

# samplesizelogisticcasecontrol Package

November 13, 2024

```
> library(samplesizelogisticcasecontrol)
```

## Random data generation functions

Let  $X_1$  and  $X_2$  be two variables with a bivariate normal distribution 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,]	0.23437095	1.4286535	0.33483487
[2,]	0.69891960	0.2692693	0.18819761
[3,]	0.35311765	2.4478229	0.86436947
[4,]	1.62422754	-0.2287488	-0.37154008
[5,]	0.08057511	-0.9034103	-0.07279238

## Examples of univariate calculations

We have the logistic model  $\text{logit} = \mu + \beta X$  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] 4472
```

```
$ss.wald.2
[1] 4498
```

```
$ss.score.1
[1] 4467
```

```
$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] 4472
```

```
$ss.wald.2
[1] 4498
```

```
$ss.score.1
[1] 4467
```

```
$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] 975
```

```
$ss.wald.2
[1] 985
```

```
$ss.score.1
[1] 973
```

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

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

```
> sampleSize_continuous(prev, logOR)
```

```
$ss.wald.1  
[1] 625
```

```
$ss.wald.2  
[1] 644
```

```
$ss.score.1  
[1] 621
```

```
$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  $\text{var}(X) = 2$ .

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

```
$ss.wald.1  
[1] 170
```

```
$ss.wald.2  
[1] 242
```

```
$ss.score.1  
[1] 168
```

```
$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] 170
```

```
$ss.wald.2  
[1] 242
```

```
$ss.score.1  
[1] 168
```

```
$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] 170

$ss.wald.2
[1] 242

$ss.score.1
[1] 168

$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] 547

$ss.wald.2
[1] 565

$ss.score.1
[1] 544

$ss.score.2
[1] 526
```

## Examples with confounders

We have the logit model  $\text{logit} = \mu + \beta_1 X_1 + \beta_2 X_2$  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] 1399

$ss.wald.2
[1] 1420

$ss.score.1
[1] 1395
```

```
$ss.score.2
[1] 1373
```

Now we would like to perform a test of interaction,  $\beta_3 = 0$ , where  $\text{logit} = \mu + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3$  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] 1518
```

```
$ss.wald.2
[1] 1558
```

```
$ss.score.1
[1] 1499
```

```
$ss.score.2
[1] 1461
```

## 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/RtmpNoGKbm/Rinst24834fb4bc5/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
> data2 <- 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")
> data2 <- data2[, 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, data2)
```

```
$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
```

## Power calculation using pilot data

Using the pilot data, estimate the log-odds ratios from a logistic regression:

```
> fit <- glm(Casecontrol ~ Gender_Male + Treatment + Interaction, data=data, family=binomi
> summary(fit)
```

Call:

```
glm(formula = Casecontrol ~ Gender_Male + Treatment + Interaction,
     family = binomial(), data = data)
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-0.94369	0.13890	-6.794	1.09e-11 ***
Gender_Male	0.08297	0.19627	0.423	0.672
Treatment	0.21373	0.19447	1.099	0.272
Interaction	-0.30629	0.27771	-1.103	0.270

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1213.1 on 999 degrees of freedom

Residual deviance: 1211.5 on 996 degrees of freedom

AIC: 1219.5

Number of Fisher Scoring iterations: 4

Extract the estimates needed

```
> coef <- fit$coefficients
> logOR <- coef[-1]
> logOR
```

```
Gender_Male Treatment Interaction
0.08296883 0.21372855 -0.30628697
```

Estimate the power assuming a study size of 15000 subjects with 10 percent of them cases.

```
> power_data(prev, logOR, data[, vars], sampleSize=15000, cc.ratio=0.1)
```

```
$pow.wald.1
[1] 0.8020756
```

```
$pow.wald.2
[1] 0.8010627
```

```
$pow.score.1
[1] 0.8035882
```

```
$pow.score.2
[1] 0.8045952
```

## Session Information

```
> sessionInfo()
```

```
R version 4.4.2 (2024-10-31)  
Platform: x86_64-pc-linux-gnu  
Running under: Ubuntu 24.04.1 LTS
```

```
Matrix products: default
```

```
BLAS: /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
```

```
LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/libopenblas-p-r0.3.26.so; LAPACK version
```

```
locale:
```

```
[1] LC_CTYPE=en_US.UTF-8      LC_NUMERIC=C  
[3] LC_TIME=en_US.UTF-8      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
```

```
time zone: Etc/UTC
```

```
tzcode source: system (glibc)
```

```
attached base packages:
```

```
[1] stats      graphics  grDevices  utils      datasets  methods    base
```

```
other attached packages:
```

```
[1] samplesizelogisticcasecontrol_2.0.2 mvtnorm_1.3-2
```

```
loaded via a namespace (and not attached):
```

```
[1] compiler_4.4.2  tools_4.4.2     maketools_1.3.1  buildtools_1.0.0  
[5] knitr_1.49      xfun_0.49       sys_3.4.3        evaluate_1.0.1
```