Package 'riskPredictClustData'

October 14, 2022

Type Package

Title Assessing Risk Predictions for Clustered Data

Version 0.2.6

Date 2018-11-18

Maintainer Weiliang Qiu <Weiliang.Qiu@gmail.com>

Depends R (>= 3.4.0)

Imports MASS, stats, gee, Hmisc, mvtnorm, utils

Description
Assessing and comparing risk prediction rules for clustered data. The method is based on the paper: Rosner B, Qiu W, and Lee MLT.(2013) <doi:10.1007/s10985-012-9240-6>.

License GPL (>= 2)

NeedsCompilation no

Author Bernard Rosner [aut, ctb],
Weiliang Qiu [aut, cre],
Meiling T. Lee [aut, ctb]

Repository CRAN

Date/Publication 2018-11-28 08:50:03 UTC

R topics documented:

---

genSimDataGLMEM   *Generate simulated data from logistic mixed effects model based on the AMD data*

Description

Generate simulated data from logistic mixed effects model based on the AMD data.

Usage

```
genSimDataGLMEM(
  nSubj = 131,
  beta0 = -6,
  sd.beta0i = 1.58,
  beta1 = 1.58,
  beta2 = -3.95,
  beta3 = 3.15,
  beta4 = 2.06,
  beta5 = 0.51,
  beta6 = 1.47,
  beta7 = 3.11,
  p.smkcur = 0.08,
  p.inieye31 = 0.44,
  p.inieye32 = 0.42,
  p.inieye41 = 0.12,
  p.inieye42 = 0.11,
  sd.lncalorc = 0.33)
```

Arguments

nSubj	integer. Number of subjects. Each subject would have data for 2 eyes.
beta0	mean of intercept β_{0i}, which is assumed random and follows normal distribution $N(\beta_0, \sigma_\beta^2)$
sd.beta0i	standard deviation σ_β^2 of the random intercept β_{0i}.
beta1	slope for the binary covariate $cursmk$ (current smoking status). $cursmk = 1$ indicates current smokers. $cursmk = 0$ indicates past smokers or never smokers.
beta2	slope for the continuous mean-centered covariate $lncalor_c$.
beta3	slope for the binary covariate $inieye3$ indicating if an eye of a subject has initial grade equal to 3. $inieye3 = 1$ indicates the eye has initial grade equal to 3.
beta4	slope for the binary covariate $inieye4$ indicating if an eye of a subject has initial grade equal to 4. $inieye4 = 1$ indicates the eye has initial grade equal to 4.
beta5	slope for the binary covariate $rtotfat_1$ indicating if the subject's total fat intake is in the 2nd quartile of total fat intake. $rtotfat_1 = 1$ indicates the subject is in the 2nd quartile.

beta6	slope for the binary covariate $rtotfat_2$ indicating if the subject's total fat intake is in the 3rd quartile of total fat intake. $rtotfat_2 = 1$ indicates the subject is in the 3rd quartile.
beta7	slope for the binary covariate $rtotfat_3$ indicating if the subject's total fat intake is in the 4th quartile of total fat intake. $rtotfat_3 = 1$ indicates the subject is in the 4th quartile.
p.smkcur	proportion of current smokers.
p.inieye31	proportion of left eye having inital grade equal to 3.
p.inieye32	proportion of right eye having inital grade equal to 3.
p.inieye41	proportion of left eye having inital grade equal to 4.
p.inieye42	proportion of right eye having inital grade equal to 4.
sd.lncalorc	standard deviation for $lncalor_c$.

Details

We generate simulated data set from the following generalized linear mixed effects model:

$$\log\left(\frac{p_{ij}}{(1 - p_{ij})}\right) = \beta_{0i} + \beta_1 smkcur_i + \beta_2 lncalor_{ci} + \beta_3 inieye3_{ij} + \beta_4 inieye4_{ij} + \beta_5 rtotfat_{1i} + \beta_6 rtotfat_{2i} + \beta_7 rtotfat_{3i},$$

$$i = 1, \ldots, N, j = 1, 2, \beta_{0i} \sim N\left(\beta_0, \sigma_\beta^2\right).$$

Value

A data frame with 8 columns: cid, subuid, prog, smkcur, lncalorc, inieye3, inieye4, and rtotfat, where cid is the subject id, subuid is the unit id, and prog is the progression status. $prog = 1$ indicates the eye is progressed. $prog = 0$ indicates the eye is not progressed. There are nSubj*2 rows. The first nSubj rows are for the left eyes and the second nSubj rows are for the right eyes.

Author(s)

Bernard Rosner <stbar@channing.harvard.edu>, Weiliang Qiu <Weiliang.Qiu@gmail.com>, Meiling Ting Lee <MLTLEE@umd.edu>

References

Rosner B, Qiu W, and Lee MLT. Assessing Discrimination of Risk Prediction Rules in a Clustered Data Setting. Lifetime Data Anal. 2013 Apr; 19(2): 242-256.

Examples

```
set.seed(1234567)
datFrame = genSimDataGLMEM(nSubj = 30, beta0 = -6, sd.beta0i = 1.58,
                           beta1 = 1.58, beta2 = -3.95, beta3 = 3.15, beta4 = 2.06,
                           beta5 = 0.51, beta6 = 1.47, beta7 = 3.11,
                           p.smkcur = 0.08, p.inieye31 = 0.44, p.inieye32 = 0.42,
                           p.inieye41 = 0.12, p.inieye42 = 0.11, sd.lncalorc = 0.33)
print(dim(datFrame))
print(datFrame[1:2,])
```

---

getScore *Get data frame for the function riskPredict*

Description

Get data frame for the function riskPredict.

Usage

```
getScore(fmla, cidVar, subuidVar, statusVar, datFrame, mycorstr = "exchangeable",
    verbose = FALSE)
```

Arguments

`fmla`	A formula object for the function gee
`cidVar`	character. Phenotype variable name for cluster id
`subuidVar`	character. Phenotype variable name for unit id
`statusVar`	character. Phenotype variable name for progression status
`datFrame`	A data frame with at least 3 columns corresponding to `cid` (indicated by `cidVar`), `subuid` (indicated by `subuidVar`), `status` (indicated by `statusID`). `cid` indicates cluster id; `subuid` indicates unit ID within a cluster; `status=1` indicates an eye is progressed; `status=0` indicates an eye is not progressed.
`mycorstr`	character. indicates correlation structure. see the manual for the function gee in the R library gee
`verbose`	logical. indicating if summary of gee results should be printed out.

Value

A list with two elements: `frame` and `gee.obj`. `frame` is a data frame with at least 4 columns: cid, subuid, status, and score. `cid` indicates cluster id; `subuid` indicates unit ID within a cluster; `status=1` indicates an eye is progressed; `status=0` indicates an eye is not progressed; `score` represents the risk score.

`gee.obj` is the object returned by gee function.

Author(s)

Bernard Rosner <stbar@channing.harvard.edu>, Weiliang Qiu <Weiliang.Qiu@gmail.com>, Meiling Ting Lee <MLTLEE@umd.edu>

References

Rosner B, Qiu W, and Lee MLT. Assessing Discrimination of Risk Prediction Rules in a Clustered Data Setting. Lifetime Data Anal. 2013 Apr; 19(2): 242-256.

Examples

```
set.seed(1234567)
datFrame = genSimDataGLMEM(nSubj = 30, beta0 = -6, sd.beta0i = 1.58,
                           beta1 = 1.58, beta2 = -3.95, beta3 = 3.15, beta4 = 2.06,
                           beta5 = 0.51, beta6 = 1.47, beta7 = 3.11,
                           p.smkcur = 0.08, p.inieye31 = 0.44, p.inieye32 = 0.42,
                           p.inieye41 = 0.12, p.inieye42 = 0.11, sd.lncalorc = 0.33)

print(dim(datFrame))
print(datFrame[1:2,])

tt1 = getScore(fmla = prog~smkcur+lncalorc+inieye3+inieye4+factor(rtotfat),
  cidVar = "cid", subuidVar = "subuid", statusVar = "prog",
  datFrame = datFrame, mycorstr = "exchangeable",
  verbose = FALSE)
myframe1=tt1$frame

gee.obj=tt1$gee.obj
print(summary(gee.obj))

print(dim(myframe1))
print(myframe1[1:3,])
```

---

powerCal	*Calculate the power for testing $\delta = 0$*

Description

Calculate the power for testing $\delta = 0$.

Usage

```
powerCal(
  nSubj,
  mu1,
  triangle,
  rho,
  rho11,
  rho22,
  rho12,
  p11,
  p10,
  p01,
  alpha = 0.05)
```

Arguments

nSubj	integer. number of subjects to be generated. Assume each subject has two observations.
mu1	$\mu_1 = H(Y) - H(Y_c)$ is the difference between probit transformation $H(Y)$ and probit-shift alternative $H(Y_c)$, where Y is the prediction score of a randomly selected progressing subunit, and Y_c is the counterfactual random variable obtained if each subunit that had progressed actually had not progressed.
triangle	the difference of the expected value the the extended Mann-Whitney U statistics between two prediction rules, i.e., $\triangle = \eta_c^{(1)} - \eta_c^{(2)}$
rho	$\rho = corr\left(H\left(Z_{ij}\right), H\left(Z_{k\ell}\right)\right)$, where $H = \Phi^{-1}$ is the probit transformation.
rho11	$\rho_{11} = corr\left(H_{ij}^{(1)}, H_{i\ell}^{(1)}\right)$, where $H = \Phi^{-1}$ is the probit transformation.
rho22	$\rho_{22} = corr\left(H_{ij}^{(2)}, H_{i\ell}^{(2)}\right)$, where $H = \Phi^{-1}$ is the probit transformation.
rho12	$\rho_{12} = corr\left(H_{ij}^{(1)}, H_{i\ell}^{(2)}\right)$, where $H = \Phi^{-1}$ is the probit transformation.
p11	$p_{11} = Pr(\delta_{i1} = 1 \& \delta_{i2} = 1)$, where $\delta_{ij} = 1$ if the j-th subunit of the i-th cluster has progressed.
p10	$p_{10} = Pr(\delta_{i1} = 1 \& \delta_{i2} = 0)$, where $\delta_{ij} = 1$ if the j-th subunit of the i-th cluster has progressed.
p01	$p_{01} = Pr(\delta_{i1} = 0 \& \delta_{i2} = 1)$, where $\delta_{ij} = 1$ if the j-th subunit of the i-th cluster has progressed.
alpha	type I error rate

Value

the power

Author(s)

Bernard Rosner <stbar@channing.harvard.edu>, Weiliang Qiu <Weiliang.Qiu@gmail.com>, Meiling Ting Lee <MLTLEE@umd.edu>

References

Rosner B, Qiu W, and Lee MLT. Assessing Discrimination of Risk Prediction Rules in a Clustered Data Setting. Lifetime Data Anal. 2013 Apr; 19(2): 242-256.

Examples

```
set.seed(1234567)
mu1 = 0.8

power = powerCal(nSubj = 30, mu1 = mu1,
```

```
  triangle = 0.05, rho = 0.93, rho11 = 0.59, rho22 = 0.56, rho12 = 0.52,
  p11 = 0.115, p10 = 0.142, p01 = 0.130, alpha = 0.05)

print(power)
```

---

powerCalData	*Calculate the power for testing $\delta = 0$ based on a dataset*

Description

Calculate the power for testing $\delta = 0$ based on a dataset.

Usage

```
powerCalData(
  nSubj,
  triangle,
  frame,
  alpha = 0.05)
```

Arguments

nSubj	integer. number of subjects to be generated. Assume each subject has two observations.
triangle	the difference of the expected value the the extended Mann-Whitney U statistics between two prediction rules, i.e., $\triangle = \eta_c^{(1)} - \eta_c^{(2)}$
frame	A data frame with 5 columns: cid, subuid, status, score1, and score2. cid indicates cluster id; subuid indicates unit ID within a cluster; status=1 indicates an eye is progressed; status=0 indicates an eye is not progressed; score1 represents the score based on prediction rule 1. score2 represents the score based on prediction rule 2.
alpha	type I error rate

Value

A list with 11 elements.

power	the esstimated power
rho	$\rho = corr\left(H\left(Z_{ij}\right), H\left(Z_{k\ell}\right)\right)$, where $H = \Phi^{-1}$ is the probit transformation.
rho11	$\rho_{11} = corr\left(H_{ij}^{(1)}, H_{i\ell}^{(1)}\right)$, where $H = \Phi^{-1}$ is the probit transformation.
rho22	$\rho_{22} = corr\left(H_{ij}^{(2)}, H_{i\ell}^{(2)}\right)$, where $H = \Phi^{-1}$ is the probit transformation.
rho12	$\rho_{12} = corr\left(H_{ij}^{(1)}, H_{i\ell}^{(2)}\right)$, where $H = \Phi^{-1}$ is the probit transformation.

p11	$p_{11} = Pr(\delta_{i1} = 1 \& \delta_{i2} = 1)$, where $\delta_{ij} = 1$ if the j-th subunit of the i-th cluster has progressed.
p10	$p_{10} = Pr(\delta_{i1} = 1 \& \delta_{i2} = 0)$, where $\delta_{ij} = 1$ if the j-th subunit of the i-th cluster has progressed.
p01	$p_{01} = Pr(\delta_{i1} = 0 \& \delta_{i2} = 1)$, where $\delta_{ij} = 1$ if the j-th subunit of the i-th cluster has progressed.
p00	$p_{00} = Pr(\delta_{i1} = 0 \& \delta_{i2} = 0)$, where $\delta_{ij} = 1$ if the j-th subunit of the i-th cluster has progressed.
mu1	$\mu_1 = H(Y) - H(Y_c)$ is the difference between probit transformation $H(Y)$ and probit-shift alternative $H(Y_c)$ for the first prediction score, where Y is the prediction score of a randomly selected progressing subunit, and Y_c is the counterfactual random variable obtained if each subunit that had progressed actually had not progressed.
mu2	$\mu_2 = H(Y) - H(Y_c)$ is the difference between probit transformation $H(Y)$ and probit-shift alternative $H(Y_c)$ for the second prediction score, where Y is the prediction score of a randomly selected progressing subunit, and Y_c is the counterfactual random variable obtained if each subunit that had progressed actually had not progressed.

Author(s)

Bernard Rosner <stbar@channing.harvard.edu>, Weiliang Qiu <Weiliang.Qiu@gmail.com>, Meiling Ting Lee <MLTLEE@umd.edu>

References

Rosner B, Qiu W, and Lee MLT. Assessing Discrimination of Risk Prediction Rules in a Clustered Data Setting. Lifetime Data Anal. 2013 Apr; 19(2): 242-256.

Examples

```
set.seed(1234567)

datFrame = genSimDataGLMEM(nSubj = 30, beta0 = -6, sd.beta0i = 1.58,
                           beta1 = 1.58, beta2 = -3.95, beta3 = 3.15, beta4 = 2.06,
                           beta5 = 0.51, beta6 = 1.47, beta7 = 3.11,
                           p.smkcur = 0.08, p.inieye31 = 0.44, p.inieye32 = 0.42,
                           p.inieye41 = 0.12, p.inieye42 = 0.11, sd.lncalorc = 0.33)

print(dim(datFrame))
print(datFrame[1:2,])

# prediction rule 1
tt1 = getScore(fmla = prog~smkcur+lncalorc+inieye3+inieye4+factor(rtotfat),
  cidVar = "cid", subuidVar = "subuid", statusVar = "prog",
  datFrame = datFrame, mycorstr = "exchangeable",
  verbose = FALSE)
```

```
myframe1=tt1$frame

print(dim(myframe1))
print(myframe1[1:3,])

####
# prediction rule 2
tt2 = getScore(fmla = prog~smkcur+lncalorc+inieye3+inieye4,
  cidVar = "cid", subuidVar = "subuid", statusVar = "prog",
  datFrame = datFrame, mycorstr = "exchangeable",
  verbose = FALSE)
myframe2=tt2$frame

print(dim(myframe2))
print(myframe2[1:3,])

# combine scores from two prediction rules
myframe12=myframe1[, c("cid", "subuid", "status")]
myframe12$score1=myframe1$score
myframe12$score2=myframe2$score
print(dim(myframe12))
print(myframe12[1:3,])


res = powerCalData(nSubj = 30, triangle = 0.05, frame=myframe12, alpha = 0.05)

print(res)
```

---

riskPredict   *Assessing risk prediction performance for clustered data*

Description

Assessing risk prediction performance for clustered data.

Usage

```
riskPredict(frame, alpha=0.05)
```

Arguments

frame   A data frame with 4 columns: cid, subuid, status, and score. cid indicates
            cluster id; subuid indicates unit ID within a cluster; status=1 indicates an eye
            is progressed; status=0 indicates an eye is not progressed; score represents
            the risk score.

alpha   numeric. confidence level for eta_c.

Details

To obtain 95% confidence interval of η_c, we first obtain 95% confidence interval $[c_1, c_2]$ for $\Phi^{-1}(\eta_c)$, then transform back: $[\Phi(c_1), \Phi(c_2)]$.

Value

A list of 6 elements:

stat	the test statistics $\hat{\eta}_c^{(1)}$ hateta_c^(1) based on the prediction rule.
se.stat	standard error of the test statistic under the null hypothesis.
z	z score z=(stat - 0.5)/se.stat
pval	p-value of the test
rho	correlation between $H(Z_{ij}) and H(Z_{i\ell})$
mu.hat	estimated μ.
theta.hat	estimated θ.
theta.c.hat	estimated θ_c.
E.stat.Ha	expectation of $\hat{\eta}_c$ under the alternative hypothesis.
se.stat.Ha	standard error for $\hat{\eta}_c$ under the alternative hypothesis.
CIlow	lower confidence limit for η_c.
CIupp	upper confidence limit for η_c.
datHk	A nSubj by 2 matrix of probit transformed risk scores by using only the first 2 observations of each subject.
ci	the vector of c_i, the number of progressing subunits for the i-th subject.
di	the vector of d_i, the number of non-progressing subunits for the i-th subject.

Author(s)

Bernard Rosner <stbar@channing.harvard.edu>, Weiliang Qiu <Weiliang.Qiu@gmail.com>, Meiling Ting Lee <MLTLEE@umd.edu>

References

Rosner B, Qiu W, and Lee MLT. Assessing Discrimination of Risk Prediction Rules in a Clustered Data Setting. Lifetime Data Anal. 2013 Apr; 19(2): 242-256.

Examples

```
set.seed(1234567)
datFrame = genSimDataGLMEM(nSubj = 30, beta0 = -6, sd.beta0i = 1.58,
                        beta1 = 1.58, beta2 = -3.95, beta3 = 3.15, beta4 = 2.06,
                        beta5 = 0.51, beta6 = 1.47, beta7 = 3.11,
                        p.smkcur = 0.08, p.inieye31 = 0.44, p.inieye32 = 0.42,
                        p.inieye41 = 0.12, p.inieye42 = 0.11, sd.lncalorc = 0.33)
```

```
print(dim(datFrame))
print(datFrame[1:2,])

tt1 = getScore(fmla = prog~smkcur+lncalorc+inieye3+inieye4+factor(rtotfat),
  cidVar = "cid", subuidVar = "subuid", statusVar = "prog",
  datFrame = datFrame, mycorstr = "exchangeable",
  verbose = FALSE)
myframe1=tt1$frame

print(dim(myframe1))
print(myframe1[1:3,])

res1 = riskPredict(myframe1)
print(names(res1))
print(res1)
```

---

riskPredictDiff   *Difference of two risk prediction rules for clustered data*

Description

Difference of two risk prediction rules for clustered data.

Usage

```
riskPredictDiff(frame, alpha = 0.05)
```

Arguments

frame   A data frame with 5 columns: cid, subuid, status, score1, and score2. cid indi-
                cates cluster id; subuid indicates unit ID within a cluster; status=1 indicates
                an eye is progressed; status=0 indicates an eye is not progressed; score1 rep-
                resents the score based on prediction rule 1. score2 represents the score based
                on prediction rule 2.

alpha   numeric. The confidence level.

Value

A list of 7 elements:

diff   the difference of test statistics $\hat{\eta}_c^{(1)} - \hat{\eta}_c^{(2)}$ hateta_c^(1)-hateta_c^(2) based on the
                2 prediction rules.

se.diff   standard error of the difference under the null hypothesis.

z   z score z=diff/se.diff

pval   p-value of the test

res1	output object of the function `riskPredict` for prediction rule 1.
res2	output object of the function `riskPredict` for prediction rule 2.
rhoVec	A vector of 4 correlations: $\rho = cov(H_{ij}^{(1)}, H_{ij}^{(2)})$, $\rho_{11} = cov(H_{ij}^{(1)}, H_{it}^{(1)})$, $\rho_{22} = cov(H_{ij}^{(2)}, H_{it}^{(2)})$, and $\rho_{12} = cov(H_{ij}^{(1)}, H_{it}^{(2)})$
E.diff.Ha	expectation of the difference under the alternative hypothesis.
se.diff.Ha	standard error of the difference under the alternative hypothesis.
CIlow.diff	Lower confidence limit.
CIup.diff	Upper confidence limit.

Author(s)

Bernard Rosner <stbar@channing.harvard.edu>, Weiliang Qiu <Weiliang.Qiu@gmail.com>, Meil-ing Ting Lee <MLTLEE@umd.edu>

References

Rosner B, Qiu W, and Lee MLT. Assessing Discrimination of Risk Prediction Rules in a Clustered Data Setting. Lifetime Data Anal. 2013 Apr; 19(2): 242-256.

Examples

```
set.seed(1234567)
datFrame = genSimDataGLMEM(nSubj = 30, beta0 = -6, sd.beta0i = 1.58,
                           beta1 = 1.58, beta2 = -3.95, beta3 = 3.15, beta4 = 2.06,
                           beta5 = 0.51, beta6 = 1.47, beta7 = 3.11,
                           p.smkcur = 0.08, p.inieye31 = 0.44, p.inieye32 = 0.42,
                           p.inieye41 = 0.12, p.inieye42 = 0.11, sd.lncalorc = 0.33)

print(dim(datFrame))
print(datFrame[1:2,])

# prediction rule 1
tt1 = getScore(fmla = prog~smkcur+lncalorc+inieye3+inieye4+factor(rtotfat),
  cidVar = "cid", subuidVar = "subuid", statusVar = "prog",
  datFrame = datFrame, mycorstr = "exchangeable",
  verbose = FALSE)
myframe1=tt1$frame

print(dim(myframe1))
print(myframe1[1:3,])

####
# prediction rule 2
tt2 = getScore(fmla = prog~smkcur+lncalorc+inieye3+inieye4,
  cidVar = "cid", subuidVar = "subuid", statusVar = "prog",
  datFrame = datFrame, mycorstr = "exchangeable",
```

```
   verbose = FALSE)
myframe2=tt2$frame

print(dim(myframe2))
print(myframe2[1:3,])

# combine scores from two prediction rules
myframe12=myframe1[, c("cid", "subuid", "status")]
myframe12$score1=myframe1$score
myframe12$score2=myframe2$score
print(dim(myframe12))
print(myframe12[1:3,])

####
resDiff = riskPredictDiff(frame=myframe12)
print(names(resDiff))
print(resDiff)
```

Index