Explorando o mundo dos dados

Análise e Visualização com R

Lucas de Carvalho de Amorim

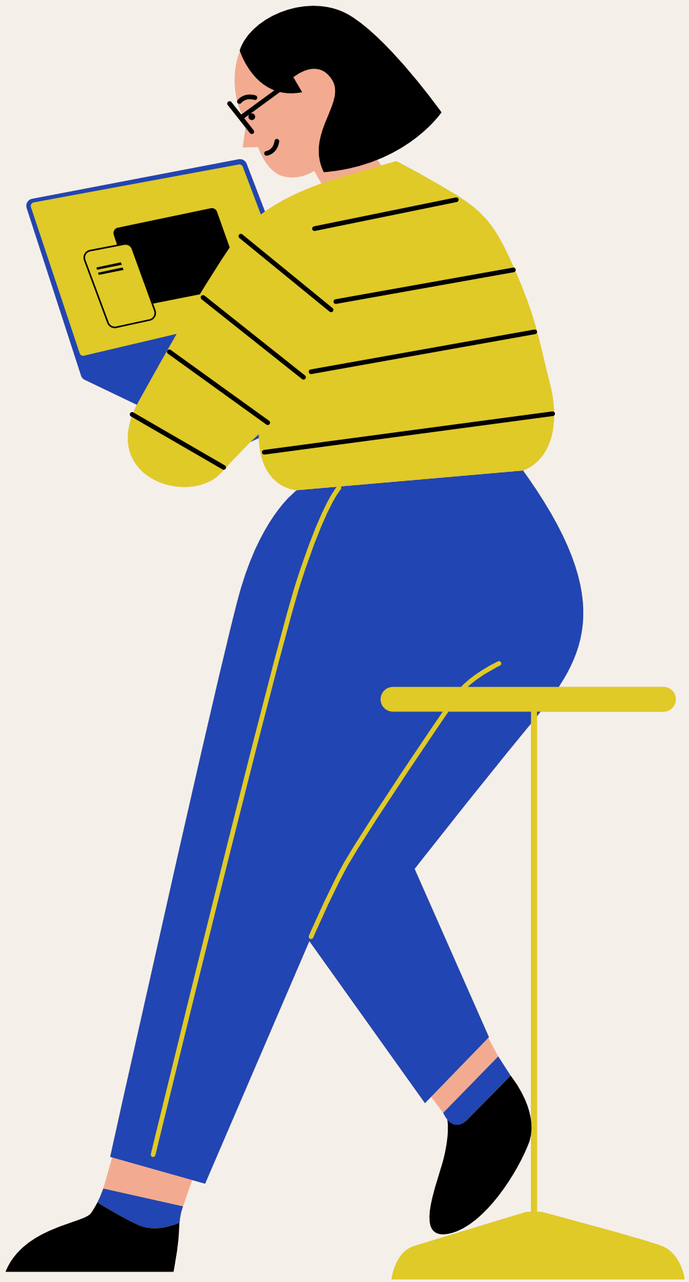
SEPEX
21ª Semana de Ensino, Pesquisa, Extensão e Inovação da UFSC

01 - Dowload

02 - Introdução

03 - Mensuração

04 - Previsão



UFSC

Sepex 2024

Redes Sociais

Lucas de Carvalho de Amorim

Youtube: @Lucasamorim0
Twitter: @amorimdf
LinkedIn: Lucas de Carvalho de Amorim
GitHub: @lucasamorimcp

SEPEX 2024: https://sepex.ufsc.br/
Inscrições minicurso Novembro/2024: https://sgsepex.ufsc.br/

AGENDAS ABERTAS PARA CONSULTORIA ACADÊMICA E DE PESQUISA

UFSC

Sepex 2024

Dowload R

R é uma poderosa linguagem de programação que é também gratuita e de código aberto.

Você pode fazer o download através do site R Project:

https://www.r-project.org/

Dowload R Studio

RStudio é uma interface conveniente para o R. É também gratuita para download:

https://posit.co/downloads/

01 - Introdução

Introdução ao R

said Hal Varian, chief economist at Google. “And you have a lot of prepackaged stuff that’s already available, so **you’re standing on the shoulders of giants**”..

UFSC

SEPEX 2024

01 - Introdução

Introdução ao R

OPERAÇÕES ARITMÉTICAS

Podemos digitar, por exemplo, 5+3 e, em seguida, pressionar Enter no teclado.

Começamos utilizando o R **como uma calculadora** com operadores aritméticos padrão.

01 - Introdução

Introdução ao R

OPERAÇÕES ARITMÉTICAS

O R ignora os espaços, de modo que 5 + 3 retornará o mesmo resultado.

Começamos utilizando o R **como uma calculadora** com operadores aritméticos padrão.

01 - Introdução

Introdução ao R

OPERAÇÕES ARITMÉTICAS

*A função **sqrt**() recebe um número não negativo e retorna sua raiz quadrada.*

Começamos utilizando o R **como uma calculadora** com operadores aritméticos padrão.

01 - Introdução

Introdução ao R

OBJETOS

O R pode **armazenar informações como um objeto** com um nome à nossa escolha. Depois de criar um objeto, basta nos referirmos a ele pelo nome.

Ele **não pode começar com um número** (mas pode conter números). Os nomes de objetos também **não devem conter espaços**. Devemos **evitar caracteres especiais**, como % e $, que possuem significados específicos no R. Os nomes de objetos **são sensíveis a maiúsculas e minúsculas**.

01 - Introdução

Introdução ao R

OBJETOS

```
result <-5+3

result

print(result)
```

*Observe que, se atribuirmos um valor diferente ao mesmo nome de objeto, o valor do objeto será alterado

01 - Introdução

Introdução ao R

OBJETOS

```
Lucas <- "professor"

Lucas
```

Podemos armazenar uma sequência de caracteres usando aspas.

01 - Introdução

Introdução ao R

OBJETOS

```
Result <- "5"

Result
```

O R trata números como caracteres quando pedimos para fazê-lo. No entanto, **operações aritméticas como adição e subtração não podem ser usadas para cadeias de caracteres**.

01 - Introdução

Introdução ao R

OBJETOS

class(result)

class(Result)

class(sqrt)

O R reconhece diferentes tipos de objetos ao atribuir cada objeto a uma classe. **Separar objetos em classes permite que o R execute operações apropriadas dependendo da classe dos objetos**.

01 - Introdução

Introdução ao R

VETORES

Um **vetor** ou um *array* unidimensional simplesmente representa uma **coleção de informações armazenadas em uma ordem específica**. Usamos a **função c**(), que significa "concatenate", para inserir um vetor de dados contendo múltiplos valores, com vírgulas separando os diferentes elementos do vetor que estamos criando.

01 - Introdução

Introdução ao R

VETORES

```
world.pop <- c(2525779, 3026003, 3691173, 4449049, 5320817, 6127700, 6916183)

world.pop
```

Year	*World population* (thousands)
1950	2,525,779
1960	3,026,003
1970	3,691,173
1980	4,449,049
1990	5,320,817
2000	6,127,700
2010	6,916,183

Source: United Nations, Department of Economic and Social Affairs, Population Division (2013). *World Population Prospects: The 2012 Revision, DVD Edition*.

01 - Introdução

Introdução ao R

VETORES

```
pop.first <- c(2525779, 3026003, 3691173)
pop.second <- c(4449049, 5320817, 6127700, 6916183)
pop.all <- c(pop.first, pop.second)
pop.all
```

A função c() pode ser usada para combinar múltiplos vetores.

Year	*World population* (thousands)
1950	2,525,779
1960	3,026,003
1970	3,691,173
1980	4,449,049
1990	5,320,817
2000	6,127,700
2010	6,916,183

Source: United Nations, Department of Economic and Social Affairs, Population Division (2013). *World Population Prospects: The 2012 Revision, DVD Edition.*

01 - Introdução

Introdução ao R

VETORES

```
pop.million <- world.pop / 1000
pop.million

pop.rate <- world.pop / world.pop[1]
pop.rate
```

Uma vez que cada elemento deste vetor é um valor numérico, podemos aplicar operações aritméticas a ele. As operações serão repetidas para cada elemento do vetor.

Year	*World population* (thousands)
1950	2,525,779
1960	3,026,003
1970	3,691,173
1980	4,449,049
1990	5,320,817
2000	6,127,700
2010	6,916,183

Source: United Nations, Department of Economic and Social Affairs, Population Division (2013). *World Population Prospects: The 2012 Revision, DVD Edition.*

01 - Introdução

Introdução ao R

VETORES

```
pop.rate[c(2,3)] <- c(19.8 , 46.1)
pop.rate
```

Também podemos substituir os valores associados a índices específicos usando o operador de atribuição habitual (<-).

Year	*World population* (thousands)
1950	2,525,779
1960	3,026,003
1970	3,691,173
1980	4,449,049
1990	5,320,817
2000	6,127,700
2010	6,916,183

Source: United Nations, Department of Economic and Social Affairs, Population Division (2013). *World Population Prospects: The 2012 Revision, DVD Edition.*

01 - Introdução

Introdução ao R

FUNÇÕES

Funções são objetos importantes no R e realizam uma ampla variedade de tarefas. **Uma função geralmente recebe múltiplos objetos de entrada e retorna um objeto de saída.** Já vimos várias funções: sqrt(), print(), class() e c(). No R, uma função geralmente é executada como **funcname**(**input**), onde funcname é o nome da função e input é o objeto de entrada. Em programação (e em matemática), chamamos esses inputs de **argumentos**. Por exemplo, na sintaxe sqrt(4), sqrt é o nome da função e 4 é o argumento ou o objeto de entrada.

01 - Introdução

Introdução ao R

FUNÇÕES

length(world.pop)

min(world.pop)

max(world.pop)

range(world.pop)

mean(world.pop)

sum(world.pop) / length(world.pop)

Year	*World population* (thousands)
1950	2,525,779
1960	3,026,003
1970	3,691,173
1980	4,449,049
1990	5,320,817
2000	6,127,700
2010	6,916,183

Source: United Nations, Department of Economic and Social Affairs, Population Division (2013). *World Population Prospects: The 2012 Revision, DVD Edition.*

01 - Introdução

Introdução ao R

FUNÇÕES

```
year <- seq(from = 1950, to = 2010, by = 10)
year

seq(from = 2010, to = 1950, by = -10)
```

Podemos criar um objeto para a variável ano da tabela.

Year	*World population* (thousands)
1950	2,525,779
1960	3,026,003
1970	3,691,173
1980	4,449,049
1990	5,320,817
2000	6,127,700
2010	6,916,183

Source: United Nations, Department of Economic and Social Affairs, Population Division (2013). *World Population Prospects: The 2012 Revision, DVD Edition.*

01 - Introdução

Introdução ao R

FUNÇÕES

```
names(world.pop)

names(world.pop) <- year
names(world.pop)
```

A função names() pode acessar e atribuir nomes aos elementos de um vetor. Os nomes dos elementos não fazem parte dos dados em si, mas são atributos úteis do objeto R.

Year	*World population* (thousands)
1950	2,525,779
1960	3,026,003
1970	3,691,173
1980	4,449,049
1990	5,320,817
2000	6,127,700
2010	6,916,183

Source: United Nations, Department of Economic and Social Affairs, Population Division (2013). *World Population Prospects: The 2012 Revision, DVD Edition.*

01 - Introdução

Introdução ao R

FUNÇÕES

Em muitas situações, queremos criar nossas próprias funções e usá-las repetidamente. Isso nos permite evitar a duplicação de conjuntos de códigos idênticos (ou quase idênticos), tornando nosso código mais eficiente e facilmente interpretável. A função function() pode criar uma nova função. A sintaxe tem a seguinte forma.

```
myfunction <- function(input1, input2, ..., inputN) {

    DEFINE "output" USING INPUTS

    return(output)
}
```

01 - Introdução

Introdução ao R

FUNÇÕES

```
my.summary <- function(x){

s.out <- sum(x)
l.out <- length(x)
m.out <- s.out / l.out
out <- c(s.out, l.out, m.out)
names(out) <- c("sum", "length", "mean")

return(out)
}
z <- 1:10

my.summary(z)
my.summary(world.pop)
```

Criando uma função para calcular um resumo de um vetor numérico.

01 - Introdução

Introdução ao R

ARQUIVOS DE DADOS

Até agora, os únicos dados que usamos foram inseridos manualmente no R. Mas, **na maioria das vezes, carregaremos dados de um arquivo externo.**

01 - Introdução

Introdução ao R

ARQUIVOS DE DADOS

```
setwd("~/Curso_R_2024")
getwd()
```

É possível alterar o diretório de trabalho usando a função setwd() especificando o caminho completo para a pasta de nossa escolha como uma string de caracteres.

01 - Introdução

Introdução ao R

ARQUIVOS DE DADOS

```
UNpop <- read.csv("UNpop.csv")
class(UNpop)
```

No RStudio, podemos ler ou carregar arquivos de dados.

01 - Introdução

Introdução ao R

ARQUIVOS DE DADOS

names(UNpop)
nrow(UNpop)
ncol(UNpop)
dim(UNpop)
summary(UNpop)

Um objeto data frame é uma coleção de vetores, mas podemos pensá-lo como uma planilha. Muitas vezes é útil inspecionar visualmente os dados.

01 - Introdução

Introdução ao R

ARQUIVOS DE DADOS

UNpop$world.pop

O operador $ é uma forma de acessar uma variável individual dentro de um objeto data frame. Ele retorna um vetor contendo a variável especificada.

01 - Introdução

Introdução ao R

ARQUIVOS DE DADOS

```
UNpop[, "world.pop"]

UNpop[c(1, 2, 3),]

UNpop[1:3, "year"]
```

Outra forma de recuperar variáveis individuais é usar indexação dentro de colchetes [ ], como feito para um vetor.

01 - Introdução

Introdução ao R

ARQUIVOS DE DADOS

```
UNpop$world.pop[seq(from = 1, to = nrow(UNpop), by = 2)]
```

Ao extrair observações específicas de uma variável em um objeto data frame, fornecemos apenas um índice, uma vez que a variável é um vetor.

01 - Introdução

Introdução ao R

ARQUIVOS DE DADOS

```
UNpop$world.pop[seq(from = 1, to = nrow(UNpop), by = 2)]
```

Ao extrair observações específicas de uma variável em um objeto data frame, fornecemos apenas um índice, uma vez que a variável é um vetor.

01 - Introdução

Introdução ao R

ARQUIVOS DE DADOS

```
world.pop <- c(UNpop$world.pop, NA)
world.pop
mean(world.pop)

mean(world.pop, na.rm = TRUE)
```

No R, valores ausentes são representados por NA. Quando aplicadas a um objeto com valores ausentes, as funções podem ou não remover automaticamente esses valores antes de realizar operações.

01 - Introdução

Introdução ao R

SALVANDO OBJETOS

```
save.image("~/Curso_R_2024/Aula1.RData")
```

No RStudio, podemos salvar o ambiente de trabalho clicando no ícone de Salvar na janela Environment no canto superior direito. Como alternativa, na barra de navegação, clique em Session > Save Workspace As... e escolha um local para salvar o arquivo.
Certifique-se de usar a extensão de arquivo .RData. Para carregar o mesmo ambiente de trabalho na próxima vez que iniciarmos o RStudio, clique no ícone Abrir Arquivo na janela Environment no canto superior direito, selecione Session > Load Workspace... ou use a função load() como antes.

01 - Introdução

Introdução ao R

SALVANDO OBJETOS

```
save(UNpop, file = "UNpop.RData")
```

Às vezes, desejamos salvar apenas um objeto específico (por exemplo, um objeto data frame) em vez de todo o ambiente de trabalho.

01 - Introdução

Introdução ao R

SALVANDO OBJETOS

```
write.csv(UNpop, file = "UNpop2.csv")
```

Em outros casos, podemos querer salvar um objeto data frame como um arquivo CSV em vez de um arquivo RData.

01 - Introdução

Introdução ao R

SALVANDO OBJETOS

```
load("UNpop.RData")
```

Para acessar os objetos salvos no arquivo RData, basta usar a função load().

01 - Introdução

Introdução ao R

said Hal Varian, chief economist at Google. “And you have a lot of prepackaged stuff that’s already available, so **you’re standing on the shoulders of giants**”..

UFSC

SEPEX 2024

01 - Introdução

Introdução ao R

PACOTES

Uma das forças do R é a existência de uma grande comunidade de usuários que contribuem com diversas funcionalidades na forma de pacotes R. **Esses pacotes estão disponíveis através da Comprehensive R Archive Network (CRAN; http://cran.r-project.org).**

01 - Introdução

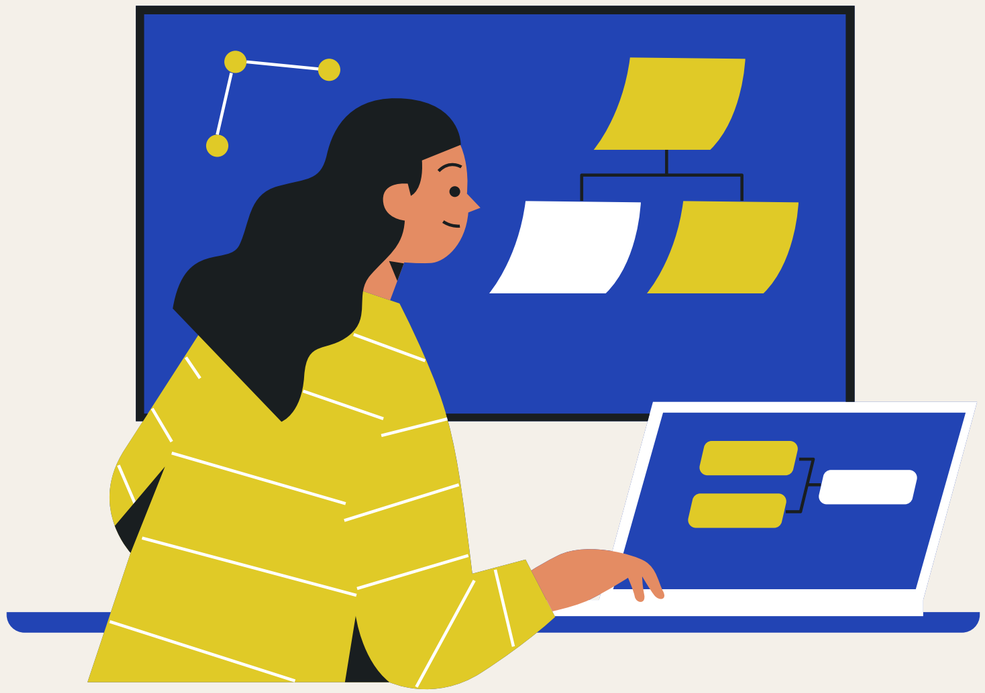
Introdução ao R

PACOTES

```
install.packages("foreign")

library("foreign")

read.dta("WVS_Wave_7_Brazil_Stata_v5.0.dta")

write.dta(UNpop, file = "UNpop.dta")
```

O pacote foreign é útil para lidar com arquivos de outros softwares estatísticos.

02 - Mensuração



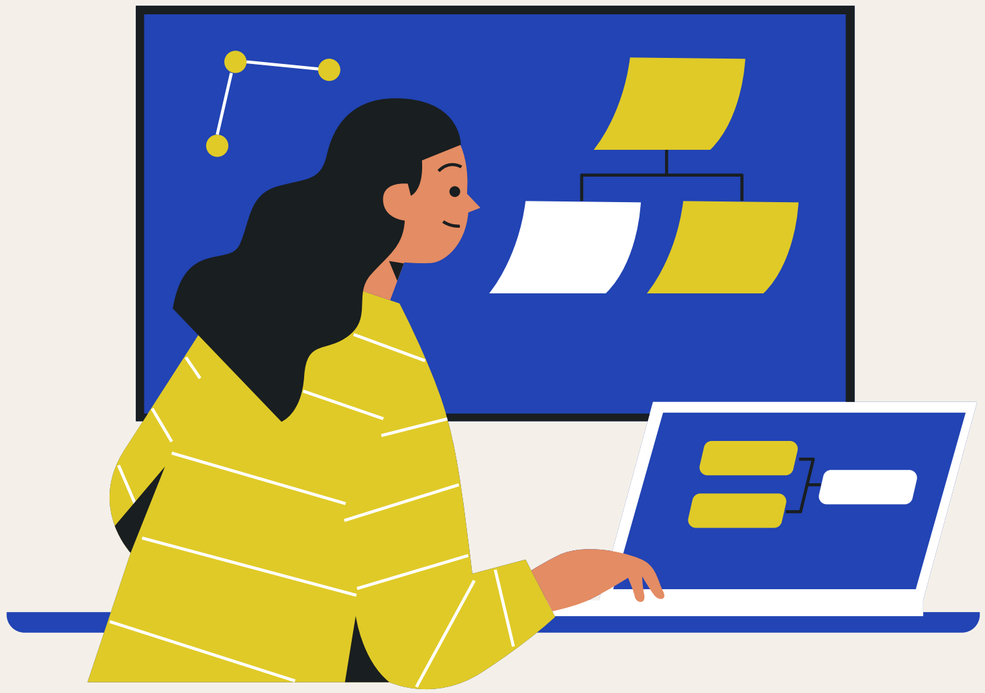
O que é econometria?

Econometria é uma área do conhecimento que utiliza **métodos estatísticos e matemáticos para analisar dados e testar teorias**. Ela serve para medir e entender relações entre variáveis, como o impacto de políticas públicas, o comportamento dos consumidores, e o desempenho do mercado. Basicamente, a econometria ajuda a **transformar observações do mundo real em informações úteis para prever tendências e tomar decisões informadas.**

UFSC

SEPEX 2024

02 - Mensuração



Questões de mensuração frequentemente ocupam a interseção entre análises teóricas e empíricas no estudo do comportamento humano. Introduzimos um método básico de **agrupamento**, que permite aos pesquisadores conduzir uma análise exploratória dos dados, descobrindo padrões interessantes. Também aprendemos a plotar dados de diversas maneiras e a calcular estatísticas descritivas relevantes no R.

A mensuração desempenha um papel central na pesquisa em ciências sociais.

UFSC

SEPEX 2024

02 - Mensuração

Explaining Support for Combatants during Wartime: A Survey Experiment in Afghanistan

Publication Year: 2013

Publisher: American Political Science Review

Author: Graeme Blair, Jason Lyall, Kosuke Imai

Download

UFSC

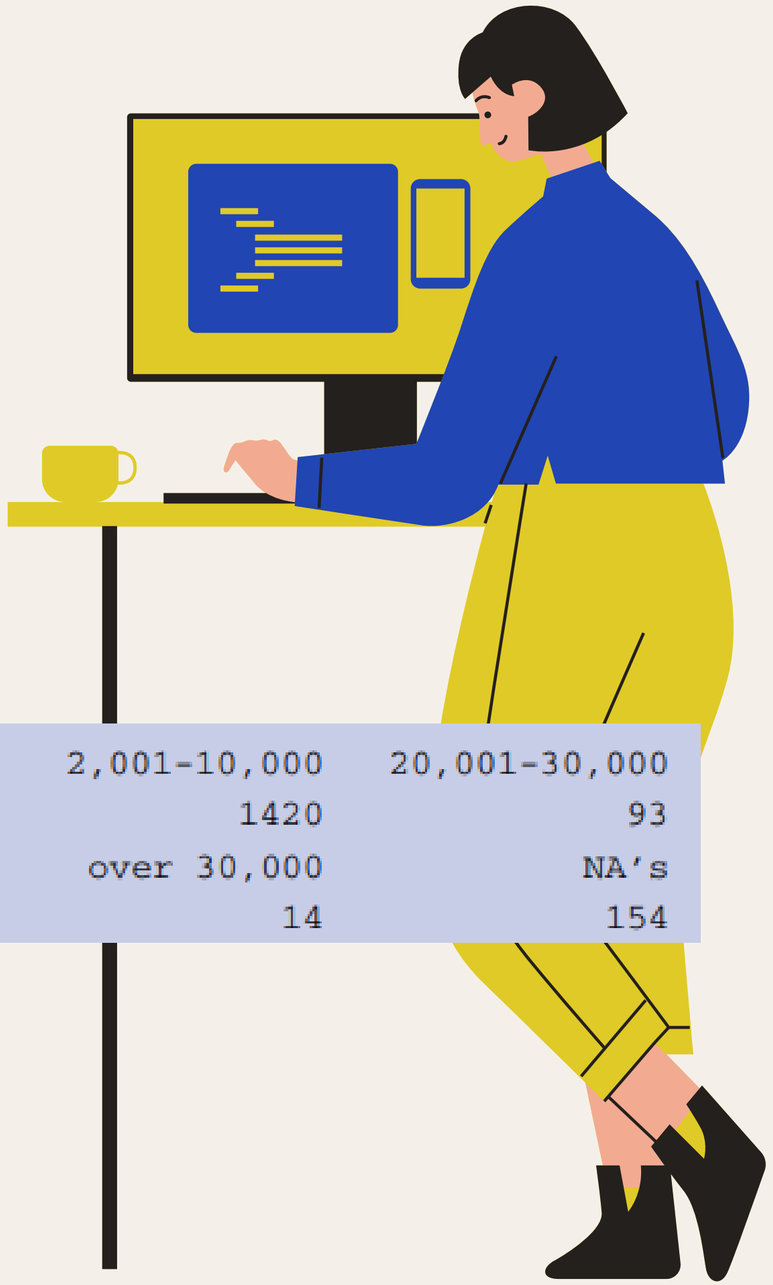
SEPEX 2024

Como as atitudes de civis em relação aos combatentes são afetadas pela vitimização em tempos de guerra? Esses efeitos dependem de qual combatente infligiu o dano?

02 - Mensuração

```
summary(data$age)

summary(afghan$educ.years)

summary(afghan$employed)

table(afghan$income)
```

```
##   10,001-20,000    2,001-10,000   20,001-30,000
##             616            1420              93
## less than 2,000     over 30,000            NA's
##             457              14             154
```



Resumindo as características dos entrevistados em termos de idade, anos de escolaridade, emprego e renda mensal em afeganes (a moeda local).

02 - Mensuração

```
prop.table(table(ISAF =
afghan$violent.exp.ISAF,
Taliban = afghan$violent.exp.taliban))
```

“No último ano, você ou alguém de sua família sofreu algum dano devido às ações das Forças Estrangeiras / do Talibã?”. Os entrevistadores explicaram aos entrevistados que a palavra “dano” se refere tanto a lesões físicas quanto a danos materiais.

02 - Mensuração

DADOS AUSENTES

```
head(afghan$income, n = 10)

head(is.na(afghan$income), n = 10)
```

Em muitas pesquisas, os pesquisadores podem se deparar com a não resposta, seja porque os entrevistados se recusam a responder algumas perguntas ou simplesmente não sabem a resposta. **No R, dados ausentes são codificados como NA.**

02 - Mensuração

DADOS AUSENTES

```
afghan.sub <- na.omit(afghan)
nrow(afghan.sub)
length(na.omit(afghan$income))
```

A função na.omit() oferece uma maneira simples de remover todas as observações com pelo menos um valor ausente de um conjunto de dados.

02 - Mensuração

VISUALIZAÇÃO DE DISTRIBUIÇÃO UNIVARIADA

Até agora, estivemos resumindo a distribuição de cada variável em um conjunto de dados usando **estatísticas descritivas**, como a média, a mediana e os quantis. No entanto, muitas vezes **é útil visualizar a própria distribuição**.

02 - Mensuração

VISUALIZAÇÃO DE DISTRIBUIÇÃO UNIVARIADA: BAR PLOT

```
ISAF.ptable <- prop.table(table(ISAF =
afghan$violent.exp.ISAF,
exclude = NULL))

barplot(ISAF.ptable,
names.arg = c("No harm", "Harm",
"Nonresponse"),
main = "Civilian victimization by the ISAF",
xlab = "Response category",
ylab = "Proportion of the respondents",
ylim = c(0, 0.7))
```

02 - Mensuração

VISUALIZAÇÃO DE DISTRIBUIÇÃO UNIVARIADA: HISTOGRAM

```
ISAF.ptable <- prop.table(table(ISAF =
afghan$violent.exp.ISAF,
exclude = NULL))

barplot(ISAF.ptable,
names.arg = c("No harm", "Harm",
"Nonresponse"),
main = "Civilian victimization by the ISAF",
xlab = "Response category",
ylab = "Proportion of the respondents",
ylim = c(0, 0.7))
```

02 - Mensuração

VISUALIZAÇÃO DE DISTRIBUIÇÃO UNIVARIADA: HISTOGRAM

```
hist(afghan$educ.years, freq = FALSE,
breaks = seq(from = -0.5, to = 18.5, by =
1),
xlab = "Years of education",
main = "Distribution of respondent's
education")
text(x = 3, y = 0.5, "median")
abline(v = median(afghan$educ.years))
```

02 - Mensuração

VISUALIZAÇÃO DE DISTRIBUIÇÃO UNIVARIADA: BOXPLOT

```
boxplot(afghan$age, main = "Distribution
of age", ylab = "Age",
ylim = c(10, 80))

boxplot(educ.years ~ province.id, data =
afghan,
main = "Education by province", ylab =
"Years of education")

tapply(afghan$violent.exp.taliban,
afghan$province.id, mean, na.rm = TRUE)
```

02 - Mensuração

VISUALIZAÇÃO DE DISTRIBUIÇÃO UNIVARIADA: BOXPLOT (TRANSFORMAÇÃO LOG NATURAL

```
boxplot(population ~ village.surveyed,
data = villages.balance,
        ylab = "log population", names =
c("Nonsampled", "Sampled"))

boxplot(log(population) ~
village.surveyed, data = villages.balance,
        ylab = "log population", names =
c("Nonsampled", "Sampled"))
```

02 - Mensuração

RELAÇÕES BIVARIADAS

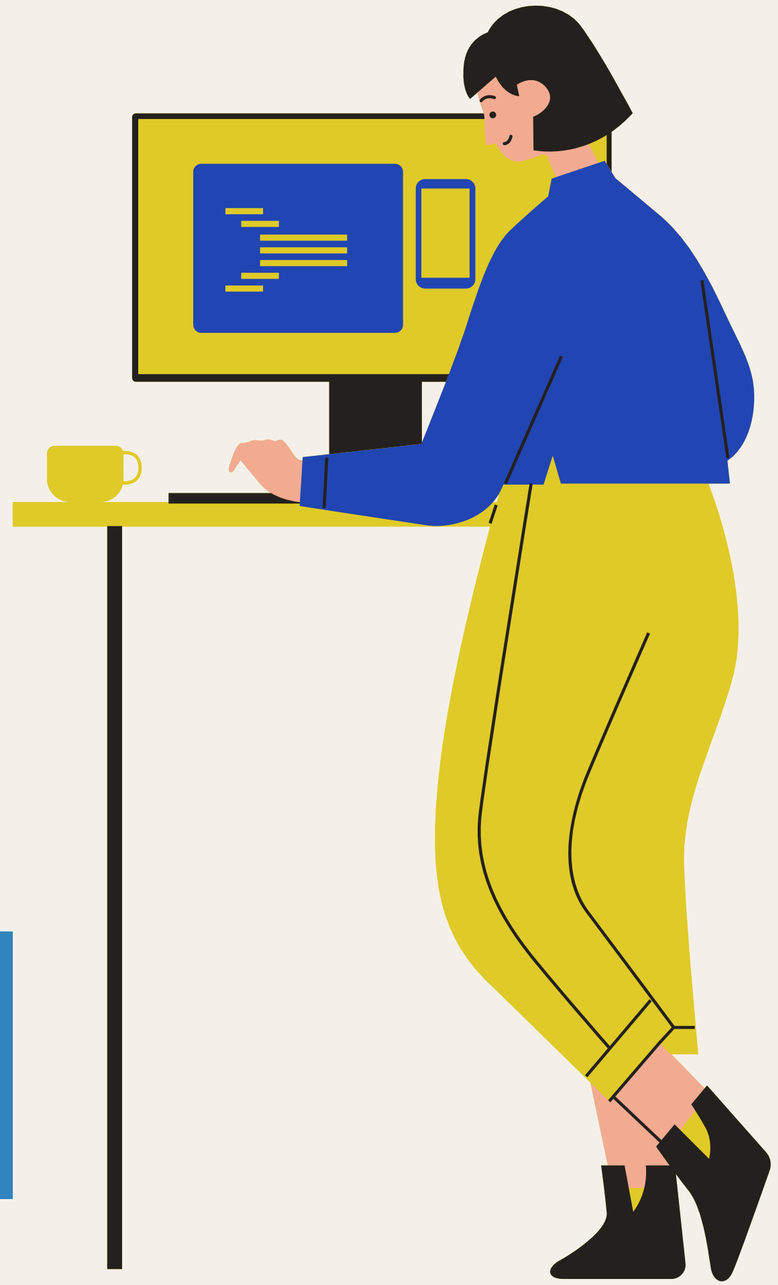
Até agora, estivemos resumindo a distribuição de cada variável em um conjunto de dados usando **estatísticas descritivas**, como a média, a mediana e os quantis. No entanto, muitas vezes **é útil visualizar a própria distribuição**.

02 - Mensuração

RELAÇÕES BIVARIADAS

VOTEVIEW PROJECT - UCLA

O Voteview permite que os usuários vejam todos os registros de votos de cada sessão do Congresso na história americana em um mapa dos Estados Unidos e em um mapa ideológico liberal-conservador, incluindo informações sobre as posições ideológicas dos senadores e representantes que votaram.



02 - Mensuração

RELAÇÕES BIVARIADAS: SCATER PLOT

```
rep <- subset(congress, subset = (party_code == 200))
dem <- congress[congress$party_code == 100, ]
rep80 <- subset(rep, subset = (congress == 80))
dem80 <- subset(dem, subset = (congress == 80))
rep112 <- subset(rep, subset = (congress == 112))
dem112 <- subset(dem, subset = (congress == 112))

xlab <- "Economic liberalism/conservatism"
ylab <- "Racial liberalism/conservatism"
lim <- c(-1.5, 1.5)
```

02 - Mensuração

RELAÇÕES BIVARIADAS: SCATER PLOT

```
plot(dem80$nominate_dim1, dem80$nominate_dim2, pch = 16, col = "blue",
     xlim = lim, ylim = lim, xlab = xlab, ylab = ylab,
     main = "80th Congress")
points(rep80$nominate_dim1, rep80$nominate_dim2, pch = 17, col = "red")
text(-0.75, 1, "Democrats")
text(1, -1, "Republicans")
```