**Tarefa pre-curso**

**Após seguir o seguinte tutorial, envia um e-mail para agus.camacho@gmail com os seguintes exercícios.**

1. **Qual é a differença entre um vetor, uma matriz, um data.frame e uma lista, no R?**
2. **Escreva o código para fazer uma regressão simples e plotar um scatterplot das variáveis envolvidas.**

**Tutorial básico de R.**

Este document está baseado no [R for Beginners](http://cran.r-project.org/doc/contrib/Paradis-rdebuts_en.pdf) by Emmanuel Paradis, <http://ecologia.ib.usp.br/bie5782/doku.php?id=bie5782:02_tutoriais:start>

e o tutorial de Liam Revel.

**Uma rápida apresentação ao R**

O R é uma linguagem de programação desenhada para realizar análises estatísticos. Programar pode ser algo assustador, ou até frustrante, no começo, mas constitui uma habilidade fundamental para os pesquisadores. Fique tranquilo, é mais simples do que parece e os tutoriais estão pensados para que você possa usar o programa e fazer todos os exercícios corretamente.

O R não e mais do que uma forma de comunicar-se com o computador, escrevendo as ordens no lugar de apertar botões. Isso pode ser chato até que escrevemos bem os comando, mas depois permite repetir tudo muito mais facilmente. Além disso, o R nos permite realizar virtualmente qualquer tipo de análise estatístico, havendo já muitos tutorais disponíveis. Os principais componentes de uma sesso de analise são os Objetos e as funções. Os objetos são as variáveis (vetores), dados (em forma de tabela, matriz, data.frame, imagem, áudio, raster, etc), ou resultados de um calculo. Os objetos são criados e nomeados no R, cada um com sua estrutura, para serem incorporados nas funções. As funções são podem ser considerados “objetos especiais” que fazem operações com os objetos que carregam dados.

O R utiliza paquetes de funções e objetos necessários para realizar conjuntos de análises particulares. Tem milhares e estão disponíveis ao público(ex. the *Comprehensive R Archive Network* ([CRAN](http://cran.us.r-project.org/)))

**Instalando o R**

Siga os passos neste link:

<https://cran.r-project.org/bin/windows/base/>

<https://cran.r-project.org/bin/macosx/>

Usuários de MAC terão que aprender a realizar alguns procedimentos de modo diferente. Este tutorial está pensado para usuários de PC.

**Instalando Rstudio**

Esta interface facilita interagir com o R e é obrigatória para esta disciplina.

Siga os passos neste link:

https://www.rstudio.com/products/rstudio/download/#download

**Comandos básicos**

Vamos começar criando objetos e realizando algumas operações. Copie o texto abaixo, linha por linha no GUI do Rstudio. Uma vez copiado lá selecione cada linha, aperte crtl-enter, observe e entenda o resultado. Linhas que tem o símbolo # na frente, são lidas apenas como comentários. Não realizam nenhum cálculo.

Criando um objeto (assignando um valor a um nome)

n <- 15 #(<- e = funcionam igual)

n

# os nomes levam em conta as maiúsculas

x <- 1

X <- 10

x

# dando um valor novo, apagamos o valor anterior

X <- 1000

# todas as operações matemáticas estão no R e podemos combinar números com objetos

(1000 + 200) \* 5

(X+200) \* 5

# ls() é uma função que mostra todos os objetos disponíveis no diretório de trabalho.

nome <- "Agus"

n1 <- 10

n2 <- 100

m <- 0.5

**ls**()

# a função ls, assim como outras funções pode tomar diferentes argumentos que afetam o modo em que trabalha

**ls**(pat = "m") # lista objetos que tem 'm' no nome

# ls.str mostra características dos objetos criados

**ls.str**()

para saber que argumentos podemos introduzir em uma função, podemos escrever

**args**(ls)

**args**(ls.str)

Porém, outra opção é buscar na ajuda do R. Todos os pacotes do CRAN tem pelo menos uma página de ajuda para cada função, e frequentemente tem tutoriais explicativos:

**help**(ls)

**help.search**("phylogeny")

Ok, existem 5 tipos principais de objetos de dados no R. Vetor, fator,matriz,data.frame, e lista. Todos eles tem valores e atributos.

Vetor: uma serie de elementos do mesmo tipo. Considera as seguintes classes de objetos: integer (número enteiro), numeric (números reais), fator (níveis de fator), character( letras) time (tempo), e date (date).

 x <- 1:5

X

Class(x)

x=c("a","b","c")

Class(x)

# é possivel fazer operações logicas com vetores e objetos

x <- 1:5

x >= 3

## [1] FALSE FALSE TRUE TRUE TRUE

z <- **c**("order", "superfamily", "family", "genus", "species")

**class**(z)

**length**(z)

Podemos acessar elementos de um objeto de multiples jeitos.

z[2]

z[**c**(1, 3)]

i <- **c**(4, 5)

z[i]

# e até eliminá-los

z[-2]

z[**c**(**TRUE**, **FALSE**, **TRUE**, **TRUE**, **TRUE**)]

Gerando um vetor ou variável usando uma função, e selecionando elementos dentro dele.

x <- **runif**(n = 6, min = 0, max = 10)

x

x >= 5

**which**(x >= 5)

x[x >= 5]

x[**which**(x >= 5)]

Um atributo útil de vectores e outros objetos são os nomes (*names*). Nós os usaremos frequentemente durante nossas análises comparativas para identificar a espécie/população a que corresponde um determinado valor de um vetor ou matriz.

x <- 1:5

**names**(x) <- z

Um fator é como um vetor, mas tem o atributo *levels*:

f <- **c**("Male", "Male", "Female", "Female", "Female")

f <- **factor**(f)

f

# ou também:

f <- **c**(0, 0, 1, 1, 1)

**levels**(f)

**table**(f)

**summary**(f)

# e agora:

f <- **factor**(f)

**levels**(f) <- **c**("Male", "Female")

**table**(f)

**summary**(f)

Uma matriz é um vetor distribuído como uma tabela. Pelo tanto agora tem o atributo dim, que representa as linhas (horizontal) e colunas (vertical):

X <- matrix(1:6, 2, 3)

X

dim(X)

# agora compara:

X <- matrix(1:9, 3, 3)

X

X <- matrix(1:9, 3, 3), byrow = TRUE)

X

Também podemos acessar elementos de uma matriz

X[3, 2]

X[, 3] # a terceira coluna

X[2, ] # a segunda fileira

Um data frame parece uma matriz, mas pode guardar diferentes tipos de vetores. É visualizado como uma folha Excel.

z <- c("order", "superfamily", "family", "genus", "species")

Y <- data.frame(z, y = 1:5, x = 5:1)

Y

Summary(Y)

Str(Y)

Y[1,1]

Finalmente, uma lista permite armazenar todo tipo de objetos (até arvores filogenéticas).

# criando uma lista com um vetor de caracteres, um vetor numérico e um data.frame

L <- **list**(z = z, 1:2, Y)

Também podemos acessar os elementos de uma lista de múltiplos modos.

L[[1]]

## [1] "order" "superfamily" "family" "genus" "species"

L$z

## [1] "order" "superfamily" "family" "genus" "species"

Com esses objetos, podemos gerar diversos tipos de gráficos no R.

<https://www.harding.edu/fmccown/r/>.

Neste tutorial, criado pelo professor Alexandre Adalardo e colaboradores, podemos encontrar como fazer uma regressão simples e plotar os resultados.

<http://ecologia.ib.usp.br/bie5782/doku.php?id=bie5782:02_tutoriais:tutorial7:start>

O próprio site fornece uma extensiva introdução à linguagem R. Se tiver tempo, aproveite para explorar esse site e ganhar confiança com o R antes do nosso curso!