

# Lista sobre regressão e medidas de associação.

Jaques E. Zanon

22/05/2021

## Regressão e medidas de associação.

1- Considere os dados de `formigas.txt` na pasta da disciplina. Importe os dados novamente, em seguida faça um plot desses dados. No plot, considere o eixo x = **Elevation** e o y = **AntSpeciesDensity**. Considere fazer uma transformação log desses dados. Depois faça uma regressão linear simples considerando como variável dependente **AntSpeciesDensity** e independente **Elevation**. Plote o resultado.

```
## Importando:
formigas<-read.table("formigas.txt",header=T)

## Plotando o grafico
plot(x = formigas$Elevation,
     y = "termie você")

## Transformando em Log na base 10
log10(formigas$"termine você"+1) #Lembra de adicionar 1 caso tenha abundancia zero.

## Agora a regressão
result<-lm("dependente"~"independente")

## Plot com resultado da regressão
plot(x = formigas$Elevation,
     y = "termie você",
     main = "Título do grafico",
     xlab = "Título do eixo x",
     ylab = "Título do eixo y")

abline(result, col='red')# resultado da regressão em vermelho.
```

2. Importe o arquivo `tabela.txt` enviado pelo e-mail. Carregue o pacote **vegan**. Quantas espécies existem na tabela? Qual o total de observações? Existem locais com zero espécies? As parcelas diferem em número de indivíduos total? E as populações de espécies diferem de tamanho? Em seguida calcule três índices de associação entre **objetos** (Q-mode): A distância de hellinger, a distância de Bray-Curtis e a distância euclidiana. Substitua o que está entre aspas sempre que conveniente.

```
## Importando:
df<-read.table("nome_do_arquivo.txt",header=T)
library(vegan)

## Nessa tabela temos informações categóricas também. Vamos deixa-las de lado por agora:
df_aux <- df[,-1:-2]# retirar as duas primeiras colunas.
```

```

## Quantas espécies?
ncol()

## Quantas obs?
nrow()

## Locais com zero espécies?
rowSums()

# E as parcelas são diferentes em abundancia? Vamos fazer um boxplot:
plot(rowSums()~factor(df$parcela),
     main="Abundancia entre parcelas")# ou também
boxplot(rowSums()~factor(df$parcela),
     main="Abundancia entre parcelas")# o comando factor() serve para indicar ao R que estamos tratando

## Tamanho das populações de sp.
colSums()

## Agora o modo Q:
vegdist("df ou df_aux?",method="bray")# distancia de bray-curtis

# Hellinger:
decostand("df ou df_aux?",method="hel")# transformação e depois:
vegdist("dados tranforamdos",method="eu")# distancia euclidiana

# Euclidiana
vegdist("df ou df_aux?",method="eu")

```

3. Agora compare as distâncias calculadas, com a distância euclidiana. Observe as diferentes *outputs* para cada índice e como eles se relacionam. Lembre-se que Hellinger e Bray-Curtis são assimétricos e a Euclidiana é simétrico.

```

## Podemos fazer isso:

a <- as.vector("distancia_hellinger")
b <- as.vector("distancia_Bray_curtis")
c <- as.vector("distancia_euclidiana")

abc <- cbind(a, b, c)

pairs(abc ,
      upper.panel = NULL)

```

4. Vamos dar um “visu” diferente para os dados. Vamos fazer um heatmap com os dados brutos (raw) e transformados (Hellinger). Observe com calma os gráficos e tente interpreta-los.

```

# Heatmap dados brutos:
heatmap(df_aux)

# Transforme os dados:
decostand("df ou df_aux?",method="hel")# transformação hellinger

```

```
# Heatmap dados transformados  
heatmap("transformados")
```

*# Compare os heatmaps, o que muda em relação as linhas? Ou seja, suas unidades amostrais.*

5. Vamos de PCA? Faça uma PCA das sp. Isso mesmo, podemos sim fazer PCA com espécies, mas lembre-se dos double-zeros. Para isso temos que transformar os dados e só depois fazer a PCA. Faça com dados brutos e depois transformados. Compare os resultados.

```
library(vegan)  
mob <- rda("df_brutos")  
mob.hel <- rda("df_transformados_hellinger")  
  
# temos uma função que plota a PCA melhor!  
biplot(mob, scaling = "symmetric", type = c("text", "points"))  
biplot(mob.hel, scaling = "symmetric", type = c("text", "points"))
```

Além de mandar por e-mail eu vou anexar todos os arquivos da aula em minha pagina no **GitHub**.