

# Introdução ao R

Professora: Anaíle Mendes Rabelo

# Introdução ao R

- Download: <http://www.r-project.org/>
  - É um ambiente de programação que opera pela **linha de comando** (digitando o código)
  - Software **100% open source**
  - Ativo: builds frequentes (**atualizações frequentes**)
  - **Extensível**: pacotes (existem milhares de pacotes disponíveis)
  - **Multi-plataforma**: Windows, Linux, Mac
  - Propósito específico de servir a computação estatística e construção de gráficos

# Introdução ao R

- **Milhares de funções de análise de dados**
- **Ambiente de produção e visualização de gráficos**
- **Processamento em memória**

Integração “out of box” com quase tudo:

- Oracle
  - SQL Server
  - .NET
  - Java
  - Python
  - Tableau
  - Power BI
  - Hadoop
- } Banco de dados
- } Linguagem de programação
- } Ambientes de BI
- } Processamento Distribuído

- **Rgui (ambiente do R)**
  - **Ambiente de linha de comando simples, instalado por padrão**
  - Interface de digitação mais visualização de gráficos
- **RStudio**
  - **IDE mais avançada**
  - Possui versão gratuita

# R-Studio

- R: <https://cran.r-project.org/mirrors.html>
- RStudio: <https://www.rstudio.com/products/rstudio/download/>

```
File Edit Code View Plots Session Build Debug Tools Help
Get to Help/Editor Addins
Untitled1.R
Source on Save Run Source
1 library(dygraphs)
2 data("nhtemp")
3 nhtemp
4 dygraph(nhtemp, main = "New Haven Temperatures") %>%
5   dyRangeSelector(dateWindow = c("1920-01-01", "1960-01-01"))
```

1.1 Top Level = R Script =

Console → C:\Users\fr-jp\OneDrive\Documents\master\ / #2

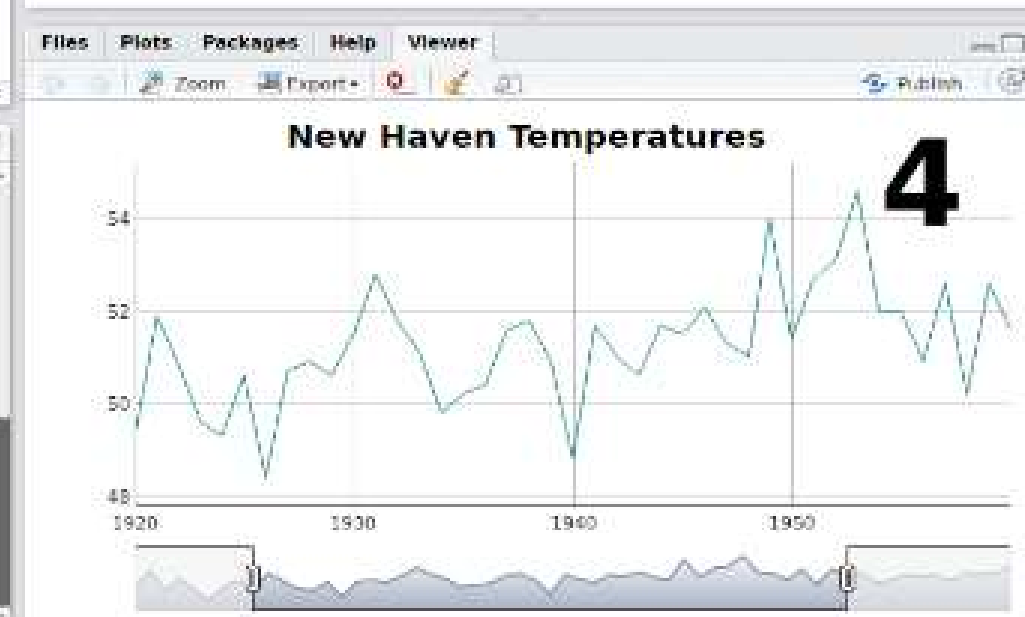
```
> library(dygraphs)
> data("nhtemp")
> nhtemp
Time Series:
Start = 1912
End = 1971
Frequency = 1
 [1] 49.9 52.3 49.4 51.1 49.4 47.9 49.8 50.9 49.3 51.9 50.8 49.6 49.3 50.6 48.4 50.7 50.9 50.6 51.1
[20] 52.8 51.8 51.1 49.8 50.2 50.4 51.6 51.8 50.9 48.8 51.7 51.0 50.6 51.7 51.5 52.1 51.3 51.0 54.0
[39] 51.4 52.7 53.1 54.6 52.0 52.0 50.9 52.6 50.2 52.6 51.6 51.9 50.5 50.9 51.7 51.4 51.7 50.8 51.9
[58] 51.0 51.9 53.0
> dygraph(nhtemp, main = "New Haven Temperatures") %>%
+   dyRangeSelector(dateWindow = c("1920-01-01", "1960-01-01"))
>
```

Environment History

Global Environment

Values

nhtemp	Time-Series [1:60] from 1912 to 1971: 49.9 52.3 49.4 51.1
--------	---



# Packages

- Implementam funções
- Desenvolvidos no mundo inteiro
- Totalmente open source
- Existem mais de 18 mil !!!



# Exemplos

- Gráficos
- Series Temporais
- Distribuições de Probabilidade
- Finanças
- Machine Learning
- Genética
- Entre outros.....

# Pacotes populares

- Dplyr: manipulação de dados
- Devtools: desenvolvimento (criação de pacotes)
- Foreign: importar dados de outras ferramentas (SAS, SPSS, WEKA, ...)
- Ggplot2: visualização de dados

## Pacotes - Instalação

- Linha de comando
  - `install.packages("NOME DO PACOTE", dependencies=TRUE)`
  - Seleciona o Espelho do CRAN e aguarda o download
  - Verifica a mensagem de instalação ou eventual problema
- Manualmente – muito utilizado com máquinas que tem bloqueios de firewall

# Instalação Manual

- Localiza a página do CRAN do pacote
- Download dos binários conforme SO


arules: Mining Association Rules and Frequent Itemsets

Provides the infrastructure for representing, manipulating and analyzing transaction data and patterns (frequent itemsets and association rules). Also provides C implementations of the association mining algorithms Apriori and Eclat.

Version: 1.5-5  
Depends: R ( $\geq 3.4.0$ ), [Matrix](#) ( $\geq 1.2-0$ )  
Imports: stats, methods, graphics, utils  
Suggests: [pmmi](#), [XML](#), [arulesViz](#), [testthat](#)  
Published: 2018-01-10  
Author: Michael Hahsler [aut, cre, cph], Christian Buchta [aut, cph], Bettina Gruen [aut, cph], Kurt Hornik [aut, cph], Ian Johnson [ctb, cph], Christian Borgelt [ctb, cph]  
Maintainer: Michael Hahsler <mhahsler@lyle.smu.edu>  
BugReports: <https://github.com/mhahsler/arules>  
License: [GPL-3](#)  
Copyright: The code for apriori and eclat in src/rapriori.c was obtained from <http://www.borgelt.net/> and is Copyright (C) 1996-2003 Christian Borgelt. All other code is Copyright (C) Michael Hahsler, Christian Buchta, Bettina Gruen and Kurt Hornik.  
URL: <https://github.com/mhahsler/arules>, <http://lyle.smu.edu/IDA/arules>  
NeedsCompilation: yes  
Classification/ACM: G.4, H.2.8, I.5.1  
Citation: [arules citation info](#)  
Materials: [README NEWS](#)  
In views: [MachineLearning](#)  
CRAN checks: [arules results](#)

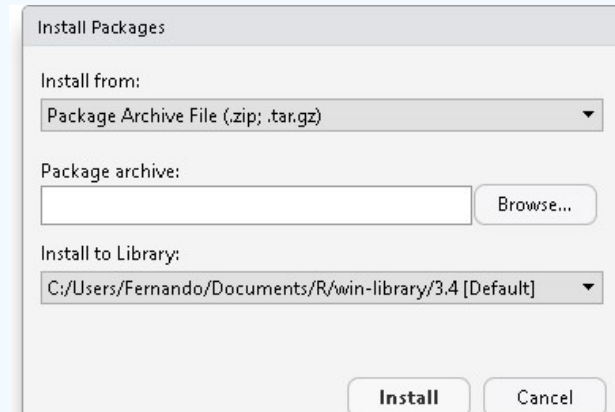
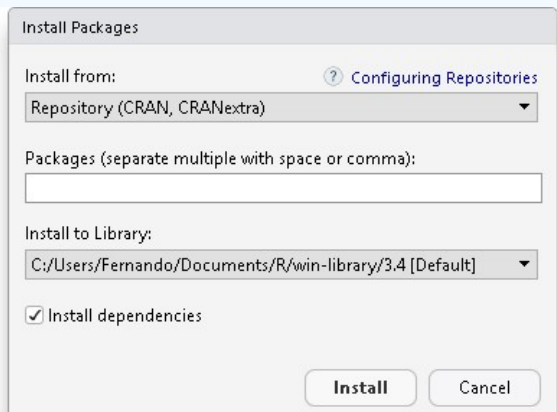
Downloads:

Reference manual: [arules.pdf](#)  
Vignettes: [Introduction to arules](#)  
Package source: [arules\\_1.5-5.tar.gz](#)  
Windows binaries: r-devel: [arules\\_1.5-5.zip](#), r-release: [arules\\_1.5-5.zip](#), r-oldrel: [arules\\_1.5-4.zip](#)  
OS X El Capitan binaries: r-release: [arules\\_1.5-5.tgz](#)  
OS X Mavericks binaries: r-oldrel: [arules\\_1.5-4.tgz](#)  
Old sources: [arules archive](#)



# Instalação Manual

- R-Sudio: Acessar menu tools, Install Packages



# Carregar e Descarregar Pacote

- `library(nome_pacote)`
- `detach("package:nome_pacote", unload=TRUE)`
- Curiosidade:
  - <https://cran.r-project.org/web/views/>

# Diretório de Trabalho

- Local onde o R busca por padrão os arquivos
- Saber o diretório padrão
  - `getwd()` -> retorna o diretório de trabalho padrão naquele momento
- Alterar o diretório padrão no R
  - `Setwd("caminho com barras duplas invertidas")`

# Visualização de Dados

- `plot()`: função genérica
- `hist()`
- `boxplot()`



# Tipos de Dados

- Caractere
- Numérico
- Inteiro
- Fator

# Atribuição de Valor

- =
- <-

–  $X <- 5$

–  $Y = 4$

# Declaração de variáveis

- Implícita
  - `num <- 8`
    - Considerada como numerica
  - `int <- 8L`
    - Considera como inteiro
  - `logico <- TRUE`
  - `logico <- F`
    - Variavel de tipo lógico
  - `caractere <- "Texto"`
    - Variavel de tipo texto

## Principais Operadores

+	Soma
-	Subtração
*	Multiplicação
/	Divisão
^	Potência
%%	Divisão de inteiros

## Operadores Lógicos

<	Menor que
>	Maior que
<=	Menor ou igual que
>=	Maior ou igual que
==	Igual
!=	Diferente
!	Not
	Ou
&	E

## Funções matemáticas nativas do R

abs	Valor absoluto
sqrt	Raiz quadrada
sum	Soma
log	Logaritmo base 10
cos	Cosseno
sin	Seno
tan	Tangente
exp	Exponencial

# Estrutura de dados

- Vetores
  - Qualquer objeto declarado
    - Vetor de uma posição
  - `X <- 5`
    - Lê a posição 1
  - `X <- c(1,2,3,4,5,6)`
    - Vetor 6 posições
  - `X`
    - Le todo o vetor
  - `X[1]`
    - Lê a posição 1
  - `X[1] <- 7`
    - Altera a posição 1

# Matrizes

- Duas dimensões (linhas e colunas)
- Permite um único tipo de dados
- Linhas e colunas podem ter nomes
- Ler ou alterar posição:
  - `nome_da_matrix[linha,coluna]`

# Data Frame

- Semelhante a Matrizes, porém:
  - Permite diferentes tipos de dados por coluna
- Duas dimensões (linhas e colunas)
- Linhas e colunas podem ter nomes
- Sintaxe para acessar coluna
  - `nome_data_frame$nome_coluna`

# Funções

- É um trecho do código que executa uma tarefa específica
- Podem ou não requer argumentos (parâmetros)

## Exemplo

- `getwd()`
- `[1] "C:/Users/XXXXX/Documents"`
  
- `sd(x)`
- `[1] 555.5555`



# Argumentos

- O R é flexível com argumentos:
  - Você pode simplesmente passar os argumentos pela ordem esperada, sem nome
  - Você pode nomear os argumentos
  - Você passar os primeiros sem nome e os últimos nomeados, omitindo intermediários

```
head(x=iris, n=2)
```

```
head(iris)
```

```
head(iris,2)
```

```
head(n=22)
```

```
Error in head.default(n = 22) : argumento "x" ausente, sem padrão
```

# Principais Funções

**head()**

Visualizar primeiras linhas de um conjunto de dados -

**tail()**

Visualizar últimas linhas de um conjunto de dados -

**summary()**

Resumo estatístico de um conjunto de dados -

**dim()**

Dimensões de um conjunto de dados (numero de colunas e número de linhas) – para matriz e data.frames

**colnames()**

Nomes das colunas de um conjunto de dados

**rownames**

Nomes das linhas de um conjunto de dados

**colbind()**

Adiciona coluna

# Pacote Tidyverse

# Tidyverse

- Coleção de pacotes para análise e ciência de dados no R
- É desenhado com uma filosofia , gramática e estrutura de dados em comum
- Cobre (quase) tudo que parecia para analisar dados

# Tidyverse



para importação de dados.



para tratamento de dados.



para organizar tabelas.



para gráficos.



para criar tabelas.



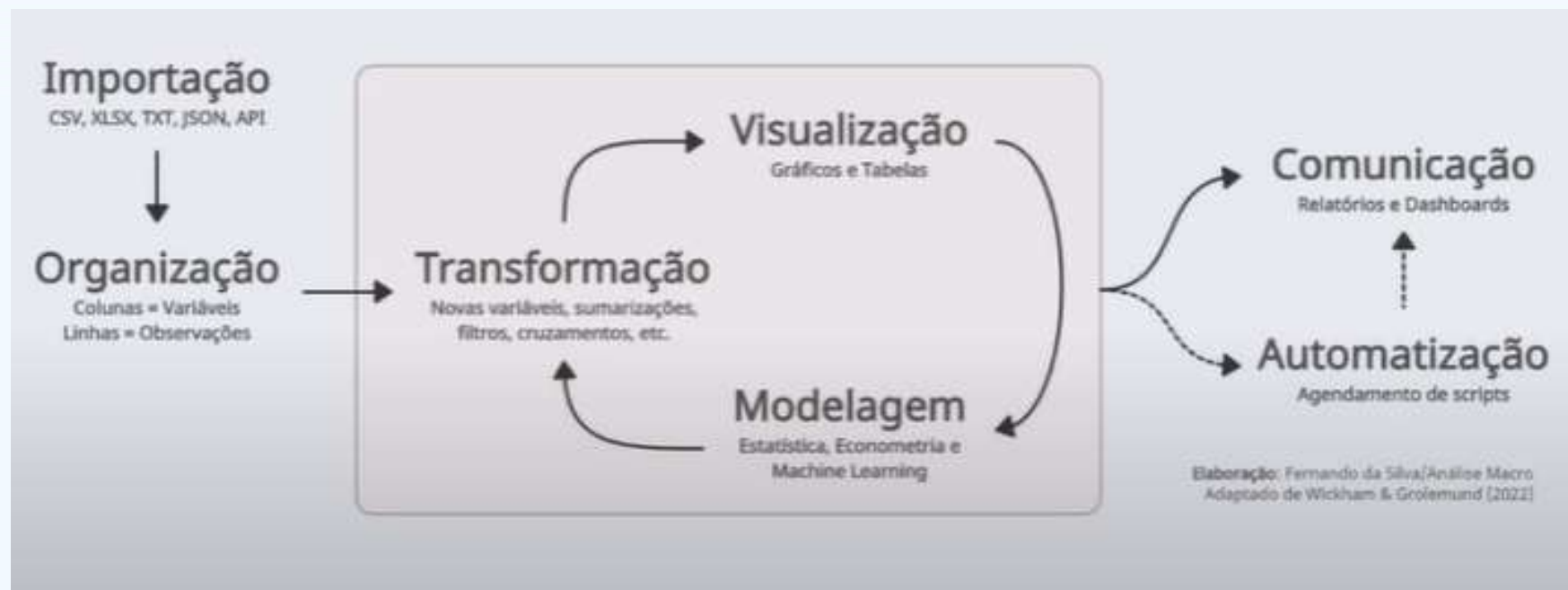
para programação funcional.



para manipulação de textos.



para tratar variáveis categóricas.



## Tidyverse

### Ciclo da análise de dados

# Operador pipe

- Ajuda na visualização do encadeamento de funções

```
mais_tres <- function(x) { x + 3 }  
sobre_dois <- function(x) { x / 2 }  
  
x <- 1:3  
  
sobre_dois(mais_tres(x))  
#> [1] 2.0 2.5 3.0
```

```
x %>% mais_tres() %>% sobre_dois()  
#> [1] 2.0 2.5 3.0
```

- A grande vantagem do pipe não é só enxergar quais funções são aplicadas primeiro, mas sim nos ajudar a programar pipelines (“encanamento” em inglês) de tratamentos de dados.

# Leitura de dados com readr

- O primeiro ponto de qualquer projeto de análise de dados é **obter os dados**.
- **Podendo ser** : .csv, .xlsx, .txt., etc.
- pacote readr possui recursos mais otimizados.



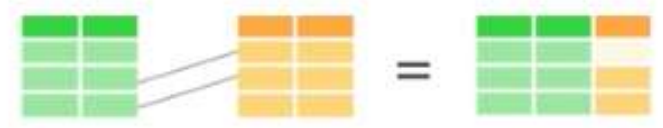
## O pacote dplyr

- Executa variadas tarefas de manuseio de dados:
  - **select()** - > selecionar ou remover colunas.
  - **filter()** - > criar filtros de observações baseado em um ou mais critérios.
  - **mutate()** -> serve para criar novas colunas que são funções de colunas já existentes no dataframe

# Principais funções

- ***arrange()*** muda a posição das linhas do dataframe baseado em uma ou mais colunas, em ordem crescente ou decrescente É como o classificar do Excel.
- O combo ***group\_by()*** e ***summarise()*** é excelente para agregar e resumir dados. Com ***group\_by()***, as funções aplicadas com ***summarise()*** ou até mesmo com ***mutate()*** ou ***filter()*** são aplicadas não em todo o dataset mas sim em cada grupo da variável especificada na função ***group\_by***
- ***join()*** para agrupar duas tabelas em uma

`left_join()`



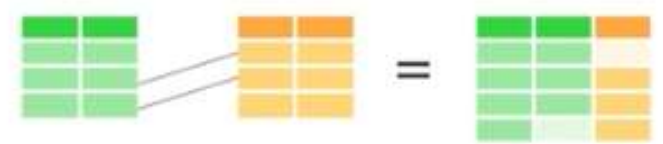
`right_join()`



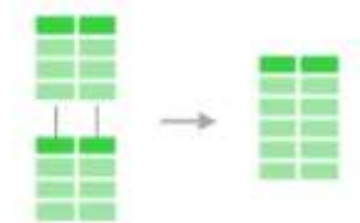
`inner_join()`



`full_join()`



`bind_rows()`



`bind_cols()`

