Package ‘baystability’

October 15, 2024

Type Package

Title Bayesian Stability Analysis of Genotype by Environment Interaction (GEI)

Version 0.2.0

Maintainer Muhammad Yaseen <`myaseen208@gmail.com`>

Description Performs general Bayesian estimation method of linear–bilinear models for genotype × environment interaction. The method is explained in Perez-Elizalde, S., Jarquin, D., and Crossa, J. (2011) (<doi:10.1007/s13253-011-0063-9>).

Depends R (>= 3.1)

Imports dplyr, ggplot2, lme4, MASS, rstiefel, rlang, scales, stats, tibble, tidyr

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

NeedsCompilation no

Author Muhammad Yaseen [aut, cre],
Diego Jarquin [aut, ctb],
Sergio Perez-Elizalde [aut, ctb],
Juan Burgueño [aut, ctb],
Jose Crossa [aut, ctb]

Repository CRAN

Date/Publication 2024-10-15 04:40:03 UTC

Contents

`bayes_ammi` *Bayesian Estimation of Genotype by Environment Interaction Model*

Description

Bayesian estimation method of linear–bilinear models for Genotype by Environment Interaction Model

Usage

```
## Default S3 method:
bayes_ammi(.data, .y, .gen, .env, .rep, .nIter)
```

Arguments

`.data`	data.frame
`.y`	Response Variable
`.gen`	Genotypes Factor
`.env`	Environment Factor
`.rep`	Replication Factor
`.nIter`	Number of Iterations

Value

Genotype by Environment Interaction Model

Author(s)

1. Muhammad Yaseen (<`myaseen208@gmail.com`>)
2. Diego Jarquin (<`diego.jarquin@gmail.com`>)
3. Sergio Perez-Elizalde (<`sergiop@colpos.mx`>)
4. Juan Burgueño (<`j.burgueno@cgiar.org`>)
5. Jose Crossa (<`j.crossa@cgiar.org`>)

References

Perez-Elizalde, S., Jarquin, D., and Crossa, J. (2011) A General Bayesian Estimation Method of Linear–Bilinear Models Applied to Plant Breeding Trials With Genotype × Environment Interaction. *Journal of Agricultural, Biological, and Environmental Statistics*, 17, 15–37. (doi:10.1007/s13253-011-0063-9)

Examples

```
## Not run:
library(baystability)
library(dplyr)
data(cultivo2008)
fm1 <-
   ge_ammi(
      .data  = cultivo2008
     , .y    = y
     , .gen  = entry
     , .env  = site
     , .rep  = rep
     )

r0 <- fm1$g
c0 <- fm1$e
n0 <- fm1$Rep
k0 <- fm1$k

mu0      <- fm1$mu
sigma20  <- fm1$sigma2
tau0     <- fm1$tau
tao0     <- fm1$tao
delta0   <- fm1$delta
lambdas0 <- fm1$lambdas
alphas0  <- fm1$alphas
gammas0  <- fm1$gammas

ge_means0 <- fm1$ge_means$ge_means

data(cultivo2008)

fm2 <-
 ge_ammi(
   .data = cultivo2009
   , .y    = y
   , .gen  = entry
   , .env  = site
   , .rep  = rep
  )

k        <- fm2$k
alphasa  <- fm2$alphas
gammasa  <- fm2$gammas
```

```
alphas1  <-
           tibble::as_tibble(fm2$alphas, .name_repair = "unique") %>%
           setNames(paste0("V", 1:ncol(.)))
gammas1  <-
           tibble::as_tibble(fm2$gammas, .name_repair = "unique") %>%
           setNames(paste0("V", 1:ncol(.)))


# Biplots OLS
library(ggplot2)
    BiplotOLS1 <-
      ggplot(data = alphas1, mapping = aes(x = V1, y = V2)) +
      geom_point() +
      geom_hline(yintercept = 0) +
      geom_vline(xintercept = 0) +
      geom_text(aes(label = 1:nrow(alphas1)), vjust = "inward", hjust = "inward") +
      scale_x_continuous(
               limits = c(-max(abs(c(range(alphas1[, 1:2]))))
                         , max(abs(c(range(alphas1[, 1:2])))))) +
      scale_y_continuous(
               limits = c(-max(abs(c(range(alphas1[, 1:2]))))
                         , max(abs(c(range(alphas1[, 1:2])))))) +
      labs(title = "OLS", x = expression(u[1]), y = expression(u[2])) +
      theme_bw() +
      theme(plot.title = element_text(hjust = 0.5))
      print(BiplotOLS1)


    BiplotOLS2 <-
      ggplot(data = gammas1, mapping = aes(x = V1, y = V2)) +
      geom_point() +
      geom_hline(yintercept = 0) +
      geom_vline(xintercept = 0) +
      geom_text(aes(label = 1:nrow(gammas1)), vjust = "inward", hjust = "inward") +
      scale_x_continuous(
                limits = c(-max(abs(c(range(gammas1[, 1:2]))))
                         , max(abs(c(range(gammas1[, 1:2])))))) +
      scale_y_continuous(
                limits = c(-max(abs(c(range(gammas1[, 1:2]))))
                          , max(abs(c(range(gammas1[, 1:2])))))) +
      labs(title = "OLS", x = expression(v[1]), y = expression(v[2])) +
      theme_bw() +
      theme(plot.title = element_text(hjust = 0.5))
      print(BiplotOLS2)


    BiplotOLS3 <-
      ggplot(data = alphas1, mapping = aes(x = V1, y = V2)) +
      geom_point() +
      geom_hline(yintercept = 0) +
      geom_vline(xintercept = 0) +
      geom_text(aes(label = 1:nrow(alphas1)), vjust = "inward", hjust = "inward") +
      geom_point(data = gammas1, mapping = aes(x = V1, y = V2)) +
```

```
      geom_segment(data = gammas1, aes(x = 0, y = 0, xend = V1, yend = V2),
                  arrow = arrow(length = unit(0.2, "cm")), alpha = 0.75, color = "red") +
      geom_text(data = gammas1,
              aes(x = V1, y = V2, label = paste0("E", 1:nrow(gammasa)))
             , vjust = "inward", hjust = "inward") +
      scale_x_continuous(
              limits = c(-max(abs(c(range(alphas1[, 1:2], gammas1[, 1:2]))))
                        , max(abs(c(range(alphas1[, 1:2], gammas1[, 1:2])))))) +
      scale_y_continuous(
              limits = c(-max(abs(c(range(alphas1[, 1:2], gammas1[, 1:2]))))
                       , max(abs(c(range(alphas1[, 1:2], gammas1[, 1:2])))))) +
      labs(title = "OLS", x = expression(PC[1]), y = expression(PC[2])) +
      theme_bw() +
      theme(plot.title = element_text(hjust = 0.5))
      print(BiplotOLS3)

data(cultivo2009)
fm3 <-
  bayes_ammi(
    .data = cultivo2009
    , .y     = y
    , .gen   = entry
    , .env   = site
    , .rep   = rep
    , .nIter = 200
  )

 Mean_Alphas <- fm3$Mean_Alphas %>% setNames(paste0("V", 1:ncol(.)))
 Mean_Gammas <- fm3$Mean_Gammas %>% setNames(paste0("V", 1:ncol(.)))


# Biplots Bayesian
BiplotBayes1 <-
  ggplot(data = Mean_Alphas, mapping = aes(x = V1, y = V2)) +
  geom_point() +
  geom_hline(yintercept = 0) +
  geom_vline(xintercept = 0) +
  geom_text(aes(label = 1:nrow(Mean_Alphas)),
             vjust = "inward"
           , hjust = "inward") +
  scale_x_continuous(
     limits = c(-max(abs(c(range(Mean_Alphas[, 1:2]))))
               , max(abs(c(range(Mean_Alphas[, 1:2])))))) +
  scale_y_continuous(
      limits = c(-max(abs(c(range(Mean_Alphas[, 1:2]))))
                , max(abs(c(range(Mean_Alphas[, 1:2])))))) +
  labs(title = "Bayes", x = expression(u[1]), y = expression(u[2])) +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5))

print(BiplotBayes1)
```

```
BiplotBayes2 <-
  ggplot(data = Mean_Gammas, mapping = aes(x = V1, y = V2)) +
  geom_point() +
  geom_hline(yintercept = 0) +
  geom_vline(xintercept = 0) +
  geom_text(aes(label = 1:nrow(Mean_Gammas)), vjust = "inward", hjust = "inward") +
  scale_x_continuous(
            limits = c(-max(abs(c(range(Mean_Gammas[, 1:2]))))
                      , max(abs(c(range(Mean_Gammas[, 1:2])))))) +
  scale_y_continuous(
            limits = c(-max(abs(c(range(Mean_Gammas[, 1:2]))))
                      , max(abs(c(range(Mean_Gammas[, 1:2])))))) +
  labs(title = "Bayes", x = expression(v[1]), y = expression(v[2])) +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5))

print(BiplotBayes2)


BiplotBayes3 <-
  ggplot(data = Mean_Alphas, mapping = aes(x = V1, y = V2)) +
  geom_point() +
  geom_hline(yintercept = 0) +
  geom_vline(xintercept = 0) +
  geom_text(aes(label = 1:nrow(Mean_Alphas)),
             vjust = "inward", hjust = "inward") +
  geom_point(data = Mean_Gammas, mapping = aes(x = V1, y = V2)) +
  geom_segment(data = Mean_Gammas,
                aes(x = 0, y = 0, xend = V1, yend = V2),
               arrow = arrow(length = unit(0.2, "cm"))
               , alpha = 0.75, color = "red") +
  geom_text(data = Mean_Gammas,
            aes(x = V1, y = V2,
            label = paste0("E", 1:nrow(Mean_Gammas))),
            vjust = "inward", hjust = "inward") +
  scale_x_continuous(
            limits = c(-max(abs(c(range(Mean_Alphas[, 1:2], Mean_Gammas[, 1:2]))))
                      , max(abs(c(range(Mean_Alphas[, 1:2], Mean_Gammas[, 1:2])))))) +
  scale_y_continuous(
           limits = c(-max(abs(c(range(Mean_Alphas[, 1:2], Mean_Gammas[, 1:2]))))
                   , max(abs(c(range(Mean_Alphas[, 1:2], Mean_Gammas[, 1:2])))))) +
  labs(title = "Bayes", x = expression(PC[1]), y = expression(PC[2])) +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5))

print(BiplotBayes3)

Plot1Mu <-
  ggplot(data = fm3$mu1, mapping = aes(x = 1:nrow(fm3$mu1), y = mu)) +
  geom_line(color = "blue") +
  scale_x_continuous(labels = scales::comma) +
  scale_y_continuous(labels = scales::comma) +
  labs(y = expression(mu), x = "Iterations") +
```

```
    theme_bw()
  print(Plot1Mu)

  Plot2Mu <-
    ggplot(data = fm3$mu1, mapping = aes(mu)) +
    geom_histogram() +
    scale_x_continuous(labels = scales::comma) +
    scale_y_continuous(labels = scales::comma) +
    labs(y = "Frequency", x = expression(mu)) +
    theme_bw()
  print(Plot2Mu)


  Plot1Sigma2 <-
    ggplot(data = fm3$tau1, mapping = aes(x = 1:nrow(fm3$tau1), y = tau)) +
    geom_line(color = "blue") +
    scale_x_continuous(labels = scales::comma) +
    scale_y_continuous(labels = scales::comma) +
    labs(y = expression(sigma^2), x = "Iterations") +
    theme_bw()
  print(Plot1Sigma2)


  Plot2Sigma2 <-
    ggplot(data = fm3$tau1, mapping = aes(tau)) +
    geom_histogram() +
    scale_x_continuous(labels = scales::comma) +
    scale_y_continuous(labels = scales::comma) +
    labs(y = "Frequency", x = expression(sigma^2)) +
    theme_bw()
  print(Plot2Sigma2)


  # Plot of Alphas
  Plot1Alpha1 <-
    ggplot(data = fm3$tao1, mapping = aes(x = 1:nrow(fm3$tao1), y = tao1)) +
    geom_line(color = "blue") +
    scale_x_continuous(labels = scales::comma) +
    scale_y_continuous(labels = scales::comma) +
    labs(y = expression(alpha[1]), x = "Iterations") +
    theme_bw()
  print(Plot1Alpha1)

  Plot2Alpha1 <-
    ggplot(data = fm3$tao1, mapping = aes(tao1)) +
    geom_histogram() +
    scale_x_continuous(labels = scales::comma) +
    scale_y_continuous(labels = scales::comma) +
    labs(y = "Frequency", x = expression(alpha[1])) +
    theme_bw()
  print(Plot2Alpha1)

  Plot1Alpha2 <-
```

```
  ggplot(data = fm3$tao1, mapping = aes(x = 1:nrow(fm3$tao1), y = tao2)) +
  geom_line(color = "blue") +
  scale_x_continuous(labels = scales::comma) +
  scale_y_continuous(labels = scales::comma) +
  labs(y = expression(alpha[2]), x = "Iterations") +
  theme_bw()
print(Plot1Alpha2)

Plot2Alpha2 <-
  ggplot(data = fm3$tao1, mapping = aes(tao2)) +
  geom_histogram() +
  scale_x_continuous(labels = scales::comma) +
  scale_y_continuous(labels = scales::comma) +
  labs(y = "Frequency", x = expression(alpha[2])) +
  theme_bw()
print(Plot2Alpha2)

# Plot of Betas
Plot1Beta1 <-
  ggplot(data = fm3$delta1, mapping = aes(x = 1:nrow(fm3$delta1), y = delta1)) +
  geom_line(color = "blue") +
  scale_x_continuous(labels = scales::comma) +
  scale_y_continuous(labels = scales::comma) +
  labs(y = expression(beta[1]), x = "Iterations") +
  theme_bw()
print(Plot1Beta1)

Plot2Beta1 <-
  ggplot(data = fm3$delta1, mapping = aes(delta1)) +
  geom_histogram() +
  scale_x_continuous(labels = scales::comma) +
  scale_y_continuous(labels = scales::comma) +
  labs(y = "Frequency", x = expression(beta[1])) +
  theme_bw()
print(Plot2Beta1)


Plot1Beta2 <-
  ggplot(data = fm3$delta1, mapping = aes(x = 1:nrow(fm3$delta1), y = delta2)) +
  geom_line(color = "blue") +
  scale_x_continuous(labels = scales::comma) +
  scale_y_continuous(labels = scales::comma) +
  labs(y = expression(beta[2]), x = "Iterations") +
  theme_bw()
print(Plot1Beta2)

Plot2Beta2 <-
  ggplot(data = fm3$delta1, mapping = aes(delta2)) +
  geom_histogram() +
  scale_x_continuous(labels = scales::comma) +
  scale_y_continuous(labels = scales::comma) +
  labs(y = "Frequency", x = expression(beta[2])) +
  theme_bw()
```

```
print(Plot2Beta2)


Plot1Beta3 <-
  ggplot(data = fm3$delta1, mapping = aes(x = 1:nrow(fm3$delta1), y = delta3)) +
  geom_line(color = "blue") +
  scale_x_continuous(labels = scales::comma) +
  scale_y_continuous(labels = scales::comma) +
  labs(y = expression(beta[3]), x = "Iterations") +
  theme_bw()
print(Plot1Beta3)

Plot2Beta3 <-
  ggplot(data = fm3$delta1, mapping = aes(delta3)) +
  geom_histogram() +
  scale_x_continuous(labels = scales::comma) +
  scale_y_continuous(labels = scales::comma) +
  labs(y = "Frequency", x = expression(beta[3])) +
  theme_bw()
print(Plot2Beta3)

## End(Not run)
```

`cultivo2008` *Data for Genotypes by Environment Interaction (GEI)*

Description

`cultivo2008` is used for performing Genotypes by Environment Interaction (GEI) Analysis.

Usage

```
data(cultivo2008)
```

Format

A `data.frame` 1320 obs. of 6 variables.

Details

- Gen Genotype
- Institute Institute
- Rep Replicate
- Block Block
- Env Environment
- Yield Yield Response

Author(s)

1. Muhammad Yaseen (<`myaseen208@gmail.com`>)
2. Diego Jarquin (<`diego.jarquin@gmail.com`>)
3. Sergio Perez-Elizalde (<`sergiop@colpos.mx`>)
4. Juan Burgueño (<`j.burgueno@cgiar.org`>)
5. Jose Crossa (<`j.crossa@cgiar.org`>)

References

Perez-Elizalde, S., Jarquin, D., and Crossa, J. (2011) A General Bayesian Estimation Method of Linear–Bilinear Models Applied to Plant Breeding Trials With Genotype × Environment Interaction. *Journal of Agricultural, Biological, and Environmental Statistics*, 17, 15–37. (doi:10.1007/s13253-011-0063-9)

Examples

```
data(cultivo2008)
```

`cultivo2009` *Data for Genotypes by Environment Interaction (GEI)*

Description

`cultivo2009` is used for performing Genotypes by Environment Interaction (GEI) Analysis.

Usage

```
data(cultivo2009)
```

Format

A `data.frame` 1320 obs. of 6 variables.

Details

- Gen Genotype
- Institute Institute
- Rep Replicate
- Block Block
- Env Environment
- Yield Yield Response

Author(s)

1. Muhammad Yaseen (<`myaseen208@gmail.com`>)
2. Diego Jarquin (<`diego.jarquin@gmail.com`>)
3. Sergio Perez-Elizalde (<`sergiop@colpos.mx`>)
4. Juan Burgueño (<`j.burgueno@cgiar.org`>)
5. Jose Crossa (<`j.crossa@cgiar.org`>)

References

Perez-Elizalde, S., Jarquin, D., and Crossa, J. (2011) A General Bayesian Estimation Method of Linear–Bilinear Models Applied to Plant Breeding Trials With Genotype × Environment Interaction. *Journal of Agricultural, Biological, and Environmental Statistics*, 17, 15–37. (doi:10.1007/s13253-011-0063-9)

Examples

```
data(cultivo2009)
```

`e_eff` *Environment Effects*

Description

Calcuates Environment Effects

Usage

```
## Default S3 method:
e_eff(.data, .y, .gen, .env)
```

Arguments

`.data`	data.frame
`.y`	Response Variable
`.gen`	Genotypes Factor
`.env`	Environment Factor

Value

Environment Effects

Author(s)

1. Muhammad Yaseen (<`myaseen208@gmail.com`>)

References

Perez-Elizalde, S., Jarquin, D., and Crossa, J. (2011) A General Bayesian Estimation Method of Linear–Bilinear Models Applied to Plant Breeding Trials With Genotype × Environment Interaction. *Journal of Agricultural, Biological, and Environmental Statistics*, 17, 15–37. (doi:10.1007/s13253-011-0063-9)

Examples

```
data(cultivo2008)
e_eff(
    .data  = cultivo2008
   , .y    = y
   , .gen  = entry
   , .env  = site
   )
```

`ge_ammi` *AMMI of Genotype by Environment Interaction Model*

Description

Performs Additive Main Effects and Multiplication Interaction Analysis of Genotype by Environment Interaction Model

Usage

```
ge_ammi(.data, .y, .gen, .env, .rep)

## Default S3 method:
ge_ammi(.data, .y, .gen, .env, .rep)
```

Arguments

`.data`	data.frame
`.y`	Response Variable
`.gen`	Genotypes Factor
`.env`	Environment Factor
`.rep`	Replication Factor

Value

Genotype by Environment Interaction Model

Author(s)

1. Muhammad Yaseen (`<myaseen208@gmail.com>`)

References

Perez-Elizalde, S., Jarquin, D., and Crossa, J. (2011) A General Bayesian Estimation Method of Linear–Bilinear Models Applied to Plant Breeding Trials With Genotype × Environment Interaction. *Journal of Agricultural, Biological, and Environmental Statistics*, 17, 15–37. (doi:10.1007/s13253-011-0063-9)

Examples

```
data(cultivo2008)
fm1 <-
   ge_ammi(
      .data  = cultivo2008
     , .y    = y
     , .gen  = entry
     , .env  = site
     , .rep  = rep
     )


data(cultivo2009)
fm2 <-
   ge_ammi(
      .data  = cultivo2009
     , .y    = y
     , .gen  = entry
     , .env  = site
     , .rep  = rep
     )
```

`ge_eff` *Genotype by Environment Interaction Effects*

Description

Calcuates Genotype by Environment Interaction Effects

Usage

```
## Default S3 method:
ge_eff(.data, .y, .gen, .env)
```

Arguments

`.data`	data.frame
`.y`	Response Variable
`.gen`	Genotypes Factor
`.env`	Environment Factor

Value

Genotype by Environment Interaction Effects

Author(s)

1. Muhammad Yaseen (`<myaseen208@gmail.com>`)

References

Perez-Elizalde, S., Jarquin, D., and Crossa, J. (2011) A General Bayesian Estimation Method of Linear–Bilinear Models Applied to Plant Breeding Trials With Genotype × Environment Interaction. *Journal of Agricultural, Biological, and Environmental Statistics*, 17, 15–37. (doi:10.1007/s13253-011-0063-9)

Examples

```
data(cultivo2008)
ge_eff(
    .data  = cultivo2008
   , .y    = y
   , .gen  = entry
   , .env  = site
   )
```

`ge_mean` *Genotype by Environment Interaction Means*

Description

Calcuates Genotype by Environment Interaction Means

Usage

```
## Default S3 method:
ge_mean(.data, .y, .gen, .env)
```

Arguments

`.data`	data.frame
`.y`	Response Variable
`.gen`	Genotypes Factor
`.env`	Environment Factor

Value

Genotype by Environment Interaction Means

Author(s)

1. Muhammad Yaseen (<`myaseen208@gmail.com`>)

References

Perez-Elizalde, S., Jarquin, D., and Crossa, J. (2011) A General Bayesian Estimation Method of Linear–Bilinear Models Applied to Plant Breeding Trials With Genotype × Environment Interaction. *Journal of Agricultural, Biological, and Environmental Statistics*, 17, 15–37. (doi:10.1007/s13253-011-0063-9)

Examples

```
data(cultivo2008)
ge_mean(
    .data  = cultivo2008
   , .y    = y
   , .gen  = entry
   , .env  = site
   )
```

`ge_model` *Genotype by Environment Interaction Model*

Description

Calcuates Genotype by Environment Interaction Model

Usage

```
ge_model(.data, .y, .gen, .env, .rep)

## Default S3 method:
ge_model(.data, .y, .gen, .env, .rep)
```

Arguments

`.data`	data.frame
`.y`	Response Variable
`.gen`	Genotypes Factor
`.env`	Environment Factor
`.rep`	Replication Factor

Value

Genotype by Environment Interaction Model

Author(s)

1. Muhammad Yaseen (<`myaseen208@gmail.com`>)

References

Perez-Elizalde, S., Jarquin, D., and Crossa, J. (2011) A General Bayesian Estimation Method of Linear–Bilinear Models Applied to Plant Breeding Trials With Genotype × Environment Interaction. *Journal of Agricultural, Biological, and Environmental Statistics*, 17, 15–37. (doi:10.1007/s13253-011-0063-9)

Examples

```
data(cultivo2008)
fm1 <-
   ge_model(
      .data  = cultivo2008
     , .y    = y
     , .gen  = entry
     , .env  = site
     , .rep  = rep
     )
```

`ge_var` *Genotype by Environment Interaction Variances*

Description

Calcuates Genotype by Environment Interaction Variances

Usage

```
## Default S3 method:
ge_var(.data, .y, .gen, .env)
```

Arguments

`.data`	data.frame
`.y`	Response Variable
`.gen`	Genotypes Factor
`.env`	Environment Factor

Value

Genotype by Environment Interaction Variances

Author(s)

1. Muhammad Yaseen (<`myaseen208@gmail.com`>)

References

Perez-Elizalde, S., Jarquin, D., and Crossa, J. (2011) A General Bayesian Estimation Method of Linear–Bilinear Models Applied to Plant Breeding Trials With Genotype × Environment Interaction. *Journal of Agricultural, Biological, and Environmental Statistics*, 17, 15–37. (doi:10.1007/s13253-011-0063-9)

Examples

```
data(cultivo2008)
ge_var(
    .data  = cultivo2008
  , .y     = y
  , .gen   = entry
  , .env   = site
  )
```

`g_eff` *Genotype Effects*

Description

Calcuates Genotype Effects

Usage

```
## Default S3 method:
g_eff(.data, .y, .gen, .env)
```

Arguments

`.data`	data.frame
`.y`	Response Variable
`.gen`	Genotypes Factor
`.env`	Environment Factor

Value

Genotype Effects

Author(s)

1. Muhammad Yaseen (<`myaseen208@gmail.com`>)

References

Perez-Elizalde, S., Jarquin, D., and Crossa, J. (2011) A General Bayesian Estimation Method of Linear–Bilinear Models Applied to Plant Breeding Trials With Genotype × Environment Interaction. *Journal of Agricultural, Biological, and Environmental Statistics*, 17, 15–37. (doi:10.1007/s13253-011-0063-9)

Examples

```
data(cultivo2008)
g_eff(
    .data  = cultivo2008
   , .y    = y
   , .gen  = entry
   , .env  = site
   )
```

`matrix_k` *k Matrix*

Description

Gives k matrix

Usage

```
matrix_k(n)

## Default S3 method:
matrix_k(n)
```

Arguments

`n`	Number of columns

Value

Matrix

Author(s)

1. Muhammad Yaseen (`<myaseen208@gmail.com>`)

References

Perez-Elizalde, S., Jarquin, D., and Crossa, J. (2011) A General Bayesian Estimation Method of Linear–Bilinear Models Applied to Plant Breeding Trials With Genotype × Environment Interaction. *Journal of Agricultural, Biological, and Environmental Statistics*, 17, 15–37. (doi:10.1007/s13253-011-0063-9)

`orthnorm` *Orthogonal Normalization*

Description

Perform Orthogonal Normalization of a matrix

Usage

```
orthnorm(u = NULL, basis = TRUE, norm = TRUE)

## Default S3 method:
orthnorm(u = NULL, basis = TRUE, norm = TRUE)
```

Arguments

`u`	Matrix
`basis`	Logical argument by default TRUE
`norm`	Logical argument by default TRUE

Value

Matrix

Author(s)

1. Muhammad Yaseen (`<myaseen208@gmail.com>`)

References

Perez-Elizalde, S., Jarquin, D., and Crossa, J. (2011) A General Bayesian Estimation Method of Linear–Bilinear Models Applied to Plant Breeding Trials With Genotype × Environment Interaction. *Journal of Agricultural, Biological, and Environmental Statistics*, 17, 15–37. (doi:10.1007/s13253-011-0063-9)

Index