite (when the red line is above the black one)

```
estimated.alpha <- sum(p.value < alpha)/B
c(estimated.alpha - sqrt(estimated.alpha*(1-estimated.alpha)/B)*qnorm(0.975),
  estimated.alpha,
  estimated.alpha + sqrt(estimated.alpha*(1-estimated.alpha)/B)*qnorm(0.975))
```

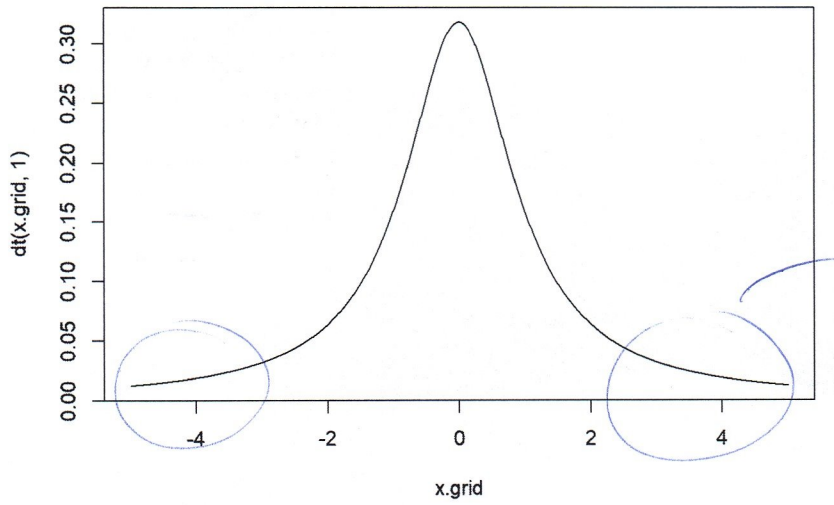
If the nominal α is $\alpha = 0.05$ the actual significance level (empirical) is $\alpha = 0.0661700$

```
## [1] 0.06462932 0.06617000 0.06771068
```

```
### ------------------------------------------------------------------------
### Platicurtic data (e.g. Student's t)
### ------------------------------------------------------------------------
x.grid <- seq(-5, 5, by=0.01)
plot(x.grid, dt(x.grid, 1), type='l')
```

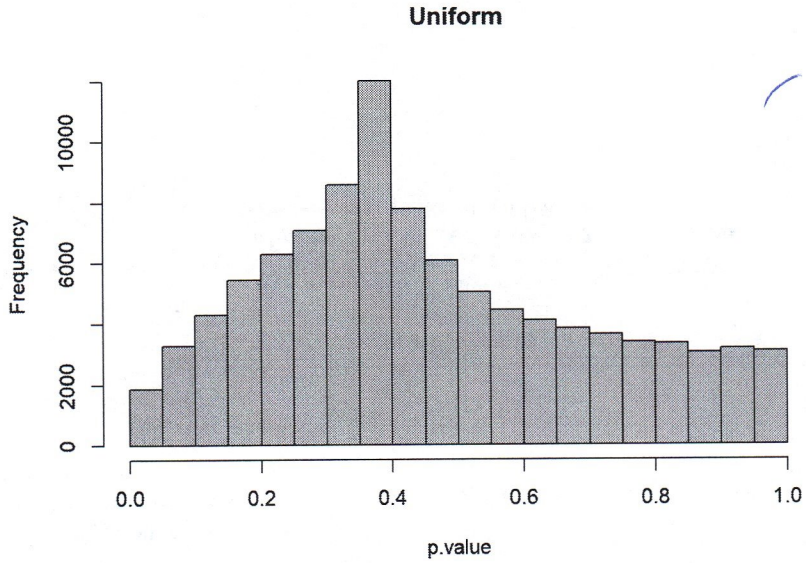
(very long tail distribution) $\Rightarrow$ we have probability $\neq 0$ of observing data which are extremely far from the main cloud

t-student with 1 degree of freedom



the tails are so heavy that neither the mean value or the variance exist
(this is a distribution for which the central limit theorem does not work)

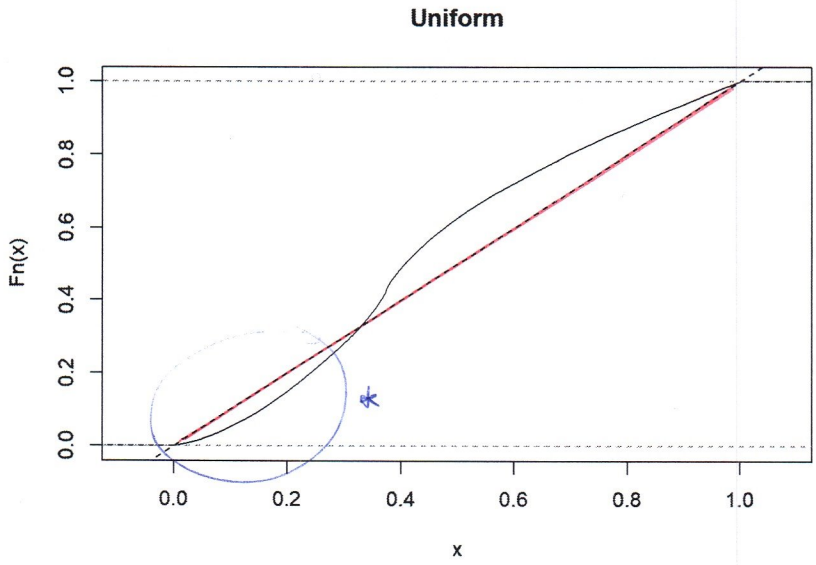
```
p.value <- numeric(B)
for(j in 1:B)
{
  x.sample <- rt(n, 1)
  p.value[j] <- t.test(x.sample)$p.value
}

hist(p.value, main = 'Uniform')
```



Distribution of the p-values under Ho

This is a lot far away from a uniform distribution.
⇒ we have no control about the true level of the test when we've using a standard

```
plot(ecdf(p.value), main = 'Uniform')
abline(0,1, lty=2, col='red')
```



* for small values of α (the more important ones) we reject a lot less: if we fix $\alpha = 0.05$ we think that we reject in average 1 time over 20 times that we do this test, instead we reject 1 time over 50+.

```
estimated.alpha <- sum(p.value < alpha)/B
c(estimated.alpha - sqrt(estimated.alpha*(1-estimated.alpha)/B)*qnorm(0.975),
  estimated.alpha,
  estimated.alpha + sqrt(estimated.alpha*(1-estimated.alpha)/B)*qnorm(0.975))
```

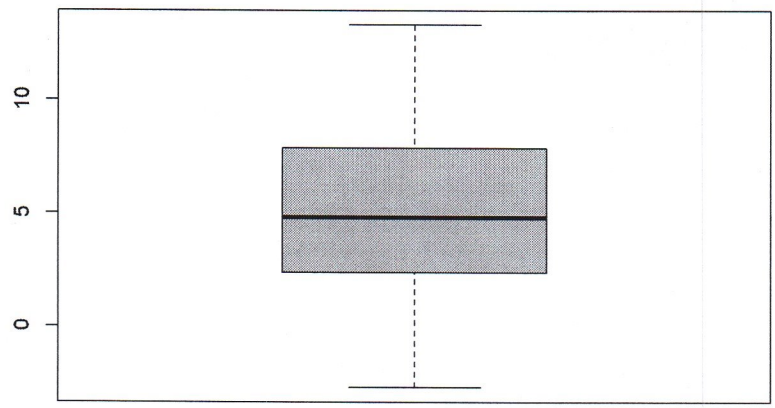
```
## [1] 0.01784085 0.01868000 0.01951915
```

02 - Sign test

```
### ---------------------------------------------------------------------------
### ---------------------------------------------------------------------------
### Example of two-sample paired sign test
### ---------------------------------------------------------------------------
### ---------------------------------------------------------------------------
X <- read.table('parziali.txt')

### ---------------------------------------------------------------------------
### right-sided test
### ---------------------------------------------------------------------------
# H0: P(PII > PI) = 0.5
# H1: P(PII > PI) > 0.5

differences <- X$PII - X$PI
boxplot(differences)
```

= Is it true that was more probable to take a larger grade on the 2nd exam?

H_0: the probability of having the grade higher during the 2nd exam is the same of the probability of having the grade higher during the 1st



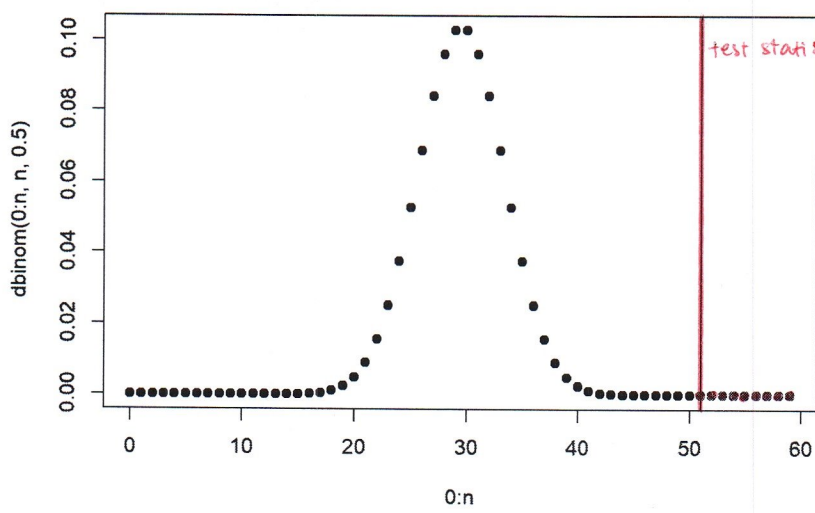
We see a lot of probability on the positive values, which means that (from the boxplot) it seems like we'll reject H_0 (because it seems that the difference of the grades is not $\sim$ norm)

```
n <- length(differences)
signs <- sign(differences)

W <- sum(signs==1)
# W <- (n+sum(signs))/2

plot(0:n, dbinom(0:n, n, 0.5))
abline(v = W, col='red')
points(0:n, dbinom(0:n, n, 0.5), col= (0:n >= W) + 1, pch=16)
```

$n = 59$

$W = 51$ (51 positive variations over 59 (!))

Distribution under H_0



The test is one sided (it is more likely to have an increment than a decrement), it means that the p-value must be computed as probability of the right tail. (Only large values (close to 59) lead to the rejection of H_0)

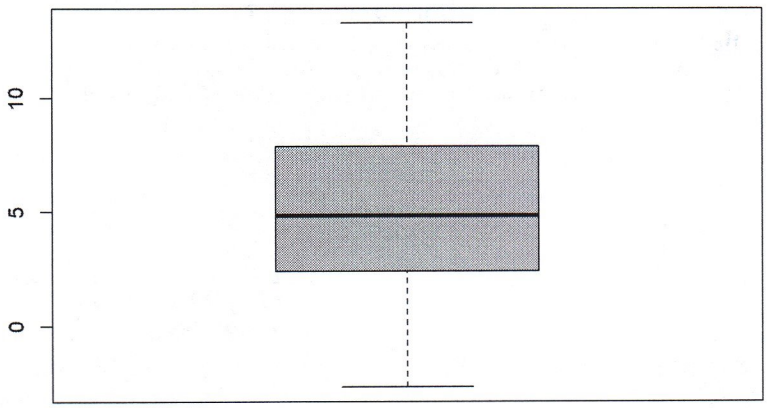
$\Rightarrow$ the p-value is just $\mathbb{P}(W=51) + \mathbb{P}(W=52) + \dots + \mathbb{P}(W=59)$

```
p.value <- 1 - pbinom(W-1, n, 0.5) ### P( B(n,0.5) >= 51 )
```

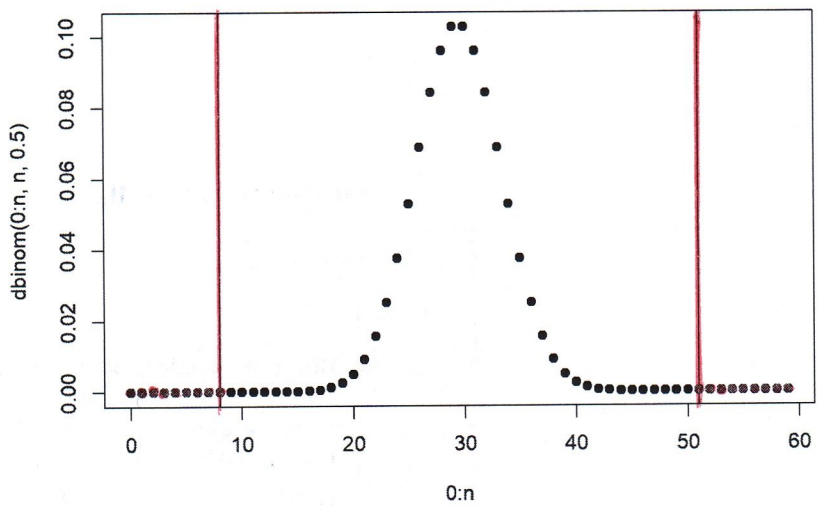
$= 4.526196 \cdot 10^{-9}$

```
### ------------------------------------------------------------------------
### two-sided test
### ------------------------------------------------------------------------
# H0: P(PII > PI)  = 0.5
# H1: P(PII > PI) != 0.5

differences <- X$PII - X$PI
boxplot(differences)
```

what do we have to change? The way we compute the p-value!



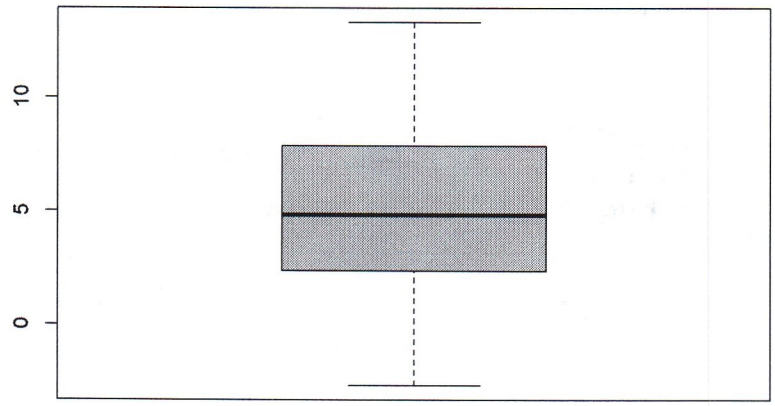

```
n <- length(differences)
signs <- sign(differences)

W <- sum(signs ==  1)
W.max <- max(W, n-W)

plot(0:n, dbinom(0:n, n, 0.5))
abline(v = c(W, n-W), col='red')
points(0:n, dbinom(0:n, n, 0.5), col= (0:n >= max(W,n-W) | 0:n <= min(W,n-W)) + 1, pch=16)
```



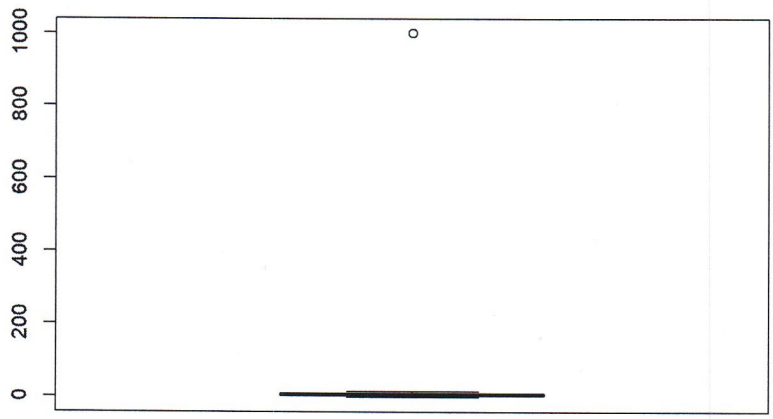
This time the p-value is twice the previous p-value

```
p.value <- 2*(1 - pbinom(W.max-1, n, 0.5) )
### P( B(n,0.5) >= 51 OR B(n,0.5) <= 8) = 2* P( B(n,0.5) >= 51 ) = 2*P( B(n,0.5) <= 8)

### ------------------------------------------------------------------------------
### two-sided test WITH OUTLIERS
### ------------------------------------------------------------------------------
# H0: P(PII > PI)  = 0.5
# H1: P(PII > PI) != 0.5

differences <- X$PII - X$PI
differences.out <- differences
#differences.out[1] <- differences.out[1] - 1000 # sign consistent outlier
differences.out[1] <- differences.out[1] + 1000 # sign non-consistent outlier
boxplot(differences)
```

we artificially modify the data (one datum)




```
boxplot(differences.out)
```



Having just an outlier everything drastically change (!) (actually only in the visualization of the boxplot; the boxplot "schiacciato" is similar to the previous one)

```r
n <- length(differences)
signs <- sign(differences)
signs.out <- sign(differences.out)

W <- sum(signs == 1)
W.max <- max(W, n-W)

W.out <- sum(signs.out == 1)
W.max.out <- max(W.out, n-W.out)

p.value <- 2*(1 - pbinom(W.max-1, n, 0.5) )
p.value.out <- 2*(1 - pbinom(W.max.out-1, n, 0.5) )

# Comparison with the paramtric t-test
p.value
```

W = 51

W.out = 52

$9.052391 \cdot 10^{-9}$

$1.358992 \cdot 10^{-9}$

```
## [1] 9.052391e-09
```

the p-value does not change too much with the outlier in the case of the sign test
⟹ the sign test is robust to outliers

```r
p.value.out
```

```
## [1] 1.358992e-09
```

```r
t.test(differences)$p.value
```

```
## [1] 8.560146e-14
```

the p-value changes a lot with the outlier in the case of the t-test
⟹ t-test is not robust to outliers

```r
t.test(differences.out)$p.value
```

```
## [1] 0.1960045
```

```r
W.max
```

```
## [1] 51
```

```r
W.max.out
```

```
## [1] 52
```

```r
t.test(differences)$statistic
```

```
##        t
## 9.727302
```

```r
t.test(differences.out)$statistic
```

```
##        t
## 1.308087
```

03 - Mann-Whitney U test

```
### ---------------------------------------------------------------------------
### ---------------------------------------------------------------------------
### Example of two-sample (independendent) rank sum test (MW U-test)
### ---------------------------------------------------------------------------
### ---------------------------------------------------------------------------
X <- read.table('parziali.txt')
G <- read.table('sesso.txt')

### ---------------------------------------------------------------------------
### two-sided test
### ---------------------------------------------------------------------------
# H0: P(PI.Male > PI.Female)  = 0.5
# H1: P(PI.Male > PI.Female) != 0.5

PI1 <- X$PI[G$MF == 1]
PI2 <- X$PI[G$MF == 0]
PI  <- X$PI
n1  <- length(PI1)
n2  <- length(PI2)
n   <- length(PI)

ranks.PI <- rank(PI)

R1 <- sum(ranks.PI[G$MF == 1])
U1 <- R1 - n1*(n1+1)/2  # Nr of wins of the 1st sample

R2 <- sum(ranks.PI[G$MF == 0])
U2 <- R2 - n2*(n2+1)/2  # Nr of wins of the 2nd sample

n1*n2 # Nr of contests
```

(again the parziali data, this time we add the gender of the students)

PI1 → grades of the males : random sample from the 1st pop.
PI2 → grades of the females : random sample from the 2nd pop.
PI → pooled sample : only under H_0 this is a random sample

ranks.PI ⟹ ranks.PI = [47 | 30 | ... | 3] : the first student is ranked 47, the second is ranked 30, ..., the last is ranked 3. The student ranked 1 is the worst student.

```
## [1] 850
```

```
U1 - n1*n2/2  # unbalance
```

```
## [1] 77
```

```
U2 - n1*n2/2  # unbalance
```

```
## [1] -77
```

if the two population were identical on average we would expect half of the contests won by females and half won by males.
The average number is $\frac{1}{2}(n_1+n_2) = 425$ and

$U_1 - \frac{1}{2}(n_1+n_2) = 77$
$U_2 - \frac{1}{2}(n_1+n_2) = -77$

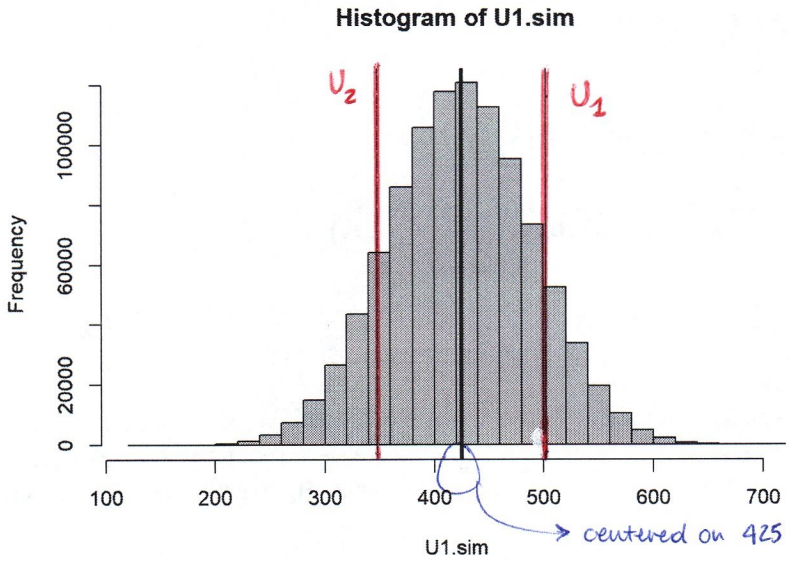
} are the deviations

```
# MC computation of the p-value
# Generation of U1 and U2 under the null hypothesis

set.seed(24021979)
B <- 1000000
U1.sim <- numeric(B)
U2.sim <- numeric(B)
for (k in 1:B)
{
  ranks.temp <- sample(1:n)
  R1.temp <- sum(ranks.temp[1:n1])
  R2.temp <- sum(ranks.temp[(n1+1):(n1+n2)])
  U1.temp <- R1.temp - n1*(n1+1)/2
  U2.temp <- R2.temp - n2*(n2+1)/2
  U1.sim[k] <- U1.temp
  U2.sim[k] <- U2.temp
}

hist(U1.sim)
abline(v = c(U1, U2), col='red')
abline(v = n1*n2/2, lwd=3)
```

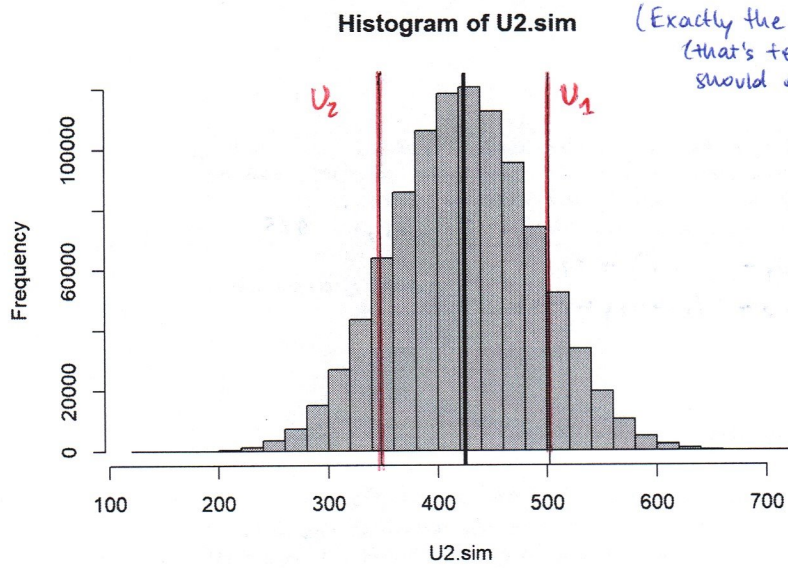
n_1+n_2

we generate 1.000.000 pooled samples, then we compute the rank of the data and make the summation of the first n_1 data and the summation of the remaining ones.

→ # contests won by the males




```
hist(U2.sim)
abline(v = c(U1, U2), col='red')
abline(v = n1*n2/2, lwd=3)
```



(Exactly the same distribution)
(that's telling us that probably we should accept H_0)

```
U.star <- max(U1, U2)

p.value <- 2 * sum(U1.sim >= U.star)/B
p.value
```

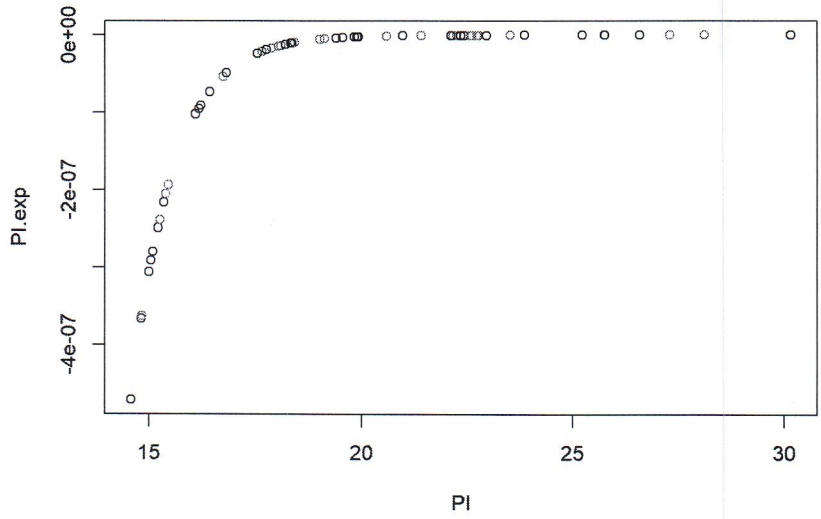
```
## [1] 0.242994
```

$\Longrightarrow$ not reject H_0 : there is not enough evidence to think that $\underline{P(PI.male > PI.female)} \neq \frac{1}{2}$

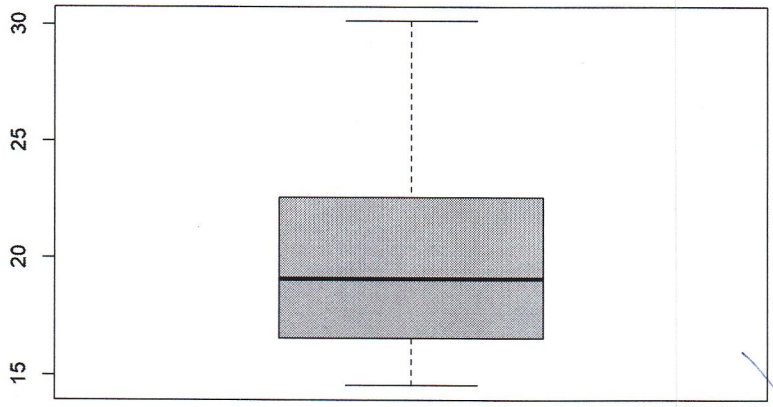
```
# Let's apply a NON-LINEAR MONOTONIC TRANSFORMATION

PI  <- X$PI
PI1 <- X$PI[G$MF == 1]
PI2 <- X$PI[G$MF == 0]
PI.exp  <- -exp(-PI)
PI1.exp <- -exp(-PI1)
PI2.exp <- -exp(-PI2)

plot(PI, PI.exp, col=G$MF+1)
```

in this case, instead of working with grades we work with $-\exp(-\text{grades})$

What changes in the ranks? Nothing.
Non-linear (monotonic) transformations do not influence this test.

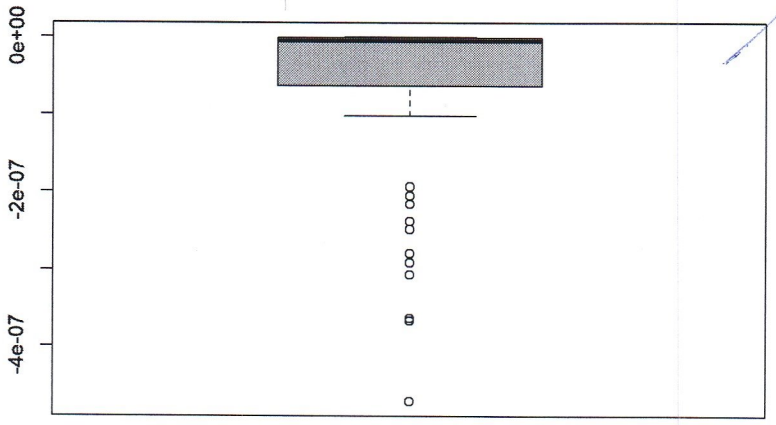



```
boxplot(PI)
```




```
boxplot(PI.exp)
```

Even if the boxplots are different it is not important: the important thing is what happens to the ranks.



```
ranks.PI <- rank(PI)
R1 <- sum(ranks.PI[G$MF == 1])
U1 <- R1 - n1*(n1+1)/2

ranks.PI.exp <- rank(PI.exp)
R1.exp <- sum(ranks.PI.exp[G$MF == 1])
U1.exp <- R1.exp - n1*(n1+1)/2
```

```
U1
```

```
## [1] 502
```

```
U1.exp
```

```
## [1] 502
```

```
U.star     <- n1*n2/2 + abs(U1     - n1*n2/2)
U.star.exp <- n1*n2/2 + abs(U1.exp - n1*n2/2)

# no need for a new MC estimation being n1 and n2 the same as before
p.value     <- 2 * sum(U1.sim >= U.star)/B
p.value.exp <- 2 * sum(U1.sim >= U.star.exp)/B
```

```
p.value
```

```
## [1] 0.242994
```

exactly the same!

```
p.value.exp
```

```
## [1] 0.242994
```

```
t.test(PI1,     PI2    )$p.value
```

```
## [1] 0.3445562
```

```
t.test(PI1.exp, PI2.exp)$p.value
```

```
## [1] 0.2510763
```

!

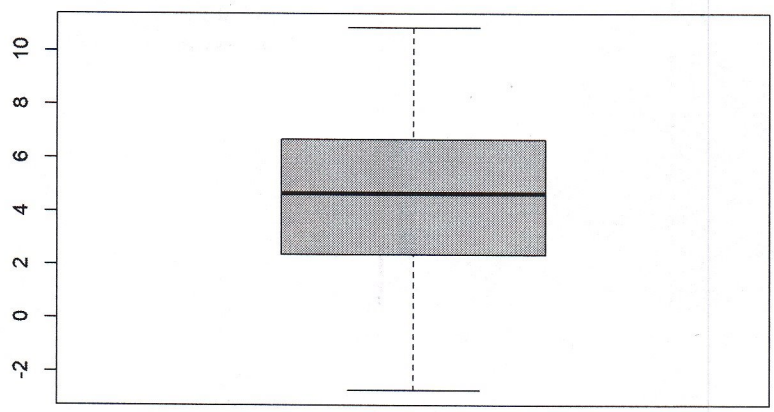
⟹ No matter the way in which we measure a phenomena, if there is a monotonic relation in the ways we measure a phenomena we'll obtain the same inference.

04 - Wilcoxon Signed Rank W test

```
### ---------------------------------------------------------------------------
### ---------------------------------------------------------------------------
### Example of two-sample paired signed rank test
### ---------------------------------------------------------------------------
### ---------------------------------------------------------------------------
X <- read.table('parziali.txt')
G <- read.table('sesso.txt')

### ---------------------------------------------------------------------------
### two-sided test (for male students)
### ---------------------------------------------------------------------------
# H0: P(PII > PI | Male)  = 0.5
# H1: P(PII > PI | Male) != 0.5

differences <- X$PII[G$MF == 1] - X$PI[G$MF == 1]
boxplot(differences)
```

Considering the median, are the grades of PII better than the grades of PI?

← pairwise difference (we consider only males) of PII and PI




```
n       <- length(differences)
ranks   <- rank(abs(differences))
W.plus  <- sum(ranks[differences > 0])
W.minus <- sum(ranks[differences < 0])

W.plus
```

← remember "abs": we're ranking data w.r.t. the magnitude, ≠ of the sign

```
## [1] 307
```

```
W.minus
```

```
## [1] 18
```

```
n*(n+1)/2
```

$= W^+ + W^-$

```
## [1] 325
```

```
W <- W.plus - W.minus
# W <- sum(sign(differences)*rank(abs(differences)))
W
```

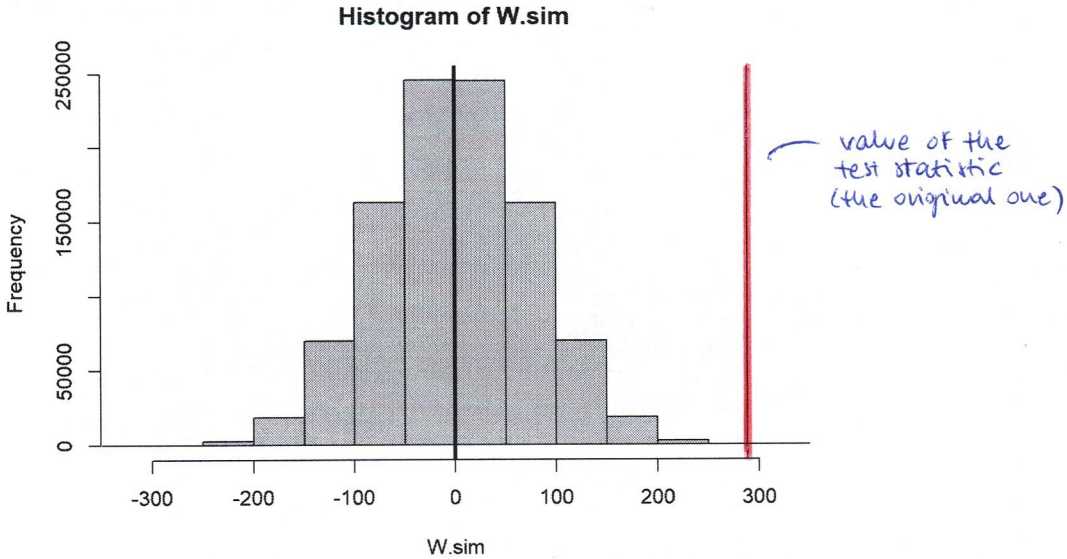
```
## [1] 289
```

```r
# MC computation of the p-value
# Generation of W under the null hypothesis
set.seed(24021979)
B     <- 1000000
W.sim <- numeric(B)
for (k in 1:B)
{
  ranks.temp <- sample(1:n)
  signs.temp <- 2*rbinom(n, 1, 0.5) - 1
  W.temp     <- sum(signs.temp*ranks.temp)
  W.sim[k]   <- W.temp
}

hist(W.sim, xlim=c(-n*(n+1)/2, n*(n+1)/2))
abline(v = W, col='red')
abline(v = 0, lwd=3)
```

we make 1.000.000 realizations of the test statistics generating values (of the test stat.) under H_0.

We randomly generate the values (just randomly permuting the sequence $[1,\dots,n]$) and we randomly generate the sign (sampling n times from a Bernoulli distribution $(\frac{1}{2})$)

this is just a transformation to have a Bernoulli generating $\{-1,+1\}$ instead of $\{0,1\}$



value of the test statistic (the original one)

```r
### ---------------------------------------------------------------------------
### two-sided test
### ---------------------------------------------------------------------------
# H0: P(PII > PI | Male)  = 0.5
# H1: P(PII > PI | Male) != 0.5
p.value <- 2 * sum(W.sim >= abs(W))/B
p.value
```

Monte Carlo estimate of the p-value:
count the number of times in which MC simulation obtain a value for the test statistic larger than $W = 289$, divided by $B = 1.000.000$ times 2 (since the test is 2 sided)

```
## [1] 1.4e-05
```

```r
### ---------------------------------------------------------------------------
### right-sided test
### ---------------------------------------------------------------------------
# H0: P(PII > PI | Male) = 0.5
# H1: P(PII > PI | Male) > 0.5
p.value <- sum(W.sim >= W)/B
p.value
```

```
## [1] 7e-06
```

(half of the previous)

```r
### ---------------------------------------------------------------------------
### lwfte-sided test
### ---------------------------------------------------------------------------
# H0: P(PII > PI | Male) = 0.5
# H1: P(PII > PI | Male) < 0.5
p.value <- sum(W.sim <= W)/B
p.value
```

```
## [1] 0.999994
```

(1 - p_value_right_sided_test)