

# Introdução ao uso do



# em Ciências da Saúde

## 4b - Análise inferencial univariada

Fúlvio Borges Nedel

**Departamento de Saúde Pública – SPB**  
**Centro de Ciências da Saúde – CCS**  
**Universidade Federal de Santa Catarina – UFSC**

*Grups de Recerca d'Amèrica i Àfrica Llatines – GRAAL*  
<http://graal.uab.cat>

19 de dezembro de 2017

- Inicie com uma sessão vazia e carregue o arquivo de dados
- Verá que ele tem três objetos,
  - os dois bancos de dados e
  - a função `tabua`, criada anteriormente

```
rm(list=ls())  
load('cursoR.RData')  
ls()  
[1] "cursoR" "cursoR2" "tabua"  
class(cursoR)  
[1] "data.frame"  
class(cursoR2)  
[1] "data.frame"  
class(tabua)  
[1] "function"
```

- Para buscar uma função que os calcule, comece com  
??`"confidence interval"`
- Temos (no mínimo)
  - `ci{epiDisplay}`
    - variáveis numéricas
    - variáveis dicotômicas, método exato
  - `prop.test{stats}`
    - variáveis dicotômicas, aproximação normal
  - `binom.test{stats}`
    - variáveis dicotômicas, método exato

- Para buscar uma função que os calcule, comece com

??`"confidence interval"`

- Temos (no mínimo)
  - `ci{epiDisplay}`
    - variáveis numéricas
    - variáveis dicotômicas, método exato
  - `prop.test{stats}`
    - variáveis dicotômicas, aproximação normal
  - `binom.test{stats}`
    - variáveis dicotômicas, método exato



confidence interval for multinomial probabilities in r



Todas Vídeos Imagens Notícias Shopping Mais Configurações Ferramentas

Aproximadamente 1.870.000 resultados (0,41 segundos)

[\[PDF\] Package 'MultinomialCI' - CRAN.R-project.org](https://cran.r-project.org/web/packages/MultinomialCI/MultinomialCI.pdf)

<https://cran.r-project.org/web/packages/MultinomialCI/MultinomialCI.pdf> Traduzir esta página

19 de fev de 2015 - Title Simultaneous confidence intervals for multinomial proportions according to the method by Sison and Glaz. Type Package. Author Pablo J. Villacorta <pjvi@decsai.ugr.es>. Description An implementation of a method for building simultaneous confidence intervals for the probabilities of a multinomial.

[MultinomCI function | R Documentation](https://www.rdocumentation.org/packages/0.99/MultinomialCI)

<https://www.rdocumentation.org/packages/0.99/MultinomialCI> Traduzir esta página

Given a vector of observations with the number of samples falling in each class of a multinomial distribution, builds the simultaneous confidence intervals for the multinomial probabilities according to the method proposed by Sison and Glaz (1995). The R code has been translated from the SAS code written by May and ...

```
attach(cursor2)
```

```
t.test(imc)
```

One Sample t-test

```
data: imc
```

```
t = 96.569, df = 298, p-value < 2.2e-16
```

```
alternative hypothesis: true mean is not equal to 0
```

```
95 percent confidence interval:
```

```
27.20038 28.33206
```

```
sample estimates:
```

```
mean of x
```

```
27.76622
```

```
library(epiDisplay)
```

```
ci(imc)
```

n	mean	sd	se	lower95ci	upper95ci
299	27.76622	4.971806	0.287527	27.20038	28.33206

### Com a variável obeso, que é dicotômica

```
table(obeso)
```

```
obeso
```

```
sim não
```

```
84 215
```

```
prop.test(table(obeso))
```

```
1-sample proportions test with continuity correction
```

```
data: table(obeso), null probability 0.5
```

```
X-squared = 56.522, df = 1, p-value = 5.558e-14
```

```
alternative hypothesis: true p is not equal to 0.5
```

```
95 percent confidence interval:
```

```
0.2314582 0.3361532
```

```
sample estimates:
```

```
p
```

```
0.2809365
```

### Com a variável obeso, que é dicotômica

```
table(obeso)
```

```
obeso
```

```
sim não
```

```
84 215
```

```
prop.test(table(obeso))
```

```
1-sample proportions test with continuity correction
```

```
data: table(obeso), null probability 0.5
```

```
X-squared = 56.522, df = 1, p-value = 5.558e-14
```

```
alternative hypothesis: true p is not equal to 0.5
```

```
95 percent confidence interval:
```

```
0.2314582 0.3361532
```

```
sample estimates:
```

```
      p  
0.2809365
```

### Com a variável obeso, que é dicotômica

```
table(obeso)
```

```
obeso
```

```
sim não
```

```
84 215
```

```
prop.test(table(obeso))
```

```
1-sample proportions test with continuity correction
```

```
data: table(obeso), null probability 0.5
```

```
X-squared = 56.522, df = 1, p-value = 5.558e-14
```

```
alternative hypothesis: true p is not equal to 0.5
```

```
95 percent confidence interval:
```

```
0.2314582 0.3361532
```

```
sample estimates:
```

```
      p  
0.2809365
```

```
ci(obeso)
```

events	total	probability	se	exact.lower95ci	exact.upper95ci
215	299	0.7190635	0.02599274	0.6644214	0.769295



### Com a variável obeso, que é dicotômica

```
table(obeso)
```

```
obeso  
sim não  
84 215
```

```
prop.test(table(obeso))
```

1-sample proportions test with continuity correction

data: table(obeso), null probability 0.5

X-squared = 56.522, df = 1, p-value = 5.558e-14

alternative hypothesis: true p is not equal to 0.5

95 percent confidence interval:

0.2314582 0.8361532

sample estimates:

p  
0.2809365

```
ci(obeso)
```

events	total	probability	se	exact.lower95ci	exact.upper95ci
215	299	0.7190635	0.02599274	0.6644214	0.769295

### Com a variável obeso, que é dicotômica

```
table(obeso)
```

```
obeso
```

```
sim não  
84 215
```

```
prop.test(table(obeso))
```

1-sample proportions test with continuity correction

data: table(obeso), null probability 0.5

X-squared = 56.522, df = 1, p-value = 5.558e-14

alternative hypothesis: true p is not equal to 0.5

95 percent confidence interval:

0.2314582 0.3361532

sample estimates:

p  
0.2809365

```
ci(obeso)
```

events	total	probability	se	exact.lower95ci	exact.upper95ci
215	299	0.7190635	0.02599274	0.6644214	0.769295

```
ci(obeso=='sim')
```

events	total	probability	se	exact.lower95ci	exact.upper95ci
84	299	0.2809365	0.02599274	0.230705	0.3355786

## Apresentar os resultados numa tabela bonita ;)

```
x <- ci(obeso)
x[5:6]

exact.lower95ci exact.upper95ci
0.6644214      0.769295

IC95 <- matrix(c(as.numeric(rev(1-x[5:6])),
                  as.numeric(x[5:6])),
               nrow = 2, byrow = T)
colnames(IC95) <- c('IC95.inf', 'IC95.sup')
tab.obeso <- cbind(tabua(obeso)[-3,1:2], round(IC95*100, 2))
```

```
options(OutDec = ',') # marcador decimal como vírgula
tab.obeso
```

	Freq	%	IC95.inf	IC95.sup
sim	84	28,1	23,07	33,56
não	215	71,9	66,44	76,93

### De volta a estado nutricional, que é politômica

```
tabua(imccat)[-4,-c(3:4)]
```

	Freq	%
normal	99	33,1
sobrepeso	116	38,8
obesidade	84	28,1

O que se faz muitas vezes é calcular o IC da probabilidade da categoria contra seu complemento, i.e., todas as outras categorias em conjunto.

```
icbin <- rbind( ci(imccat=='normal'),
                ci(imccat=='sobrepeso'),
                ci(imccat=='obesidade') )
```

```
cbind(tabua(imccat)[1:3,1:2], round(icbin[5:6]*100,2))
```

	Freq	%	exact.lower95ci	exact.upper95ci
normal	99	33,1	27,80	38,76
sobrepeso	116	38,8	33,24	44,58
obesidade	84	28,1	23,07	33,56

Entretanto, isso não é muito adequado, porque a distribuição de probabilidades das categorias de uma **variável politômica** segue uma **distribuição multinomial**, que é um pouco mais complexa que a binomial.

```
library(MultinomialCI)
round(multinomialCI(table(imccat), .05)*100, 2)

      [,1] [,2]
[1,] 27,09 39,46
[2,] 32,78 45,14
[3,] 22,07 34,44
```

Entretanto, isso não é muito adequado, porque a distribuição de probabilidades das categorias de uma **variável politômica** segue uma **distribuição multinomial**, que é um pouco mais complexa que a binomial.

Claro que a tabela pode ser melhorada:

```
icmulti <- round(multinomialCI(table(imccat), .05)*100, 2)
colnames(icmulti) <- c('IC95.inf', 'IC95.sup')
# icmulti
cbind(tabua(imccat)[1:3,1:2], icmulti)

      Freq    % IC95.inf IC95.sup
normal    99 33,1    27,09    39,46
sobrepeso 116 38,8    32,78    45,14
obesidade  84 28,1    22,07    34,44

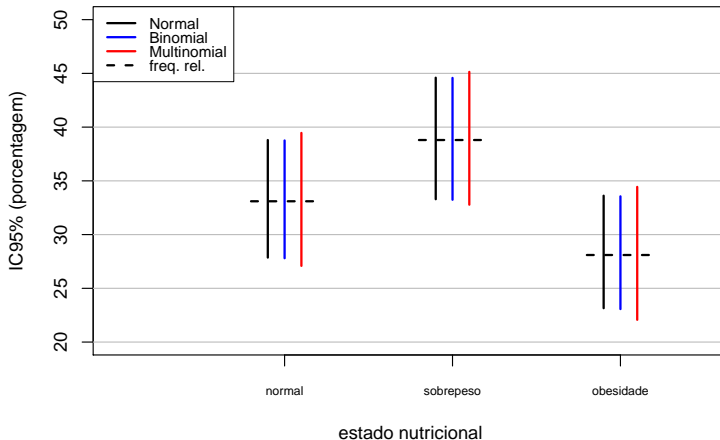
# cbind(tabua(imccat, digits = 2)[1:3,1:2], icmulti)
# cbind(tabua(imccat)[1:3,1:2], round(icmulti, 1))
```

Introdução ao R  
4.b Análise  
univariada – IC  
9/11

Fúlvio Nedel  
SPB/UFSC

	%	IC normal	IC binomial	IC multinomial
normal	33,1	27,86 - 38,8	27,8 - 38,76	27,09 - 39,46
sobrepeso	38,8	33,29 - 44,6	33,24 - 44,58	32,78 - 45,14
obesidade	28,1	23,15 - 33,62	23,07 - 33,56	22,07 - 34,44

Análise inferencial



```
icnorm <- rbind( prop.test(table(imccat!='normal'))$conf.int,  
                 prop.test(table(imccat!='sobrepeso'))$conf.int,  
                 prop.test(table(imccat!='obesidade'))$conf.int)  
icnorm <- round(icnorm*100,2)  
#  
icbin <- round(icbin[5:6]*100, 2)  
rownames(icbin) <- rownames(icnorm)  
icnorm2 <- rbind(paste(icnorm[1,1], '-', icnorm[1,2]),  
                 paste(icnorm[2,1], '-', icnorm[2,2]),  
                 paste(icnorm[3,1], '-', icnorm[3,2]))  
icbin2 <- rbind(paste(icbin[1,1], '-', icbin[1,2]),  
                paste(icbin[2,1], '-', icbin[2,2]),  
                paste(icbin[3,1], '-', icbin[3,2]))  
icmulti2 <- rbind(paste(icmulti[1,1], '-', icmulti[1,2]),  
                  paste(icmulti[2,1], '-', icmulti[2,2]),  
                  paste(icmulti[3,1], '-', icmulti[3,2]))  
tabela <- cbind(tabua(imccat)[-4,2], icnorm2, icbin2, icmulti2)  
colnames(tabela) <- c('%', 'IC normal', 'IC binomial', 'IC multinomial')  
  
xtable::xtable(tabela)
```



```
plot(NULL, xlim=c(0,3.5), ylim=c(20,50), xlab="estado nutricional",  
      ylab="IC95% (porcentagem)", xaxt='n')  
axis(1, at = 1:3, labels = levels(imccat), cex.axis=.7)  
abline(h=seq(0,45, 5), col='gray')  
legend('topleft', legend = c('Normal', 'Binomial', "Multinomial", "freq. rel."),  
       col = c(1,4,2, 1), lty = c(rep(1,3),2), lwd = 2, bg = 'white',  
       box.lwd = 0, y.intersp = .75, cex=.85)  
segments(y0 = icbin[1,1], x0 = 1, y1 = icbin[1,2], x1 = 1, col=4, lwd=2)  
segments(y0 = icbin[2,1], x0 = 2, y1 = icbin[2,2], x1 = 2, col=4, lwd=2)  
segments(y0 = icbin[3,1], x0 = 3, y1 = icbin[3,2], x1 = 3, col=4, lwd=2)  
segments(y0 = icmulti[1,1], x0 = 1.1, y1 = icmulti[1,2], x1 = 1.1, col=2, lwd=2)  
segments(y0 = icmulti[2,1], x0 = 2.1, y1 = icmulti[2,2], x1 = 2.1, col=2, lwd=2)  
segments(y0 = icmulti[3,1], x0 = 3.1, y1 = icmulti[3,2], x1 = 3.1, col=2, lwd=2)  
segments(y0 = icnorm[1,1], x0 = 0.9, y1 = icnorm[1,2], x1 = 0.9, lwd=2)  
segments(y0 = icnorm[2,1], x0 = 1.9, y1 = icnorm[2,2], x1 = 1.9, lwd=2)  
segments(y0 = icnorm[3,1], x0 = 2.9, y1 = icnorm[3,2], x1 = 2.9, lwd=2)  
segments(y0 = tabua(imccat)[1,2], x0 = .8, y1 = tabua(imccat)[1,2], x1 = 1.2,  
         lwd=2, lty=2)  
segments(y0 = tabua(imccat)[2,2], x0 = 1.8, y1 = tabua(imccat)[2,2], x1 = 2.2,  
         lwd=2, lty=2)  
segments(y0 = tabua(imccat)[3,2], x0 = 2.8, y1 = tabua(imccat)[3,2], x1 = 3.2,  
         lwd=2, lty=2)
```