

MARCIO SEIJI SUGANUMA



Biólogo formado pela Universidade Estadual de Londrina-UEL.

M.Sc. em Ciências Biológicas com ênfase Botânica (UEL).

Aluno de Doutorado em Ciências da Engenharia Ambiental (Escola de Engenharia de São Carlos-EESC/USP).

Tese: “Análise de fatores que condicionam o sucesso na restauração de matas ciliares”

Orientado pela Dr^a Giselda Durigan.

Contato: marciosuganuma@gmail.com

Exercícios

[exec](#)

Links

<http://www.eesc.usp.br/ppgsea/index.php?link=home>

Trabalho Final

Plano A

Avaliação da qualidade de um habitat pelo método: “Habitat hectares condition score” (Parkes et al., 2003)

Avaliar a “qualidade” e as condições de uma vegetação nativa ou em regeneração é uma tarefa muito complicada que envolve a coleta de muitos dados e resultados um tanto subjetivos. Essa abordagem de avaliação de “habitat hectares” desenvolvido pelo grupo de estudos da Victoria Native Vegetation Management Framework, se utiliza de sete variáveis ambientais e mais três variáveis da paisagem (Tabela.1). Sendo útil para avaliar ecossistemas naturais fragmentados, como ecossistemas em restauração (Munro et al., 2009)

As variáveis ambientais, escolhida por um grupo de consultores e especialistas em botânica e ecologia são utilizadas de maneira comparativa com um ambiente de referência não perturbado. Essa metodologia visa uma rápida avaliação do ambiente através de pontuações (Tabela.1), de maneira que possa ser aplicada por qualquer pessoa.

Tabela 1. Componentes e valores do “Habitat hectares condition score”.

	Componentes	Valor máximo (%)
Condições ambientais	Proporção de árvores grandes (DAP>80cm)	10
	Cobertura florestal	5
	Cobertura do subosque	25
	Cobertura de gramíneas invasoras	15
	Regeneração	10
	Serapilheira	5
	Troncos caídos(DAP>10cm)	5
Contexto da paisagem	Tamanho do fragmento	10
	Tipo de matriz	10
	Distância de uma fonte	5
	Total	100

(Modificado de Parkes et al.,2003)

Meu objetivo é escrever uma função no R, onde a entrada seja um data.frame com as variáveis em ordem (de acordo com as variáveis nas linhas e os locais nas colunas) em porcentagens, do ambiente estudado e do ecossistema de referência, essas porcentagens serão convertidas de acordo com as pontuações de Parkes et al. (2003), resultando em uma pontuação final para cada local. Adicional serão inseridas variáveis de escolha em algumas variáveis. Exemplo: Para a variável cobertura de gramíneas, a pontuação é diferente de ambientes 1-que naturalmente apresenta zero de gramíneas e 2-que apresenta uma cobertura natural entre zero e 50% e estava pensando em fazer uma saída gráfica de boxplot, sempre pareada entre o ecossistema estudado e o ecossistema de referência para todas as variáveis ambientais.

Referências:

Parkes, D.; Newell, G. and Cheal, D. 2003. Assessing the quality of native vegetation: The “habitat hectares” approach. [Ecological Management & Restoration](#), 2003:4, pp.29-38.

Munro, N.T.; Fischer, J.; Wood, J. and Lindenmayer, D.B. 2009. Revegetation in agricultural areas: The development of structural complexity and floristic diversity. [Ecological Applications](#), 19:5, pp.1197-1210.

Plano B

Fazer uma função que gere regressões simples (lineares, logarítmicas, exponenciais e até polinomiais), testando qual a melhor reta que se ajusta aos dados (pesnei somente em ver a confiabilidade pela variância), para comparar variáveis do ambiente (pensando em variáveis de vegetação florestal: área basal, número de regenerantes, ...) em função do tempo.

Terei como entrada um data.frame com essas várias variáveis nas colunas, sempre correlacionando-as com o tempo, que vai estar em outra coluna, assim tenho uma tabela de saída para cada uma das variáveis, com sua significância e variância.

Adicionalmente posso fazer uma saída gráfica com a melhor reta para cada variável.

Comentários

Acho que o plano A é factível, pode implementá-lo. — [Alexandre Adalardo de Oliveira](#) 2010/03/31 11:48

Página de ajuda

HHCS()

package:nenhum

R Documentation

Calcula o "Habitat hectares condition score" de Parkes et al. (2003)

Description:

Produz uma tabela com os resultados da pontuação para a avaliação de um ambiente, produzido por Parkes et al. (2003), utilizando as seguintes características do meio: comparação de número de árvores grandes (DAP>80cm), cobertura do dossel, componentes do subosque, cobertura de gramíneas invasoras, número de regenerantes, serapilheira, número de troncos caídos, tamanho do fragmento, tipo de matriz e distância de uma fonte de propágulos. Adicionalmente temos as médias e o box-plot para as sete primeiras variáveis citadas.

Usage:

```
HHCS<- function(x, y, exotic, weed, disturbed)
```

Arguments:

x: ambiente estudado

y: ambiente de referência

exotic: nível de alteração do ambiente por espécies invasoras: "0"= não alterado, "1"=alterado até 50%, "2"=alterado mais que 50%

weed: nível de vulnerabilidade à invasão de gramíneas: "0"=sem ou baixo, "1"=médio, "2"=alto

disturbed: nível de distúrbio do ambiente estudado: "0"= não alterado, "1"=alterado até 50%, "2"=alterado mais que 50%

Details:

Os resultados são comparativos, por isso deve-se entrar com os dados do ambiente estudado (x) no formato de um data.frame com 11 colunas, com os seguintes dados, nessa ordem: número de árvores grandes (DAP>80cm), % de cobertura do dossel, número de componentes do subosque (máximo de 6), % de cobertura de gramíneas invasoras, número de regenerantes, número de regenerantes com DAP> 1cm, % de cobertura de serapilheira, número de troncos caídos, tamanho do fragmento, tipo de matriz e distância de uma fonte de propágulos, onde, as três últimas variáveis (da paisagem), entram somente com um dado na primeira linha e não estão presentes no data.frame do ambiente de referência (y), que apresentará 8 colunas.

Os cálculos e as decisões de escolha foram feitas em cima das

características do ecossistema de referência(y) e nos argumentos de nível de espécies exóticas, nível de vulnerabilidade à gramíneas invasoras e nível de distúrbio do ambiente estudado (Parkes et al., 2003).

Value:

Uma matriz é gerada, com os resultados da pontuação para cada variável e as médias de cada variável para o ambiente estudado (x) e o ambiente de referência (y).

Adicionalmente é gerada uma saída gráfica, com 8 gráficos de box-plot, comparando os dados brutos do ambiente estudado com o ambiente de referência.

Warning:

Esperava construir uma função geral, mas para alguns caracteres ambientais a função ficou mais restrita para ambientes florestais.

É aconselhável uma revisão e/ou reajuste no "número de árvores grandes (DAP>80cm)" e "número de componentes de subosque" para outros ecossistemas.

Author(s):

Marcio Seiji Suganuma
marciosuganuma@gmail.com

References:

Parkes, D.; Newell, G. and Cheal, D. 2003. Assessing the quality of native vegetation: The "habitat hectares" approach.

[[<http://www.wiley.com/bw/journal.asp?ref=1442-7001>|Ecological Management & Restoration]], 2003:4, pp.29-38.\

Munro, N.T.; Fischer, J.; Wood, J. and Lindenmayer, D.B. 2009. Revegetation in agricultural areas: The development of structural complexity and floristic diversity.

[[<http://www.esapubs.org/esapubs/journals/applications.htm>|Ecological Applications]], 19:5, pp.1197-1210.

Examples:

#Para exemplo:\

```
x<-read.table(file="x.csv", header=T, sep=";", row.names=1)
```

```
y<-read.table(file="y.csv", header=T, sep=";", row.names=1) #referência não  
têm as 3 últimas colunas (paisagem)\
```

Arquivos para o exemplo:

[x.csv](#) [y.csv](#)

Código da função

```

HHCS<- function(x, y, exotic, weed, disturbed)
#onde: x=ecossistema estudado e y= ecossistema de referência
#exotic= nível de alteração do ambiente por espécies invasoras: "0"= não
alterado, "1"=alterado 50%,
"2"=alterado mais que 50%
#weed= nível de vulnerabilidade à invasão de gramíneas: "0"=sem ou baixo,
"1"=médio, "2"=alto
#disturbed= nível de distúrbio: "0"= não alterado, "1"=alterado 50%,
"2"=alterado mais que 50%
{
  nomes<-c("X","Y")
  x11()
  par(mfrow=c(2,4))
  boxplot(x[,1],y[,1], names=nomes, xlab="Large trees", ylab="Nº of large
trees")
  boxplot(x[,2],y[,2], names=nomes, xlab="Canopy cover", ylab="% of
canopy cover")
  boxplot(x[,3],y[,3], names=nomes, xlab="Understorey", ylab="Nº of
understorey components")
  boxplot(x[,4],y[,4], names=nomes, xlab="Weeds", ylab="% of weeds
cover")
  boxplot(x[,5],y[,5], names=nomes, xlab="Recruitment total", ylab="Nº of
recruitments")
  boxplot(x[,6],y[,6], names=nomes, xlab="Recruitment > 1cm", ylab="Nº of
recruitments > 1cm")
  boxplot(x[,7],y[,7], names=nomes, xlab="Litter", ylab="% of litter
cover")
  boxplot(x[,8],y[,8], names=nomes, xlab="Logs", ylab="Nº of logs")
  par(mfrow=c(1,1))
  media.x=mean(x)
  media.y=mean(y)
  recruit=round((media.x[6]/media.x[5])*100)
  percent=round(((c(media.x[1:8]))/media.y)*100)
  raio.cem.m<- (x[1,10])*0.03
  raio.um.km<- (x[2,10])*0.04
  raio.cinco.km<- (x[3,10])*0.03
  raio.total<-raio.cem.m+raio.um.km+raio.cinco.km
  result=matrix(c(NA),3,12)
  colnames(result)<-c("Large trees", "Canopy", "Understorey", "Weeds",
"Recruitment", "Recruitment.1", "Litter", "Logs", "Patch size",
"Neighbourhood", "Distance", "Total")
  rownames(result)<-c("Mean.x", "Mean.y", "Score")
  result[1,]<-c(media.x,NA)
  result[2,]<-c(media.y,NA,NA,NA,NA)

  { #Início da pontuação por árvores grandes
    if(media.y[2]>70) #Dossel mais fechado (>70%)
    {

```

```
if(porcent[1]==0)
{
  result[3,1]<-0
}
if(porcent[1]>0 & porcent[1]<=20)
{
  result[3,1]<-3
}
if(porcent[1]>20 & porcent[1]<=40)
{
  result[3,1]<-4
}
if(porcent[1]>40 & porcent[1]<=70)
{
  result[3,1]<-6
}
if(porcent[1]>70 & porcent[1]<=100)
{
  result[3,1]<-8
}
if(porcent[1]>100)
{
  result[3,1]<-10
}
}
if(media.y[2]>30 & media.y[2]<70) #Dossel entre 30 e 70%
{
  if(porcent[1]==0)
  {
    result[3,1]<-0
  }
  if(porcent[1]>0 & porcent[1]<=20)
  {
    result[3,1]<-2
  }
  if(porcent[1]>20 & porcent[1]<=40)
  {
    result[3,1]<-3
  }
  if(porcent[1]>40 & porcent[1]<=70)
  {
    result[3,1]<-5
  }
  if(porcent[1]>70 & porcent[1]<=100)
  {
    result[3,1]<-7
  }
  if(porcent[1]>100)
  {
    result[3,1]<-9
  }
}
```

```
    }
  }
  if(media.y[2]<30) #Dossel mais aberto (<30%)
  {
    if(porcent[1]==0)
    {
      result[3,1]<-0
    }
    if(porcent[1]>0 & porcent[1]<=20)
    {
      result[3,1]<-1
    }
    if(porcent[1]>20 & porcent[1]<=40)
    {
      result[3,1]<-2
    }
    if(porcent[1]>40 & porcent[1]<=70)
    {
      result[3,1]<-4
    }
    if(porcent[1]>70 & porcent[1]<=100)
    {
      result[3,1]<-6
    }
    if(porcent[1]>100)
    {
      result[3,1]<-8
    }
  } #Final da pontuação de árvores grandes
}
{ #Início da pontuação do dossel
  if(media.y[2]>70) #Dossel mais fechado(>70%)
  {
    if(porcent[2]>90)
    {
      result[3,2]<-0
    }
    if(porcent[2]>50 & porcent[2]<=90)
    {
      result[3,2]<-3
    }
    if(porcent[2]<=50)
    {
      result[3,2]<-5
    }
  }
  if(media.y[2]>30 & media.y[2]<=70) #Dossel entre 30 e 70%
  {
    if(porcent[2]>90)
    {
      result[3,2]<-0
    }
  }
}
```

```
    }
    if(porcent[2]>50 & porcent[2]<=90)
    {
        result[3,2]<-2
    }
    if(porcent[2]<=50)
    {
        result[3,2]<-4
    }
}
if(media.y[2]<=30) #Dossel mais aberto (<30%)
{
    if(porcent[2]>90)
    {
        result[3,2]<-0
    }
    if(porcent[2]>50 & porcent[2]<=90)
    {
        result[3,2]<-1
    }
    if(porcent[2]<=50)
    {
        result[3,2]<-3
    }
}
} #Final da pontuação de dossel
{ #Início da pontuação por formas de vida no subosque (utilizei 6
clasificações de forma de vida)
    if(porcent[3]==0) #Ausencia total do estrato e das formas de vida
    {
        result[3,3]<-0
    }
    if(porcent[3]>0 & porcent[3]<50) #Com 1 ou 2 classes de forma de vida
    {
        result[3,3]<-5
    }
    if(porcent[3]>=50 & porcent[3]<90) #Com 3, 4 ou 5 classes de forma de
vida
    {
        if(exotic=="2")
        {
            result[3,3]<-10
        }
        if(exotic=="1" | exotic=="0")
        {
            result[3,3]<-15
        }
    }
    if(porcent[3]>=90) #Com 6 classes de forma de vida
    {
```

```
    if(exotic=="2")
    {
        result[3,3]<-15
    }
    if(exotic=="1")
    {
        result[3,3]<-20
    }
    if(exotic=="0")
    {
        result[3,3]<-25
    }
} #Final da pontuação por formas de vida no subosque
{ #Início da pontuação por gramíneas invasoras
if(weed=="0") #Risco baixo de invasão
{
    if(media.x[4]>50)
    {
        result[3,4]<-4
    }
    if(media.x[4]>25 & media.x[4]<=50)
    {
        result[3,4]<-7
    }
    if(media.x[4]>5 & media.x[4]<=25)
    {
        result[3,4]<-11
    }
    if(media.x[4]<=5)
    {
        result[3,4]<-15
    }
} #Final de weed="0"
if(weed=="1") #Risco médio de invasão
{
    if(media.x[4]>50)
    {
        result[3,4]<-2
    }
    if(media.x[4]>25 & media.x[4]<=50)
    {
        result[3,4]<-6
    }
    if(media.x[4]>5 & media.x[4]<=25)
    {
        result[3,4]<-9
    }
    if(media.x[4]<=5)
    {
        result[3,4]<-13
    }
}
```

```
    }  
  } #Final de weed="1"  
if(weed=="2") #Risco alto de invasão  
{  
  if(media.x[4]>50)  
  {  
    result[3,4]<-0  
  }  
  if(media.x[4]>25 & media.x[4]<=50)  
  {  
    result[3,4]<-4  
  }  
  if(media.x[4]>5 & media.x[4]<=25)  
  {  
    result[3,4]<-7  
  }  
  if(media.x[4]<=5)  
  {  
    result[3,4]<-11  
  }  
}  
} #Final da pontuação por gramíneas invasoras  
{ #Início da pontuação por regenerantes  
if(porcent[5]>=50) #Proporção total de regenerantes >=50%  
{  
  if(recruit<10)  
  {  
    result[3,5]<-1  
  }  
  if(recruit>=10 & recruit<30)  
  {  
    result[3,5]<-3  
  }  
  if(recruit>=30 & recruit<70)  
  {  
    result[3,5]<-6  
  }  
  if(recruit>=70)  
  {  
    result[3,5]<-10  
  }  
}  
if (porcent[5]<50)#Proporção total de regenerantes <50%  
{  
  if(recruit<10)  
  {  
    result[3,5]<-0  
  }  
  if(recruit>=10 & recruit<30)  
  {
```

```
    result[3,5]<-1
  }
  if(recruit>=30 & recruit<70)
  {
    result[3,5]<-3
  }
  if(recruit>=70)
  {
    result[3,5]<-5
  }
}
} #Fim da pontuação por regenerantes
{ #Início da pontuação por serapilheira
  if(media.y[7]>=50 & media.y[7]<=100) #Serapilhira na floresta nativa
  acima de 50%
  {
    if(porcent[7]>=0 & porcent[7]<10)
    {
      result[3,7]<-0
    }
    if(porcent[7]>=10 & porcent[7]<50 | porcent[7]>150)
    {
      result[3,7]<-3
    }
    if(porcent[7]>=50 | porcent[7]<=150)
    {
      result[3,7]<-5
    }
  }
  if(media.y[7]>=0 & media.y[7]<50) #Serapilhira na floresta nativa
  acima de 50%
  {
    if(porcent[7]>=0 & porcent[7]<10)
    {
      result[3,7]<-0
    }
    if(porcent[7]>=10 & porcent[7]<50 | porcent[7]>150)
    {
      result[3,7]<-2
    }
    if(porcent[7]>=50 & porcent[7]<=150)
    {
      result[3,7]<-4
    }
  }
} #Fim do score por serapilheira
{ #Início do score por troncos mortos
  if(media.y[8]/(media.y[1]*60)>=25) #Proporção do comprimento dos
  troncos maior que a metade do diametro médio (60) das árvores grandes
  {
    if(porcent[8]>=0 & porcent[8]<10)
```

```
{
  result[3,8]<-0
}
if(porcent[8]>=10 & porcent[8]<50 | porcent[8]>150)
{
  result[3,8]<-2
}
if(porcent[8]>=50 & porcent[8]<=150)
{
  result[3,8]<-4
}
}
if(media.y[8]/(media.y[1]*60)<25) #Proporção do comprimento dos
troncos menor que a metade do diametro médio (60) das árvores grandes
{
  if(porcent[8]>=0 & porcent[8]<10)
  {
    result[3,8]<-0
  }
  if(porcent[8]>=10 & porcent[8]<50 | porcent[8]>150)
  {
    result[3,8]<-2
  }
  if(porcent[8]>=50 & porcent[8]<=150)
  {
    result[3,8]<-4
  }
}
} #Fim do score por troncos mortos
{ #Início do score por tamanho da área (dados em ha)
  if(x[1,9]>=0 & x[1,9]<2) #Área entre 0 e 2 ha
  {
    result[3,9]<-1
  }
  if(x[1,9]>=2 & x[1,9]<5) #Área entre 2 e 5 ha
  {
    result[3,9]<-2
  }
  if(x[1,9]>=5 & x[1,9]<10) #Área entre 5 e 10 ha
  {
    result[3,9]<-4
  }
  if(x[1,9]>=10 & x[1,9]<20) #Área entre 10 e 20 ha
  {
    result[3,9]<-6
  }
  if(x[1,9]>=20 & disturbed=="2") #Área entre 20 e 40 ha
  {
    result[3,9]<-8
  }
}
```

```
        if(x[1,9]>=20 & disturbed=="1" | disturbed=="0") #Área entre maior
que 40 ha
        {
            result[3,9]<-10
        }
    } #Fim do score por tamanho da área (dados em ha)
    { #Início do score por entorno
        if(disturbed=="0")
        {
            result[3,10]<-round(raio.total)
        }
        if(disturbed=="1")
        {
            result[3,10]<-round(raio.total-1)
        }
        if(disturbed=="2")
        {
            result[3,10]<-round(raio.total-2)
        }
    } #Fim do score por entorno
    { #Início do score por distância (dados em km)
        if(x[1,11]>5)
        {
            result[3,11]<-0
        }
        if(x[1,11]>=1 & x[1,11]<=5)
        {
            result[3,11]<-2
        }
        if(x[1,11]<1 & x[1,11]>0)
        {
            result[3,11]<-4
        }
        if(x[1,11]<=0)
        {
            result[3,11]<-5
        }
        result[3,12]<-sum(result[3, 1:5],result[3, 7:11] )
    } #Fim do score por distância
    return(result)
}
```

Arquivo da função

[Função_Marcio](#)

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

Permanent link:
http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2010:alunos:trabalho_final:msuganuma:start 

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