samplesizelogisticcasecontrol Package

September 20, 2016

```
> library(samplesizelogisticcasecontrol)
```

Random data generation functions

Let X_1 and X_2 be two variables with a bivariate normal ditribution with mean (0, 0) and covariance [1, 0.5; 0.5, 2]. X_2 corresponds to the exposure of interest. Let $X_3 = X_1X_2$ and define functions for generating random data from the distribution of (X_1, X_2) and (X_1, X_2, X_3).

```
> mymvn <- function(n) {
+   mu    <- c(0, 0)
+   sigma <- matrix(c(1, 0.5, 0.5, 2), byrow=TRUE, nrow=2, ncol=2)
+   dat   <- rmvnorm(n, mean=mu, sigma=sigma)
+   dat
+ }
> myF <- function(n) {
+   dat <- mymvn(n)
+   dat <- cbind(dat, dat[, 1]*dat[, 2])
+   dat
+ }
```

Generate some data

```
> data <- myF(200)
> colnames(data) <- paste("X", 1:3, sep="")
> data[1:5, ]
```

```
             X1         X2          X3
[1,] -0.03175259 -0.7062623  0.02242566
[2,]  0.57595297 -0.5576887 -0.32120245
[3,] -1.15885939  0.4438503 -0.51436007
[4,] -0.97295536 -1.3392027  1.30298446
[5,] -0.67407391 -0.2216138  0.14938408
```

Examples of univariate calculations

We have the logistic model $logit = \mu + \beta X$ and are testing $\beta = 0$. Suppose the disease prevalence is 0.01, the log-odds ratio for the exposure X is 0.26 and that the exposure follows a Bernoulli(p) distribution with p = 0.15.

```
> prev  <- 0.01
> logOR <- 0.26
> p <- 0.15
```

Compute the sample sizes

```
> sampleSize_binary(prev, logOR, probXeq1=p)

$ss.wald.1
[1] 4725

$ss.wald.2
[1] 4498

$ss.score.1
[1] 4230

$ss.score.2
[1] 4441
```

The same result can be obtained assuming X is ordinal and passing in the 2 probabilities $P(X = 0)$ and $P(X = 1)$.

```
> sampleSize_ordinal(prev, logOR, probX=c(1-p, p))

$ss.wald.1
[1] 4725

$ss.wald.2
[1] 4498

$ss.score.1
[1] 4230

$ss.score.2
[1] 4440
```

Let X be ordinal with 3 levels. The vector being passed into the probX argument below is $(P(X = 0), P(X = 1), P(X = 2))$.

```
> sampleSize_ordinal(prev, logOR, probX=c(0.4, 0.35, 0.25))

$ss.wald.1
[1] 989

$ss.wald.2
[1] 985

$ss.score.1
[1] 959

$ss.score.2
[1] 963
```

Now let the exposure X be $N(0, 1)$.

```
> sampleSize_continuous(prev, logOR)

$ss.wald.1
[1] 631

$ss.wald.2
[1] 644

$ss.score.1
[1] 615

$ss.score.2
[1] 602
```

For the univariate case with continuous exposure, we can specify the probability density function of X in different ways. Consider X to have a chi-squared distribution with 1 degree of freedom. Note that the domain of a chi-squared pdf is from 0 to infinity, and that the var(X) = 2.

```
> sampleSize_continuous(prev, logOR, distF="dchisq(x, 1)",
+                       distF.support=c(0,Inf), distF.var=2)

$ss.wald.1
[1] 283

$ss.wald.2
[1] 242

$ss.score.1
[1] 98

$ss.score.2
[1] 113

> f <- function(x) {dchisq(x, 1)}
> sampleSize_continuous(prev, logOR, distF=f, distF.support=c(0, Inf),
+                        distF.var=2)

$ss.wald.1
[1] 283

$ss.wald.2
[1] 242

$ss.score.1
[1] 98

$ss.score.2
[1] 113
```

If we do not set *distF.var*, then the variance of X will be approximated by numerical integration and could yield slightly different results.

```
> sampleSize_continuous(prev, logOR, distF="dchisq(x, 1)", distF.support=c(0,Inf))

$ss.wald.1
[1] 276

$ss.wald.2
[1] 242

$ss.score.1
[1] 100

$ss.score.2
[1] 113
```

Let X have the distribution defined by column X_1 in data.

```
> sampleSize_data(prev, logOR, data[, "X1", drop=FALSE])

$ss.wald.1
[1] 735

$ss.wald.2
[1] 754

$ss.score.1
[1] 732

$ss.score.2
[1] 714
```

Examples with confounders

We have the logit model $logit = \mu + \beta_1 X_1 + \beta_2 X_2$ and are interested in testing $\beta_2 = 0$. Here we must have log-odds ratios for X_1 and X_2, and we will use the distribution function mymvn defined above to generate 200 random samples. Note that logOR[1] corresponds to X_1 and logOR[2] corresponds to X_2.

```
> logOR <- c(0.1, 0.13)
> sampleSize_data(prev, logOR, mymvn(200))

$ss.wald.1
[1] 1149

$ss.wald.2
[1] 1167

$ss.score.1
[1] 1145
```

```
$ss.score.2
[1] 1128
```

Now we would like to perform a test of interaction, $\beta_3 = 0$, where $logit = \mu + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3$ and $X_3 = X_1 X_2$. The vector of log-odds ratios must be of length 3 and in the same order as (X_1, X_2, X_3).

```
> logOR <- c(0.1, 0.15, 0.11)
> sampleSize_data(prev, logOR, myF(1000))

$ss.wald.1
[1] 1583

$ss.wald.2
[1] 1633

$ss.score.1
[1] 1579

$ss.score.2
[1] 1530
```

Pilot data from a file

Suppose we want to compute sample sizes for a case-control study where we have pilot data from a previous study. The pilot data is stored in the file:

```
> file <- system.file("sampleData", "data.txt", package="samplesizelogisticcasecontrol")
> file

[1] "/tmp/Rtmpz1YxMa/Rinst5d577cb4ab6a/samplesizelogisticcasecontrol/sampleData/data.txt"
```

Here the exposure variable is "Treatment", and "Gender_Male" is a dummy variable for the confounder gender. We will use the data from only the controls and define a new variable of interest which is the interaction of gender and treatment. In our model, both gender and treatment will be confounders. First, read in the data.

```
> data <- read.table(file, header=1, sep="\t")
```

Create the interaction variable

```
> data[, "Interaction"] <- data[, "Gender_Male"]*data[, "Treatment"]
> data[1:5, ]

  Casecontrol Gender_Male Treatment Interaction
1           0           0         0           0
2           1           1         0           0
3           0           0         0           0
4           1           1         1           1
5           1           1         0           0
```

Now subset the data to use only the controls

```
> temp <- data[, "Casecontrol"] %in% 0
> data <- data[temp, ]
```

The data that gets passed in should only contain the columns that will be used in the analysis with the variable of interest being the last column.

```
> vars <- c("Gender_Male", "Treatment", "Interaction")
> data <- data[, vars]
```

Define the log-odds ratios for gender, treatment, and the interaction of gender and treatment. The order of these log-odds ratios must match the order of the columns in the data.

```
> logOR <- c(0.1, 0.13, 0.27)
```

Compute the sample sizes

```
> sampleSize_data(prev, logOR, data)

$ss.wald.1
[1] 9402

$ss.wald.2
[1] 9404

$ss.score.1
[1] 9403

$ss.score.2
[1] 9400
```

Note that the same results can be obtained by not reading in the data and creating a new interaction variable, but by setting the input argument of data to be of type *file.list*.

```
> data.list <- list(file=file, header=1, sep="\t",
+             covars=c("Gender_Male", "Treatment"),
+             exposure=c("Gender_Male", "Treatment"))
> data.list$subsetData <- list(list(var="Casecontrol", operator="%in%", value=0))
> sampleSize_data(prev, logOR, data.list)

$ss.wald.1
[1] 9402

$ss.wald.2
[1] 9404

$ss.score.1
[1] 9403

$ss.score.2
[1] 9400
```

Session Information

```
> sessionInfo()

R version 3.3.0 (2016-05-03)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: CentOS release 6.6 (Final)

locale:
 [1] LC_CTYPE=en_US             LC_NUMERIC=C
 [3] LC_TIME=C                  LC_COLLATE=C
 [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
 [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
 [9] LC_ADDRESS=C               LC_TELEPHONE=C
[11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] samplesizelogisticcasecontrol_0.0.5 mvtnorm_1.0-5

loaded via a namespace (and not attached):
[1] tools_3.3.0
```