

# Milena Regina Wolf



Aluna de doutorado em Zoologia, UNESP-Campus de Botucatu. Linha de pesquisa: Ecologia de Animais Aquáticos, especificamente crustáceos Decapoda.

## Exercícios

[exercicios](#)

## Trabalho Final

**Proposta A:** Estimativa populacional de populações selvagens através de captura e recaptura

Em consequência das alterações ambientais muitas espécies ficam suscetíveis a um decréscimo populacional. Informações provenientes de estimativas do tamanho da população são utilizadas pela IUCN (International Union for Conservation of Nature), como um dos critérios para avaliar o estado de conservação de determinada espécie. Um modo de estimar o tamanho de uma população é capturar e marcar os indivíduos liberá-los em seguida e criar uma nova amostra para verificar qual a fração de indivíduos que carregam marcas. Essa técnica pode ser utilizada em populações abertas (caso mais comum, em que a população muda de tamanho e composição de nascimentos, mortes e movimentos) e fechada (não ocorre mudança no tamanho durante o período de estudo, ou seja, aquele em que os efeitos de nascimentos, mortes e movimentos são insignificantes). O objetivo dessa função será realizar em uma única função, o cálculo da estimativa do tamanho da população a partir dos dados em um `data.frame`. Há uma grande variedade de métodos utilizados para calcular essa estimativa dos quais escolhi o método de Schumacher e Eschmeyer para uma população fechada. A função:

1) O que vou inserir? Um `data frame` com as minhas anotações de captura e recaptura contendo: dia(t), Nº de animais capturados (Ct) e Nº de animais recapturados (Rt) e a área em metros quadrados; \*Usar o `if` e o `cat` para informar caso os dados não sejam um `data.frame`.

2) Calcular a partir desses valores dos dados obtidos no campo os seguintes valores:

Nº de novos animais marcados (número de animais capturados em cada dia menos o número de recapturados do mesmo dia);

Nº de animais marcados que estavam soltos (Mt) (número de novos animais marcados somados ao número de animais marcados que estavam soltos do dia anterior, é uma soma cumulativa na qual eu quero representar o quanto de animais marcados tenho do dia 1 para o dia 2 .... \*Pensei em usar o `for`, para montar esse cálculo, uma vez que não tenho apenas um valor para calcular, mas sim vários valores.

3) Retornar um `data.frame` com as novas colunas com os cálculo novos, não sei se seria possível...

4) Verificar previamente se as premissas dos métodos de marcação e recaptura foram devidamente satisfeitas:

Vou fazer uma regressão para verificar se os pontos Rt/Ct (proporção de animais recapturados) em função de Mt (número de animais marcados que estavam soltos) apresentam tendência linear. Isto é testado aplicando uma regressão linear com intercepto fixo na origem (0,0).

\*Usarei o plot e abline para as saídas gráficas;

\*Usarei o `lm = lm(y ~ 0 + ., data) # force a regression through the origin in R`

\* `Summary(lm)`

5) Estimar o tamanho da população (fórmula);

6) Calcular o valor dos intervalos de confiança.

Os intervalos de confiança de 95% de todo o cálculo do tamanho da população que é o  $1/\text{valor da estimativa} \pm t_{\alpha} \cdot \text{desvio padrão}$ .

\*Para esse cálculo vou precisar do valor crítico da distribuição t de Student (nesse caso fiquei em dúvida em como fazer, pensei em inserir como um dos argumentos da função, a pessoa que estará utilizando a função, de antemão saber e informar esse valor. Não sei se isso é possível;

\*O valor da variância de  $1/\text{valor da estimativa}$ , para o cálculo do desvio padrão;

\* Para o cálculo da variância e desvio, utilizarei valores calculados a partir dos meus cálculos de Mt, Rt e Ct. São cálculos simples, mas são vários:  $Ct \cdot Mt^2$ ,  $Rt \cdot Mt^2$ ,  $Rt^2/Ct$ ,  $(Rt \cdot Mt)^2$ . Assim como para calcular previamente o Mt, Ct e Rt, pensei em usar for, e retornar algo como uma tabela ou data.frame.

7) Densidade populacional (opcional):

É o valor da estimativa dividido pelo valor do metro quadrado da área amostrada.

8) O que eu quero que retorne?

O valor da estimativa populacional, dos intervalos de confiança e da densidade.

**Proposta B:** Verificar a correlação entre matrizes para dados multivariados com o teste de hipóteses

Achei interessante a ideia de poder utilizar métodos para realizar uma análise de comparação direta quando os descritores de resposta formam uma tabela de dados multivariados. Seria possível no caso, verificar qual fator abiótico, por exemplo, poderia estar determinando na composição de uma comunidade através da correlação entre matrizes e poder avaliar essa correlação com um teste de hipóteses. Portanto, minha segunda função seria realizar uma análise de dados, com a comparação entre duas matrizes, uma com as minhas espécies por exemplo (poderia ser composição por área amostrada, ou mês) e outra com as características do ambiente, por exemplo, (para a mesma área ou mês). Usaria para comparar essas matrizes o teste de Mantel. Gostaria de fazer, mas não sei se seria algo muito complexo...

1) Não sei se na minha função as matrizes de entrada já estariam como matrizes de similaridade ou

distância, ou se na própria função isso seria possível de ser feito, o usuário escolhendo nos argumentos qual índice de similaridade usar...

- 2) Seria necessário permutar as matrizes e recalcular a correlação
- 3) Resultado do teste de Mantel;
- 4) Produção de um gráfico final entre as distâncias, com um modelo lm entre essas distâncias e um abline;
- 5) Retorno seria: o resultado do teste, com o valor do coeficiente de correlação e do p.

### Comentários

Milena, achei a sua proposta A bem interessante e viável. Alguns comentários seguindo os seus itens:

2) Não entendi totalmente o que você pretende calcular na soma cumulativa, mas as funções diff e cumsum podem te ajudar. 3) Incluir novas colunas no seu data frame original? É possível e simples. 4) E se as premissas forem violadas? Você pode usar o if e retornar uma mensagem para o usuário caso isso aconteça. 6) Para obter o valor crítico da distribuição t Student para um dado valor de graus de liberdade: função qt. 8) Defina o tipo de objeto de saída da função de forma mais específica.

A sua proposta B me pareceu interessante, mas ficou um pouco obscura para mim. Na minha opinião a sua proposta A está formulada de forma mais clara e por isso, acredito que vai ser tranquilo de implementá-la. Talvez alguém mais venha fazer comentários aqui.

— [Lucas Medeiros](#)

## Trabalho Final

**PROPOSTA A:** *Estimativa do tamanho de populações selvagens através do método de captura e recaptura*

### Página de Ajuda/Help

pop.estimates  
Documentation

package:unknown

R

Estimation of population size in wild populations.

## Description:

A function to calculate the estimation of population size in wild populations using the Schumacher and Eschmeyer mark-recapture method for closed populations. This function also calculates the population density based on the size of the sampled area.

It is assumed that the assumptions for the application of the method in the field are satisfied

## Usage:

```
pop.estimates<-function(x,col.t, col.Ct, col.Rt, y, col.lar, col.comp, s, p, df, meas)
```

## Arguments:

`x` the input data should be of class `data.frame` containing the information capture and recapture of animals (see Details).  
`col.t` the collumn number of the "day" (t), corresponding to the duration of the field study in each occasion, composed of a specified number of consecutive days (in `data.frame` x).  
`col.Ct` the collumn number of "the total number of individuals captured in the t(th) sampling" (Ct)(in `data.frame` x).  
`col.Rt` the collumn number of "the total number of individuals recaptured in the t(th) sampling" (Rt)(in `data.frame` x).  
`y` the input data should be of class `data.frame` containing the information of "length" and "width" of the area where the collections were made (see Details).  
`col.len` the collumn number of the variable "lenght" of the sampled area (in `data.frame` y).  
`col.wid` the collumn number of the variable "width " of the sampled area (in `data.frame` y).  
`s` number of samples (see Details).  
`p` vector of probabilities (see Details).  
`df` degrees of freedom (see Details).  
`meas` number of times that the trapezoids area calculation should be performed.

## Details:

The assumptions that must be followed in the field for the application of the Schumacher and Eschmeyer mark-recapture method for closed populations are: a) the population should be closed (isolated area); b) the capture of animals is random; c) the mark is not lost during the study period; d) marked and unmarked animals have the same chance of being caught and e) the mark does not affect the chances of capture.

The adequacy of the Schumacher and Eschmeyer method for the estimation was tested by applying through-the-origin regression on the scatter-plot of the proportion of marked animals in samples ( $Rt/Ct$ ) against the accumulated

number of marked animals ( $M_t$ ).

When the assumptions required to validate the method are met, a positive linear relationship between these variables is expected with significant results (significant line checked by Student's t test,  $\alpha = 0.05$ ).

The data contained in data.frame x should be: the day information (t) (this calculation is usually performed with field data from a week in the month), number of captured animals ( $C_t$ )

that corresponds to how many animals were captured on a given day (and are not marked) and number of recaptured animals ( $R_t$ ), where every day the animals caught (unmarked) are marked and can be caught again in the next day,

therefore it is necessary to measure (count) captured animals with marks and also include this information in the worksheet.

The data contained in data.frame y should be: the length and width of the sampled area. The measures should be taken of sessions in order to find the trapezoids area and finally the total area. The results of the total area and the population size estimate will be used to calculate population density.

The number of samples (s) corresponds to a short period of consecutive 24-hour interval sampling events (the population size estimate is usually performed with 6 field days in the month).

The confidence intervals around the population size estimate were obtained by taking reciprocals of the results obtained from the normal approximation by using the critical value of the Student's t distribution for  $s-2$  degrees of freedom. The function qt was used to access the critical value of the Student's t distribution. The probability value (p) is one of the arguments of qt function and should be determined previously.

The degrees of freedom (df) is also used in qt function, and should be considered as the result of subtracting 2 from the whole number of samples (Example: 6 minus 2 = 4). Are commonly used the the 95% confidence intervals with the critical value of the Student's t distribution ( $\alpha = 0.05$ ).

Value:

The result of the function will be displayed in R-console (list) and in a new graphic window (with a regression graphic). A list will be displayed, containing:

Regression summary: a summary of the regression results (through the origin) will be provided.  
Regression probability value (The p value): the p-value of the linear model.

Anova of Regression: an Anova performed based on the results obtained from regression.

Population size estimate: the result from the population size estimate equation of the Schumacher and Eschmeyer mark-recapture method.

The Critical Value (Student's t Distribution): the critical value of the t distribution, which is based on the adopted value of significance and degrees of freedom. The critical value will be used in order to calculate the confidence intervals.

Variance: the variance value. The value will be used in order to calculate the standard error

Standard Error: the standard error value of the estimation. The standard error will be used in order to calculate the confidence intervals.

Confidence Intervals: the confidence intervals values around the population size estimate.

Population density: the population density value.

Warning:

The input data of the mark-recapture of the species and the length and width of the area should be a data.frame.

The assumptions for the application of the method in the field should be satisfied. For verify this, the line of regression should be significant.

Author(s):

Milena Regina Wolf  
milenarwolf@gmail.com

References:

Krebs, C. J. 1999. Ecological Methodology. 2nd Edition. Benjamin/Cummings, Menlo Park, CA.

Seber, G. A. F. 1982. The Estimation of Animal Abundance and Related Parameters. 2nd Edition. Charles Griffin, London.

Zar, J. H. 1996. Biostatistical Analysis. 3rd Edition. Prentice Hall, Upper Saddle River, NJ.

See Also:

TDist, lm (to access regression through the origin)

Examples:

```
# Download the "popul.txt" and "area.txt" to run the example below
and save it in the working directory that will be used in R. Save them on
objects called popul and area.
```

```
# Function example
```

```
pop.estimates(x= popul, col.t= 1, col.Ct= 2, col.Rt= 3, y= area,
col.wid= 1, col.len= 2, s= 6, p= 0.975, df= 4, meas= 16)
```

## Código da Função pop.estimate

```

pop.estimate<-function(x, col.t, col.Ct, col.Rt, y, col.wid, col.len, s, p,
df, meas) # Nome da função e seus argumentos
{
  if(class(x)!= "data.frame") # Esse comando verifica se os dados de
  entrada do argumento "x" pertencem a classe data.frame.
  {
    stop("input data does not belong to class data.frame") #
    Interrompe a execução da expressão atual e executa uma ação de erro se os
    dados não pertencerem a classe data.frame
  }
  if(class(y)!= "data.frame") # Esse comando verifica se os dados
  de entrada do argumento "y" pertence a classe data.frame.
  {
    stop("input data does not belong to class data.frame") #
    Interrompe a execução da expressão atual e executa uma ação de erro se os
    dados não forem um data.frame
  }

#####
#####
# Parte 1: cálculos de NOVOS ANIMAIS MARCADOS e ANIMAIS MARCADOS SOLTOS
(Mt) #####
  # Nessa parte novos cálculos serão inseridos a partir dos dados de
  animais capturados (Ct) e recapturados (Rt) no data.frame (x).####
#####
#####
  N.novos.marc<-(x[,col.Ct]-x[,col.Rt]) # O objetivo desse comando é
  verificar a diferença do número de animais capturados em cada dia e do
  número de animais recapturados do mesmo dia.
  soma.cumulativa<-cumsum(N.novos.marc) # Comando para calcular o
  número de animais marcados que estavam soltos (Mt). A soma cumulativa tem
  como objetivo mensurar o Mt e o seguinte exemplo será utilizado para a
  explicação:
  # o número de animais novos marcados é anotado por dia durante 6 dias
  (corresponde aos dias de amostragem), mas para saber quantos animais
  marcados estavam soltos (em um momento determinado) é preciso fazer uma soma
  cumulativa,
  # no dia 1 considera-se 0 porque foi a primeira coleta e ainda não
  havia nenhum animal marcado, no dia 2 vão ter os animais marcados do dia 1
  (139); no dia 3 os novos marcados do dia 2 + o número de marcados que
  estavam soltos do dia 1 (137+139=276);
  # no dia 4 os novos marcados do dia 3 + o número de marcados que
  estavam soltos dos dias 1 e 2 (120+276=396).
  # Nos três objetos seguintes, esse novo vetor resultante da soma
  cumulativa será corrigido, para chegar ao que realmente é preciso que é o
  objeto N.marcados.soltos.Mt. Porque a função cumsum apenas faz as somas
  cumulativas, mas não considera o dia 1 como 0
  # e nem desconsidera a última soma que seria referente ao 7º dia, que
  não existe nos dados porque no exemplo as coletas foram realizadas apenas

```

durante 6 dias.

```
novo.vetor<-c(0,soma.cumulativa) # 0 valor 0 é colocado na primeira posição do objeto da soma cumulativa, porque no 1º dia de captura não havia nenhum animal marcado solto, só vão existir animais marcados a partir do 2º dia.
```

```
novo.vetor1 = novo.vetor[1 : length(novo.vetor) - 1]# 0 último valor do objeto da soma.cumulativa esta sendo retirado, porque é preciso considerar que são apenas 6 dias (nesse exemplo), porque se somar o 52+578, teria que ter coleta no 7º dia.
```

```
N.marcados.soltos.Mt<-novo.vetor1 # Finalmente, esse é o objeto criado a partir da: soma.cumulativa.
```

```
x<-data.frame(x,N.novos.marc,N.marcados.soltos.Mt) # Os novos valores calculados (Novos animais marcados e animais marcados soltos Mt) são adicionados ao data.frame original (x).
```

```
#####  
#####  
#####
```

```
# Parte 2: REGRESSÃO
```

```
#####  
#####  
#####
```

```
# Objetivo dessa parte é verificar se as premissas de amostragem em algum momento foram violadas.#####  
#####
```

```
# Uma regressão da proporção de recapturados/capturados em função do nº de marcados que estavam soltos será realizada.#####  
###
```

```
#####  
#####  
#####
```

```
proporcao<-c(x[,col.Rt]/x[,col.Ct]) # É realizada uma proporção do número de animais recapturados (Rt) pelo número de animais capturados (Ct) no total por dia.
```

```
regressao.premissas<-lm(proporcao~0+N.marcados.soltos.Mt) # É realizada uma regressão da proporção em função do número de animais marcados que estavam soltos (Mt) forçada à origem (through the origin).
```

```
# A reta passa pela origem porque quando foi realizada a primeira coleta não há nenhum animal recapturado.
```

```
sumario.regressao<-summary(regressao.premissas) # Função que apresenta um resumo do modelo linear (estatísticas descritivas dos resíduos; teste t dos coeficientes de regressão;erro padrão da estimativa;
```

```
# coeficiente de determinação e coef. de det. ajustado e teste F geral do modelo).
```

```
valor.p<-summary(regressao.premissas)$coefficients[1,4] # Retira o valor do p calculado com o teste t de Student no sumário do modelo linear.
```

```
if(valor.p>0.05)# Verifica se o valor do p do modelo é
```



```

significativo, baseado em um alpha=0.05
    {
        warning("The line of the regression was not
significant") # Dá um aviso se o valor do p for maior do que 0.05.
    }

# Se o aviso anterior não foi dado e se a reta do modelo for
significativa, o usuário poderá prosseguir com tranquilidade uma vez que as
premissas do método não foram violadas.
    anova.regressao<-anova (regressao.premissas)# Foi utilizada a função
anova que apresenta a tabela de análise de variância, tendo as variáveis
preditoras como fatores.
    x11() # Essa função foi utilizada para abrir uma janela gráfica
(dispositivo de tela) quando a função for rodada para que o usuário possa
ver o gráfico da regressão.
    gráfico.regressao<-plot(proporcao~N.marcados.soltos.Mt) # O gráfico
da regressão: proporção em função do número de animais marcados que estavam
soltos (Mt).
    abline(regressao.premissas, col="red") # Esse comando foi utilizado
para inserir uma linha de tendência criada a partir do modelo linear
(regressao.premissas).

#####
#####
#####
# Parte 3: Estimativa do tamanho populacional
#####
#####
#####
# Nessa parte o cálculo da estimativa será realizado. O método
utilizado será o de Schumacher &
Eschmeyer.#####
#####
# A fórmula da estimativa é: a somatória do número de animais
capturados (Ct) multiplicado pelo número de animais marcados que estavam
soltos (Mt) elevado ao quadrado e esse cálculo é dividido pela somatória do
#####
# número de animais recapturados (Rt) multiplicado pelo número de
animais marcados que estavam soltos (Mt).Essa fórmula esta dividida em
partes:
#####
#####
#####
#####
Mt.quadrado<-N.marcados.soltos.Mt^2 # Esse objeto corresponde ao
número de animais marcados que estavam soltos elevado ao quadrado.
    Ct.Mt.quadrado<-sum(x[,col.Ct]* Mt.quadrado) # Foi realizado a
somatória do número de animais capturados (Ct) multiplicada pelo
Mt.quadrado.
    Rt.Mt<-sum(x[,col.Rt]* N.marcados.soltos.Mt) # Foi realizado a
somatória do número de animais recapturados (Rt) multiplicada pelo número de
animais marcados que estavam soltos (Mt).

```

```
#####  
#####  
##### RESULTADO DA ESTIMATIVA DO TAMANHO DA POPULAÇÃO  
#####  
#####  
#####  
#####  
N<-Ct.Mt.quadrado/Rt.Mt # Cálculo final da fórmula da estimativa do  
tamanho da população.  
  
#####  
#####  
#####  
#####  
# Parte 4: Cálculo da VARIÂNCIA  
#####  
#####  
#####  
#####  
# A variância de 1/N é calculada com a somatória do número de animais  
recapturados (Rt) elevado ao quadrado e dividido pelo número de animais  
capturados (Ct); subtraindo da somatória do Rt multiplicado pelo número de  
animais marcados soltos (Mt) elevado ao quadrado; #####  
# e dividindo pela somatória de Ct multiplicado por Mt elevado ao  
quadrado. Todo esse cálculo é dividido pelo número de amostras/dias (s)  
menos 2.  
#####  
#####  
#####  
#####  
#####  
#####  
#####  
# Os dois comandos seguintes serão realizados porque fazem parte da  
fórmula da variância.  
Rt.quadrado<-x[,col.Rt]^2 # Número de animais recapturados (Rt)  
elevado ao quadrado.  
Rt.quadrado.Ct<-sum(Rt.quadrado/x[,col.Ct]) # A somatória do  
Rt.quadrado dividido pelo Ct.  
Variancia<- (Rt.quadrado.Ct- (Rt.Mt^2/Ct.Mt.quadrado))/(s-2)# Resultado  
da fórmula da variância.  
  
#####  
#####  
#####  
# Parte 5: Cálculo dos INTERVALOS DE CONFIANÇA  
#####  
#####  
#  
# Os intervalos de confiança de 1/N (partir da estimativa do tamanho  
da população serão calculados). Na fórmula o valor crítico da distribuição t
```

```

e o erro padrão serão utilizados. #####
#####
#####
#####
# A fórmula do erro padrão de 1/N é o resultado da variância dividida
pela somatória de Ct multiplicado por Mt elevado ao quadrado. A fórmula será
dividida em duas partes:
erro.padrao1<-Variancia/Ct.Mt.quadrado # Primeira parte da fórmula a
divisão da variância pelo Ct*(Mt^2).
erro.padrao<-sqrt(erro.padrao1) # Segunda parte da fórmula, raiz
quadrada do valor calculado no vetor "erro.padrao1".
# Os intervalos de confiança em torno da estimativa do tamanho da
população foram obtidos tomando recíprocos dos resultados obtidos a partir
de uma aproximação normal, utilizando o valor crítico de distribuição t de
Student na fórmula dos intervalos.
critical.tvalue<-qt(p,df,lower.tail=TRUE,log.p=FALSE)# Função "qt"
(The Student t Distribution) é usada para encontrar o valor crítico da
distribuição t, para determinado grau de liberdade e valor do alpha
oferecidos pelo usuário. O argumento lower.tail é usado para determinar se
# a cauda superior e inferior farão parte do cálculo ou se a função
retornará a probabilidade de uma das partes da distribuição. Nesse caso são
as duas partes lower.tail=TRUE.
intervalo1<-1/N+critical.tvalue*erro.padrao # Fórmula do Intervalo de
confiança.
intervalo2<-1/N-critical.tvalue*erro.padrao # Fórmula do Intervalo de
confiança.
# Os próximos dois passos são realizados porque o uso do recíproco
(1/valor) no cálculo do intervalo de confiança da estimativa de tamanho
populacional ocorre porque o que acontece é que as fórmulas envolvidas nos
cálculos da variância e do erro padrão são definidos para 1/N(chapeu) e não
diretamente para N(chapeu).
# Assim, os valores calculados dos limites inferior e superior do
intervalo de confiança também referem-se a 1/N(chapeu) e é por isso que deve
se obter o recíproco para se chegar aos valores diretamente relacionados com
o N(chapeu) estimado.
intervalo3<-1/intervalo1 # Resultado do cálculo dos intervalos de
confiança.
intervalo4<-1/intervalo2 # Resultado do cálculo dos intervalos de
confiança.

#####
#####
#####
#####
# Parte 6: Cálculo da DENSIDADE POPULACIONAL
#####
#####
#####
# A densidade populacional baseia-se no resultado da estimativa do
tamanho da população em relação ao tamanho total da área (precisa ser
delimitada) e deve ser medida para que esse cálculo possa ser realizado. As

```

```
medidas do comprimento e largura da área são tomadas por seções formando
#####
#####
#####
#####
#####
result.trapezio<-numeric(0) # Vetor vazio para armazenar os
resultados do que será calculado um número determinado de número de vezes
por meio da função "for" (que cria ciclos de eventos).
    for (i in 1:meas) # Repete a operação de acordo com o número
de medidas tomadas e deverá ser fornecida pelo usuário por meio do argumento
"meas" e armazena no vetor "result.trapezio".
        {
            result.trapezio[i]<-
((y[,col.wid][i]+y[,col.wid][i+1])/2)*y[,col.len][i] # Calcula a área dos
trapézios pela soma das larguras superior e inferior dividido por 2 e vezes
o comprimento.
        }
    area.total<-sum(result.trapezio) # Com a soma de toda a área dos
trapézios calculados a área total é calculada.
#####
###
##### RESULTADO DA DENSIDADE
POPULACIONAL#####
#####
#
    Densidade<-N/area.total # Resultado da fórmula da densidade
populacional
    resulta<-list("Sumário da Regressão"= sumario.regressao," Valor da
Probabilidade da Regressão (Valor do p)"= valor.p, "Anova da Regressão"=
anova.regressao, "Estimativa do Tamanho da População"= N, "Variância"=
Variancia, "Erro padrão"= erro.padrao,"O Valor Crítico da Distribuição t de
Student"= critical.tvalue,"Intervalo de confiança"= intervalo3, "Intervalo
de confiança"=intervalo4, "Densidade Populacional"= Densidade)# Objeto que
contém uma lista com os resultados da minha função que serão visualizados no
R-console.
    return(resulta)# Retorna os resultados do objeto resulta que possui
os resultados da função na forma de uma lista.
}
```

### Arquivos para Download *Example do Help*

[popul.txt](#)

[area.txt](#)

### Arquivos para Download

[help\\_da\\_funcao\\_pop.estimates.txt](#)

[codigo\\_da\\_funcao\\_pop.estimate.r](#)

From:

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

Permanent link:

[http://ecor.ib.usp.br/doku.php?id=05\\_curso\\_antigo:r2014:alunos:trabalho\\_final:milena:start](http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2014:alunos:trabalho_final:milena:start) 

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