

# Package ‘gexp’

June 14, 2019

**Title** Generator of Experiments

**Version** 1.0-0

**Date** 2019-06-14

**Author** Ivan Bezerra Allaman <ivanalaman@gmail.com> and  
José Claudio Faria <joseclaudio.faria@gmail.com>

**Maintainer** Ivan Bezerra Allaman <ivanalaman@gmail.com>

**Depends** R (>= 3.5.0)

**Imports** mvtnorm, tcltk, jpeg, png

**Description** Generates experiments - simulating structured or experimental data as:  
completely randomized design, randomized block design, latin square design,  
factorial and split-plot experiments (Ferreira, 2008, ISBN:8587692526;  
Naes et al., 2007 <doi:10.1002/qre.841>; Rencher et al., 2007, ISBN:9780471754985;  
Montgomery, 2001, ISBN:0471316490).

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**License** GPL (>= 2)

**URL** <https://github.com/ivanalaman/gexp>

**Encoding** latin1

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2019-06-14 11:40:12 UTC

## R topics documented:

Generator of Experiments . . . . .	2
gexp . . . . .	2
plot . . . . .	22
print . . . . .	28
summary . . . . .	32

<b>Index</b>	<b>36</b>
--------------	-----------

---

Generator of Experiments

*gexp: Generator of Experiments*

---

## Description

The package `gexp` was created to make it possible to plan, create and to model structured experiments, that is, under a experimental design. In the modeling it is possible to simulate results of experiments with possibility of user to report effects and random error(s). The designs are: Completely Randomized Design (CRD), Randomized Complete Block Design (RCBD) and Latin Squares Design (LSD). The types of experiments are: Simple (SIMPLE), Factorial Experiment (FE) and Split-plot Experiment (SPE).

The experiments can be generated with one or more response variables, in the latter case, a strict covariance structure can be imposed. It is also possible to plan experiments using the graphic functions for use in planning from pictures or pictures of the experimental area.

The possible uses are multiple: in the planning it makes possible to distribution and randomization of treatments and experimental units; in the data analysis allows to generate experiments for application in evaluations individual and can also be used to generate experiments for validations of new computational resources in the area of structured data analysis.

In summary, the package provides computational resources useful in planning and modeling of structured experiments in the R.

## Details

In some situations, we are interested in simulating a variable randomized according to the experimental procedure where the differences between treatments are predetermined. In a completely randomized design with two treatments for example, we may have an interest in simulating a variable random whose treatment A will have a 1-deviation effect and treatment B a effect of 3 deviations from a given overall average. In addition, may be interested in imposing a pre-established error structure for purposes evaluation in the various analysis strategies.

## Author(s)

Ivan Bezerra Allaman <[ivanalaman@gmail.com](mailto:ivanalaman@gmail.com)> and José Cláudio Faria <[joseclaudio.faria@gmail.com](mailto:joseclaudio.faria@gmail.com)>.

Maintainer: Ivan Bezerra Allaman <[ivanalaman@gmail.com](mailto:ivanalaman@gmail.com)>

---

`gexp`

*Generator of Experiments*

---

## Description

The package provides computational resources useful in planning and modeling of structured experiments in the R environment.

The generic function S3 `gexp` was created to enable plan, create and model structured experiments, that is, under a design experimental. In the modeling it is possible to simulate results of experiments with possibility of user informing the effects and the random error(s). The designs are: Completely Randomized Design (CRD), Randomized complete block design (RCBD) and Latin Squares Design (LSD). The types of experiments are: Factorial Experimentation (FE) and Split-plot experiment (SPE).

The experiments can be generated with one or more variable response, in the latter case, it may be important for a structure covariance between them. It is also possible to plan experiments with of graphic parts for use in planning from figures or pictures of the experimental area.

## Usage

```
gexp(x, ...)  
  
## Default S3 method:  
gexp(  
  x      = NULL,  
  mu     = 26,  
  err    = NULL,  
  errp   = NULL,  
  r      = 5L,  
  fl     = NULL,  
  blk1   = NULL,  
  row1   = NULL,  
  coll   = NULL,  
  fe     = NULL,  
  inte   = NULL,  
  blke   = NULL,  
  rowe   = NULL,  
  cole   = NULL,  
  contrasts = NULL,  
  type   = c('SIMPLE', 'FE', 'SPE'),  
  design = c('CRD', 'RCBD', 'LSD'),  
  round  = 2L,  
  ...)  
  
## S3 method for class 'simple_crd'  
gexp(  
  x, ...)  
  
## S3 method for class 'simple_rcbd'  
gexp(  
  x, ...)  
  
## S3 method for class 'simple_lsd'
```

```

gexp(
  x, ...)

## S3 method for class 'fe_crd'
gexp(
  x, ...)

## S3 method for class 'fe_rcbd'
gexp(
  x, ...)

## S3 method for class 'fe_lsd'
gexp(
  x, ...)

## S3 method for class 'spe_crd'
gexp(
  x, ...)

## S3 method for class 'spe_rcbd'
gexp(
  x, ...)

## S3 method for class 'spe_lsd'
gexp(
  x, ...)

```

### Arguments

x	An object of gexp class.
mu	Is a numeric scalar, or a vector to Multivariate Data (MD), that represent the mean of each factor. Required NULL if the effect of at least one factor of the experiment is quantitative.
err	It is a vector, or matrix for MD, that represents the experimental error. The default value is NULL, that is, for each response variable a normal error is added with mean 0 and variance 1 generated by <code>rmvnorm(sigma = diag(length(mu)))</code> .
errp	It is a vector, or a matrix for MD, of the error associated with the plots if type is equal to SPE (Split-Plot Experiments). The default value is NULL, that is, for each response variable a normal error is added with mean 0 and variance 1 generated by <code>rmvnorm(sigma = diag(length(mu)))</code> .
r	It is a scalar of the number of repetitions of the experiment.
f1	List of a vector of characters, or a matrix (MD). It's a list of factor names.
blk1	List of a vector of characters, or an array for MD, of block names.
row1	List a vector of characters, or an array for MD, of the line names in case type is equal to LSD (Latin Square Design).
coll	List of a vector of characters, or an array for MD, of the column names in case the type is equal to LSD (Latin Square Design).

fe	It is a numerical vector, or a matrix (MD). It's a list of the effect of a factor.
inte	It is a numerical vector, or a matrix for MD, of the effects of the interaction.
blke	It is a numerical vector, or a matrix for MD, of the effects of the blocks.
rowe	It is a numerical vector, or an array for MD, of the effects of the lines in case the type is equal to LSD (Latin Square Design).
cole	Is a numeric vector, or a matrix for MD, of the effects of the columns in case the type is equal to LSD (Latin Square Design).
contrasts	A list, whose entries are values (numeric matrices or character strings naming functions) to be used as replacement values for the contrasts function and whose names are the names of the columns. See "contrasts.arg" argument of the model.matrix function to more details. Required if the effect of at least one factor of the experiment is quantitative.
type	It is a vector of strings that contains the type of experiment to be used: Simple (SIMPLE), Factorial Experiment (FE) and Split-plot Experiment (SPE). SIMPLE is the default.
design	It is a vector of strings that contains the type of design to be used: Completely Randomized Design (CRD), Randomized Complete Block Design (RCBD), Latin Squares Design (LSD). CRD is the default.
round	This is a numeric scalar for rounding of the response variable.
...	Further arguments (required by generic).

### Value

The method gexp returns the list of class gexp with the slots:

X	It is the incidence matrix of the design.
Z	It is the incidence matrix of the error of the main parcel in the case of type equal to SPLIT.
Y	It is a vector, or a matrix for MD, with the values of the random variable(s).
dfm	It is a data.frame with all experiment information: treatments, repetitions, and the random response variable.

### References

- Ferreira, Daniel Furtado. 2008. *Estatística Multivariada*. 1 ed. Lavras: Ed. UFLA.
- Aquino, Luiz Henrique. *Técnica Experimental com Animais I*. Apostila da disciplina "Técnica Experimental com Animais" da Universidade Federal de Lavras, 1992.
- Rencher, Alvin C. and Schaalje, Bruce G. 2007. *Linear Models in Statistics, second edition*. Hoboken: John Wiley & Sons.
- Naes, T.; Aastveit, A.H.; Sahni, N.S. 2007. "Analysis of split-plot designs: An Overview and Comparison of Methods". *Qual. Reliab. Engng. Int.* 23, 801-820.

### See Also

[plot.gexp.simple\\_crd](#)

## Examples

```

#####
#   UNIVARIATE APPROACH   #
#####
#!_____
#! Qualitative Factor(s) (QL)
#!_____

#! Completely Randomized Design (CRD)
#! 1 factor - CRD - QLF
# Nonsense(experimental error = 0)
#  $Y_i = \mu + fe + e$ 
r <- 2 # (repet. number)
fln <- 3 # (factor levels number)

crd00 <- gexp(mu = 0,
             r = r,
             fe = list(f1 = c(1, 2, 3)),
             err = matrix(0,
                          nrow = r*fln),
             round = 0)

crd00$X
print(crd00)
summary(crd00)
str(crd00)

#! 1 factor - CRD - QL
# Nonsense(error is 0)
#  $Y_i = \mu + fe + e$ 
r <- 3 # (repet. number)
fln <- 5 # (factor levels number)

crd01 <- gexp(mu = 1,
             r = r,
             fe = list(f1 = c(0, 2, 4, 6, 8)),
             err = matrix(0,
                          nrow = r*fln),
             round = 2)

summary(crd01)

#! 1 factor - CRD - QL
# Default error: rmvnorm(sigma = diag(ncol(as.matrix([[fe]]))))
crd_1f <- gexp(mu = 1,
             r = 3,
             fe = list(f1 = c(1, 1, 5, 1, 1)),
             fl = list(Treat = LETTERS[1:5]),
             round = 2)

crd_1f$X
summary(crd_1f)

```

```

#! Binomial error - CRD - QL
e_binom <- as.matrix(rbinom(n = 15,
                           size = 5,
                           prob = 0.1))

crd_bin <- gexp(mu = 20,
               err = e_binom,
               r = 5,
               fe = list(f1 = c(1, 4, 1)))

summary(crd_bin)

mod <- aov(Y1 ~ X1,
           data = crd_bin$dfm)

shapiro.test(mod$res)

#! Factorial Experiment (FE) - CRD - QL
fe_crd00 <- gexp(mu = 0,
                r = 2,
                fe = list(f1 = c(1, 1, 5),
                          f2 = c(1, 1),
                          f3 = c(2, 2, 1)),
                fl = list(A = paste('a',
                                     1:3,
                                     sep = ''),
                          B = paste('b',
                                     1:2,
                                     sep = ''),
                          C = paste('c',
                                     1:3,
                                     sep = '')),
                round = 0,
                type = 'FE')
fe_crd00$X
summary(fe_crd00)

#! Factorial Experiment (FE) - With interaction - CRD - QL
fe_crd01 <- gexp(mu = 30,
                fe = list(f1 = c(1, 1, 3),
                          f2 = c(1, 1)),
                fl = list(A = paste('a',
                                     1:3,
                                     sep = ''),
                          B = paste('b',
                                     1:2,
                                     sep = '')),
                inte = c(3, 1, 1, 1, 1, 5), # (3*2)
                round = 1,
                type = 'FE')
summary(fe_crd01)

#! Split-plot Experiment (SPE) - CRD - QL

```

```

split_crd <- gexp(mu = 30,
  fe = list(f1 = c(1, 1),
            f2 = c(2, 3)),
  fl = list(P = paste('p',
                    1:2,
                    sep = '')),
          SP = paste('sp',
                    1:2,
                    sep = '')),
  inte = c(1, 15, 1, 1), # (2*2)
  round = 1,
  type = 'SPE',
  design = 'CRD')

split_crd$X
split_crd$Z
summary(split_crd)

split_crd01 <- gexp(mu = 30,
  r = 3,
  fe = list(f1 = c(1, 1),
            f2 = c(2, 3),
            f3 = c(1, 1, 1)),
  fl = list(P = paste('p',
                    1:2,
                    sep = '')),
          A = paste('a',
                    1:2,
                    sep = '')),
          B = paste('b',
                    1:3,
                    sep = '')),
  round = 1,
  type = 'SPE',
  design = 'CRD')

split_crd01$X
split_crd01$Z
summary(split_crd01)

#! Randomized Complete Block Design (RCBD) - QL
# 1 factor, 3 blocks
rcbd <- gexp(mu = 0,
  r = 2,
  fe = list(f1 = c(5, 1, 1)),
  fl = list(TR = LETTERS[1:3]),
  blke = c(1, 2, 3),
  blk1 = list(BLK = paste('B',
                        1:3,
                        sep = '')),
  round = 1,
  design = 'RCBD')

rcbd$X
summary(rcbd)

```



```

#! Factorial Experiment (FE) - RCBD - QL
fe_rcbd <- gexp(mu = 30,
  r = 2,
  fe = list(f1 = c(1, 1, 1),
            f2 = c(2, 3)),
  blke = c(1, 3),
  inte = c(1, 15, 1, 1, 5, 1), # (3*2)
  round = 1,
  type = 'FE',
  design = 'RCBD')
summary(fe_rcbd)

#! Multivariated - RCBD - QL
rcbd_m <- gexp(mu = c(0, 2),
  fe = list(f1 = matrix(c(1, 1,
                          5, 1,
                          1, 1),
                        ncol = 2,
                        byrow = TRUE)),
  blke = matrix(c(2, 1,
                  1, 2,
                  1, 1),
                ncol = 2,
                byrow = TRUE),
  round = 1,
  design = 'RCBD')
summary(rcbd_m)

#! Split-plot Experiment (SPE) - RCBD - QL
split_rcbd <- gexp(mu = 30,
  r = 2,
  fe = list(f1 = c(1, 1),
            f2 = c(2, 3),
            f3 = c(1, 1, 1)),
  fl = list(P = paste('p',
                      1:2,
                      sep = ''),
            B = paste('b',
                      1:2,
                      sep = ''),
            C = paste('c',
                      1:3,
                      sep = '')),
  blke = c(1, 2),
  blk1 = list(BLK = paste('B',
                          1:2,
                          sep = '')),
  round = 1,
  type = 'SPE',
  design = 'RCBD')
split_rcbd$Z
summary(split_rcbd)

```

```

#! Latin Square Design (LSD) - QL
#!. Warning!!!! r = 5 by default
lsd00 <- gexp(design = 'LSD')

#Set r = 1 to omitting warning
lsd01 <- gexp(mu = 30,
  r = 1,
  fe = list(f1 = c(1, 1, 10)),
  rowe = c(1, 1, 1),
  cole = c(1, 1, 1),
  row1 = list(Row = paste('r',
    1:3,
    sep = '')),
  coll = list(Col = paste('c',
    1:3,
    sep = '')),
  round = 0,
  design = 'LSD')
summary(lsd01)

#! Factorial Experiment (FE) - LSD - QL
fe_1sd <- gexp(mu = 30,
  r = 1,
  fe = list(f1 = c(1, 1),
    f2 = c(2, 3)),
  rowe = c(1, 3, 2, 1),
  cole = c(2, 2, 1, 1),
  row1 = list(Row = paste('r',
    1:4,
    sep = '')),
  coll = list(Col = paste('c',
    1:4,
    sep = '')),
  inte = c(1, 15, 1, 1), # (2*2)
  round = 1,
  type = 'FE',
  design = 'LSD')
summary(fe_1sd)

#! Split-plot Experiment (SPE) - LSD - QL
split_1sd <- gexp(mu = 30,
  r = 1,
  fe = list(f1 = c(1, 1, 2),
    f2 = c(2, 3, 1)),
  fl = list(P = paste('p',
    1:3,
    sep = '')),
  SP = paste('sp',
    1:3,
    sep = '')),
  inte = c(1, 15, 1, 1, 1, 1, 1, 1, 1), # (3*3)
  rowe = c(1, 1, 1),
  cole = c(1, 1, 1),

```

```

row1 = list(Row = paste('r',
                        1:3,
                        sep = '')),
coll = list(Col = paste('c',
                        1:3,
                        sep = '')),

round = 1,
type = 'SPE',
design = 'LSD')
summary(split_1sd)

#!-----
#! Quantitative Factor(s) (QT)
#!-----

#! CRD - Orthogonal polynomials
# Linear effect
# Nonsense(error is 0)
# Default contrasts: Orthogonal contrasts
r <- 4 # (repet. number)
fln <- 5 # (factor levels number)
level <- c(0, 10, 20, 30, 40)

crd_lo <- gexp(mu = 1, #in this case, mu=beta0 (intercept)
              r = r,
              fe = list(f1 = c(2, 0, 0, 0)), #b1 #b2 #b3 #b4
              fl = list(Dose = level),
              err = matrix(0,
                          nrow = r*fln),
              round = 2)
crd_lo$X
summary(crd_lo)

plot(Y1 ~ Dose,
     crd_lo$dfm)

# Quadratic effect
crd_qo <- gexp(mu = 2,
              r = r,
              fe = list(f1 = c(0, 3, 0, 0)), #b1 #b2 #b3 #b4
              fl = list(Dose = level),
              err = matrix(0,
                          nrow = r*fln))
summary(crd_qo)
plot(Y1 ~ Dose,
     crd_qo$dfm)

# Cubic effect
crd_co <- gexp(mu = 2,
              r = r,
              fe = list(f1 = c(1, 1, 3, 0)), #b1 #b2 #b3 #b4
              fl = list(Dose = level),
              err = matrix(0,

```

```

                                nrow = r*fln))
summary(crd_co)
plot(Y1 ~ Dose,
      crd_co$dfm)

# Not orthogonal polynomials
# Linear
cont_crd <- matrix(c(level,
                      level^2,
                      level^3,
                      level^4),
                   ncol = 4)

crd_l <- gexp(mu = 2,
             r = 2,
             fe = list(f1 = c(10, 0, 0, 0)), #b1 #b2 #b3 #b4
             fl = list(Dose = level),
             contrasts = list(Dose = cont_crd))

crd_l$X
summary(crd_l)

plot(Y1 ~ Dose,
      crd_l$dfm)

reg <- lm(Y1 ~ Dose + I(Dose^2) + I(Dose^3) + I(Dose^4),
          data = crd_l$dfm)

summary(reg)

# Linear and quadratic
level1 <- seq(0,30,by = 10)
cont_crd1 <- matrix(c(level1,
                      level1^2,
                      level1^3),
                   ncol = 3)

level2 <- 1:4
cont_crd2 <- matrix(c(level2,
                      level2^2,
                      level2^3),
                   ncol = 3)

crd_lq <- gexp(mu = 1,
             r = 2,
             fe = list(f1 = c(10, 0, 0), #b1 #b2 #b3
                      f2 = c(1, 8, 0)),
             fl = list(P = level1,
                      N = level2),
             contrasts = list(N = cont_crd2,
                              P = cont_crd1))

crd_lq$X
summary(crd_lq)

with(crd_lq$dfm,

```

```

    plot(Y1 ~ P))

with(crd_lq$dfm,
     plot(Y1 ~ N))

# Multivariate
crd_m <- gexp(mu = c(2, 10),
             r = 4,
             fe = list(f1 = matrix(c(10, 0, #L   Q
                                   0, 10,
                                   0, 0),
                                   ncol = 2,
                                   byrow = TRUE)),
             f1 = list(Dose = level1),
             contrasts = list(Dose = cont_crd1))

with(crd_m$dfm,
     plot(Y1 ~ Dose))

with(crd_m$dfm,
     plot(Y2 ~ Dose))

# RCBD - Orthogonal polynomials
level3 <- c(0, 2, 4, 6)
rcbd <- gexp(mu = 1,
            fe = list(f1 = c(3, 0, 0)), #b1 #b2 #b3
            blke = c(1, 2, 3),
            r = 2,
            f1 = list(Dose = level3),
            blk1 = list(Blk = c('B1', 'B2', 'B3')),
            design = 'RCBD')

rcbd$X
summary(rcbd)

# Not orthogonal
cont_crd3 <- matrix(c(level3, level3^2, level3^3),
                   ncol = 3)

rcbd_01 <- gexp(mu = 1,
               fe = list(f1 = c(3, 0, 0)), #b1 #b2 #b3
               blke = c(1, 2, 3),
               r = 2,
               f1 = list(Dose = level3),
               blk1 = list(Blk = c('B1', 'B2', 'B3')),
               contrasts = list(Dose = cont_crd3),
               design = 'RCBD')

rcbd_01$X
summary(rcbd_01)

# Orthogonal polynomials - LSD
lsd <- gexp(mu = 1,
            r = 1,
            fe = list(f1 = c(3, 0, 0)), #b1 #b2 #b3

```

```

        rowe = rep(1, 4),
        cole = rep(1, 4),
        fl = list(Dose = level1),
        design = 'LSD')

lsd$X
summary(lsd)

lsd_01 <- gexp(mu = 1,
              r = 1,
              fe = list(f1 = c(3, 0, 0)), #b1 #b2 #b3
              rowe = rep(1, 4),
              cole = rep(1, 4),
              rowl = list(row = paste('r',
                                      1:4,
                                      sep = ' ')),
              fl = list(Dose = level1),
              design = 'LSD')

lsd_01$X
summary(lsd_01)

# Not orthogonal
lsd_02 <- gexp(mu = 1,
              r = 1,
              fe = list(f1 = c(3, 0, 0)), #b1 #b2 #b3
              rowe = rep(1, 4),
              cole = rep(1, 4),
              fl = list(Dose = level3),
              contrasts = list(Dose = cont_crd3),
              design = 'LSD')

lsd_02$X
str(lsd_02)

#!-----
#! Hibrid: qualitative and quantitative factors in the same experiment - HB
#!-----
#! CRD - HB
r <- 2 # (repet. number)
f11 <- 4# (first factor levels number)
f12 <- 3# (second factor levels number)

crd_hb <- gexp(mu = 1, #in this case, mu=beta0 (intercept)
              r = r,
              fe = list(f1 = c(2, 0, 0), #b1 #b2 #b3
                        f2 = c(1, 1, 3)),
              fl = list(Dose = seq(0,30,
                                   by = 10),
                        Trat = LETTERS[1:3]),
              err = matrix(0,
                           nrow = r*f11*f12),
              round = 2)

crd_hb$X
summary(crd_hb)

```

```

#Only one contrasts!
crd_hb2 <- gexp(mu = 1, #in this case, mu=beta0 (intercept)
              r = r,
              fe = list(f1 = c(2, 0, 0), #b1 #b2 #b3
                       f2 = c(1, 1, 3)),
              fl = list(Dose = level1,
                       Trat = LETTERS[1:3]),
              err = matrix(0,
                          nrow = r*f1*f12),
              contrasts = list(Dose = cont_crd1),
              round = 2)

crd_hb2$X
summary(crd_hb)

#! RCBD - HB
r <- 2
blke <- c(1, 2)
level <- c(0, 10, 20, 30)
(error <- matrix(rep(0,
                    4^1*3^1*r*length(blke)),
                ncol=1))

rcbd_hb <- gexp(mu = 2,
              err = error,
              r = r,
              fe = list(f1 = c(0, 1, 0), # Qualitative
                       f2 = c(1, 0, 0)), # Quantitative linear
              fl = list(Var = LETTERS[1:3],
                       Dose = level),
              blke = blke,
              blk1 = list(Blk = c('B1', 'B2')),
              design = 'RCBD')

rcbd_hb$X
summary(rcbd_hb)
str(rcbd_hb)

#! LSD - QT
set.seed(3)
lsd <- gexp(mu = 100,
           r = 1,
           fe = list(f1 = c(10, # b1
                          20, # b2
                          0, # b3
                          0)), # b4
           fl = list(tra = seq(0,
                              40,
                              by = 10)),
           rowe = c(1, 2, 3, 4, 5),
           rowl = list(row = paste('r',
                                   1:5,
                                   sep = '')),
           cole = c(5, 4, 3, 2, 1),
           coll = list(col = paste('c',

```

```

                                1:5,
                                sep = '')),
                                design = 'LSD')
summary(lsd)
plot(Y1 ~ tra, lsd$dfm)

#! FE - LSD - QT
fe_1sd <- gexp(mu = 10,
              fe = list(f1 = c(2, 3),
                       f2 = c(5, # b1*
                              0, # b2
                              0, # b3
                              0)), # b4
              rowe = rep(1, 10),
              cole = rep(1, 10),
              f1 = list(var = paste('v',
                                    1:2,
                                    sep = '')),
              tra = seq(0,
                       40,
                       by = 10)),
              coll = list(col = paste('c',
                                      1:10,
                                      sep = '')),
              rowl = list(row = paste('r',
                                      1:10,
                                      sep = '')),
              type = 'FE',
              design = 'LSD')

fe_1sd$X
str(fe_1sd)
summary(fe_1sd)
plot(Y1 ~ tra,
      fe_1sd$dfm)

#! SPE - QL - QT
spe_1sd <- gexp(mu = 100,
               r = 1,
               fe = list(f1 = c(2, 3, 1),
                        f2 = c(1, # b1
                              5, # b2*
                              1)), # b3
               f1 = list(p = paste('p',
                                    1:3,
                                    sep = '')),
               sp = seq(0,
                       30,
                       by = 10)),
               rowe = c(1, 2, 3),
               cole = c(3, 2, 1),
               rowl = list(row = paste('r',
                                       1:3,
                                       sep = '')),

```



```

coll = list(col = paste('c',
                        1:3,
                        sep = ')),

round = 1,
type = 'SPE',
design = 'LSD')
summary(spe_1sd)
plot(spe_1sd)

#####
# MULTIVARIATE APPROACH #
#####
#! CRD - QL
# Error = 0 - Nonsense (you can easily understand the effects)
r <- 2 # (repet. number)
fln <- 3 # (factor levels number)

crd_m01 <- gexp(mu = c(0,10),
              r = r,
              fe = list(f1 = matrix(c(1, 0, #Y1 Y2
                                     2, 1,
                                     3, 3),
                                   ncol = 2,
                                   byrow = TRUE)),
              err = mvtnorm::rmvnorm(n = fln * r,
                                     sigma = matrix(c(0, 0,
                                                     0, 0),
                                                  ncol = 2)),
              round = 0)
summary(crd_m01)

#! FE - CRD - QL
r <- 2
crd_m_fe01 <- gexp(mu = c(0, 0),
                  r = r,
                  err = mvtnorm::rmvnorm(n = 3^1 * 2^1 * r,
                                          sigma = matrix(c(0, 0,
                                                          0, 0),
                                                       ncol = 2)),
                  fe = list(f1 = matrix(c(0, 3, #X1 X1
                                          1, 4, #X2 X2
                                          2, 5), #X3 X3
                                        ncol = 2,
                                        byrow = TRUE),
                            f2 = matrix(c(0, 2, #X1 X1
                                          1, 3), #X2 X2
                                        ncol = 2,
                                        byrow = TRUE)),
                  type = 'FE',
                  round = 1)
summary(crd_m_fe01)

```

```

#! FE - CRD - QL
# Using default error
set.seed(30)
crd_m_fe02 <- gexp(mu = c(0, 2),
                  r = 3,
                  fe = list(f1 = matrix(c(1, 1,
                                          5, 1,
                                          1, 1),
                                       ncol = 2,
                                       byrow = TRUE),
                            f2 = matrix(c(1, 3,
                                          2, 2),
                                       ncol = 2,
                                       byrow = TRUE)),
                  type = 'FE',
                  round = 1)
summary(crd_m_fe02)

#! SPE - CRD - QL
# Using default error
crd_m_spe01 <- gexp(mu = c(0, 2),
                   r = 3,
                   fe = list(f1 = matrix(c(1, 1,
                                           5, 1,
                                           1, 1),
                                        ncol = 2,
                                        byrow = TRUE),
                              f2 = matrix(c(1, 3,
                                           2, 2),
                                        ncol = 2,
                                        byrow = TRUE)),
                   type = 'SPE',
                   round = 1)
summary(crd_m_spe01)

#! RCBD - QL
r <- 2 # (repet. number)
fln <- 3 # (factor levels number)
bln <- 3 # (block levels number)

rcbd_m01 <- gexp(mu = c(0,10),
                 r = r,
                 fe = list(f1 = matrix(c(1, 0, #Y1 Y2
                                         2, 1,
                                         3, 3),
                                       ncol = 2,
                                       byrow = TRUE)),
                 blke = matrix(c(2, 1,
                                 4, 1,
                                 6, 1),
                               ncol = 2,
                               byrow = TRUE),
                 err = mvtnorm::rmvnorm(n = fln * r * bln,

```

```

                                sigma = matrix(c(0, 0,
                                                    0, 0),
                                                    ncol = 2)),
                                design = 'RCBD',
                                round = 0)
summary(rcbd_m01)

#! FE - RCBD - QL
rcbd_m_fe01 <- gexp(mu = c(0, 0),
                   r = r,
                   err = mvtnorm::rmvnorm(n = 3^1 * 2^1 * r * bln,
                                           sigma = matrix(c(0, 0,
                                                           0, 0),
                                                           ncol = 2)),
                   fe = list(f1 = matrix(c(0, 3, #X1 X1
                                           1, 4, #X2 X2
                                           2, 5), #X3 X3
                                           ncol = 2,
                                           byrow = TRUE),
                              f2 = matrix(c(0, 2, #X1 X1
                                           1, 3), #X2 X2
                                           ncol = 2,
                                           byrow = TRUE)),
                   blke = matrix(c(2, 1,
                                    4, 1,
                                    6, 1),
                                   ncol = 2,
                                   byrow = TRUE),
                   type = 'FE',
                   design = 'RCBD',
                   round = 1)
summary(rcbd_m_fe01)

#! SPE - RCBD - QL
rcbd_m_spe01 <- gexp(mu = c(0, 2),
                    r = 2,
                    fe = list(f1 = matrix(c(1, 1,
                                             5, 1,
                                             1, 1),
                                             ncol = 2,
                                             byrow = TRUE),
                              f2 = matrix(c(1, 3,
                                             2, 2),
                                             ncol = 2,
                                             byrow = TRUE),
                              f3 = matrix(c(1, 3,
                                             2, 2),
                                             ncol = 2,
                                             byrow = TRUE)),
                    blke = matrix(c(2, 1,
                                    4, 1,
                                    6, 1),
                                   ncol = 2,
                                   byrow = TRUE),
                    type = 'SPE',
                    design = 'RCBD',
                    round = 1)
summary(rcbd_m_spe01)

```

```

                                byrow = TRUE),
                                type = 'SPE',
                                design = 'RCBD',
                                round = 1)
summary(rcbd_m_spe01)

#! LSD - QL
lsd_m01 <- gexp(mu = c(0,10),
               r = 1,
               fe = list(f1 = matrix(c(1, 0,
                                       2, 1,
                                       3, 3),
                                       ncol = 2,
                                       byrow = TRUE)),
               rowe = matrix(rep(1, 6),
                              ncol = 2),
               cole = matrix(rep(1, 6),
                              ncol = 2),
               err = mvtnorm::rmvnorm(n = 3^2,
                                       sigma = matrix(c(0, 0,
                                                       0, 0),
                                                       ncol = 2)),
               design = 'LSD',
               round = 0)
summary(lsd_m01)

#! LSD/FE - QL
lsd_m_fe01 <- gexp(mu = c(0, 0),
                  r = 1,
                  err = mvtnorm::rmvnorm(n = 3^1 * 2^1 * 6,
                                          sigma = matrix(c(0, 0,
                                                          0, 0),
                                                          ncol = 2)),
                  #Y1 Y2
                  fe = list(f1 = matrix(c(0, 3, #X1 X1
                                          1, 4, #X2 X2
                                          2, 5), #X3 X3
                                          ncol = 2,
                                          byrow = TRUE),
                              #Y1 Y2
                              f2 = matrix(c(0, 2, #X1 X1
                                              1, 3), #X2 X2
                                              ncol = 2,
                                              byrow = TRUE)),
                  rowe = matrix(rep(1, 12),
                                ncol = 2),
                  cole = matrix(rep(1, 12),
                                ncol = 2),
                  type = 'FE',
                  design = 'LSD',
                  round = 1)
summary(lsd_m_fe01)

```

```

#! SPE - LSD - QL
# Using default error
lsd_m_spe01 <- gexp(mu = c(0, 2),
  r = 1,
  fe = list(f1 = matrix(c(1, 1,
    5, 1,
    1, 1),
    ncol = 2,
    byrow = TRUE),
    f2 = matrix(c(1, 3,
    2, 2),
    ncol = 2,
    byrow = TRUE)),
  rowe = matrix(rep(1, 6),
    ncol = 2),
  cole = matrix(rep(1, 6),
    ncol = 2),
  type = 'SPE',
  design = 'LSD',
  round = 1)
summary(lsd_m_spe01)

#! FE - RCBD - QL
r = 1
bln = 3
fe_rcbd_m <- gexp(mu = c(0, 0),
  r = 1,
  err = mvtnorm::rmvnorm(n = 3^1 * 2^1 * r * bln,
    sigma = matrix(c(0, 0,
    0, 0),
    ncol = 2)),
  fe = list(f1 = matrix(c(0, 3, #X1 X1
    1, 4, #X2 X2
    2, 5), #X3 X3
    ncol = 2,
    byrow = TRUE),
    f2 = matrix(c(0, 2, #X1 X1
    1, 3), #X2 X2
    ncol = 2,
    byrow = TRUE)),
  blke = matrix(c(2, 1,
    4, 1,
    6, 1),
    ncol = 2,
    byrow = TRUE),
  type = 'FE',
  design = 'RCBD')
str(fe_rcbd_m)
summary(fe_rcbd_m)

#! SPE - RCBD - QL
spe_rcbd_m <- gexp(mu = c(0, 2),
  r = 3,

```

```

fe = list(f1 = matrix(c(1, 1,
                       5, 1,
                       1, 1),
                    ncol = 2,
                    byrow = TRUE),
          f2 = matrix(c(1, 3,
                       2, 2),
                    ncol = 2,
                    byrow = TRUE),
          f3 = matrix(c(1, 3,
                       2, 2),
                    ncol = 2,
                    byrow = TRUE)),
blke = matrix(c(2, 1,
                4, 1,
                6, 1),
              ncol = 2,
              byrow = TRUE),
type = 'SPE',
design = 'RCBD')
str(spe_rcbd_m)
summary(spe_rcbd_m)

```

---

plot

*Plot methods for gexp objects*


---

### Description

These are methods for objects of class `gexp.simple_crd` - Completely Randomized Design (CRD), `gexp.simple_rcbd` - Randomized Complete Block Design (RCBD), `gexp.simple_lsd` - Latin Squares Design (LSD), `gexp.fe_crd` - Factorial Experiment (FE) to CRD, `gexp.fe_rcbd` - FE to RCBD and `gexp.fe_lsd` - FE to LSD and `gexp.spe_crd` - Split-plot Experiment (SPE) to CRD, `gexp.spe_rcbd` - SPE to RCBD, `gexp.spe_lsd` - SPE to LSD. The main objective of these methods is to produce an experimental croqui with randomized treatments according with the design and type of experiment. It can be very useful in experiment planning.

### Usage

```

## S3 method for class 'gexp.simple_crd'
plot(x,
     main      = NULL,
     sub       = NULL,
     colgrid  = 'red',
     coltext  = 'blue',
     ltygrid  = 'dotted',
     lwdgrid  = par('lwd'),
     xleftimg = par()$usr[1],
     ybottomimg = par()$usr[3],
     xrightimg = par()$usr[2],

```

```
    ytopimg = par()$usr[4],
    dynamic = FALSE,
    random = TRUE, ...)

## S3 method for class 'gexp.simple_rcbd'
plot(x,
     main = NULL,
     sub = NULL,
     colgrid = 'red',
     coltext = 'blue',
     ltygrid = 'dotted',
     lwdgrid = par('lwd'),
     xleftimg = par()$usr[1],
     ybottomimg = par()$usr[3],
     xrightimg = par()$usr[2],
     ytopimg = par()$usr[4],
     dynamic = FALSE,
     random = TRUE, ...)

## S3 method for class 'gexp.simple_lsd'
plot(x,
     main = NULL,
     sub = NULL,
     colgrid = 'red',
     coltext = 'blue',
     ltygrid = 'dotted',
     lwdgrid = par('lwd'),
     xleftimg = par()$usr[1],
     ybottomimg = par()$usr[3],
     xrightimg = par()$usr[2],
     ytopimg = par()$usr[4],
     dynamic = FALSE,
     random = TRUE, ...)

## S3 method for class 'gexp.fe_crd'
plot(x,
     main = NULL,
     sub = NULL,
     colgrid = 'red',
     coltext = 'blue',
     ltygrid = 'dotted',
     lwdgrid = par('lwd'),
     xleftimg = par()$usr[1],
     ybottomimg = par()$usr[3],
     xrightimg = par()$usr[2],
     ytopimg = par()$usr[4],
     dynamic = FALSE,
     random = TRUE, ...)
```

```
## S3 method for class 'gexp.fe_rcbd'
```

```
plot(x,  
      main      = NULL,  
      sub       = NULL,  
      colgrid   = 'red',  
      coltext   = 'blue',  
      ltygrid   = 'dotted',  
      lwdgrid   = par('lwd'),  
      xleftimg  = par()$usr[1],  
      ybottomimg = par()$usr[3],  
      xrightimg = par()$usr[2],  
      ytopimg   = par()$usr[4],  
      dynamic   = FALSE,  
      random    = TRUE, ...)
```

```
## S3 method for class 'gexp.fe_lsd'
```

```
plot(x,  
      main      = NULL,  
      sub       = NULL,  
      colgrid   = 'red',  
      coltext   = 'blue',  
      ltygrid   = 'dotted',  
      lwdgrid   = par('lwd'),  
      xleftimg  = par()$usr[1],  
      ybottomimg = par()$usr[3],  
      xrightimg = par()$usr[2],  
      ytopimg   = par()$usr[4],  
      dynamic   = FALSE,  
      random    = TRUE, ...)
```

```
## S3 method for class 'gexp.spe_crd'
```

```
plot(x,  
      main      = NULL,  
      sub       = NULL,  
      colgrid   = 'red',  
      coltext   = 'blue',  
      srttext   = 30,  
      ltygrid   = 'dotted',  
      lwdgrid   = par('lwd'),  
      xleftimg  = par()$usr[1],  
      ybottomimg = par()$usr[3],  
      xrightimg = par()$usr[2],  
      ytopimg   = par()$usr[4],  
      dynamic   = FALSE,  
      random    = TRUE, ...)
```

```
## S3 method for class 'gexp.spe_rcbd'
```



```

plot(x,
      main      = NULL,
      sub       = NULL,
      colgrid   = 'red',
      coltext   = 'blue',
      srttext   = 30,
      ltygrid   = 'dotted',
      lwdgrid   = par('lwd'),
      xleftimg  = par()$usr[1],
      ybottomimg = par()$usr[3],
      xrightimg = par()$usr[2],
      ytopimg   = par()$usr[4],
      dynamic   = FALSE,
      random    = TRUE, ...)

## S3 method for class 'gexp.spe_1sd'
plot(x,
      main      = NULL,
      sub       = NULL,
      colgrid   = 'red',
      coltext   = 'blue',
      srttext   = 30,
      ltygrid   = 'dotted',
      lwdgrid   = par('lwd'),
      xleftimg  = par()$usr[1],
      ybottomimg = par()$usr[3],
      xrightimg = par()$usr[2],
      ytopimg   = par()$usr[4],
      dynamic   = FALSE,
      random    = TRUE, ...)

```

### Arguments

<code>x</code>	A class <code>gexp</code> object.
<code>main</code>	An overall title for the plot.
<code>sub</code>	An sub title for the plot.
<code>coltext</code>	A color to the text on the plot.
<code>srttext</code>	The string rotation in degrees. See <code>srt</code> argument of the <code>par</code> function.
<code>colgrid</code>	A color to the grid on the plot.
<code>ltygrid</code>	A <code>lty</code> to the grid on the plot.
<code>lwdgrid</code>	A <code>lwd</code> to the grid on the plot.
<code>xleftimg</code>	A vector (or scalar) of left x positions.
<code>ybottomimg</code>	A vector (or scalar) of bottom y positions.
<code>xrightimg</code>	A vector (or scalar) of right x positions.
<code>ytopimg</code>	A vector (or scalar) of top y positions.

dynamic	A logical argument to plot experimental design using image.
random	It is a logical argument when the purpose is to plan experiments so that randomisation of treatments occurs in the experimental units. TRUE is the default.
...	Further arguments (required by generic).

**See Also**

[plot.default](#), [rasterImage](#)

**Examples**

```

#! CRD
crd <- gexp()
plot(crd) # Default

plot(crd, # Changing some arguments
      main = '',
      sub = '',
      coltext = 'black',
      colgrid = 'darkred',
      ltygrid = 'solid',
      lwdgrid = 3)

#! FE - CRD
crd_fe <- gexp(type = 'FE')
plot(crd_fe)

#! SPE - CRD
split_crd <- gexp(type = 'SPE')
plot(split_crd)

#! RCBD
rcbd <- gexp(design = 'RCBD')
plot(rcbd)

#! FE - RCBD
fe_rcbd <- gexp(r = 2,
               type = 'FE',
               design = 'RCBD')
plot(fe_rcbd)

fe_rcbd1 <- gexp(r = 1,
                blke = c(1, 2),
                blk1 = list(Blk = c('B1', 'B2')),
                type = 'FE',
                design = 'RCBD')
plot(fe_rcbd1)

#! SPE - RCBD
split_rcbd <- gexp(r = 2,
                  type = 'SPE',
                  design = 'RCBD')

```

```

plot(split_rcbd)

#! LSD
lsd <- gexp(r = 1,
           design = 'LSD')
plot(lsd)

#! FE - LSD
fe_lsd <- gexp(r = 1,
              fe = list(f1 = rep(1, 2),
                       f2 = rep(1, 2)),
              rowe = c(1, 3, 2, 1),
              cole = c(2, 2, 1, 1),
              rowl = list(Row = paste('r',
                                      1:4,
                                      sep = '')),
              coll = list(Col = paste('c',
                                      1:4,
                                      sep = '')),
              type = 'FE',
              design = 'LSD')
plot(fe_lsd)

#! SPE - LSD
split_lsd <- gexp(r = 1,
                 type = 'SPE',
                 design = 'LSD')
plot(split_lsd)

## Not run:
#! Using images in plannig
# CRD
# Dynamic
# Open picture 'crd.jpg' when requested
crd_i <- gexp(r = 3,
             fe = list(f1 = c(1, 1)),
             f1 = list(Treat = LETTERS[1:2]))
plot(crd_i) # Default

# dynamic plot require 'jpeg' package!
plot(crd_i,
     dynamic = TRUE,
     xleftimg = 0.6,
     ybottomimg = .6,
     xrightimg = 1.4,
     ytopimg = 1.4)

# RCBD
rcbd_i <- gexp(fe = list(f1 = c(5, 1, 1)),
              f1 = list(TR = LETTERS[1:3]),
              blke = c(1, 2, 3),
              blk1 = list(BLK = paste('B',
                                      1:3),

```

```

                                sep = ')),
    design = 'RCBD')

plot(rcbd_i,
     dynamic=TRUE)

# LSD
lsd <- gexp(r = 1,
           fe = list(f1 = c(1, 1, 10)),
           rowe = c(1, 1, 1),
           cole = c(1, 1, 1),
           rowl = list(Row = paste('r',
                                   1:3,
                                   sep = ')),
           coll = list(Col = paste('c',
                                   1:3,
                                   sep = ')),
           design = 'LSD')

#TODO: dynamic plot require 'png' package!
plot(lsd,
     dynamic=TRUE)

## End(Not run)

```

---

print

*Print for gexp objects.*


---

## Description

Print gexp objects.

## Usage

```

## S3 method for class 'gexp'
print(x,
      digits=3L, ...)

```

## Arguments

x	A class gexp object.
digits	Number of decimal digits in the results. The default is 3.
...	Further arguments (required by generic).

## See Also

[gexp](#)

**Examples**

```

#! CRD
#! 1 factor - CRD
crd <- gexp(mu = 1,
           r = 3,
           fe = list(f1 = c(1, 1, 5, 1, 1)),
           fl = list(Treat = LETTERS[1:5]),
           round = 2)

print(crd)
crd

#! FE - CRD
# 5 factors (f1..f5, level^factor): 3^1 * 2^1 * 3^1 * 2^1 * 4^1 * 2 = 360 experimental units
# 5 factors (f1..f5, level^factor): 3^2 * 2^2 * 5^1 = 180 * 2 = 360 experimental units
crd_fe <- gexp(mu = 0,
              r = 2,
              fe = list(f1 = c(1, 1, 5),
                        f2 = c(1, 1),
                        f3 = c(2, 2, 1),
                        f4 = c(1, 5),
                        f5 = c(1, 2, 3, 4, 5)),
              fl = list(A = paste('a',
                                   1:3,
                                   sep = ''),
                        B = paste('b',
                                   1:2,
                                   sep = ''),
                        C = paste('c',
                                   1:3,
                                   sep = ''),
                        D = paste('d',
                                   1:2,
                                   sep = ''),
                        E = paste('e',
                                   1:5,
                                   sep = '')),
              round = 0,
              type = 'FE')

print(crd_fe)
crd_fe

#! SPE - CRD
split_crd <- gexp(mu = 30,
                 fe = list(f1 = c(1, 1),
                           f2 = c(2, 3)),
                 fl = list(P = paste('p',
                                       1:2,
                                       sep = ''),
                           SP = paste('sp',
                                       1:2,
                                       sep = '')),
                 inte = c(1, 15, 1, 1),

```

```

        round = 1,
        type = 'SPE')
print(split_crd)
split_crd

#! RCBD
# 1 factor, 3 blocks
rcbd <- gexp(mu = 0,
            fe = list(f1 = c(5, 1, 1)),
            fl = list(TR = LETTERS[1:3]),
            blke = c(1, 2, 3),
            blk1 = list(BLK = paste('B',
                                   1:3,
                                   sep = '')),
            round = 1,
            design = 'RCBD')
print(rcbd)
rcbd

#! FE - RCBD
fe_rcbd <- gexp(mu = 30,
               fe = list(f1 = c(1, 1, 1),
                       f2 = c(2, 3)),
               blke = c(1, 3),
               round = 1,
               type = 'FE',
               design = 'RCBD')
print(fe_rcbd)
fe_rcbd

#! SPE - RCBD
split_rcbd <- gexp(mu = 30,
                  fe = list(f1 = c(1, 1),
                          f2 = c(2, 3),
                          f3 = c(1, 1, 1)),
                  fl = list(A = paste('a',
                                      1:2,
                                      sep = ''),
                          B = paste('b',
                                      1:2,
                                      sep = ''),
                          C = paste('c',
                                      1:3,
                                      sep = '')),
                  blke = c(1, 2),
                  blk1 = list(BLK = paste('B',
                                          1:2,
                                          sep = '')),
                  inte = c(1, 15, 1, 1, 1, 3, 4, 2, 1, 1, 4, 1,
                          1, 2, 1, 1,
                          1, 1, 1, 1, 1, 1,
                          1, 1, 3, 3, 3, 3),
                  round = 1,

```

```

                                type = 'SPE',
                                design = 'RCBD')
print(split_rcbd)
split_rcbd

#! LSD
lsd <- gexp(mu = 30,
            r = 1,
            fe = list(f1 = c(1, 1, 10)),
            rowe = c(1, 1, 1),
            cole = c(1, 1, 1),
            rowl = list(Row = paste('r',
                                    1:3,
                                    sep = '')),
            coll = list(Col = paste('c',
                                    1:3,
                                    sep = '')),
            round = 1,
            design = 'LSD')
print(lsd)
lsd

#! FE - LSD
fe_lsd <- gexp(mu = 30,
               r = 1,
               fe = list(f1 = c(1, 1),
                           f2 = c(2, 3)),
               rowe = c(1, 3, 2, 1),
               cole = c(2, 2, 1, 1),
               rowl = list(Row = paste('r',
                                       1:4,
                                       sep = '')),
               coll = list(Col = paste('c',
                                       1:4,
                                       sep = '')),
               inte = c(1, 15, 1, 1),
               round = 1,
               type = 'FE',
               design = 'LSD')
print(fe_lsd)
fe_lsd

#! SPE - LSD
split_lsd <- gexp(mu = 30,
                  r = 1,
                  fe = list(f1 = c(1, 1, 2),
                              f2 = c(2, 3, 1)),
                  fl = list(P = paste('p',
                                       1:3,
                                       sep = '')),
                  SP = paste('sp',
                              1:3,
                              sep = '')),

```

```

inte = c(1, 15, 1, 1, 1, 1, 1, 1, 1),
rowe = c(1, 1, 1),
cole = c(1, 1, 1),
rowl = list(Row = paste('r',
                        1:3,
                        sep = '')),
coll = list(Col = paste('c',
                        1:3,
                        sep = '')),
round = 1,
type = 'SPE',
design = 'LSD')
print(split_1sd)
split_1sd

```

---

summary

*Summary gexp objects.*


---

## Description

Summary gexp objects.

## Usage

```

## S3 method for class 'gexp'
summary(object,
        digits=3L, ...)

```

## Arguments

object	A class gexp object.
digits	Number of decimal digits in the results. The default is 3.
...	Further arguments (required by generic).

## Examples

```

#! CRD
#! 1 factor - CRD
crd <- gexp(mu = 1,
           r = 3,
           fe = list(f1 = c(1, 1, 5, 1, 1)),
           fl = list(Treat = LETTERS[1:5]),
           round = 2)
summary(crd)

#! FE - CRD
# 5 factors (f1..f5, level^factor): 3^1 * 2^1 * 3^1 * 2^1 * 4^1 * 2 = 360 experimental units
# 5 factors (f1..f5, level^factor): 3^2 * 2^2 * 5^1 = 180 * 2 = 360 experimental units
crd_fe <- gexp(mu = 0,

```



```

r = 2,
fe = list(f1 = c(1, 1, 5),
          f2 = c(1, 1),
          f3 = c(2, 2, 1),
          f4 = c(1, 5),
          f5 = c(1, 2, 3, 4, 5)),
fl = list(A = paste('a',
                    1:3,
                    sep = ''),
          B = paste('b',
                    1:2,
                    sep = ''),
          C = paste('c',
                    1:3,
                    sep = ''),
          D = paste('d',
                    1:2,
                    sep = ''),
          E = paste('e',
                    1:5,
                    sep = '')),
round = 0,
type = 'FE')
summary(crd_fe)

#! SPE - CRD
split_crd <- gexp(mu = 30,
                  fe = list(f1 = c(1, 1),
                            f2 = c(2, 3)),
                  fl = list(P = paste('p',
                                      1:2,
                                      sep = ''),
                            SP = paste('sp',
                                       1:2,
                                       sep = '')),
                  inte = c(1, 15, 1, 1),
                  round = 1,
                  type = 'SPE')
summary(split_crd)

#! RCBD
# 1 factor, 3 blocks
rcbd <- gexp(mu = 0,
             fe = list(f1 = c(5, 1, 1)),
             fl = list(TR = LETTERS[1:3]),
             blke = c(1, 2, 3),
             blk1 = list(BLK = paste('B',
                                     1:3,
                                     sep = '')),
             round = 1,
             design = 'RCBD')
summary(rcbd)

```

```

#! FE - RCBD
fe_rcbd <- gexp(mu = 30,
  fe = list(f1 = c(1, 1, 1),
            f2 = c(2, 3)),
  blke = c(1, 3),
  round = 1,
  type = 'FE',
  design = 'RCBD')
summary(fe_rcbd)

#! SPE - RCBD
split_rcbd <- gexp(mu = 30,
  fe = list(f1 = c(1, 1),
            f2 = c(2, 3),
            f3 = c(1, 1, 1)),
  fl = list(A = paste('a',
                      1:2,
                      sep = '')),
        B = paste('b',
                  1:2,
                  sep = '')),
        C = paste('c',
                  1:3,
                  sep = '')),
  blke = c(1, 2),
  blk1 = list(BLK = paste('B',
                          1:2,
                          sep = '')),
  inte = c(1, 15, 1, 1, 1, 3, 4, 2, 1, 1, 4, 1,
            1, 2, 1, 1,
            1, 1, 1, 1, 1, 1,
            1, 1, 3, 3, 3, 3),
  round = 1,
  type = 'SPE',
  design = 'RCBD')
summary(split_rcbd)

#! LSD
lsd <- gexp(mu = 30,
  r = 1,
  fe = list(f1 = c(1, 1, 10)),
  rowe = c(1, 1, 1),
  cole = c(1, 1, 1),
  row1 = list(Row = paste('r',
                           1:3,
                           sep = '')),
  coll = list(Col = paste('c',
                           1:3,
                           sep = '')),
  round = 1,
  design = 'LSD')
summary(lsd)

```

```
#! FE - LSD
fe_ksd <- gexp(mu = 30,
  r = 1,
  fe = list(f1 = c(1, 1),
            f2 = c(2, 3)),
  rowe = c(1, 3, 2, 1),
  cole = c(2, 2, 1, 1),
  row1 = list(Row = paste('r',
                           1:4,
                           sep = '')),
  coll = list(Col = paste('c',
                           1:4,
                           sep = '')),
  inte = c(1, 15, 1, 1),
  round = 1,
  type = 'FE',
  design = 'LSD')
summary(fe_ksd)

#! SPE - LSD
split_ksd <- gexp(mu = 30,
  r = 1,
  fe = list(f1 = c(1, 1, 2),
            f2 = c(2, 3, 1)),
  fl = list(P = paste('p',
                       1:3,
                       sep = ''),
            SP = paste('sp',
                       1:3,
                       sep = '')),
  inte = c(1, 15, 1, 1, 1, 1, 1, 1, 1),
  rowe = c(1, 1, 1),
  cole = c(1, 1, 1),
  row1 = list(Row = paste('r',
                           1:3,
                           sep = '')),
  coll = list(Col = paste('c',
                           1:3,
                           sep = '')),
  round = 1,
  type = 'SPE',
  design = 'LSD')
summary(split_ksd)
```

# Index

Generator of Experiments, [2](#)

`gexp`, [2](#), [28](#)

`gexp-package` (Generator of Experiments), [2](#)

`plot`, [22](#)

`plot.default`, [26](#)

`plot.gexp.simple_crd`, [5](#)

`print`, [28](#)

`rasterImage`, [26](#)

`summary`, [32](#)