```
# Κατασκευή παραγοντικού σχεδιασμού 2^2

# A = factor(rep(c(-1,1),  times = 6))
# B = factor(rep(c(-1,1),each = 2, times=3))
# Rep = factor(rep(c(1:3), each=4))
# design=data.frame(A, B, Rep)
# design

# library(agricolae)
# trt=c(2,2)
# design=design.ab(trt, r=4, design=c("crd"), seed=1,randomization=T)

# library(DoE.base)
# design=fac.design(2, 2, replications= 3, randomize=FALSE)

design=expand.grid(A=c(-1,1),B=c(-1,1),Rep=c(1,2,3))
design

Y=c(28, 36, 18, 31, 25, 32, 19, 30, 27, 32, 23, 29)
data=cbind(design, Y)
data

fit=aov(Y~(A+B)^2,data)
plot(fit)
# par(mfrow=c(2,2))
plot(fit)

anova(fit)

fit=lm(Y~(A+B)^2,data)
summary(fit)

library (rsm)
# data=as.coded.data(data,A~(Conc-20)/5,B~(Cat-1.5)/0.5)
data

datarsm=rsm(Y ~ FO(A,B), data)
persp(datarsm, A ~B)

contour(datarsm, ~ A + B, image=TRUE)

# Κατασκευή παραγοντικού σχεδιασμού 2^3

data=expand.grid(A=c(-1,1),B=c(-1,1),C=c(-1,1),R=c(1,2))
data


Y=c(550,669,633,642,1037,749,1075,729,604,650,601,635,1052,868,1063,860)
data=cbind(design, Y)
data

anova(fit)

fit=lm(Y~A*B*C, data)
summary(fit)

datarsm=rsm(Y ~ FO(A, C) + TWI(A, C), data)
```

```
contour(datarsm, ~ A + C, image=TRUE)

persp(datarsm, C~A)

# Μη επαναλαμβανόμενος σχεδιασμός 2^4

library(FrF2)
design=FrF2(16,4,randomize = FALSE)
design

Y=c(45,71,48,65,68,60,80,65,43,100,45,104,75,86,70,96)
data=add.response(design, Y)
data

anova(fit)

fit=lm(Y~A*B*C*D,data)
summary(fit)

DanielPlot(fit)

library(BsMD)
LenthPlot(fit)


fit=lm(Y~A*C*D, data)
anova(fit)

summary(fit)

# Προσθήκη κεντρικών σημείων

datac=add.center(data, 4)
datac

Y=c(45,71,48,65,68,60,80,65,43,100,45,104,75,86,70,96,73,75,66,69)
datac=add.response(datac, Y, replace=TRUE)
datac

fit=lm(Y~(A+B+C+D)^4, datac)
anova(fit)

# library(remotes)
# remotes::install_github("cran/alr3")

library(alr3)
pureErrorAnova(fit)


# Κατασκευή παραγοντικού σχεδιασμού με ανάμειξη ABCD

library(conf.design)
design=conf.design(c(A = 1, B = 1, C = 1, D = 1), 2)
design

Y=c(45,65,60,80,100,45,75,96,71,48,68,65,43,104,86,70)
data=cbind(design, Y)
```

```
data


fit=aov(Y ~ (A+B+C+D)^ 4 + Error(Blocks), data)
summary(fit)

fit=lm(Y~(A+B+C+D)^4, data)

DanielPlot(fit)

fit=aov(Y ~ (A+B+C+D)^ 2 + Error(Blocks), data)
summary(fit)

# Κατασκευή κλασματικού παραγοντικού σχεδιασμού I=ABCDE

design = FrF2(nfactors=5, resolution=5, randomize = F)
design = FrF2(16, generators = "ABCD", randomize = F)
 summary(design)

Y=c(8,9,34,52,16,22,45,60,6,10,30,50,15,21,44,63)
data=cbind(design, Y)
data

fit=lm(Y~ (A+B+C+D+E)^ 4, data=data)
aliases(fit)

fit=lm(Y~ (A+B+C+D+E)^2 , data=data)
# fit=lm(Y~ (A+B+C+D+E)^4 , data=data)
 summary(fit)

anova(fit)
DanielPlot(fit)

 fit=lm(Y~A+B+C+A*B, data=data)
anova(fit)

# 1/4 κλάσμα σχεδιασμού 2^6

design=FrF2(16,generators=c("ABC","BCD"),randomize=F)
Y=c(6,10,32,60,4,15,26,60,8,12,34,60,16,5,37,52)
data=cbind(design,Y)
fit=lm(Y~(A+B+C+D+E+F)^6, data=data)
summary(fit)
aliases(fit)
DanielPlot(fit)

fit=lm(Y~(A+B)^2, data=data)
summary(fit)


# Κατασκευή Plackett-Burman designs

library(FrF2)
pb(12,randomize=FALSE)

Y=rnorm(12)
design=pb(12,randomize=FALSE)
```

```
data=cbind(design,Y)
fit=lm(Y~(A+B+C+D+E+F+G+H+J+K+L)^2,data)
alias(fit)

# Κατασκευή παραγοντικού σχεδιασμού 3^2

library(DoE.base)
design32=fac.design(3, 2, replications=2, randomize=F)
design32

Y=c(-2,0,-1,-3,1,5,2,4,0,-1,2,0,0,3,6,3,6,-1)
data=add.response(design32, Y)
data

#  Κατασκευή παραγοντικού σχεδιασμού 3^2 με ανάμειξη AB^2

library(conf.design)
design=conf.design(c(A = 1, B = 2), 3)
design

#  Κατασκευή παραγοντικού σχεδιασμού 3^3 με ανάμειξη AB^2C^2

design=conf.design(c(A = 1, B = 2, C = 2), 3)
design

fract.design=design[c(19:27),]
fract.design
```