

# MARIANA R QUEIROZ

 — [Mariana Ramos Queiroz](#)

Doutoranda em epidemiologia no Laboratório de Epidemiologia e Bioestatística (LEB), pelo Departamento de Medicina Veterinária Preventiva e Saúde Animal da Faculdade de Medicina Veterinária e Zootecnia, Universidade de São Paulo - VPS/FMVZ/USP. [Laboratório de Epidemiologia e Bioestatística](#)

Meu projeto refere-se a uma análise espacial da disseminação de brucelose pelo Brasil. Já participei de pesquisas com desenvolvimento de animais transgênicos para a expressão de proteínas recombinantes. Atualmente estou envolvida com saúde pública e animal, programas sanitários e epidemiologia.

A bioestatística me encanta, mas meu conhecimento é ínfimo.

## Meus Exercícios

Link para a página com os meus exercícios resolvidos [exec](#)

## Minha Proposta

### Plano A \*essa que prefiro :-D

Desculpem-me, mas hoje (21/03/2013) em plena aula de epidemiologia surgiu a ideia, por isso mais uma opção pra ver se vocês preferem.... (mesmo porque os planos A e B não estão dos melhores)

Cálculos para a avaliação de testes diagnósticos de doenças e agravos são realizados e criar uma função que simule o comportamento, a maneira que os diagnóstico irão impactar nos programas de saúde é fantástico!!! Logo, a ideia é fazer uma função que simule os cenários em diferentes situações, em que número de animais positivos, negativos e valor de prevalência seriam os inputs. Como output, provavelmente, seria o valor preditivo positivo (VPP), valor preditivo negativo (VPN), sensibilidade e especificidade.

Quanto aos outputs, tenho algumas dúvidas, qual seriam os melhores dados, pois a simulação iria ajudar na tomada de decisão de como agir para efetuar um rápido ou preciso diagnóstico. Dependeria também se é, por exemplo, um caso sobre zonas de erradicação da doença - de prevalência muito baixa - como efetuar o diagnóstico, fazer testes em série, que levam mais tempo. Ou em paralelo, que geralmente são bons em casos de surtos ou epidemias. Enfim o uso para fins didáticos seria incrível, pois essas simulações podem ser usadas para explicar essas taxas as pessoas quando tem que aprender sobre o assunto e além disso ajuda bastante nos programas de saúde pra tomada de decisão na maneira de agir, pois dessa forma podemos verificar dentre os testes diagnósticos existentes, quais são mais viáveis economicamente e viáveis em relação a mão de obra e tempo.

## Plano B

A função que pretendo criar vai “montar” uma pergunta de um questionário, sendo que essa pergunta pode ser para marcar entre múltiplas escolhas, uma lista de opções de resposta ou caixas de seleção de respostas (nesse caso, pode-se marcar mais que uma opção). Os inputs serão as perguntas e as respostas presentes no formulário que será criado e como output, serão as respostas escolhidas, que serão contagens das respostas - da classe vetor. A intenção da função é de criar questionários, para que sejam utilizados em inquéritos como alguns do Ministério da Agricultura, Pecuária e Abastecimento - MAPA.

## Plano C

Como segunda opção, a ideia é fazer uma função que tenha como input equações de retas ou curvas para formar desenhos, símbolos já contidos no R, além de texto para montar cartões de visita ou qualquer outro que sua imaginação te proporcionar.

## Comentários

A proposta A é muito legal, mesmomesmo, mas é um exercício de modelagem, não de programação! (acho que seria um ótimo projeto para ecopop ou biologia matemática, mas não para esta disciplina). Quanto às suas outras propostas, acho que não está claro o conceito de input e output de uma função: dado um certo input, o computador deve realizar certas operações e devolver um resultado, que é o output. Que tipo de operações uma função faria com uma pergunta para obter uma resposta? Isso me parece impossível. Talvez você possa rephrasing essa proposta para que ela fique mais clara. - Fernando Pizza

## Proposta final (mesmo!rs)

Ao tentar fazer algo mais genérico, pensei em uma função para comparar visualmente a distribuição de uma variável quantitativa com diferentes grupos (variáveis categóricas) por gráficos de densidade no mesmo dispositivo gráfico.

## Página de Ajuda / Help

cd.grupos

package:MRQ

R Documentation

Function to show density graph of two variables

Description:

The function is to show the behavior when two variables are grouped according to a categorical variable.

Usage:

```
cd.grupos(x,col=2,grupos=1)
```

#### Arguments:

`x` only accepts as argument a database of class `data.frame`  
`col` by default, `col = 2`, which means the number of the column containing the sample of the interest variable  
`grupos` by default, `grupos = 1`, which means the column number that is of the categorical variable

#### Details:

Used to check visually as "n" different groups behavior

#### Value:

By default, the function returns:

By default, the function returns:

A graph, which shows density plots of each group at the same device, to make comparison

#### Warning:

`col` serves only for the values `[[1, 2, 3 and 4`, corresponding to the column number it will use

`grupos` serves only for the values `[[1, 2, 3 and 4`, corresponding to the column number it will use

#### Author(s):

Mariana Ramos Queiroz (mrq.vet@gmail.com)

#### See Also:

`ggplot()`

#### Examples:

```
# Creating a random values for a data base
sexo <- c(rep('M',each=50),rep('H',50))
peso.m.h <- c(rnorm(50,60,12),rnorm(50,70,16))
altura.m.h <- c(rnorm(50,160.07,3.55),rnorm(50,172.2,2.5))
idade <- sample(rnorm(100,26.85,0.55))
```

```
# Creating data.frame
vari1 <- data.frame(sexo,peso.m.h,altura.m.h,idade)
vari2 <- data.frame(peso.m.h,sexo,altura.m.h,idade)
vari3 <- data.frame(peso.m.h,altura.m.h,sexo,idade)
vari4 <- data.frame(peso.m.h,altura.m.h,idade,sexo)
# Groups in column 1
```

```
x<-vari1
cd.grupos(vari1,col=3,grupos=1)
# Groups in column 2
x<-vari2
cd.grupos(vari2,col=3,grupos=2)

# Groups in column 3
x<-vari3
cd.grupos(vari3,col=2,grupos=3)

# Groups in column 4
x<-vari4
cd.grupos(vari4,col=2,grupos=4)
  # Loading a data frame
  esaligna <- read.csv('esaligna.csv',header=TRUE,as.is=TRUE)
  # Groups in column 2
  x <- esaligna
  cd.grupos(esaligna,col=3,grupos=2)
```

## Código da Função cd.grupos

```
cd.grupos <- function(x,col=2,grupos=1) # Não usei o ciclo 'for', pois no
ggplot não tava dando certo e a condicional 'if' funcionou bem
{
  require(ggplot2)
  cat('0 pacote -ggplot2- é carregado')
  if (col==2 & grupos==1)
  {
    tmp <- ggplot(x, aes(x=x[,2])) +
geom_density(aes(fill=x[,1],group=x[,1],color=x[,1]),alpha=0.3)
    return(tmp)
  }
  if (col==3 & grupos==1)
  {
    tmp <- ggplot(x, aes(x=x[,3])) +
geom_density(aes(fill=x[,1],group=x[,1],color=x[,1]),alpha=0.3)
    return(tmp)
  }
  if (col==4 & grupos==1)
  {
    tmp <- ggplot(x, aes(x=x[,4])) +
geom_density(aes(fill=x[,1],group=x[,1],color=x[,1]),alpha=0.3)
    return(tmp)
  }
  if (col==1 & grupos==2)
  {
    tmp <- ggplot(x, aes(x=x[,1])) +
geom_density(aes(fill=x[,2],group=x[,2],color=x[,2]),alpha=0.3)
    return(tmp)
  }
}
```

```
}
if (col==3 & grupos==2)
{
  tmp <- ggplot(x, aes(x=x[,3])) +
geom_density(aes(fill=x[,2],group=x[,2],color=x[,2]),alpha=0.3)
  return(tmp)
}
if (col==4 & grupos==2)
{
  tmp <- ggplot(x, aes(x=x[,4])) +
geom_density(aes(fill=x[,2],group=x[,2],color=x[,2]),alpha=0.3)
  return(tmp)
}
if (col==1 & grupos==3)
{
  tmp <- ggplot(x, aes(x=x[,1])) +
geom_density(aes(fill=x[,3],group=x[,3],color=x[,3]),alpha=0.3)
  return(tmp)
}
if (col==2 & grupos==3)
{
  tmp <- ggplot(x, aes(x=x[,2])) +
geom_density(aes(fill=x[,3],group=x[,3],color=x[,3]),alpha=0.3)
  return(tmp)
}
if (col==4 & grupos==3)
{
  tmp <- ggplot(x, aes(x=x[,4])) +
geom_density(aes(fill=x[,3],group=x[,3],color=x[,3]),alpha=0.3)
  return(tmp)
}
if (col==1 & grupos==4)
{
  tmp <- ggplot(x, aes(x=x[,1])) +
geom_density(aes(fill=x[,4],group=x[,4],color=x[,4]),alpha=0.3)
  return(tmp)
}
if (col==2 & grupos==4)
{
  tmp <- ggplot(x, aes(x=x[,2])) +
geom_density(aes(fill=x[,4],group=x[,4],color=x[,4]),alpha=0.3)
  return(tmp)
}
if (col==3 & grupos==4)
{
  tmp <- ggplot(x, aes(x=x[,3])) +
geom_density(aes(fill=x[,4],group=x[,4],color=x[,4]),alpha=0.3)
  return(tmp)
}
else (print('Erro'))
```

```
}
```

## Exemplo de output

Utilizando o banco de dados esaligna.csv - primeiramente visualizar as colunas do seu banco de dados #head(), depois de definida a coluna com a variável que quero #col=4 e em qual coluna estão meus diferentes grupos #grupos=2.

```
esaligna <- read.csv('esaligna.csv',header=TRUE,as.is=TRUE)
head(esaligna)
x <- esaligna
cd.grupos(esaligna,col=4,grupos=2)
```

O output são as diferentes distribuições para a variável 'diâmetro na altura do peito':



## Arquivos para download

[Função cd.grupos\(\)](#)

[Help cd.grupos](#)

[Exemplo de banco de dados a ser carregado \(esaligna.csv\)](#)

[Exemplo de output - esaligna/dap](#)

From:  
<http://ecor.ib.usp.br/> - **ecoR**

Permanent link:  
[http://ecor.ib.usp.br/doku.php?id=05\\_curso\\_antigo:r2013:alunos:trabalho\\_final:mariana.ramos.queiroz:start](http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2013:alunos:trabalho_final:mariana.ramos.queiroz:start)

Last update: **2020/08/12 06:04**