

Matheus



Nome completo: Matheus Pinheiro Ferreira

Curso: Doutorado em Sensoriamento Remoto - Instituto Nacional de Pesquisas Espaciais (INPE)

Projeto de Pesquisa: Diversidade química e espectral de espécies arbóreas da Mata Atlântica

Motivação: Necessidade do conteúdo da disciplina para realização de minha tese de doutoramento.

Currículo Lattes [indexmenu_n_1http://lattes.cnpq.br/9686528152912455](http://lattes.cnpq.br/9686528152912455)

Exercícios

Respostas dos Exercícios da Aula 1

[resposta_exercicio1_matheusferreira.r](#)

Respostas dos Exercícios da Aula 4

[exercicios_aula4.r](#)

Respostas dos Exercícios da Aula 5

[exercicios_5.r](#)

Respostas dos Exercícios da Aula 6

[exercicio_6.r](#)

Respostas dos Exercícios da Aula 7

[exercicio_aula7_ok.r](#)

Respostas dos Exercícios da Aula 8

[exercicio_8_ok.r](#)

Minha Proposta Inicial

Possuo dados de reflectância e transmitância espectral de folhas de sete espécies arbóreas da Mata Atlântica. Os espectros foram coletados com um radiômetro portátil chamado FieldSpec que abrange o intervalo espectral 350-2500 nm. Os dados possuem 2150 bandas e, por isso, permitem a detecção de variações sutis entre os alvos. Foram amostradas 15 folhas de três árvores de cada espécie. Pretendo elaborar uma função no R para avaliar as variabilidades inter e intra-espécies a partir da

métrica proposta por Price (1994), que simplesmente é a raiz quadrada da diferença entre dois espectros (referente a duas medidas), sobre o intervalo espectral de observação. As variabilidades calculadas serão:

- Inter-árvores: Estima a variabilidade entre as 15 folhas da mesma espécie.
- Intra-árvores/Inter-espécies: Estima a variabilidade entre árvores da mesma espécie.
- Intra-espécies: Estima a variabilidade entre as diferentes espécies.

Com estes resultados será elaborado um gráfico com a média e barras de desvio padrão para estas classes de valores. A constatação de que as variabilidades inter-espécies são menores do que as intra-espécies, é um passo importante na tentativa de identificar estas árvores com base em seu comportamento espectral.

Referência: J.C. Price, "How unique are spectral signatures", Remote Sensing of Environment, vol. 49, no. 3, pp. 181-186, 1994.

Plano B

Com os mesmos dados citados acima. Também gostaria de identificar as regiões ou bandas ao longo do intervalo espectral 350-2500 nm em que as espécies mais se diferem entre si. Como possuo dados de três árvores da mesma espécie, estes poderiam ser utilizados em uma análise de variância (ANOVA) como repetições e os tratamentos (ou grupos) seriam as próprias espécies. Iria-se então estimar um valor-p para cada banda, o que indicaria as regiões onde as espécies em questão diferem ou não estatisticamente (a um nível de 95%). Adicionalmente poderia ser realizado um teste de Tukey para computar o número de pares estatisticamente separáveis por banda. A análise destes resultados apontaria as bandas que concentram a variabilidade espectral das espécies.

Minha Proposta Inicial + Plano B

A Função **spectral.var**, une minha proposta inicial e o Plano B, como sugerido nos comentários da Hamanda. Além disso, **spectral.var** realiza um teste de randomização, de acordo com Manly (1997), para verificar se a probabilidade das diferenças significativas observadas entre as médias das espécies é fruto do acaso.

Minha Função

spectral.var foi desenvolvida para calcular a variabilidade espectral de plantas, utilizando a métrica proposta por Price (1994). A função também realiza o test One-way ANOVA seguido por comparações múltiplas de Tukey para selecionar bandas em que as espécies estudadas mais diferem entre si. Além disso, **spectral.var** realiza um teste de randomização, de acordo com Manly (1997), para verificar se as diferenças significativas observadas entre as médias das espécies é fruto do acaso. Os dados a serem inseridos na função devem prover de medições realizadas com um espectrorradiômetro ou então de sensores hiperespectrais.

Referências:

J.C. Price, "How unique are spectral signatures", *Remote Sensing of Environment*, vol. 49, no. 3, pp. 181-186, 1994.

B. Manly. *Randomization, Bootstrap and Monte Carlo Methods in Biology* (2nd ed.), 1997.

Comentários

Matheus, acho que a proposta é adequada, mas como vc mesmo disse a variabilidade é simplesmente a raiz quadrada da diferença dos espectros. Acredito que se vc agregar as propostas A e B em uma única função ela ficará mais elegante! Hamanda

Help (Ajuda) da Função

```
spectral.var      package:nenhum      R Documentation
```

```
Variabilidade espectral de plantas
```

Description:

`spectral.var` foi desenvolvida para calcular a variabilidade espectral de plantas, utilizando a métrica proposta por Price (1994).

A função também realiza o test One-way ANOVA seguido por comparações múltiplas de Tukey para selecionar bandas em que as espécies estudadas mais diferem entre si.

Além disso, `spectral.var` realiza um teste de randomização, de acordo com Manly (1997), para verificar se as diferenças significativas observadas entre as médias das espécies é fruto do acaso.

Os dados a serem inseridos na função devem prover de medições realizadas com um espectraloradiômetro ou então de sensores hiperespectrais.

Usage:

```
spectra.var(x,y,spp.f)
```

Arguments:

`x` matrix. Deve ser uma matrix em que as linhas são as bandas (comprimentos de onda) e as colunas são as folhas medidas.

`y` data frame. Deve ser um data frame organizado da seguinte forma:

	# Wavelength	Espécie-1	Espécie-1	Espécie-1	Espécie-2
Espécie-2	Espécie-2				
	400	0.0312	0.0339	0.0297	0.0253
0.0236	0.0279				

spp.f factor. Deve conter o nome das espécies.

Details:

Na função `spectral.var` o argumento "x" será utilizado para calcular a métrica "D" proposta por Price (1994). Esta métrica é dada pela raiz quadrada da diferença entre dois espectros (referente a duas medidas), sobre o intervalo espectral de observação. Ela é computada para todas as combinações de pares possíveis (pairwise combinations) entre as medidas (colunas de x). A métrica "D" é utilizada para avaliar as variabilidades inter e intra-espécies das medidas de reflectância ou transmitância. O resultado será arquivo "D.RData" salvo no diretório corrente.

Já o argumento y, deve ser um data frame em que as espécies são organizadas nas colunas de acordo com as repetições das medidas. Por exemplo:

```
# Wavelength  ARACA  ARACA  ARACA  AROEIRA  AROEIRA  AROEIRA  .  .  
.  
    400      0.0312  0.0339  0.0297  0.0253      0.0236      0.0279  
    .        .        .        .        .        .        .  
    .        .        .        .        .        .        .  
    .        .        .        .        .        .        .
```

spp.f deve ser um objeto do tipo fator que contém o nome das espécies, exatamente como são repetidas em y. Por exemplo:

```
spp.f=factor(rep(c("ARACA", "AROEIRA", "CANELA_BRANCA", "IPE_ROXO", "PAINEIRA", "PATA_DE_VACA", "PITANGA"), each=3))
```

Caso o usuário queira realizar o teste de randomização, implementado segundo Manly(1997), a função resultará em um gráfico com os resultados das randomizações. Caso contrário, os resultados serão salvos no diretório corrente como "y_final.RData".

Boa parte das operações da função estão especificadas.

Author:

Matheus Pinheiro Ferreira
PhD Candidate in Remote Sensing - National Institute for Space Research (INPE)
mpf@dsr.inpe.br
pferreira.matheus@gmail.com

References:

J.C. Price, "How unique are spectral signatures", Remote Sensing of Environment, vol. 49, no. 3, pp. 181–186, 1994.

B. Manly. Randomization, Bootstrap and Monte Carlo Methods in Biology (2nd ed.), 1997.

Examples:

```
setwd()

x=as.matrix(read.table("Folhas_AROEIRA_AV1.txt", header=TRUE, sep="\t"))

y=read.table("Reflec_spp.txt", header=FALSE, sep="\t")

colnames(y)=c("Wavelength", rep(c("ARACA", "AROEIRA", "CANELA_BRANCA", "IPE_ROXO",
", "PAINEIRA", "PATA_DE_VACA", "PITANGA"), each=3))

spp.f=factor(rep(c("ARACA", "AROEIRA", "CANELA_BRANCA", "IPE_ROXO", "PAINEIRA", "
PATA_DE_VACA", "PITANGA"), each=3))

spectral.var(x,y)
```

Código da Função

```
#####
##### D-Spectral Amplitude #####
#####

spectral.var=function(x,y)
{
  if(is.matrix(x)==FALSE) # Para a função caso x não seja uma matriz
  {
    stop("x deve ser uma matriz, \n consulte o help da função para mais
informações")
  }
  index=cbind(rep(1:ncol(x), each = ncol(x)), 1:ncol(x)) # Cria um
índice para a diferença entre colunas
  dif.cols=index[index[, 1] < index[, 2], , drop = FALSE] # Ordena o
índice para a combinação em pares
  dif.comb=x[, dif.cols[, 1]] - x[, dif.cols[, 2], drop = FALSE] #
Calcula a diferença combinada (pairwise difference) entre as colunas
  colnames(dif.comb) <- paste(colnames(x)[dif.cols[, 1]],
".vs.", colnames(x)[dif.cols[, 2]], sep = "") # Adicionando nome as colunas
para saber quais foram os pares
  dif.comb2=dif.comb^2 # Elevando as diferenças ao quadrado
  sum.col=0 # É necessário definir esse vetor onde os resultados do loop
serão guardados
  for (i in 1:ncol(dif.comb)) # Calculando D para cada coluna
  {
    sum.col[i]=sum(dif.comb2[,i])
    D=sqrt(sum.col/nrow(x)) # Root Mean Square Difference
  }
  save(D,file="D.RData") # Salva o arquivo D no diretório corrente
#####
#####
```

```
##### One-way ANOVA seguido por comparações múltiplas de Tukey #####  
#####  
#####  
# Verificando a homocedasticidade por banda, usando o Fligner-Killen test of homogeneity of variances ("The R Book", p.450)  
valor.p.fligner=0  
for(i in 1:nrow(y))  
{  
  var.resp=as.numeric(y[i,2:ncol(y)])  
  tab.fligner=fligner.test(var.resp~spp.f)  
  valor.p.fligner[i]=tab.fligner$'p.value'[1]  
}  
if(unique(table(valor.p.fligner<=0.01))!=nrow(y)) # Para a função caso haja algum valor-p significativo, indicando heterocedasticidade em alguma banda.  
{  
  stop("Condição de heterocedasticidade identificada em alguma das bandas")  
}  
# Calculando um valor-p para cada banda  
valor.p=0  
for(i in 1:nrow(y))  
{  
  var.resp=as.numeric(y[i,2:ncol(y)]) # 0 termo 2:ncol(y.reflec) pois a primeira coluna refere-se aos comprimentos de onda  
  tab.anova=summary(aov(var.resp~spp.f))  
  valor.p[i]=tab.anova[[1]]$'Pr(>F)'[1]  
}  
y.p=data.frame(y,valor.p)  
#####  
#####  
##### Teste Tukey HSD #####  
#####  
#####  
#####  
  
  pares.sep=0 # Criar o resultado do ciclo sempre antes dele  
  for (i in 1:nrow(y.p))  
  {  
    var.resp2=as.numeric(y.p[i,2:(ncol(y.p)-1)])  
    tukey1=aov(var.resp2~spp.f)  
    tukey2=TukeyHSD(tukey1)  
    pares.sep[i]=as.numeric(sum(tukey2$spp.f[,4]<=0.05)) # Número de pares estatisticamente separáveis por banda  
  }  
  
  y.final=data.frame(y.p,pares.sep) # Tabela com os valores-p da ANOVA e o número de pares separáveis por comprimento de onda.
```

```

save(y.final,file="y_final.RData")

#####
##### Teste de Randomização #####
#####
  perg=readline("Deseja realizar o teste de randomização com 5000
simulações? (Pode demorar mais do que 1 hora dependendo do computador) \n
Para SIM aperte S \n Para NÃO aperte N \n")

  if(perg=="N")
  {
    stop("A função acaba aqui, os resultados estão salvos no diretório
corrente")
  }

  if(perg=="S")
  {
#####
##### Valor de F para cada banda dos dados OBSERVADOS #####
#####
y.rand=y.final[y.final$valor.p<=0.01&y.final$pares.sep>=round(0.5*(ncol(y)-1
)),] # Seleciona apenas as bandas com 50% ou mais de pares separáveis
  simula.f=matrix(NA, nrow=nrow(y.rand),ncol=5000)
  for(i in 1:nrow(y.rand))
  {
    var.resp=as.numeric(y.rand[i,2:(ncol(y.rand)-2)])
    tab.anova=summary(aov(var.resp~spp.f))
    simula.f[i]=tab.anova[[1]]$'F value'[1]
  }
#####
##### Valor de F para cada banda para os dados RANDOMIZADOS #####
#####

  for (j in 2:5000)
  {
    for(i in 1:nrow(y.rand))
    {
      var.resp=sample(as.numeric(y.rand[i,2:(ncol(y.rand)-2)]))
      tab.anova=summary(aov(var.resp~spp.f))
      simula.f[i,j]=tab.anova[[1]]$'F value'[1]
    }
  }
save(simula.f,file="simula_f.RData")
}

# Construção do gráfico com os resultados das simulações
par(mfrow=c(1,1))
par(family="serif")
par(mai=c(1.3,1.3,0.3,0.3))
par(cex=1.2)
par(cex.lab=1.2)

```

```
par(xaxs="i") # Fixa x e y em 0
plot(density(simula.f[1,],adjust=1.4,kernel="cosine",from=0,to=6),main="",xlab="",bty="l",xlim=c(0,6),ylim=c(0,1.5))
  mtext("F Statistic", side=1, line=3, at=2.8, cex=1.3, font=3,
family="serif")
  for (i in 1:nrow(simula.f))
  {
    cores=gray(seq(0.1,0.9,length=nrow(simula.f)))
    lines(density(simula.f[i,],adjust=1.4),col=cores[i])
  }
}
```

Arquivos para realizar o exemplo

[folhas_aroeria_av1.txt](#)

[reflec_spp.txt](#)

From:

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

Permanent link:

http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2013:alunos:trabalho_final:pferreira.matheus:start



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