

# Leandro Reverberi Tambosi



## Doutorando em Ecologia

### LEPAC - Laboratório de Ecologia da Paisagem e Conservação Biológica

### Departamento de Ecologia - Instituto de Biociências - USP

Tenho especial interesse em trabalhos relacionados aos efeitos da fragmentação florestal, regeneração de florestas secundárias e restauração de áreas degradadas voltados para a conservação da biodiversidade em paisagens fragmentadas.

[exerc](#)

## Proposta de Trabalho Final

### Plano A

#### **Definição do cronograma de restauração de propriedades rurais a partir de critérios de paisagem.**

Contexto: Diversas propriedades agrícolas estão procurando se adequar a legislação ambiental para evitarem autuações e para terem acesso a créditos de organismos internacionais. Uma das ações mais frequentes para a adequação ambiental de propriedades rurais é a restauração florestal de áreas irregularmente ocupadas como áreas de preservação permanente e áreas com baixa aptidão agrícola. As propriedades rurais de usinas produtoras de cana-de-açúcar geralmente apresentam grandes áreas irregulares como passivos ambientais e os projetos de restauração chegam a durar mais de 10 anos. Geralmente as ações de restauração são planejadas por propriedade, mas não levam em conta o benefício para a estrutura e configuração da paisagem que a restauração de cada propriedade proporcionará.

Tendo em vista este contexto, será desenvolvida uma função no R que permitirá calcular o benefício que cada propriedade proporcionará à estrutura da paisagem. Esse cálculo será feito pela razão entre a área restaurada em cada propriedade e a área contínua formada após a restauração. Dessa forma pretende-se sugerir uma ordem de restauração que priorize as propriedades que trarão mais benefícios à estrutura da paisagem. Além disso, a função retornará um gráfico que possuirá o benefício à estrutura da paisagem em função do tempo de restauração, além de uma curva de benefício à estrutura da paisagem para restaurações das propriedades de forma aleatória. Dessa forma será possível comparar a estratégia adotada para definição do cronograma de restauração com cenários aleatórios.

### Comentário

O plano A parece bom e o B tb. Ambos são bons desafios. Faltou definir melhor qual o formato dos dados de entrada, de saída e quais serão os procedimentos adotados na função. Sem isso fica difícil avaliar a viabilidade. Como já conversamos pessoalmente, parece-me que o plano A é factível,

portanto defina qual o formato do objeto de entrada, saída e procedimentos na função. Outra ponto importante e definir se irá usar algum argumento para possibilitar mudanças de procedimentos ditada pelo usuário.

— [Alexandre Adalardo de Oliveira](#) 2010/03/31 13:18

## Plano B

### Função para analisar a mudança de cobertura das terras

Criar uma função que trabalha com mapas de uso e cobertura das terras em diferentes datas, no formato matricial, para identificar as áreas em que ocorreram desmatamentos e regeneração natural na paisagem. A função ira gerar imagens de saída identificando os locais de desmatamento e regeneração natural, bem como a idade estimada dos fragmentos de vegetação natural remanescentes na paisagem. Além disso, será apresentado o resultado da quantificação das áreas de desmatamento, regeneração natural e remanescentes de vegetação na paisagem.

## PÁGINA DE AJUDA DA FUNÇÃO

restore.strat  
Documentation

package:unknown

R

Comparação do ganho de conectividade para diferentes cronogramas de restauração

#### Description:

A partir dos dados de mapeamento de uso das terras de um conjunto de propriedades rurais, verifica o ganho de conectividade dos fragmentos florestais durante a execução de projetos de restauração florestal e compara com possíveis situações sem planejamento para definição do cronograma de restauração.

#### Usage:

```
restore.strat(x)
```

#### Arguments:

x    tabela de dados gerada a partir do mapeamento de uso e cobertura das propriedade

A tabela de entrada de dados deve conter sete colunas de dados da seguinte forma:

coluna 1: deve possuir a identificação da classe da área mapeada, sendo 1 para vegetação natural e 4 para áreas a serem restauradas

coluna 2: identificador numérico e exclusivo para cada mancha mapeada, independente da classe da mancha

coluna 3: área da mancha mapeada e identificada na coluna 2. O valor de área pode estar em qualquer unidade desde que a mesma unidade seja mantida para os outros campos da tabela que deverão conter valor de área

coluna 4: identificador exclusivo para cada propriedade a ser restaurada

coluna 5: área de cada mancha da coluna 2 localizada no interior da propriedade da coluna 4. A área deve estar na mesma unidade da coluna 3.

coluna 6: área de uma mancha contínua formada após a restauração completa das propriedades. A área deve estar na mesma unidade das outras colunas com valores de área.

coluna 7: identificados numérico e exclusivo para cada mancha contínua formada após a restauração das propriedades.

#### Details:

A tabela requerida para executar a função geralmente é obtida com cruzamento de diferentes planos de informação em ambiente SIG. Neste caso, a função foi planejada para rodar com dados gerados do cruzamento de limites de propriedades com mapeamentos de vegetação e áreas a serem restauradas e por fim com o cenário após a restauração total. Nesse caso ocorre uma divisão das manchas das manchas das diferentes camadas de informação geográfica e alguns valores aparecerão repetidos (por exemplo, fragmentos de vegetação que apresentam uma área no mapeamento e ao serem cruzados com os limites das propriedades eles poderão ser divididos e os valores de área e identificação desses fragmentos aparecerão uma vez em cada propriedade.

A função compara o ganho de conectividade de fragmentos florestais em função da quantidade de área restaurada no conjunto de propriedade. Essa ganho de conectividade é medido para uma situação na qual se prioriza a restauração de manchas que resultarão em uma maior relação área conectada/área restaurada e uma outra forma de priorização na qual se prioriza a restauração de áreas que formarão maiores áreas contínuas de vegetação após a restauração. Nesse caso a ganho de conectividade é calculado pela quantidade de vegetação previamente existente na área contínua formada após a restauração e subtrai-se o valor da maior mancha de vegetação pré-existente dentro da mesma área contínua após a restauração.

#### Value:

A função retorna os gráficos do ganho de conectividade em função da área de restauração executada e futuramente apresentará nos mesmos gráficos a comparação dessas duas estratégias com cenários de restauração das áreas de maneira aleatória.

A função retornará as tabelas com a ordem indicada de restauração para as

diferentes estratégias simuladas (ainda não implementado)

Warning:

A função apresenta os resultados de duas estratégias de restauração, mas é necessário conhecer as condições locais para adequar o cronograma à necessidade de restauração em função de outros fatores que não são considerados nas análises da estrutura da paisagem.

Note:

A função deverá passar por modificações frequentes para incorporar outras classes de cobertura à serem restauradas, bem como outras estratégias de definição de cronograma. Sugestões de desenvolvimento e notificações de problemas encontrados serão muito bem recebidas pelo autor.

Author(s):

Leandro Reverberi Tambosi

letambosi at hotmail.com

References:

Por favor, visite os sites dos laboratórios para lista de publicações e trabalhos em andamentos

<http://eco.ib.usp.br/lepac/paisagem.htm>

<http://lerf.esalq.usp.br>

See Also:

~~objects to See Also as 'help', ~~~

Examples:

```
restore.strat(app)
```

## CÓDIGO DA FUNÇÃO

```
restore.strat <- function(x)
{
  x=dados
  rest.id<-unique(dados[,7])          #cria o vetor com os id dos frags a
  serem restaurados
  final.area<-c(rep(NA,length(rest.id))) #cria o vetor com a area final
```

```

continua apos a restauracao
  rest.area<-c(rep(NA,length(rest.id)))    #cria a area a ser restaurada
  forest.area<-c(rep(NA,length(rest.id)))  #cria a area de floresta dentro
da area que sera restaurada
  maxforest.area<-c(rep(NA,length(rest.id))) # cria a maior area de
floresta ja existente na area a ser restaurada
  connected.area<-c(rep(NA,length(rest.id))) # cria o vetor para o
calculado da area que sera conectada( fragmentos ja existentes na area a ser
restaurada - maio mancha de vegetacao ja existente
  pct.connected.area<-c(rep(NA, length(rest.id))) # porcentagem de area
que sera conectada durante a restauracao restaurada
  pct.restored.area<-c(rep(NA,length(rest.id))) # porcentagem da area
total a ser restaurada
  con.rest.ratio<-c(rep(NA,length(rest.id))) #relacao ganho de
conectividade por area restaurada
  for (i in 1:length(rest.id))
  {
    final.area[i]<-tapply(dados[,6],dados[,7]==rest.id[i],max)[2]
#calcula area final
    rest.area[i]<-
tapply(dados[,5],dados[,1]==4&dados[,7]==rest.id[i],sum)[2] ##verificar se o
valor de area a restaurar vai ficar 4
    forest.area[i]<-
tapply(dados[,5],dados[,1]==1&dados[,7]==rest.id[i],sum)[2] #calculado de
area de floresta
    maxforest.area[i]<-
tapply(dados[,3],dados[,1]==1&dados[,7]==rest.id[i],max)[2] #calculado da
maior mancha de floresta
    connected.area[i]<-
tapply(dados[,5],dados[,1]==1&dados[,7]==rest.id[i],sum)[2]-
maxforest.area[i] #calculado de area conectada
    rest.area[is.na(rest.area)]<- 0 #remover os 0s e NAs
    forest.area[is.na(forest.area)]<-0
    maxforest.area[is.na(maxforest.area)]<-0
    connected.area[is.na(connected.area)]<-0
    pct.connected.area[i]<-(connected.area[i]/sum(connected.area))*100
#calculado da porcentagem de area conectada
    pct.connected.area[is.nan(pct.connected.area)]<-0
    pct.restored.area[i]<-(rest.area[i]/sum(rest.area))*100 #calculado
da porcentagem da area conectada
    con.rest.ratio[i]<-connected.area[i]/rest.area[i] #calculado da razao
entre conectividade e area restaurada
    con.rest.ratio[is.na(con.rest.ratio)]<-0
    pct.connected.area<-round(pct.connected.area,2) #arredondando os
valores para menos casas decimais
    pct.restored.area<-round(pct.restored.area,2)
    con.rest.ratio<- round(con.rest.ratio,3)
  }
  rest.table<-data.frame(final.area, rest.area, forest.area,
maxforest.area, connected.area, pct.connected.area, pct.restored.area,
con.rest.ratio)

```

```
#criou o data frame com os dados do cronograma de restauracao
ord.rest.table<-order(rest.table[,8], decreasing=TRUE) #ordena a
restauracao por maior ganho de conectividade/area restaurada
ord.large.area<-order(rest.table[,1], decreasing=TRUE) #ordena a
restauracao por tamanho da area conectada a ser formada
ord.pct.restored<-rep(NA, length(ord.rest.table))
ord.pct.connected<-rep(NA, length(ord.rest.table))
ord.pct.restored.large<-rep(NA, length(ord.rest.table))
ord.pct.connected.large<-rep(NA, length(ord.rest.table))
ord.pct.restored[1]<-rest.table[ord.rest.table[1],7] #gera o
primeiro valor dos valores acumulados de ganho e de area restaurada
ord.pct.connected[1]<-rest.table[ord.rest.table[1],6]
ord.pct.restored.large[1]<-rest.table[ord.large.area[1],7] #idem
comentario acima, mas para cronograma baseado por tamanho da area conectada
ord.pct.connected.large[1]<-rest.table[ord.large.area[1],6]
for(j in 2:length(ord.rest.table))
{
  k=j-1
  ord.pct.restored[j]<-rest.table[ord.rest.table[j],7] +
ord.pct.restored[k] #gera os outros valores de ganho acumulado >> nao esta
gerando valores acumulados e esta atrapalhando todo o resto
  ord.pct.connected[j]<-rest.table[ord.rest.table[j],6] +
ord.pct.connected[k]
  ord.pct.restored.large[j]<-rest.table[ord.large.area[j],7] +
ord.pct.restored.large[k]
  ord.pct.connected.large[j]<-rest.table[ord.large.area[j],6] +
ord.pct.connected.large[k]
}

graf.ord.rest<-data.frame(ord.pct.connected,ord.pct.restored) #junta
os dados num dataframe para fazer o grafico
graf.ord.large<-
data.frame(ord.pct.restored.large,ord.pct.connected.large) #junta a outra
ordenacao em um dataframe para fazer o grafico
par(mfrow=c(2,1))
plot(ord.pct.connected~ord.pct.restored, data= graf.ord.rest,
type="b", main="Cronograma baseado na relação \n ganho de conectividade X
área restaurada",ylab= "Conectividade alcançada", xlab="Área restaurada",
col="black")
plot(ord.pct.connected.large~ord.pct.restored.large, type="b",
data=graf.ord.large, main= "Cronograma definido pela priorização de \n
formação de maiores áreas contínuas", ylab="Conectividade alcançada",
xlab="Área restaurada", col="red")
par(mfrow=c(1,1))
#gerar 1000 cronogramas de restauracao aleatoria
#sim=1000
aleat<- list()
for (l in 1:1000)
{
  sample(rest.id, length(rest.id), replace=FALSE)->a
```

```

ord.restored.a<-rep(NA, length(a))
ord.connected.a<-rep(NA, length(a))
ord.restored.a[1]<-rest.table[a[1],7] #gera o primeiro valor de
porcentagem de area restaurada para o ciclo aleatorio
ord.connected.a[1]<-rest.table[a[1],6] #gera o primeiro valor de
porcentagem de conexao para o ciclo aleatorio
for(m in 2:length(ord.rest.table))
{
  n=m-1
  ord.restored.a[m]<-rest.table[a[m],7] + ord.restored.a[n] #cria do
segundo ao elemento# esta parte não esta funcionando pois nao deixa inserir
os valores no vetor
  ord.connected.a[m]<-rest.table[a[m],6] + ord.connected.a[n] # cria
do 2 ao n elemento# idem obs acima
  ord.restored.a[is.na(ord.restored.a)]<-0
  ord.connected.a[is.na(ord.connected.a)]<-0
}

ord.aleat<-data.frame(ord.connected.a, ord.restored.a)
ord.aleat->aleat[[1]]
a<-c()
}

#Gerar a curva para o crítico de 95%. Ainda não consegui fazer isso
#É preciso gerar intervalos de quantidade de restauração por exemplo
2,5% da restauração executada, 5%, 7.5%..100%
# cada um desses intervalos terá cerca de 1000 valores de
conectividade obtidos pelo cronograma ao acaso e a partir deles poderemos
#estimar o p de 95% de ganho de conectividade para um cronograma de
restauração aleatorio
#pts<-seq(from=2.5, to=100, by=2.5)
#for (o in 1:1000)
# {
#   for (p in 1:length(pts))
#   {
#     z<-aleat[[o]][2]<=pts[p]
#   }
}

```

**Arquivo do Help d função** [help\\_restore\\_strat.txt](#)

**Arquivo da função**[restore.strat.R](#)

**Arquivo de dados para rodar a função**[app.csv](#)

From:

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

Permanent link:

[http://ecor.ib.usp.br/doku.php?id=05\\_curso\\_antigo:r2010:alunos:trabalho\\_final:letambosi:start](http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2010:alunos:trabalho_final:letambosi:start) 

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