

```

X1=rnorm(n=20, mean=10, sd=3)
X2=rnorm(n=30, mean=14, sd=2)
X3=rexp(n=15, rate=1/2)

par(mfrow=c(3,3))

hist(X1, probability = T)
hist(X2, probability = T, horizontal=T)
hist(X3, probability = T, horizontal=T)

barplot(X1, horizontal=T)
barplot(X2, horizontal=T)
barplot(X3, horizontal=T)

qqnorm(X1); qqline(X1)
qqnorm(X2); qqline(X2)
qqnorm(X3); qqline(X3)

qqplot.das(X1, "norm"); qqplot.das(X2, "norm"); qqplot.das(X3, "norm");

sh1=shapiro.test(X1); sh2=shapiro.test(X2); sh3=shapiro.test(X3)
library(tseries); library(nortest)
jb1=jarque.bera.test(X1); jb2=jarque.bera.test(X2); jb3=jarque.bera.test(X3)
ad1=ad.test(X1); ad2=ad.test(X2); ad3=ad.test(X3)

shapiro = (sh1$p.value>0.05 && sh2$p.value>0.05 && sh3$p.value>0.05)
jarque = (jb1$p.value>0.05 && jb2$p.value>0.05 && jb3$p.value>0.05)
anderson= (ad1$p.value>0.05 && ad2$p.value>0.05 && ad3$p.value>0.05)

if(shapiro && jarque && anderson) print("all normal=>one side anova") else print("at least one not
normal=>kruskal")

DF=data.frame(gr=c(rep("X1", length(X1)), rep("X2", length(X2)), rep("X3",
length(X3))),resp=c(X1,X2,X3))

k=kruskal.test(formula=resp~gr, data=DF)
if(k$p.val > 0.05) print("no evidence to reject H0") else print("reject H0 in favour of HA")

model=aov(formula=resp~gr, data=DF)
pval=summary(model)[[1]][["Pr(>F)"]]
if(pval>0.05) print("no evidence to reject H0") else print("reject H0 in favour of HA")

-----

```

Корелация: Най-точна е тази на Pearson, но е най-чувствителна към outliers. Най-неточна е тази на Kendall и е най-нечувствителна към outliers. Spearman се използва, когато искаме да видим дали имаме монотонност на данните, докато Pearson дори се опитва да я оцени тази монотонност. Корелацията варира от -1 до 1, като колкото повече модула ѝ се приближава до 1-ца, толкова по-голяма линейна зависимост имаме.

И корелацията и регресията изследват взаимовръзка между две непрекъснати променливи. За разлика от корелацията, регресията се опитва да сложи и самата оценка, т.е. с колко изменението на едната променлива ефектира на другата променлива.

```
n=100
set.seed(99)
x1=runif(n,0,1)
x2=rnorm(n,2,4)
x3=rexp(n,1/4)
y=5+6*x1+x2^2/3+3*rnorm(n)
DF=data.frame(x1,x2,x3,y)
plot(DF)
model=lm(y ~ x1+l(x2^2)+x3, data=DF)
summary(model)
```