Análise de Dados com o Software R: Métodos Estatísticos, Computacionais e Econométricos

João Pedro Albino

24 de outubro de 2018

Introdução

Vivemos em uma era onde há um volume imenso de dados disponíveis – sejam eles estruturados ou não estruturados – criados e armazenados em nível global e que impactam as decisões tomadas por pessoas e empresas no nosso dia a dia, afirma Silveira (2016). A evolução das tecnologias da comunicação e informação (TICs) e, portanto, a massiva informatização dos serviços, desde as sofisticadas transações em bolsa à simples compra de um café, associada às redes sociais e aos dispositivos móveis (tablets e smartphones) produzem uma enorme quantidade de dados. Para além da quantidade de dados, a taxa de atualização desses mesmos dados é também muito grande. Em média, de acordo com Cavique (2014), em cada 10 minutos são gerados mais dados do que todos os dados gerados desde a pré-história até ao ano de 2003.

O grande volume de dados, compensado pelo aumento da capacidade de processamento das tecnologias de informação e comunicação (TICs), tem originado novos conceitos, como a criação de novas profissões como os cientistas de dados (*data scientists*), e novos termos, tais como *Big Data* - que se refere a esta incontável quantidade de dados produzida diariamente em todo o mundo. (SILVEIRA, 2016 e CAVIQUE, 2014).

A *Ciência de Dados*, outro novo conceito, é o estudo e geração de conhecimento a partir destes grandes volumes de dados. Para isto, incorpora técnicas e teorias das mais diversas áreas de conhecimento como computação, engenharia, matemática, estatística, economia, mineração de dados, programação de computadores, inteligência artificial, entre outros. De acordo com Thor (2018), ciência de dados é um método para coletar insights de dados estruturados e não estruturados, utilizando para tanto de abordagens que vão desde a análise estatística até o *machine learning* (aprendizagem de máquina). Para a maior parte das organizações, a ciência de dados é empregada para transformar dados em valor, transformando-os em receita aprimorada, custos reduzidos, agilidade nos negócios, melhor experiência do cliente, desenvolvimento de novos produtos, entre outros.(THOR, 2018).

Segundo Silveira (2016), a Ciência de Dados é uma área que já existe há mais de 30 anos, mas vem ganhando destaque nos últimos anos, devido ao Big Data. O desenvolvimento de áreas como as machine learning e a importância da Ciência de Dados fez com que cada vez mais empresas e a academia se beneficiassem da ciência e da análise dos dados para a tomada de decisões.

A ciência de dados é uma área de conhecimento que combina habilidades estatísticas e quantitativas avançadas com a capacidade de programação do mundo real. Desta forma, para o desenvolvimento de projetos em ciência de dados, existem muitas linguagens de programação em potencial, tais como *Python*, *Scala*, *Julia* e a linguagem *R*.

Muito popular na academia, a linguagem R é utilizada por muitos pesquisadores e estudiosos para experimentar a ciência de dados. Muitos livros populares e recursos de aprendizagem em ciência de dados também. utilizam a linguagem R para análise estatística. Todas estas características contribuíram para criar um grande grupo de pessoas que possuem um bom conhecimento prático da programação R.

O objetivo deste trabalho é o de apresentar e discutir alguns conceitos sobre ciência de dados e *prova de conceito* (**POC**), utilizando como base um projeto de análise de dados utilizando a *linguagem R*, buscando desta forma determinar a viabilidade prática do processo. Uma POC é uma demonstração, cujo propósito é verificar se certos conceitos ou teorias têm potencial para aplicação no mundo real. (JANSEN e JANSEN, 2018).

Fundamentação Teórica

Prova de Conceito (PoC)

Uma prova de conceito, ou PoC (*Proof of Concept*) é um termo utilizado para denominar um modelo prático que possa provar um conceito teórico estabelecido por uma pesquisa ou artigo técnico, afirma Pineiro (2010) . A POC é uma demonstração, com o propósito de verificar se determinados conceitos ou teorias têm potencial para aplicação no mundo real, de acordo com Jansen e Jansen (2018). Uma POC é, portanto, segundo Jensen e Jensen (2018), um *protótipo* projetado para determinar a viabilidade, mas não representa os elementos produzidos (ou *entregas*). A prova de conceito é também conhecida como *prova de princípio*.

De acordo com Jansen e Jansen (2018), uma POC é um termo com várias interpretações em diferentes áreas. O POC no desenvolvimento de software descreve processos distintos com diferentes objetivos e funções dos participantes. A POC também pode se referir a soluções parciais envolvendo um pequeno número de usuários atuando em funções comerciais para estabelecer se um sistema satisfaz determinados requisitos.

O objetivo geral do POC é encontrar soluções para problemas técnicos, como a forma como os sistemas podem ser integrados ou o **throughput** (taxa de transferência) pode ser alcançada através de uma determinada configuração.

Segundo Pinheiro (2010), uma POC pode ser considerada também uma implementação, resumida ou incompleta, *de um método ou de uma ideia*, realizada com o propósito de verificar que o conceito ou teoria em questão é suscetível de ser explorada de forma prática.

De acordo com Robinson (2017), a linguagem R é utilizada por muitos pesquisadores e estudiosos para *experimentar* a ciência de dados além de encontrarmos disponíveis vários livros e recursos de aprendizagem sobre ciência de dados e a linguagem R, utilizando-a para análise estatística.

Neste trabalho, iremos utilizar o método proposto em Judd et. ali. (2017) para análise, inspeção, limpeza, transformação e modelagem de dados com o objetivo de demonstrar que o método descrito nos permite descobrir informações úteis, gerar conclusões e apoiar o ensino de ciência de dados por meio das características oferecidas pela linguagem R.

O Ecossistema R

O **R**, algumas vezes grafado em minúscula, como **r**, é uma linguagem e também um ambiente de desenvolvimento integrado voltado para cálculos estatísticos e gráficos. De acordo com Ishaka (1998), originalmente a linguagem foi criada por Ross Ihaka e Robert Gentleman no departamento de Estatística da universidade de Auckland, Nova Zelândia, e foi desenvolvida por meio de um esforço colaborativo de pessoas distribuídas em vários locais do mundo.

O nome R, segundo Hornik (2017), provém em parte das iniciais dos nomes dos criadores (**R**oos e **R**obert) e também de um jogo figurado com a **linguagem S**, desenvolvida no Bell Laboratories. (BECKER e CHAMBERS, 1984).

O R é uma linguagem e um ambiente MUITO similar ao S - podendo ser considerado uma implementação ampliada e atualizada da *New S Language* (Becker et. ali., 1988) embora com diferenças importantes. Entretanto, afirma Hornik, 2017), códigos escritos em S podem ser executados inalterados no R.

O código fonte do R está disponível sob a licença GNU/GPL e as versões binárias pré-compiladas são fornecidas para Windows, Macintosh, e muitos sistemas operacionais Unix/Linux. O R é também altamente expansível com o uso dos **pacotes** - bibliotecas com sub-rotinas específicas ou voltadas para áreas específicas de estudo. Um conjunto de pacotes básico (*base*) está incluso em toda instalação do R, porém, muitos outros pacotes estão disponíveis na rede de distribuição do R (CRAN - *Comprehensive R Archive Network*).

Todas as características do R citadas anteriormente formam o conceito do **ecossistema R** (R Ecosystem). De acordo com Plakidasa et ali. (2017), um *ecosistema* representa uma mudança no conceito de desenvolvimento de projetos: passa-se do desenvolvimento monolítico verticalmente integrado de produtos para modelos mais

abertos, modulares e colaborativos. Os ecossistemas de software, afirmam os autores, representam o passo mais recente no desenvolvimento de aplicações onde uma comunidade de desenvolvedores colabora de forma assíncrona, e muitas vezes sem uma direção centralizada, em uma plataforma única ou mercado de software. O objetivo de se investir e trabalhar em prol de um *ecossistema* é que todos os membros obterão mais benefícios por fazer parte dele, em comparação com uma abordagem mais tradicional de desenvolvimento de produtos de software com papéis segregados, baixo nível de colaboração e processos fechados, afirmam Plakidasa et. ali. (2017).

A linguagem R é largamente utilizada entre estatísticos e analistas de dados para realizar análise computacional sistemática de dados ou estatísticas. (MUENCHEN,2017). De acordo com Robinson (2017), pesquisas e levantamentos realizados junto a profissionais da área de ciência de dados mostraram que a popularidade do R aumentou substancialmente nos últimos dez anos. Segundo pesquisas realizadas em 2017, o R é bastante utilizado nas universidades americanas, onde é uma escolha padrão para pesquisa acadêmica, especialmente nas áreas de *ciências sociais* e *biologia*.

A área com o segundo maior número de usuários em R, afirma Robinson (2017), por uma margem muito próxima, é a da área de *saúde*. Este dado provavelmente não será uma surpresa para os bioestatísticos, já que o R é a ferramenta escolhida em muitos métodos estatísticos utilizados em estudos clínicos e em bioinformática.

Entretanto, segundo Robinson (2017), uma área que não utiliza muito o R, em relação a outras tecnologias, é a área de desenvolvimento de software em empresas de web. Isso ocorre parcialmente, segundo o autor, porque a análise de dados representa uma parte relativamente pequena do setor, em comparação outras áreas de desenvolvimento de tecnologia.

Por que utilizar *R* para Análise de Dados?

R é uma linguagem usada para cálculos estatísticos, análise de dados e representação gráfica de dados. Criado na década de 1990 o R foi projetado como uma plataforma estatística para limpeza, análise e representação de dados. De acordo com pesquisa da Burtch Works, realizada em 2107, de todos os cientistas de dados pesquisados, 40% preferem R, 34% preferem SAS e 26% Python. De acordo com a 18ª pesquisa anual da KDNuggets sobre o uso de software de ciência de dados, o R é a segunda linguagem mais popular em ciência de dados. Isso mostra como a programação R é popular em ciência de dados. Até mesmo as tendências do Google mostram a crescente popularidade da Programação R. (NewGenApps, 2017).

Quando se desenvolve um projeto em de ciência de dados, muito provavelmente, como visto na pesquisa da KDnuggets, se opta entre os ambientes R e Python. Embora cada um delas seja igualmente competente e tenha prós e contras, segundo NewGenApps (2017), existem vantagens distintas associadas a cada uma delas. Neste trabalho apresentaremos as vantagens do R e seu uso em ciência de dados e por que este ecossistema se mostra uma escolha ideal.

Nos tópicos seguintes serão apresentadas e discutidas algumas razões para termos optado por escolher a linguagem R nesta prova de conceito de ciência de dados, utilizando como base o trabalho apresentado em NewGenApps (2017).

1. Academia

R é uma linguagem muito popular na academia. Muitos pesquisadores e estudiosos usam o R para realizar seus ensaios em ciência de dados. Por ser a linguagem preferida pelos acadêmicos, isso cria um grande grupo de pessoas que têm um bom conhecimento prático de programação em R. Também existe uma vasta literatura disponível em ciência de dados em R (NewGenApps, 2017 e ROBINSON, 2017).

2. Data wrangling - Preparação dos Dados

O termo **data wrangling** - em algumas literaturas também definido como *data mugging* ou *data preparation* – pode ser traduzido como **preparação de dados**. É o processo de limpeza dos _ data sets_ (conjuntos

de dados) complexos e confusos para permitir o consumo conveniente e o processo posterior de análises. (NewGenApp, 2017). Este é um processo muito importante e demorado em ciência de dados. O conceito é relativamente recente e diz respeito ao ato de coletar, limpar, normalizar, combinar, estruturar e organizar os dados que serão analisados. É o primeiro passo para que a mensuração das informações extraídas com o trabalho de análise seja bem sucedido e traga significativos apontamentos para o processo em geral. (PROFAP, 2018).

Dentro do processo de preparação de dados são realizados processos, tais como: visualização de dados; agregação de dados; treinamento de um modelo estatístico; bem como outros métodos potenciais. A manipulação de dados como um processo normalmente segue um conjunto de etapas gerais que começam com a extração da fonte dos dados **brutos** , "filtragem" desses dados brutos utilizando-se algoritmos (por exemplo, de classificação) ou análise dos dados em estruturas de dados predefinidas e, finalmente, depositar o conteúdo resultante em um *data set* "limpo" (preparado) para armazenamento e uso futuro.

O ecossistema R possui uma extensa biblioteca de ferramentas para preparação e manipulação de dados "brutos". Alguns dos pacotes populares para manipulação de dados em R incluem: (a) pacote *dplyr*, conhecido por seus recursos de exploração e transformação de dados e sintaxe de encadeamento altamente adaptável; (b) pacote *data.table* - permite uma manipulação mais rápida do conjunto de dados com codificação mínima, simplificando a agregação de dados e reduzindo drasticamente o tempo de computação; e (c) pacote *readr* - auxilia na leitura de várias formas de dados executando a leitura em rápida velocidade. (NewGenApp, 2017).

3. Visualização de dados

Visualização de dados é a representação visual dos dados em forma gráfica. A visualização permite analisar dados em ângulos que não podem ser observados em dados não organizados ou tabulados. O R possui um grande grupo de ferramentas (pacotes) que podem auxiliar na visualização, análise e representação dos dados. Os pacotes **ggplot2** e **ggedit** praticamente se tornaram os pacotes padrão para a plotagem de gráficos em R. Enquanto o pacote ggplot2 é focado na visualização dos dados, o ggedit ajuda os usuários a preencher a lacuna entre fazer um gráfico e obter todas as incômodas estéticas necessárias do gráfico precisamente corretas. (NewGenApp, 2017).

4. Especificidade

O R é uma linguagem projetada especialmente para análise estatística e reconfiguração de dados. Todas as bibliotecas do R se concentram em tornar a análise de dados mais fácil, acessível e detalhada. Qualquer novo método estatístico é primeiro ativado por meio de bibliotecas no R. Isso faz do R uma boa escolha para a análise e projeção de dados. Os membros da comunidade R (CRAN-R) são ativos, oferecem suporte técnico e possuem grande conhecimento em estatística e programação. Isso tudo dá ao R uma vantagem especial, tornando-o uma boa opção para o desenvolvimento de projetos de ciência de dados.

5. Aprendizado de máquina

Em algum momento no porocesso de ciência de dados, um programador pode precisar treinar o algoritmo e trazer capacidades de automação e aprendizagem de máquina para realizar previsões. O R oferece muitas ferramentas aos desenvolvedores para treinar e avaliar um algoritmo e prever eventos futuros. Assim, o R torna o aprendizado de máquina (um ramo da ciência de dados) muito mais fácil e acessível. A lista de pacotes R para aprendizado de máquina é extensa. Os pacotes de aprendizado de máquina em R contém: **MICE** (para lidar com valores ausentes); **rpart** e **PARTY** (para criar partições de dados); **CARET** (para treinamento de classificação e regressão); **randomFOREST** (para criar árvores de decisão); além de outros.

6. Disponibilidade

A linguagem de programação R é um software de código aberto (**open source**). Isso faz com que seja utilizável em projetos de qualquer tamanho. Como é de código aberto, os desenvolvimentos em R produzem

os códigos em uma rápida escala e a comunidade de desenvolvedores é bastante grande. Em agosto de 2018, o R ficou em 18º lugar no índice TIOBE, uma medida de popularidade das linguagens de programação. (Thieme, 2018). Em conjunto com uma vasta quantidade de recursos para o aprendizado, faz da programação em R uma escolha adequada para começar a aprender programação para ciência de dados. Como há muitos novos desenvolvedores explorando o cenário de programação em R, é mais fácil e econômico recrutar ou terceirizar desenvolvedores de R.

Análise de Dados: Fundamentos da Metodologia

De acordo com Judd et. ali. (2017), a análise de dados é o processo de inspeção, limpeza, transformação e modelagem de dados com o objetivo de descobrir informações úteis, gerar conclusões e apoiar a tomada de decisões. A análise de dados tem múltiplas facetas e abordagens, abrangendo diversas técnicas sob vários nomes, sendo usada em diferentes domínios tais como negócios, ciências e ciências sociais.

Ainda segundo Judd et ali. (2017), em aplicações estatísticas, a análise de dados pode ser dividida em estatística descritiva, análise exploratória de dados (EDA) e análise confirmatória de dados (CDA). O EDA se concentra na descoberta de novos recursos nos dados, enquanto o CDA se concentra na confirmação ou falsificação de hipóteses existentes. A análise preditiva se concentra na aplicação de modelos estatísticos para previsão ou classificação preditiva, enquanto a análise de texto aplica técnicas estatísticas, linguísticas e estruturais para extrair e classificar informações de fontes textuais, uma espécie de dados não estruturados. Todas as opções acima, de acordo com os autores, são variedades de análise de dados.

O processo de Análise de Dados

Um processo de *análise* busca dividir um todo em componentes distintos e interdependentes para se realizar um exame individual. A análise de dados é um processo para obter dados brutos e convertê-los em informações úteis para a tomada de decisão pelos usuários. Os dados são coletados e analisados para responder perguntas, testar hipóteses ou refutar teorias. (JUDD et. al., 2017).

Tukey (1961) definiu a análise de dados em 1961 como: "procedimentos para analisar dados, técnicas para interpretar os resultados de tais procedimentos, formas de planejar a coleta de dados para tornar sua análise mais fácil, mais precisa ou mais depurada, e todas as engenharia e resultados de estatísticas (matemáticas) que se aplicam à análise de dados".

Existem várias fases no processo de análise de dados que podem ser discriminadas. De acordo com O'Neil e Schutt (2013) as fases são iterativas, de tal forma que o feedback das fases posteriores pode resultar em trabalho adicional nas fases anteriores. As principais fases de um processo de análise serão descritas nos subtópicos definidos no trabalho de O'Neil e Schutt (2013).

Requisitos de dados

Os dados necessários são utilizados como entradas para a análise, que é especificada com base nos requisitos daqueles que dirigem a análise ou clientes (que usarão o produto acabado da análise). O tipo geral de entidade sobre o qual os dados serão coletados é referido como uma unidade experimental (por exemplo, uma pessoa ou população de pessoas). Variáveis específicas relativas a uma população (por exemplo, idade e renda) podem ser especificadas e obtidas. Os dados podem ser *numéricos* ou *categóricos* (ou seja, um rótulo de texto para números). (O'NEIL e SCHUTT, 2013).

Coleta de dados

Os dados são coletados de várias fontes. Os requisitos podem ser comunicados pelos analistas aos guardiões dos dados, como o pessoal de tecnologia da informação dentro de uma organização. Os dados também podem ser coletados de sensores no ambiente, como câmeras de tráfego, satélites, dispositivos de gravação, etc.

Também podem ser obtidos por meio de entrevistas, downloads de fontes on-line ou leitura de documentação. (O'NEIL e SCHUTT, 2013).

Processamento de dados

Os dados inicialmente obtidos devem ser processados ou organizados para análise. Esta etapa pode envolver a inserção de dados em linhas e colunas em um formato de tabela (ou seja, dados estruturados) para análise posterior, como em uma planilha ou software estatístico. (O'NEIL e SCHUTT, 2013).

Limpeza de dados

Depois de processados e organizados, os dados podem estar incompletos, conter duplicatas ou conter erros. A necessidade de limpeza de dados surgirá por problemas na forma como os dados foram inseridos e armazenados. Limpeza de dados é o processo de prevenir e corrigir esses erros. (The SunTec India Blog, 2016).

Tarefas comuns incluem correspondência de registros, identificação de imprecisão de dados, qualidade geral de dados existentes, de duplicação e segmentação de colunas. (Microsoft Research, 2012).

Tais problemas nos dados, segundo Koomey (2006), também podem ser identificados através de uma variedade de técnicas analíticas. Por exemplo, em informações financeiras, os totais de variáveis específicas podem ser comparados com números publicados separadamente considerados confiáveis. Quantidades incomuns acima ou abaixo dos limites predeterminados também podem ser revisadas. Existem vários tipos de limpeza de dados que dependem do tipo de dados, como números de telefone, endereços de e-mail, empregadores, etc.

De acordo com Hellerstein (2008), os métodos de dados quantitativos para detecção de outliers podem ser usados para se livrar de possíveis dados inseridos incorretamente. Os verificadores ortográficos de dados textuais podem ser usados para diminuir a quantidade de palavras digitadas incorretamente, mas é mais difícil dizer se as palavras estão corretas.

Análise exploratória de dados (AED)

Depois que os dados estão limpos, podem ser *analisados*. Os analistas podem aplicar uma diversidade de técnicas referidas como **análise exploratória de dados** para começar a entender as mensagens contidas nos dados. Segundo Bahrens (1997), o processo de exploração pode resultar em limpeza adicional de dados ou solicitações adicionais de dados, portanto, essas atividades podem ser de natureza iterativa. Estatísticas descritivas, como a média ou mediana, podem ser geradas para ajudar a entender os dados.

A *visualização de dados* também pode ser usada para examinar os dados em formato gráfico, para obter informações adicionais sobre as mensagens dentro dos dados. (O'NEIL e SCHUTT, 2013).

Modelagem e Algoritmos

Fórmulas matemáticas ou modelos chamados *algoritmos* podem ser aplicados nos dados para identificar relações entre as variáveis, tais como correlação ou causa. Em termos gerais, os modelos podem ser desenvolvidos para avaliar uma variável específica nos dados com base em outra(s) variável(is) nos dados, com algum erro residual dependendo da precisão do modelo (ou seja, Dados = Modelo + Erro). (O'NEIL e SCHUTT, 2013).

Estatísticas inferenciais possuem técnicas para medir as relações entre variáveis específicas. Por exemplo, a *análise de regressão* pode ser usada para modelar se uma mudança na propaganda (variável independente X) explica a variação nas vendas (variável dependente Y). Em termos matemáticos, Y (vendas) é uma função de X (publicidade). Pode ser descrito como $\mathbf{Y = aX + b + erro}$, onde o modelo é projetado de tal forma que a e b minimizam o erro quando o modelo prevê Y para um determinado intervalo de valores de X. Os analistas podem tentar construir modelos descritivos dos dados para simplificar a análise e comunicar os resultados. (O'NEIL e SCHUTT, 2013).

Produto de dados

Um produto de dados é um aplicativo computacional que recebe dados de entrada e gera saídas, realimentando-os de volta ao ambiente. Pode ser baseado em um modelo ou algoritmo. Um exemplo é um aplicativo que analisa dados sobre o histórico de compras do cliente e recomenda outras compras que o cliente possa desfrutar. (O'NEIL e SCHUTT, 2013).

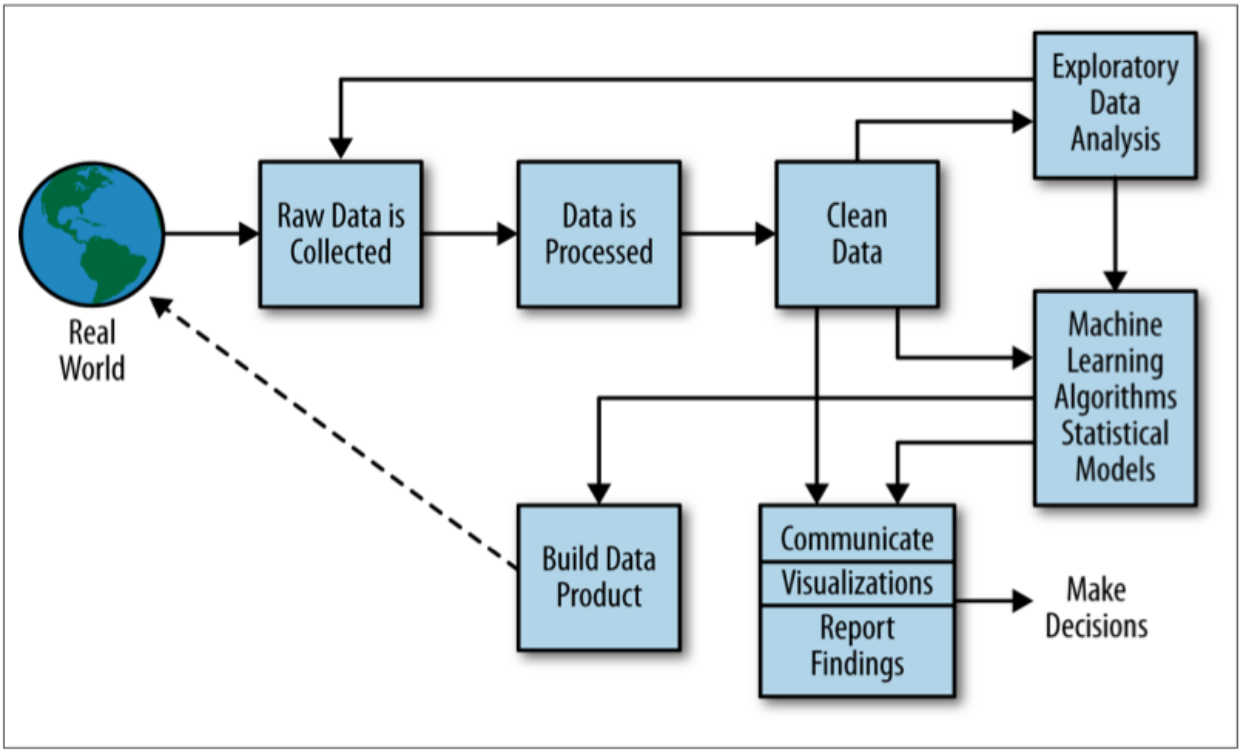
Comunicação

Depois que os dados são analisados, podem ser elaborados relatórios em vários formatos para os usuários da análise buscando dar suporte aos seus requisitos. Os usuários podem obter feedback, o que resulta em análises adicionais. Como tal, grande parte do ciclo analítico é iterativo.(O'NEIL e SCHUTT, 2013).

Ao determinar como comunicar os resultados, o analista pode considerar técnicas de visualização de dados para ajudar a comunicar de forma clara e eficiente a mensagem ao público-alvo. A visualização de dados usa exibições de informações (como tabelas e gráficos) para ajudar a comunicar as principais mensagens contidas nos dados. As tabelas são úteis para um usuário que pode pesquisar números específicos, enquanto gráficos (por exemplo, gráficos de barras ou gráficos de linhas) podem ajudar a explicar as mensagens quantitativas contidas nos dados.

Os tópicos do Processo de Ciência de Dados discutidos podem ser visualizados na Figura 1.

Figura 1. Processo de Ciência de Dados



Fonte: Schutt e O'Neil (2014).

Realizando a Análise Exploratória dos Dados

A finalidade da Análise Exploratória de Dados (AED) é a de *examinar os dados* antes da aplicação de qualquer técnica estatística. Seu principal objetivo é o de obter um entendimento básico dos dados, seu

formato e as relações existentes entre as variáveis carregadas no R.

Leitura e carga dos dados

A primeira ação a ser realizada na Análise Exploratória dos Dados (AED, ou Exploratory Data Analysis, EAD) é a carga dos dados para o R. Os dados que serão utilizados nesta AED estão armazenados no Amazon Web Services e estão disponíveis para acesso no link especificado no trecho de código em R. Grande parte do código utilizado neste trabalho foi desenvolvido pelo proessor da USP/ESALQ, Dr. Adriano Azevedo Filho e estão disponíveis no link:http://rstudio-pubs-static.s3.amazonaws.com/7342_3aee84b4bc9549adb3080f06c69174e1.html.

```
## especifica a localização do arquivo
endereco <-
"http://s3.amazonaws.com/ihbs-html/dados/ODB2013originalcorrigido.csv"
 ## "carga" da tabela em formato .csv para o R
df <- read.csv2(endereco, fileEncoding = "latin1")
## mostrará o número de linhas (casos) e colunas (variáveis) já em formato de data.frame em Rdim(df)
## mostra linhas 1 a 3 e as colunas de 1 a 8 de "df""
df[1:3, 1:8]
```

```
##   Indicação.de.data.e.hora Número Locais.principais.de.trabalho      Sexo
## 1        7/11/2013 8:26:49  67788                      UCR, UAT Masculino
## 2        7/11/2013 8:56:32  65790                           UCR  Feminino
## 3       7/16/2013 12:46:07  65788                           UCR Masculino
##   Data.de.nascimento Altura  Peso Número.do.calçado.que.calça
## 1         10/13/1982   1.71 102.0                          41
## 2           3/9/1988   1.60  58.2                          35
## 3           4/3/1986   1.89  84.0                          44
```

No trecho do código (data chunk) mostrado o símbolo "<-" é um símbolo de atribuição, ou seja, atribui um valor a uma variável. Este símbolo pode também ser substituídos pelo operador de atribuição "=".

O comando (ou função) "read.csv2" lê um arquivo no formato de tabela ".csv" e cria um **data frame** com os casos ou observações correspondo às linhas e as variáveis (correspondendo às colunas) nos campos da estrutura de dados nomeada como " *df* ". Um data frame (estrutura de dados) é uma estrutura interna do R utilizado para armazenar dados em formato correspondente a de uma tabela. São coleções fortemente acopladas de variáveis que compartilham muitas das propriedades das matrizes e de listas, utilizadas como a estrutura de dados fundamental para a maioria dos softwares de modelagem em R.

A opção *fileEncoding="latin1"* em geral não é necessária em sistemas Windows configurados para o Brasil, mas pode ser importante para outros sistemas, como no MAcOs ou Linux, que normalmente estão configurados para o padrão UTF-8.

Como podemos observar o *data frame*, com o nome "df" atribuído, contém vinte e três observações (ou casos) e cinquenta variáveis.

É sempre oportuno verificar se o arquivo foi lido corretamente. Existem muitas funções no R que facilitam a verificação de potenciais problemas. O comando *dim* mostrado no trecho de código anterior permite verificar algumas características do *data frame*. Outros comandos que oferecem mais opções são str(df) - que exibe a estrutura interna de um objeto R de forma compacta -, ou *df[1:3, 1:4]* - que exibe as 3 primeiras linhas do *data frame* "df", e as 8 primeiras variáveis.

Preparação dos Dados

```
names(df)
```

```
##  [1] "Indicação.de.data.e.hora"
##  [2] "Número"
##  [3] "Locais.principais.de.trabalho"
##  [4] "Sexo"
##  [5] "Data.de.nascimento"
##  [6] "Altura"
##  [7] "Peso"
##  [8] "Número.do.calçado.que.calça"
##  [9] "Circunferência.da.barriga..em.cm...na.altura.do.umbigo"
## [10] "Com.relação.ao.uso.das.mãos.você.é"
## [11] "Ensino.fundamental.e.médio..número.de.anos.em.escola.pública"
## [12] "Formação.acadêmica"
## [13] "Como.se.classificaria.como.aluno.a..na.graduação."
## [14] "Status.de.sua.formação.na.graduação"
## [15] "Tipo.de.escola.de.graduação.cursada"
## [16] "Estudos.de.pós.graduação"
## [17] "Descreva.as.áreas.em.que.cursou.pós.e.local..caso.tenha.cursado."
## [18] "Como.avalia.sua.habilidade.de.comunicação."
## [19] "Como.avalia.a.sua.habilidade.com.métodos.quantitativos."
## [20] "Estado"
## [21] "Tipo.de.cidade.em.que.viveu.a.maior.parte.de.sua.vida"
## [22] "classe"
## [23] "Grau.de.instrução.maior.do.pai.ou.da.mãe"
## [24] "Conhecimento.de.Inglês...Leitura"
## [25] "Conhecimento.de.Inglês...compreenção.auditiva.da.lingua.falada"
## [26] "Conhecimento.de.inglês...conversação"
## [27] "Conhecimento.de.inglês...habilidade.em.escrever"
## [28] "Outras.línguas.com.proficiência.elementar.ou.intermediária"
## [29] "Outras.línguas.com.proficiência.muito.boa.ou.excelente"
## [30] "Religião"
## [31] "Se.respondeu..Outra...na..pergunta.anterior..especifique.qual.é"
## [32] "Tipo.de.música.preferida"
## [33] "Indique.outras.músicas.que.gosta.caso.não.tenham.sido.especificadas.no.ítem.anterior"
## [34] "Hobbies.prediletos"
## [35] "Indique.outros.Hobbies..caso.não.tenham.sido.descritos.na.pergunta.anterior"
## [36] "Fumo"
## [37] "Consumo.de.bebida.alcoólica..indique.o.número.de.doses.consumidas..por.semana."
## [38] "Animal.de.estimação"
## [39] "Time.de.futebol.para.o.qual.torçe"
## [40] "Satisfação.pessoal.com.a.profissão.que.escolheu"
## [41] "Quando.ingressou.na.Odebrecht"
## [42] "Forma.de.ingresso.na.Odebrecht"
## [43] "Quantos.lançamentos.teve.desde.o.seu.ingresso.na.Odebrecht"
## [44] "Área.em.que.trabalha"
## [45] "Indique.o.tempo.em.minutos.que.gasta.por.dia.para.ir.e.voltar.da.sua.casa.para.a.empresa"
## [46] "Qual.o.seu.custo.mensal.de.moradia.e.alimentação."
## [47] "O.que.já.contribuiu.para.o.sucesso.da.empresa.em.que.trabalha"
## [48] "Se.tiver..descreva.outras.habilidades.profissionais.importantes.que.têm.e.que..não.tenham.sido
## [49] "Quantos.livros.leu.em.2013"
## [50] "Caso.tenha.lido.algum.livro.em.2013..indique.o.título.dos.livros.que.leu.em.2013..separados.po
```

Os dados inicialmente carregados devem ser processados ou organizados antes da realização da análise. Como demonstrado no trecho de código exibido, neste caso específico, verifica-se que o nome das variáveis são longos e podem dificultar a sua caracterização e/ou manipulação no R.

Os nomes longos das variáveis serão modificados para atributos mais compactos afim de facilitar sua manipulação nas futuras análises. Além disso, como os nomes são usados como rótulos para identificação das variáveis nos resultados (tabelas, gráficos, etc.), nomes muito longos podem ser inconvenientes.

Antes de fazer a alteração, por garantia, armazena-se os nomes das variáveis originais num vetor:

```
nomesorig <- names(df)  # preservando os nomes originais das variáveis
```

A alteração dos nomes é realizada pelO seguinte trecho de código:

```
novonome <- c("dh", "num", "loc", "sex", "dan", "alt", "pes", "cal", "cir",
    "mao", "pub", "fac", "alu", "sta", "uni", "pg1", "pg2", "hco", "hmq", "est",
    "cid", "cso", "ipa", "in1", "in2", "in3", "in4", "ol1", "ol2", "rel1", "rel2",
    "mu1", "mu2", "ho1", "ho2", "fum", "alc", "ani", "tim", "sat", "odi", "odf",
    "odl", "oda", "odt", "cus", "con", "out", "nlv", "liv")

names(df) <- novonome ## atribuindo novos nomes de variáveis

names(df)  ## exibindo os novos nomes
```

```
##  [1] "dh"   "num"  "loc"  "sex"  "dan"  "alt"  "pes"  "cal"  "cir"  "mao"
## [11] "pub"  "fac"  "alu"  "sta"  "uni"  "pg1"  "pg2"  "hco"  "hmq"  "est"
## [21] "cid"  "cso"  "ipa"  "in1"  "in2"  "in3"  "in4"  "ol1"  "ol2"  "rel1"
## [31] "rel2" "mu1"  "mu2"  "ho1"  "ho2"  "fum"  "alc"  "ani"  "tim"  "sat"
## [41] "odi"  "odf"  "odl"  "oda"  "odt"  "cus"  "con"  "out"  "nlv"  "liv"
```

Com relação ao *code chunk* anterior, deve-se observar que a *função "c"* do R cria um *vetor*, concatenando elementos separados por vírgulas. No caso, criou-se um vetor de textos ou "string". Cada "string" substituirá o "string" original que definia o nome da variável, na ordem mostrada quando da primeira execução de **names(alunos)**.

Da mesma forma que se pode alterar os nomes das variáveis, podemos também modificar os nomes ou identificadores que caracterizam as *categorias*" ou *níveis* de variáveis qualitativas chamadas de fatores (factor).

Para ter acesso a cada variável, precedemos o nome da variável com o nome do *data.frame* ao qual a variável pertence, separando os dois nomes com o símbolo **$**.

Podemos ver os valores da variável qualitativa sexo (no caso *df$sex*), usando o seguinte trecho de código em R:

```
df$sex
```

```
##  [1] Masculino Feminino  Masculino Feminino  Masculino Masculino Masculino
##  [8] Masculino Feminino  Masculino Feminino  Feminino  Masculino Feminino
## [15] Masculino Feminino  Feminino  Feminino  Masculino Masculino Masculino
## [22] Masculino Masculino
## Levels: Feminino Masculino
```

Para alterar os nomes (*levels*) dessas categorias, que indicam o sexo do aluno (*Masculino e Feminino*), para **"f"** e **"m"**, com letras minúsculas, deve-se inicialmente observar a ordem em que os níveis aparecem e realizar as modificações desejadas como indicado no trecho de código:

```
levels(df$sex)  ## mostra os níveis ou categorias da variável sex no data.frame df
```

```
## [1] "Feminino"  "Masculino"
```

```
levels(df$sex) <- c("f", "m")  # troca por identificadores mais sintéticos
levels(df$sex)  ## exibe os novos nomes, já alterados na variável
```

```
## [1] "f" "m"
```

O R assume uma ordem para os níveis, a qual é a apresentada quando o *comando levels* é utilizado, como foi feito. Para mudar essa ordem, que é algo que pode ser interessante em algumas análises, podemos usar (nesse caso):

```
df$sex <- factor(df$sex, levels(df$sex)[c(2, 1)])  # reordenação de níveis
levels(df$sex)  # observar a reordenação exibida
```

```
## [1] "m" "f"
```

O vetor **c(2,1)** mostra as novas posições para os níveis originais. O nível 1 vai para 2 e o nível 2 vai para 1. Pode-se proceder de forma similar se existirem mais níveis. Para retornar à forma anterior, usamos novamente o trecho de código:

```
df$sex <- factor(df$sex, levels(df$sex)[c(2, 1)])
levels(df$sex)  # note o retorno à ordem anterior
```

```
## [1] "f" "m"
```

Nos próximo trecho de código, serão modificados os nomes de níveis de variáveis cuja denominação é muito longa e também será realizada a redefinição de níveis:

```
levels(df$loc)
```

```
## [1] "UAT"      "UCR"      "UCR, UAT"
```

```
levels(df$loc) <- c("at", "cr", "po")

levels(df$mao)
```

```
## [1] "Canhoto (usa a mão esquerda para escrever)"
## [2] "Destro (usa a mão direita para escrever)"
```

```
levels(df$mao) <- c("c", "d")

levels(df$fac)
```

```
##  [1] "Administração de Empresas"    "Engenharia Agrícola"
##  [3] "Engenharia Agronômica"        "Engenharia Ambiental"
##  [5] "Engenharia de Alimentos"      "Engenharia de Automação"
##  [7] "Engenharia de Meio Ambiente"  "Engenharia Eletrica"
##  [9] "Engenharia Mecânica"          "Engenharia Mecatrônica"
## [11] "Engenharia Produção Mecânica" "Engenharia Química"
```

```
levels(df$fac) <- c("adm", "eagri", "eagro", "eamb", "eali", "eauto", "emambi", "ee", "emec", "emeca",

## Criando novas variáveis (tempg e cures) para facilitar a análise dos cursos de pg
levels(df$pg1)
```

```
## [1] "Completei 1 ou mais cursos de especialização"
## [2] "Completei 1 ou mais cursos de especialização, Estou cursando a pós em Engenharia de Segurança"
## [3] "Estou cursando a pós em Engenharia de Segurança"
## [4] "Estou cursando a pós em Engenharia de Segurança, Estou cursando um ou mais cursos de especializa
## [5] "Estou cursando um ou mais cursos de especialização"
## [6] "Nunca cursei Pós graduação"
## [7] "Tenho mestrado, Estou cursando a pós em Engenharia de Segurança, Estou cursando um ou mais curso
```

```
df$tempg <- df$pg1
levels(df$tempg) <- c("esp", "esp", "cur", "cur", "cur", "nc", "msc")
df$cures <- df$pg1
levels(df$cures) <- c("n", "s", "s", "s", "n", "n", "s")
```

```r
# Criando variável df$itot com os pontos totais no inglês
df$itot <- df$in1 + df$in2 + df$in3 + df$in4

levels(df$ani)
```

```
## [1] "Não tenho animal de estimação" "Tenho um ou mais cachorros"
```

```r
levels(df$ani) <- c("n", "s")
levels(df$tim)
```

```
##  [1] "Atlético Mineiro"             "Corinthians"
##  [3] "Cruzeiro"                     "E C Vitória"
##  [5] "Flamengo"                     "Goiás"
##  [7] "Grêmio"                       "Não me interesso por futebol"
##  [9] "Outro time"                   "Palmeiras"
## [11] "Santos"                       "São Paulo"
```

```r
levels(df$tim) <- c("am", "co", "cr", "vi", "fl", "go", "gr", "ni", "ot","pa", "sa", "sp")
levels(df$odf)
```

```
## [1] "Jovem Parceiro" "Outras formas"
```

```r
levels(df$odf) <- c("j", "o")
levels(df$oda)
```

```
## [1] "Agrícola (Operação)"
## [2] "Agrícola (Planejamento e/ou Controle)"
## [3] "Ambiente"
## [4] "Indústria (Planejamento e/ou Controle)"
## [5] "Manutenção Automotiva"
## [6] "Manutenção Automotiva, Manutenção Industrial"
## [7] "Manutenção Industrial"
## [8] "Parcerias e Fornecedores"
```

```r
levels(df$oda) <- c("agop", "agpc", "ambi", "inpc", "mana", "manai", "mani","parf")
```

Análise Descritiva

Após a *limpeza* e a *modificação* (preparação dos dados) no data.frame, o próximo passo é a realização de uma *análise descritiva*. Esta etapa é fundamental, pois uma análise descritiva detalhada permite ao pesquisador familiarizar-se com os dados, organizá-los e sintetizá-los de forma a obter as informações necessárias para responder as questões estudadas.

Filtros, seleções e estatísticas (medidas-resumo) incondicionais e condicionais

Um recurso forte da linguagem do R é a facilidade de se observar, modificar e filtrar observações de variáveis a partir de critérios lógicos definidos, assim como possibilitar a obtenção de estatísticas condicionais. Alguns exemplos serão realizados a seguir para ilustrar as possibilidades.

Observação e modificação

```r
## Acesso a observação 3 da variável df$alt
df$alt[3]
```

```
## [1] 1.89
```

```
## Acesso às observações 2, 3 e 7
df$alt[c(2, 3, 7)]
```

```
## [1] 1.60 1.89 1.87
```

```
## Modificando valores de vetores
alt2 <- df$alt[c(2, 3, 7)]  ## criando uma réplica de df$alt
alt2[c(2, 3, 7)] <- c(1.5, 1.72, 1.8)  ## alterando as observações 2, 3 e 7
prop.table(table(df$sex))
```

```
##
##         f         m
## 0.3913043 0.6086957
```

Filtros

Quando usamos um vetor com valores lógicos como argumento para os índices das variáveis, extraímos os valores da variável que coincidem com o resultado lógico TRUE (Verdade), obtido pela aplicação do teste.

Por exemplo, se quisermos obter os valores de altura para as observações associadas a mulheres usaríamos o trecho de código:

```
## Observações de altura dos alunos do sexo feminino
df$alt[df$sex == "f"]
```

```
## [1] 1.60 1.65 1.69 1.64 1.60 1.70 1.58 1.64 1.58
```

Observa-se que a avaliação de **alunos$sex=="f"** resultará em:

```
## Observações de altura dos alunos do sexo feminino
df$sex == "f"
```

```
##  [1] FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE
## [12]  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE
## [23] FALSE
```

Somente os valores de **df$alt** correspondentes às posições que têm o resultado TRUE foram selecionadas no comando anterior.

Para obtermos as observações de altura correspondentes às mulheres (f) trabalhando em Alto Taquari (at) utilizamos:

```
## Observações de altura dos alunos do sexo feminino
df$alt[df$sex == "f" & df$loc == "at"]  # (& corresponde ao **e** lógico)
```

```
## [1] 1.65 1.69 1.58 1.64
```

Para observações da altura correspondentes às mulheres (f) ou pessoas com peso igual ou acima de 70 kg:

```
## Observações de altura dos alunos do sexo feminino
df$alt[df$sex == "f" | df$pes >= 70]  # (| corresponde ao **ou** lógico)
```

```
##  [1] 1.71 1.60 1.89 1.65 1.83 1.87 1.75 1.69 1.74 1.64 1.60 1.72 1.70 1.80
## [15] 1.58 1.64 1.58 1.85 1.82 1.84 1.70
```

Em R, alguns operadores lógicos mais usuais são:

1. == (igual exatamente)
2. is.equal() (igual aproximadamente)
3. (maior) >, < (menor)
4. (maior ou igual)>=, <= (menor ou igual)

5. <> (diferente)
6. & (e lógico), | (ou lógico)
7. parêntesis podem ser utilizados para deixar clara a prioridade das operações

Estatísticas elementares condicionais e incondicionais (variáveis quantitativas)

Pode-se trabalhar com os resultados da filtragem, que também será um vetor. O *code chunk* a seguir mostra o uso dos comandos mean (média), sd (desvio padrão), median (mediana), max (máximo) e min (mínimo), which.max (qual o índice do máximo valor) e which.min (qual é o índice do menor valor) a partir da utilização de filtros (em situações que denominamos de estatísticas condicionais)

```
## estatísticas elementares incondicionais
mean(df$alt)  # média
```

```
## [1] 1.718261
```

```
median(df$alt)  # mediana
```

```
## [1] 1.71
```

```
sd(df$alt)  # desvio padrão
```

```
## [1] 0.09929992
```

```
max(df$alt)  # máximo
```

```
## [1] 1.89
```

```
min(df$alt)  # mínimo
```

```
## [1] 1.58
```

```
## estatísticas elementares condicionais
mean(df$alt[df$sex == "f"])  # média da altura das mulheres
```

```
## [1] 1.631111
```

```
mean(df$alt[df$sex == "m"])  # média da altura dos homens
```

```
## [1] 1.774286
```

```
# algumas estatísticas do núm do sapato para pessoas
# com altura maior que 1,6 trabalhando em Alto Taquari
median(df$cal[df$alt > 1.6 & df$loc == "at"])
```

```
## [1] 39.5
```

```
sd(df$cal[df$alt > 1.6 & df$loc == "at"])
```

```
## [1] 2.386719
```

```
max(df$cal[df$alt > 1.6 & df$loc == "at"])
```

```
## [1] 43
```

```
min(df$cal[df$alt > 1.6 & df$loc == "at"])
```

```
## [1] 36
```

Para encontrar-se a altura da pessoa com maior peso utiliza-se o seguinte trecho de código:

```
df$alt[which.max(df$pes)]
```

```
## [1] 1.82
```

Deve-se explorar os dados do data.frame *df* para se familiarizar com essas e outras opções. Uma opção interessante no R para estatísticas elementares condicionais, é o *tapply* ilustrado no *data chunk*, com o cômputo da média de altura por sexo e da média de altura por sexo e local de trabalho.

```
tapply(df$alt, df$sex, mean)
```

```
##        f        m
## 1.631111 1.774286
```

```
tapply(df$alt, list(df$sex, df$loc), mean)
```

```
##     at       cr    po
## f 1.64 1.605000 1.700
## m 1.76 1.791667 1.765
```

Análise básica de frequências - variáveis qualitativas

Para calcular as frequências de cada sexo nas observações do conjunto de dados (variável df$sex, com categorias "f" e "m"), podemos usar os filtros e comandos como *length* (comprimento do vetor) e *sum* (soma) dos elementos (a soma de um vetor lógico assume o valor 1 para os resultados TRUE e 0 para os resultados FALSE).

```
n <- length(df$sex)  ## definindo o número de observações
sum(df$sex == "f")  ## frequência absoluta de mulheres (possibilidade 1)
```

```
## [1] 9
```

```
length(df$sex[df$sex == "f"])  ## frequência absoluta de mulheres (possibilidade 1)
```

```
## [1] 9
```

```
sum(df$sex == "f")/n  ## frequência relativa de mulheres
```

```
## [1] 0.3913043
```

```
length(df$sex[df$sex == "m"])  ## frequência absoluta de homens
```

```
## [1] 14
```

```
length(df$sex[df$sex == "m"])/n  ## frequência relativa de mulheres
```

```
## [1] 0.6086957
```

Há muitas formas de análise de frequências no R, além da forma mostrada nos no trecho de código anterior. Uma forma elementar, mas prática, utiliza os comandos *table* e *prop.table*:

```
## Sexo dos df
table(df$sex)
```

```
##
##  f  m
##  9 14
```

```
prop.table(table(df$sex))
```

```
##
##         f         m
## 0.3913043 0.6086957
```

```
table(df$loc)
```

```
##
```

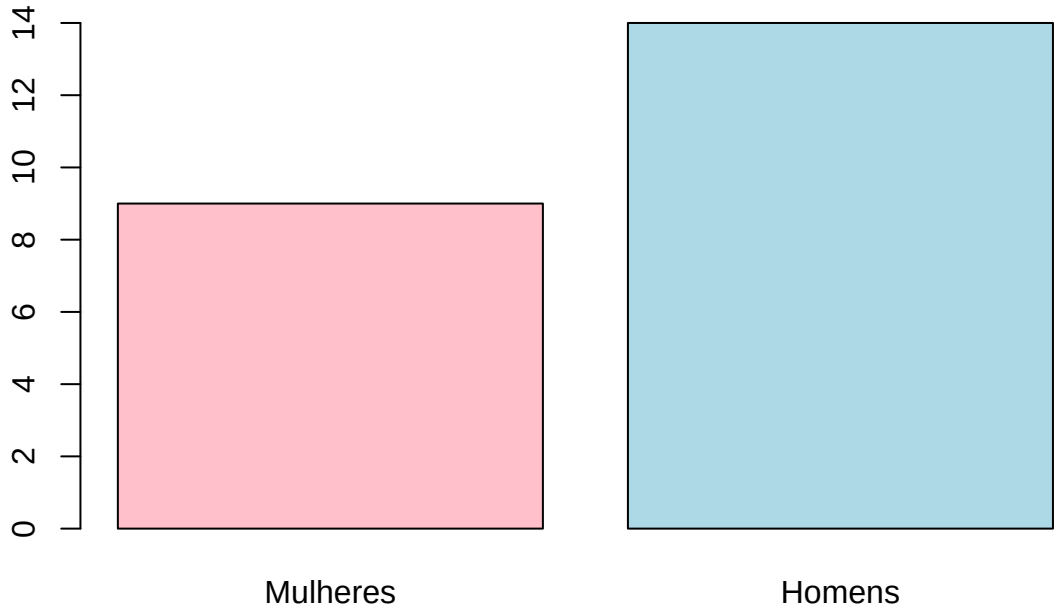
```
## at cr po
## 10 10  3
```

```
prop.table(table(df$loc))
```

```
##
##        at        cr        po
## 0.4347826 0.4347826 0.1304348
```

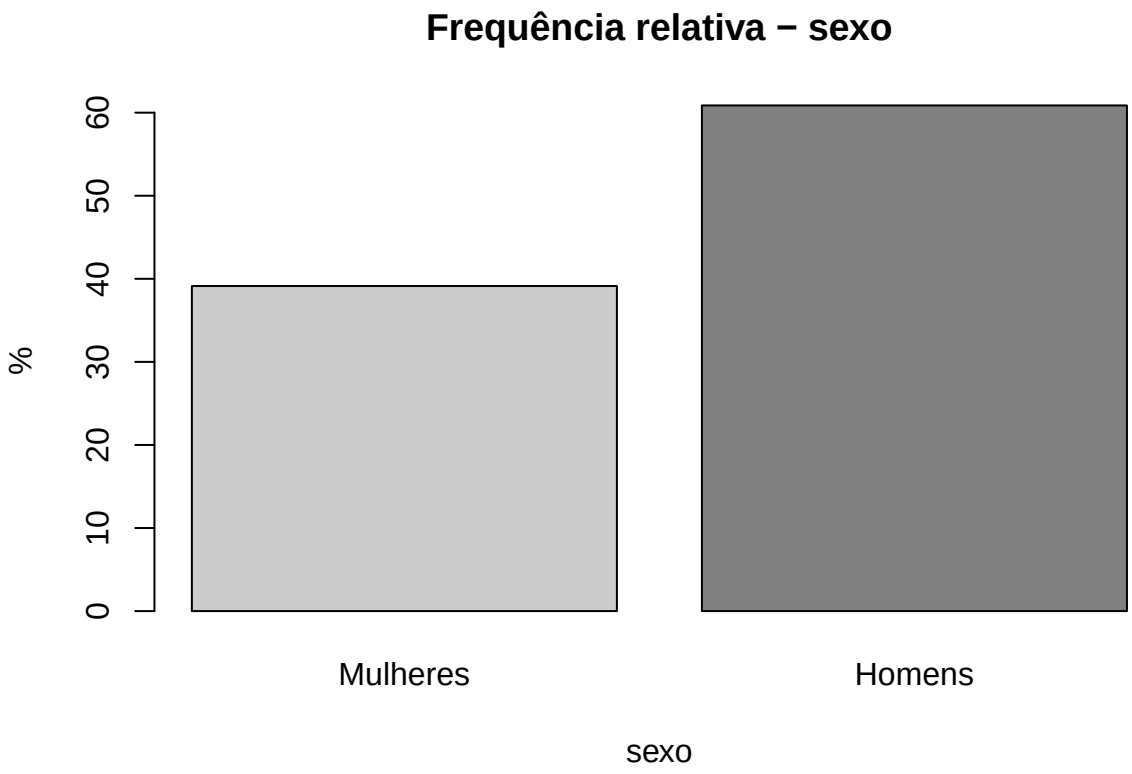
As frequências podem ser visualizadas graficamente, usando gráficos de barras elementares, que se aplicam à descrição de qualquer vetor de dados ou tabelas, como nos exemplo:

```
ident <- c("Mulheres", "Homens")
barplot(table(df$sex), names.arg = ident, col = c("pink", "lightblue"))
```



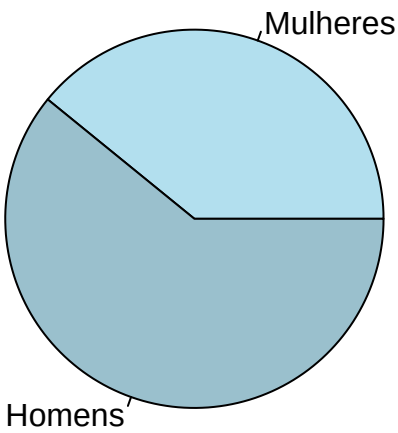
Pode-se definir cores em tonalidades de cinza, usando a função gray(x) em que **x** é um valor entre 0 e 1 (0 é o preto e 1 é o branco). O próximo gráfico ilustra essa e outras opções:

```
ident <- c("Mulheres", "Homens")
barplot(prop.table(table(df$sex)) * 100, names.arg = ident, col = c(gray(0.8), gray(0.5)))
title(main = "Frequência relativa - sexo", xlab = "sexo", ylab = "%")
```



É comum também a apresentação de dados de frequências em gráficos do tipo *pizza*, mas esses são em geral não são recomendados por especialistas em visualização de dados quando o número de níveis é muito grande (o gráfico de barras também é mais recomendado para a visualização de diferenças).

```
ident <- c("Mulheres", "Homens")
pie(prop.table(table(df$sex)) * 100, label = ident, col = c("lightblue2","lightblue3"))
title(main = "Frequência relativa - sexo")
```



Também pode-se utilizar filtros, para obter, por exemplo, as frequências conjuntas absolutas (e daí as relativas), associadas às variáveis **df***sex* ∗ ∗*e* ∗ ∗*df***loc**:

```
sum(df$sex == "f" & df$loc == "at")
```

```
## [1] 4
```

```
sum(df$sex == "m" & df$loc == "at")
```

```
## [1] 6
```

```
sum(df$sex == "f" & df$loc == "cr")
```

```
## [1] 4
```

```
sum(df$sex == "m" & df$loc == "cr")
```

```
## [1] 6
```

```
sum(df$sex == "f" & df$loc == "po")
```

```
## [1] 1
```

```
sum(df$sex == "m" & df$loc == "po")
```

```
## [1] 2
```

Ao dividir-se os valores obtidos no comando anterior por *n<-length(df$sex)*, pode-se obter as frequências relativas conjuntas:

```
table(df$sex, df$loc)
```

```
##
##     at cr po
##   f  4  4  1
##   m  6  6  2
```

```
prop.table(table(df$sex, df$loc))
```

```
##
##            at         cr         po
##   f 0.17391304 0.17391304 0.04347826
##   m 0.26086957 0.26086957 0.08695652
```

Pode-se alterar o número de dígitos significativos apresentado (para 3 por exemplo), usando a opção:

```
oldoptions <- options()  # preservando as opções existentes
options(digits = 3)
```

Não há restrição com relação ao número de variáveis em tabelas de frequência conjunta. O próximo trecho de código mostra as frequências conjuntas de todas as possibilidades envolvendo 3 variáveis:

```
prop.table(table(df$odf, df$loc, df$sex))
```

```
## , ,  = f
##
##
##         at     cr     po
##   j 0.1739 0.0870 0.0435
##   o 0.0000 0.0870 0.0000
##
## , ,  = m
##
##
##         at     cr     po
##   j 0.1304 0.1304 0.0000
##   o 0.1304 0.1304 0.0870
```

As frequências condicionais podem se obtidas através de *prop.table*. Por exemplo, a frequência de homens (m) condicional ao local de trabalho ser Alto Taquari (at) será dada por:

```
prop.table(table(df$sex, df$loc), 2)
```

```
##
##        at    cr    po
##   f 0.400 0.400 0.333
##   m 0.600 0.600 0.667
```

Situações envolvendo 3 variáveis (condicional em sexo):

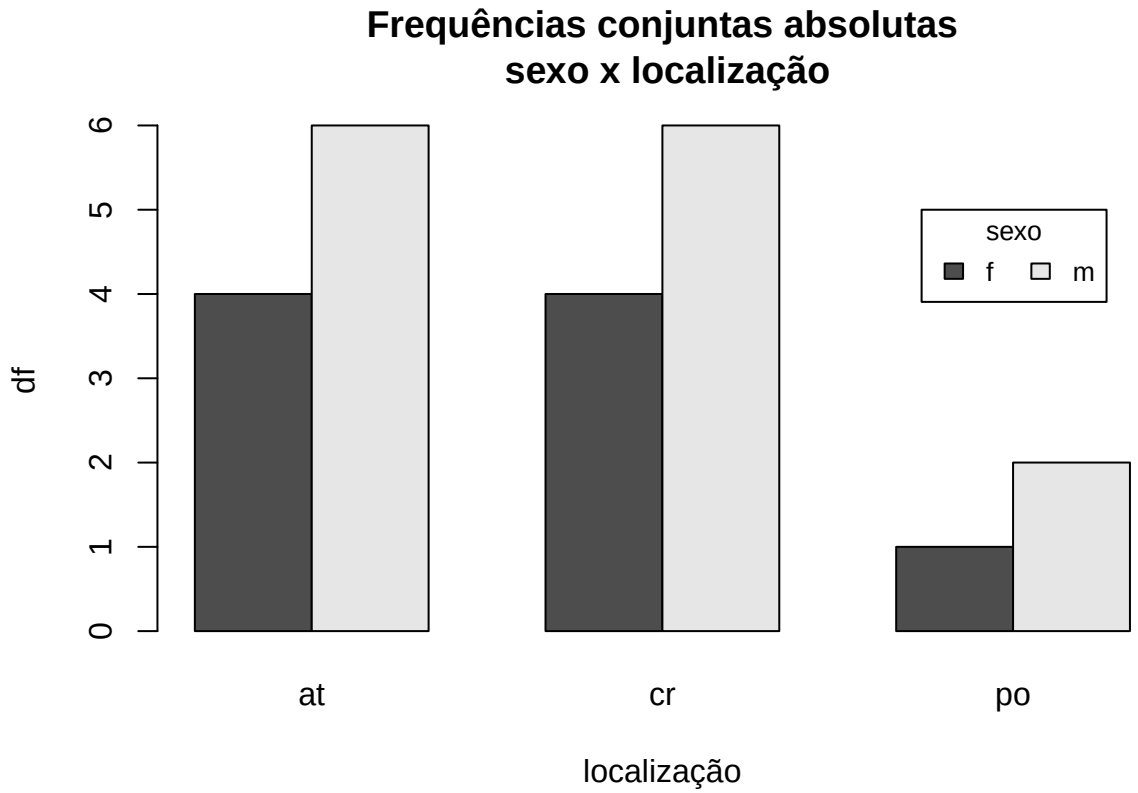
```
prop.table(table(df$odf, df$loc, df$sex), 3)
```

```
## , ,  = f
##
##
##        at    cr    po
##   j 0.444 0.222 0.111
##   o 0.000 0.222 0.000
##
## , ,  = m
##
```

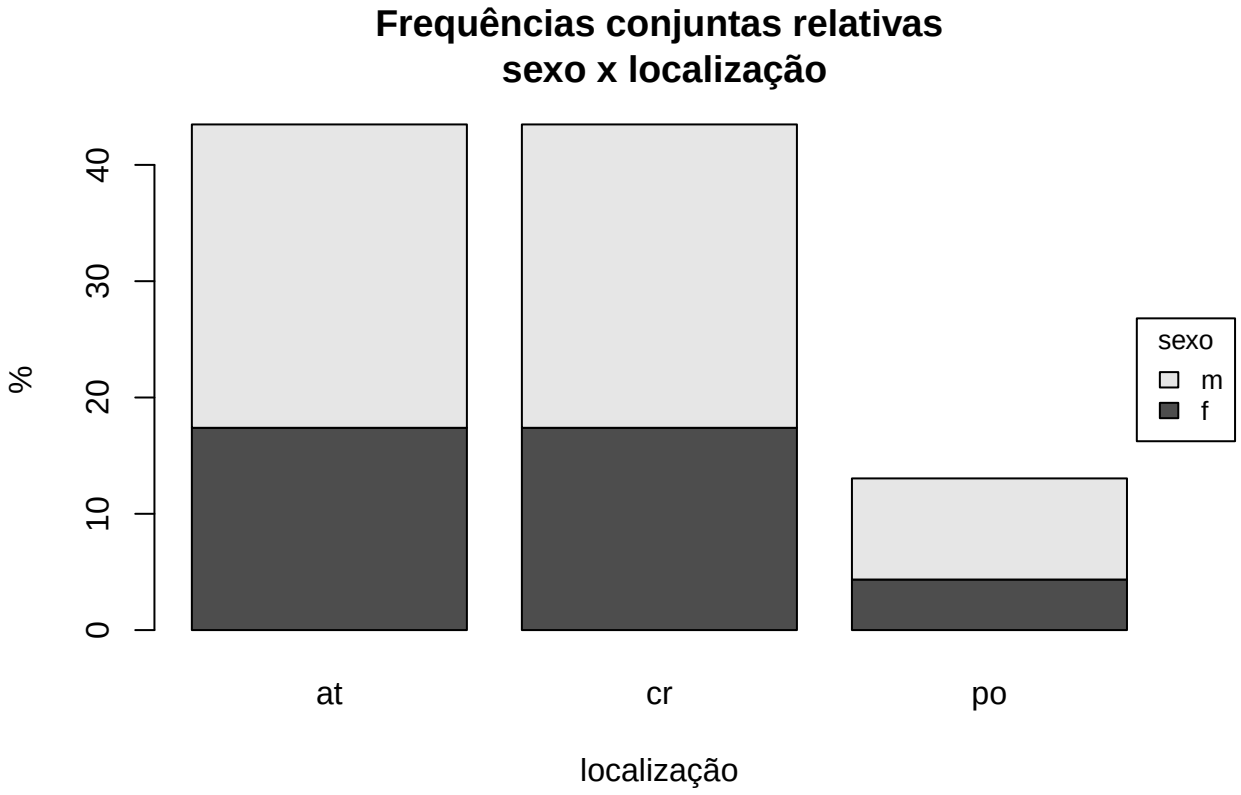
```
##
##        at    cr    po
##   j 0.214 0.214 0.000
##   o 0.214 0.214 0.143
```

Há muitos recursos importantes (**packages** específicos, como **vcd** por exemplo) especializados na análise e visualização gráfica de frequências conjuntas e condicionais. Mostraremos alguns recursos elementares fundamentados nas funções básicas e um breve exemplo do uso do *package* **vcd** (que deve ser instalado no computador) e uso elementar de funções básicas.

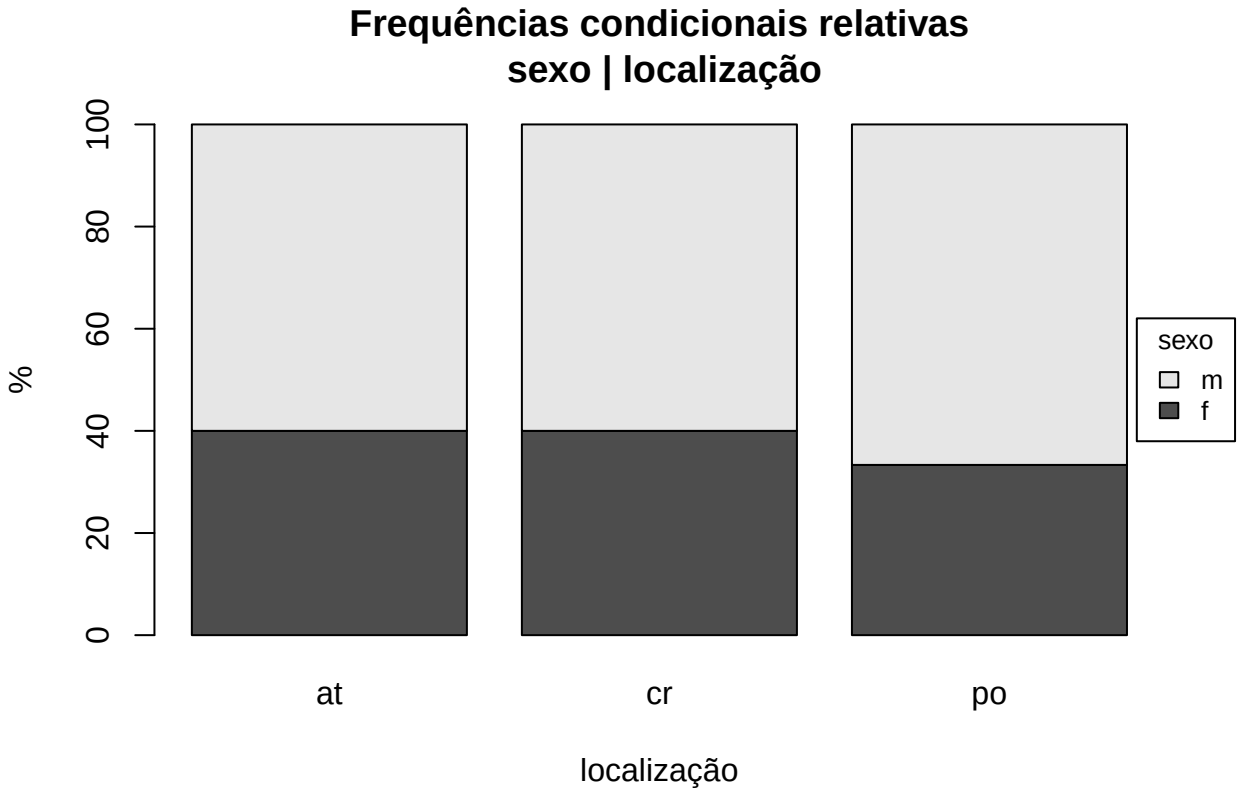
```
## gráfico de barras justapostas (segunda variável no eixo x) - Frequência
## conjunta absoluta
barplot(table(df$sex, df$loc), beside = TRUE, legend.text = TRUE, args.legend = list(x = 8.8,
title("Frequências conjuntas absolutas\n sexo x localização", xlab = "localização",
      ylab = "df")
```




```
## gráfico de barras empilhadas (segunda variável no eixo x) - Frequência
## conjunta relativa
barplot(prop.table(table(df$sex, df$loc)) * 100, legend.text = TRUE,
        xpd = TRUE, args.legend = list(x = "right", title = "sexo", horiz = FALSE,
                                       inset = -0.07, cex = 0.8))
title("Frequências conjuntas relativas\n sexo x localização", xlab = "localização",
      ylab = "%")
```

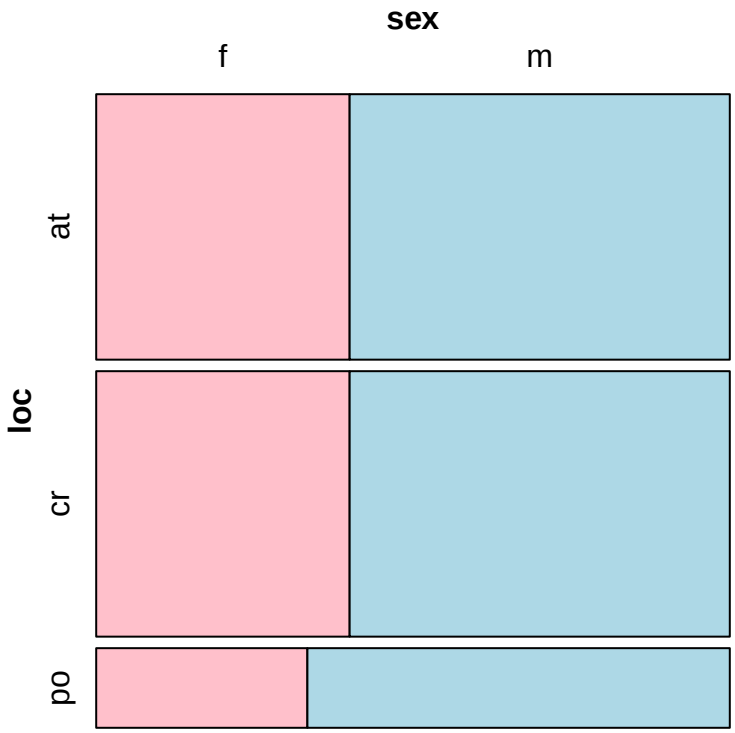



```
## Gráfico de frequência condicional (df$sex|df$loc)
barplot(prop.table(table(df$sex, df$loc), 2) * 100, legend.text = TRUE,
        xpd = TRUE, ylim = c(0, 100), args.legend = list(x = "right", title = "sexo",
                                                         horiz = FALSE, inset = -0.07, cex = 0.8))
title("Frequências condicionais relativas\n sexo | localização", xlab = "localização",
      ylab = "%")
```

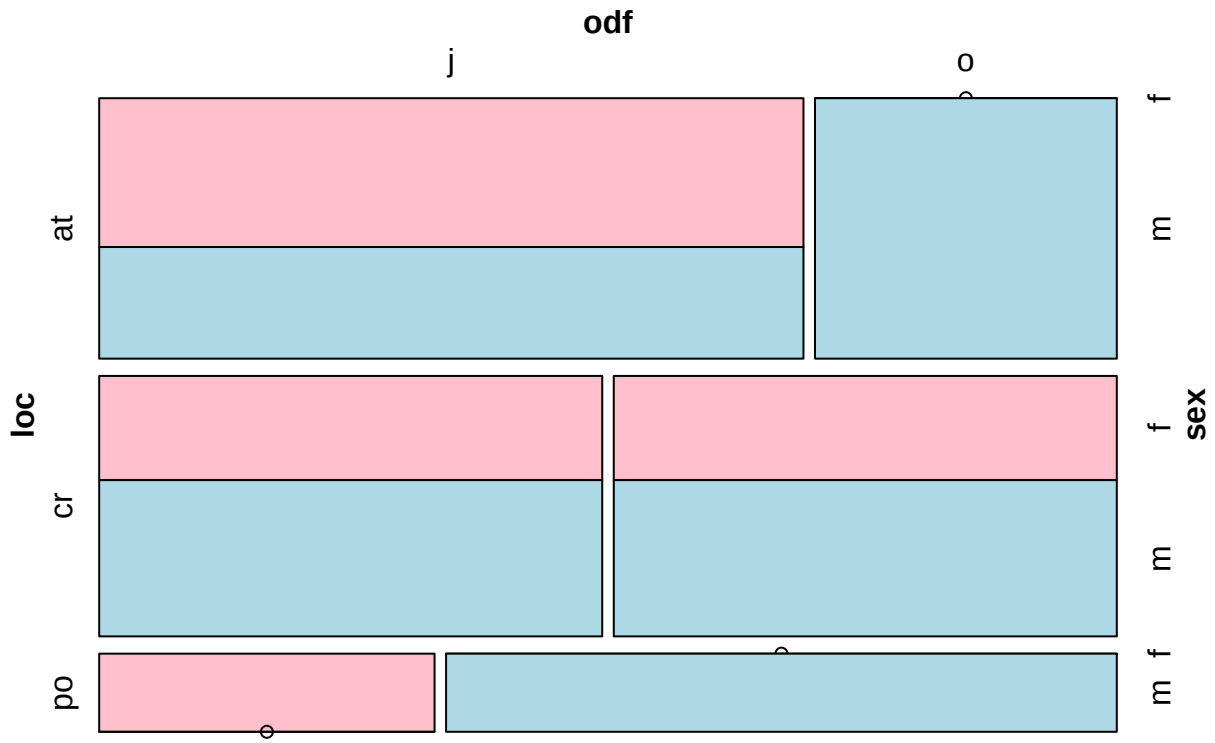


Os próximos exemplos utilizam o gráfico tipo Mosaico do *package* **vcd** que deve estar instalado para ser executado. Nos gráficos as regiões são proporcionais ao número de pessoas em cada categoria.

```
## carregamento do package vcd (deve ser instalado antes)
require(vcd)
## gráfico tipo mosaico - frequências condicionais sexo | localização
mosaic(sex ~ loc, data = df, highlighting_fill = c("pink", "lightblue"))
```

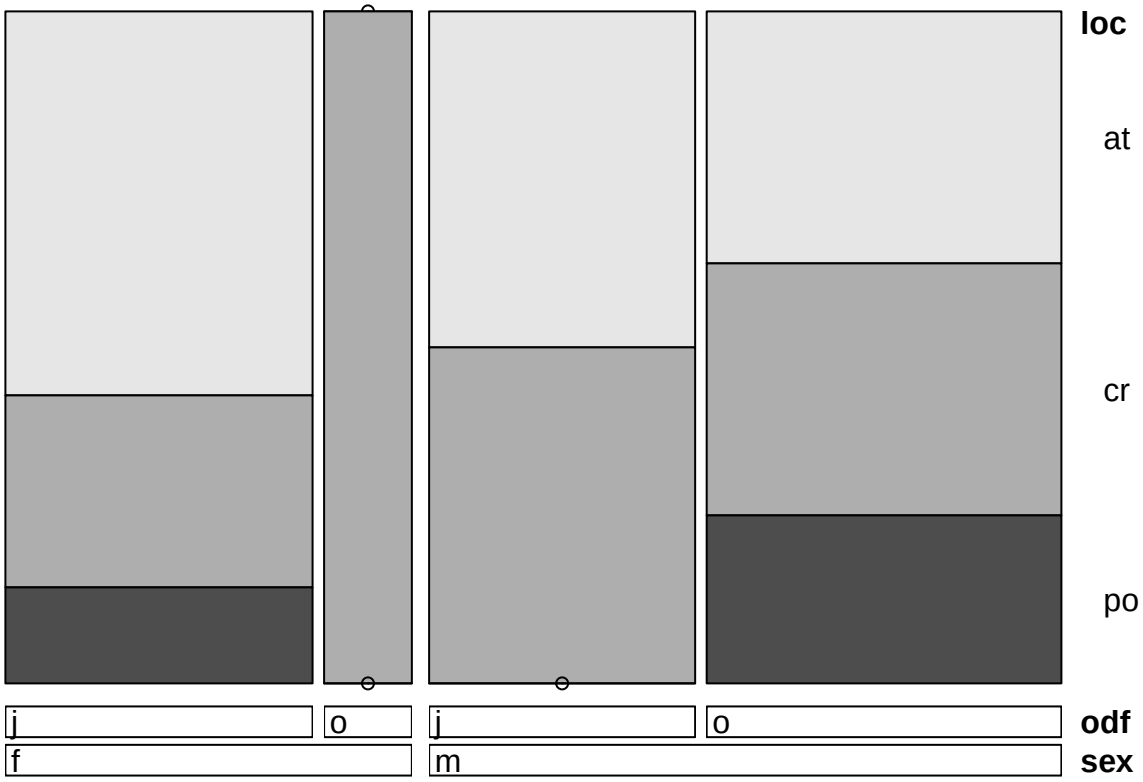



```
## gráfico tipo mosaico - frequências condicionais sexo | localização,
## tipo de ingresso
mosaic(sex ~ loc + odf, data = df, highlighting_fill = c("pink", "lightblue"))
```



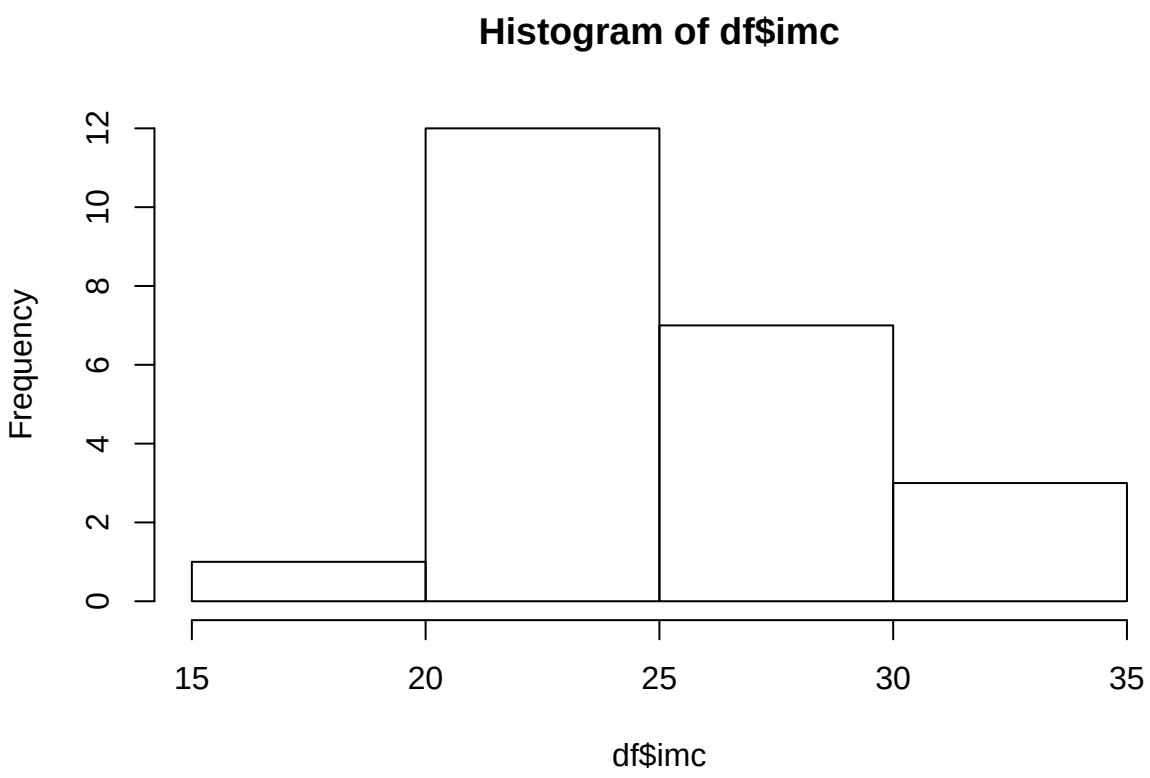
Também do *package vcd* há o gráfico tipo **doubledecker** (de dois andares) que é útil para apresentar dados de frequências condicionais. Da mesma forma que no caso anterior, as regiões são proporcionais ao número de pessoas em cada categoria.

```
## carregamento do package vcd (deve ser instalado antes)
require(vcd)
## gráfico tipo doubledecker, frequência condicional sexo | localização,
## tipo de ingresso
doubledecker(loc ~ sex + odf, data = df)
```

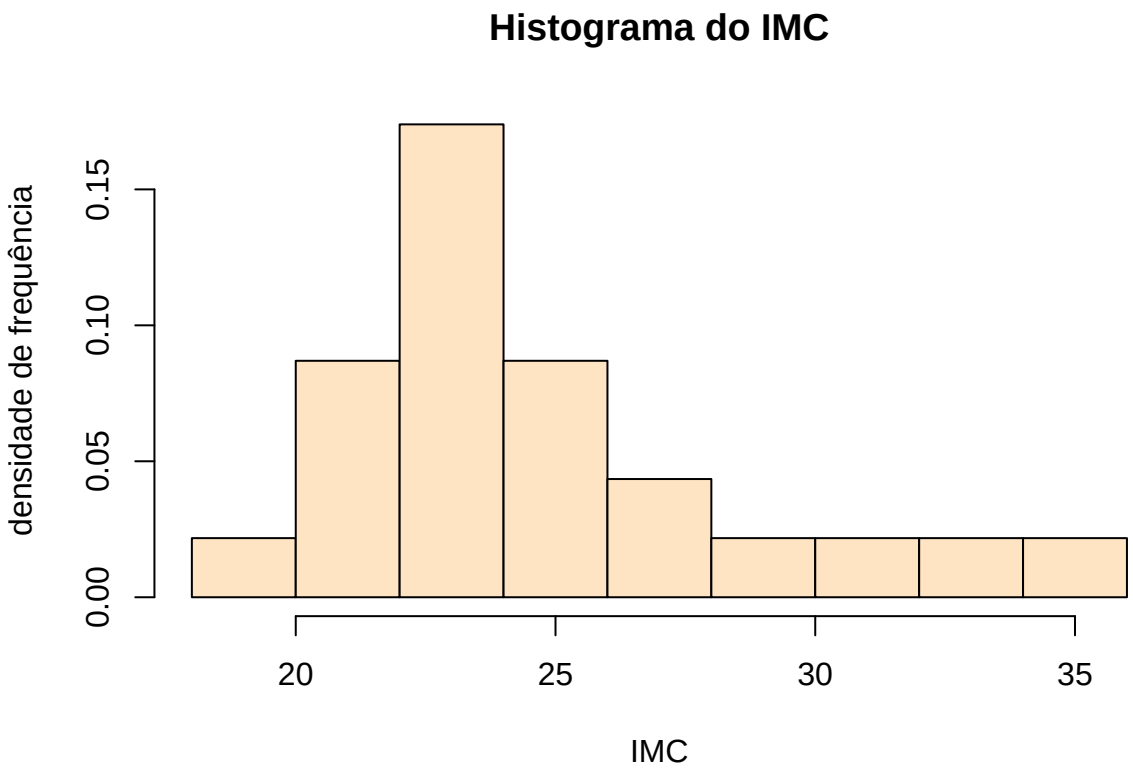


A visualização de variáveis contínuas através de gráficos pode considerar muitos conceitos diferentes. Dois deles são bem fundamentais: o *histograma* e o *boxplot* (diagrama de Tuckey) os quais serão ilustrados na descrição de variáveis quantitativas do levantamento. Para tanto examinaremos o índice de massa corporal dos alunos (que denominaremos imc). Caso seja definido, o argumento breaks tenta especificar o número de categorias que o histograma irá considerar.

```
df$imc <- df$pes/df$alt^2
## histograma do peso dos df - básico
hist(df$imc, breaks = 5)
```

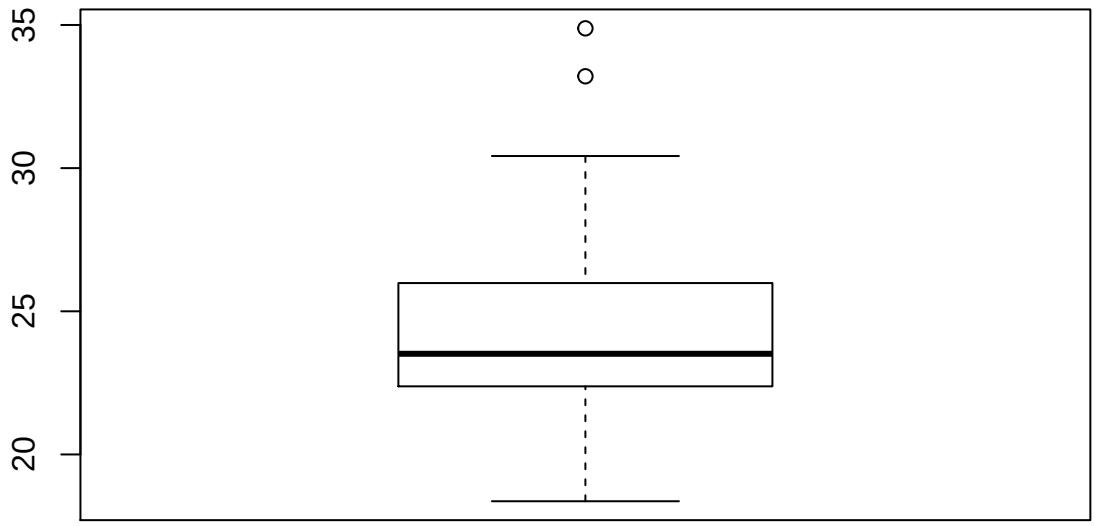



```
## a próxima implementação incorpora algumas opções específicas e deixa o
## número de categorias para o R especificar
hist(df$imc, xlab = "IMC", ylab = "densidade de frequência", main = "Histograma do IMC",
     col = "bisque", freq = FALSE)
```

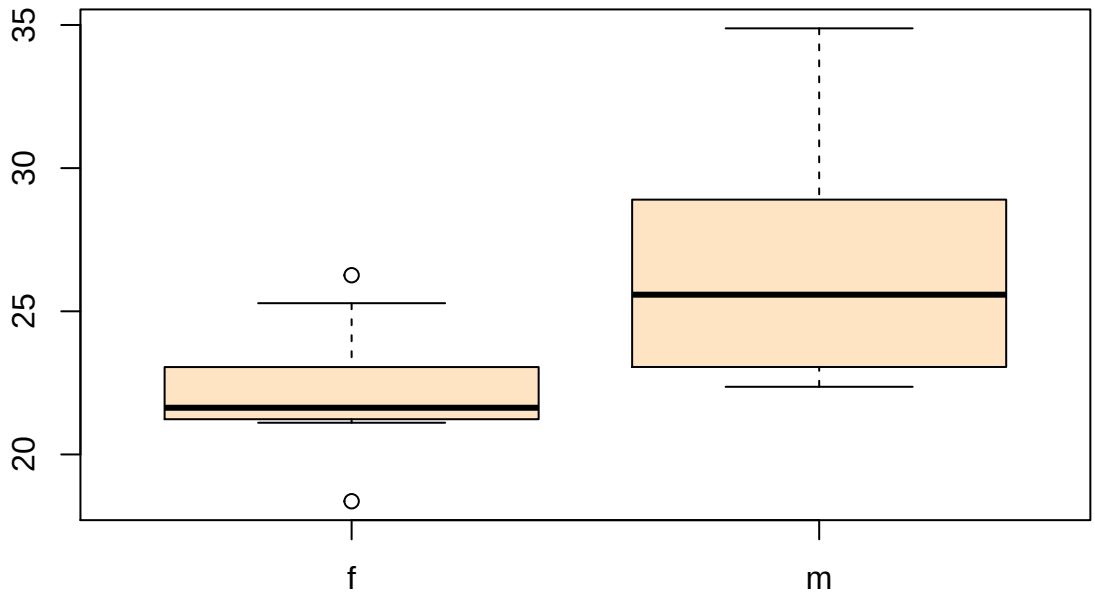


Uma outra opção é o **boxplot**, que mostra o máximo, mínimo, a mediana e os quartis 25% e 75%. Pode ser um gráfico incondicional ou condicional.

```
boxplot(df$imc)
```




```r
boxplot(df$imc ~ df$sex, col = "bisque")
```



Em muitos casos, há mais de uma informação indicada por resposta (ex. religião, hobbies, livros). Como tratar essa situação?

Há várias formas. Uma delas, a mais simples é organizar a informação em uma lista que registra a ocorrência de cada caso. Assim pode-se saber, pelo menos, o total de pessoas que informou uma dada possibilidade.

Para fazermos essa conversão, usaremos uma *função* que converte um vetor com as respostas separadas por um dado separador, em um vetor com as respostas já separadas.

```
abrestring <- function(mvec, sep) {
  n <- length(mvec)
  nvec <- list()
  for (i in 1:n) {
    nvec <- list(nvec, strsplit(as.character(mvec[i]), sep))
  }
  nvec <- unlist(nvec)
  return(nvec)
}
```

Usaremos essa função a seguir para identificar as respostas associadas à informação de hobbies.

```
relhobbies <- abrestring(df$ho1, ", ")
table(relhobbies)
```

```
## relhobbies
##                                  Academia
##                                         6
## Acessar internet para buscar conhecimento
##                                        15
```

```
##                                 Corrida
##                                       3
##                                Leitura
##                                       9
##                                 Outras
##                                       9
##                             Ouvir música
##                                      10
##                          Reuniões sociais
##                                       6
##                             Tocar musica
##                                       4
```

Para religião usaremos:

```
totrelig <- as.factor(abrestring(df$rel1, ", "))
levels(totrelig) <- c("at", "ca", "es", "ev", "ad", "ou", "pr")
table(totrelig)
```

```
## totrelig
## at ca es ev ad ou pr
##  1 11  4  4  3  1  1
```

No caso dos títulos dos livros pode-se utilizar:

```
livvec <- as.factor(abrestring(df$liv, ";"))
levels(livvec)
```

```
##  [1] " Cinquenta Tons de Cinza mais Escuro"
##  [2] " Diablo III"
##  [3] "100 anos de solidão"
##  [4] "A cabana"
##  [5] "A Cabana"
##  [6] "A Cilada"
##  [7] "A Cómedia Trágica ou a Tragédia Cômica de Mr. Punch"
##  [8] "A Travessia"
##  [9] "ABC da adubação"
## [10] "As Esganadas"
## [11] "Boas práticas"
## [12] "Cana de açúcar"
## [13] "Cinquenta Tons de Cinza"
## [14] "Cinquenta Tons de Liberdade"
## [15] "Como conviver com os Outros"
## [16] "Conhecer Jesus é Tudo"
## [17] "Educação pelo trabalho"
## [18] "Einstein por Ele Mesmo"
## [19] "Equador"
## [20] "Filosofia Sentimental, ensaios de lucidez"
## [21] "Fortaleza Digital"
## [22] "Lugar Nenhum"
## [23] "Mandela"
## [24] "Manutenção Mecânica"
## [25] "Meninas Normais Vão ao Shopping: Meninas Iradas Vão à Bolsa"
## [26] "O Contador de Lágrimas"
## [27] "O livro de Ouro da Mitologia"
## [28] "O Que Steve Jobs Faria"
```

```
## [29] "O Último Reino"
## [30] "O Vendedor de Sonhos, a revolução dos anônimos"
## [31] "Pai Rico Pai Pobre"
## [32] "Sementeira de Luz"
## [33] "Senhores da Escuridão"
## [34] "Sobreviver, crescer e perpetuar"
## [35] "Sobreviver, Crescer e Perpetuar vol.1"
## [36] "The Diary of a Wimpy Kid (vol 2,3,4)"
## [37] "Viva para Contar"
```

Observa-se que há muitas correções a fazer (espaços, títulos mal padronizados, letras com caixa diferente). A correção pode se processar pela substituição dos valores originais por valores padronizados como operacionalizado a seguir:

```
levels(livvec)[c(1, 2, 4, 8, 34, 35)] <-
c("Cinquenta Tons de Cinza mais Escuro",
  "Diablo III", "A Cabana", "A Cabana", "Sobreviver, Crescer e Perpetuar",
  "Sobreviver, Crescer e Perpetuar")
```

Verificando se está tudo correto:

```
levels(livvec)
```

```
##  [1] "Cinquenta Tons de Cinza mais Escuro"
##  [2] "Diablo III"
##  [3] "100 anos de solidão"
##  [4] "A Cabana"
##  [5] "A Cilada"
##  [6] "A Cómedia Trágica ou a Tragédia Cômica de Mr. Punch"
##  [7] "ABC da adubação"
##  [8] "As Esganadas"
##  [9] "Boas práticas"
## [10] "Cana de açúcar"
## [11] "Cinquenta Tons de Cinza"
## [12] "Cinquenta Tons de Liberdade"
## [13] "Como conviver com os Outros"
## [14] "Conhecer Jesus é Tudo"
## [15] "Educação pelo trabalho"
## [16] "Einstein por Ele Mesmo"
## [17] "Equador"
## [18] "Filosofia Sentimental, ensaios de lucidez"
## [19] "Fortaleza Digital"
## [20] "Lugar Nenhum"
## [21] "Mandela"
## [22] "Manutenção Mecânica"
## [23] "Meninas Normais Vão ao Shopping: Meninas Iradas Vão à Bolsa"
## [24] "O Contador de Lágrimas"
## [25] "O livro de Ouro da Mitologia"
## [26] "O Que Steve Jobs Faria"
## [27] "O Último Reino"
## [28] "O Vendedor de Sonhos, a revolução dos anônimos"
## [29] "Pai Rico Pai Pobre"
## [30] "Sementeira de Luz"
## [31] "Senhores da Escuridão"
## [32] "Sobreviver, Crescer e Perpetuar"
## [33] "The Diary of a Wimpy Kid (vol 2,3,4)"
```

```
## [34] "Viva para Contar"
```

Com a reestruturação, pode-se agora mostrar as frequências absolutas, já ordenadas em ordem decrescente:

```
sort(table(livvec), decreasing = TRUE)
```

```
## livvec
##                                                    A Cabana
##                                                           3
##                              Sobreviver, Crescer e Perpetuar
##                                                           2
##                         Cinquenta Tons de Cinza mais Escuro
##                                                           1
##                                                  Diablo III
##                                                           1
##                                           100 anos de solidão
##                                                           1
##                                                  A Cilada
##                                                           1
##         A Cómedia Trágica ou a Tragédia Cômica de Mr. Punch
##                                                           1
##                                               ABC da adubação
##                                                           1
##                                                 As Esganadas
##                                                           1
##                                                Boas práticas
##                                                           1
##                                               Cana de açúcar
##                                                           1
##                                       Cinquenta Tons de Cinza
##                                                           1
##                                    Cinquenta Tons de Liberdade
##                                                           1
##                                    Como conviver com os Outros
##                                                           1
##                                          Conhecer Jesus é Tudo
##                                                           1
##                                       Educação pelo trabalho
##                                                           1
##                                       Einstein por Ele Mesmo
##                                                           1
##                                                    Equador
##                                                           1
##                     Filosofia Sentimental, ensaios de lucidez
##                                                           1
##                                           Fortaleza Digital
##                                                           1
##                                                Lugar Nenhum
##                                                           1
##                                                    Mandela
##                                                           1
##                                          Manutenção Mecânica
##                                                           1
## Meninas Normais Vão ao Shopping: Meninas Iradas Vão à Bolsa
##                                                           1
```

```
##                                     O Contador de Lágrimas
##                                                          1
##                                O livro de Ouro da Mitologia
##                                                          1
##                                     O Que Steve Jobs Faria
##                                                          1
##                                               O Último Reino
##                                                          1
##              O Vendedor de Sonhos, a revolução dos anônimos
##                                                          1
##                                           Pai Rico Pai Pobre
##                                                          1
##                                            Sementeira de Luz
##                                                          1
##                                         Senhores da Escuridão
##                                                          1
##                         The Diary of a Wimpy Kid (vol 2,3,4)
##                                                          1
##                                           Viva para Contar
##                                                          1
```

Concluindo

Salvando o *data.frame* modificado em arquivo no seu computador

Para salvar o *data.frame* **limpo**, a fim de utilizá-lo em uma futuras análises, sem a necessidade de realizar a preparação dos dados, deve-se gravá-lo em um arquivo externo ao R.

Neste exemplo, o arquivo a ser salvo terá o nome **ODB2018.csv** e será utilizado o comando **write.table**. O comando *write.table(x, file=...)* grava o argumento **x** depois de convertê-lo em um *data frame*, se o argumento já não estiver no formato de uma matriz, para um arquivo ou conexão especificado em *file=....*:

```
diretorio <-"../dados/ODB2018.csv" ## Em Windows
#diretorio = "~/Downloads/AED/dados/ODB2018.csv" ## Em MacOsX
# Salvando...
write.table(df, file= diretorio, sep = ";", dec = ",",row.names = FALSE)
```

O *data.frame* alunos será salvo no arquivo "ODB2018.csv", com as opções corretas que: - definem um arquivo **csv** no formato BR desejado; - **";"** separando os valores; - **","** separando decimais; - os nomes das variáveis compactadas na primeira linha do arquivo; e - sem nomes identificando linhas (por *default*, a opção *row.names=NA*).

Referências

Becker, R.A, Chambers, J.M. e Wilks, A.R. (1988), The New S Language: A Programming Environment for Data Analysis and Graphics, Wadsworth & Brooks/Cole.

Becker, R.A.e Chambers, J.M. (1984). S: An Interactive Environment for Data Analysis and Graphics. Wadsworth & Brooks/Cole.

Behrens, John T. (1997), Principles and Procedures of Exploratory Data Analysis, Psychological Methods, Vol. 2, No. 2, pp.131-16.

Cavique, Luís (2014),Big Data e Data Science, Boletim 51.11-14, Repositório Aberto, Universidade Aberta de Portugal.

Hellerstein, Joseph (2008), Quantitative Data Cleaning for Large Databases, EECS Computer Science Division, UC Berkeley.

Hornik, Kurt (2017), The R FAQ: Why is R named R?. [Online]. Disponível em:https://cran.r-project.org/doc/FAQ/R-FAQ.html#Why-is-R-named-R_003f, Acessado em: 09/10/2018.

Ihaka, Ross (1998), R: Past and Future History, Statistics Department, The University of Auckland, Auckland, New Zealand. [Online]. Disponível em:https://cran.r-project.org/doc/html/interface98-paper/paper.html, Acessado em: 09/10/2018.

Janssen, Dale e Janssen, Cory (2018), Proof of Concept (POC), Blog Techopedia - The IT Education Site. [Online]. Disponível em:https://www.techopedia.com/definition/4066/proof-of-concept-poc. Acessado em: 15/10/2018.

Judd, Charles, McCleland, Gary e Ryan, Carey S. (2017). Data Analysis: A Model Comparison Approach To Regression, ANOVA, and Beyond, Third Edition, Harcourt.

Koomey, Jonathan G. (2006), Best Practices for Understanding Quantitative Data, Research Paper, Visual Business Intelligence Newsletter. Disponível em:http://www.perceptualedge.com/articles/b-eye/quantitative_data.pdf. Acessado em: 09/10/2018.

Microsoft Research (2012), Data Cleaning. [Online]. Disponível em:https://www.microsoft.com/en-us/research/project/data-cleaning/?from=http%3A%2F%2Fresearch.microsoft.com%2Fen-us%2Fprojects%2Fdatacleaning%2F. Acessado em: 09/10/2018.

Muenchen, Robert A. (2017), The Popularity of Data Science Software. [Online]. DIsponível em:http://r4stats.com/articles/popularity/. Acessado em: 09/10/2018.

NewGenApps (2017), 6 Reasons: Why Choose R Programming for Data Science Projects? Blog Newgwnapps, Sep 18, 2017. [Online]. Disponível em:https://www.newgenapps.com/blog/6-reasons-why-choose-r-programming-for-data-scien
Acessado em: 15/10/2018.

Olavsrud, Thor (2018), Ciência de dados: tudo sobre o método que transforma dados em valor, Computerworld. [Online]. Disponível em:https://computerworld.com.br/2018/07/02/ciencia-de-dados-tudo-sobre-o-metodo-que-transforma-da
Acessado em: 13/10/2018.

O'Neil, Cathy e Schutt, Rachel (2013), Doing Data Science: Straight Talk from the Frontline, O'Reilly Media.

Pinheiro, José Maurício Santos (2010), Prova de Conceito (PoC) no Projeto de Redes de Computadores, Blog Desmonta & CIA. [Online]. DIsponível em:https://desmontacia.wordpress.com/2010/12/21/prova-de-conceito-poc-no-projeto-de-redes-de-computadores/. Acessado em: 15/10/2018.

Plakidasa, Konstantinos, Schallb, Daniel e Zdun, Uwe (2017), Evolution of the R software ecosystem: Metrics, relationships, and their impact on qualities, Journal of Systems and Software, Vol. 132, pp. 119-146.

Profap (2018), Data wrangling: por que o big data depende dessa metodologia? Blog Profap. [Online]. Disponível em:http://profap.com.br/data-wrangling-por-que-o-big-data-depende-dessa-metodologia/. Acessado em: 13/10/2018.

Robinson, David (2017) The Impressive Growth of R, Stack Ovwerflow, Octobe, 10, 2017. [Online]. Disponível em:https://stackoverflow.blog/2017/10/10/impressive-growth-r/. Acessado em: 09/10/2018.

Schutt, Rachel e O'Neil, Cathy (2014), Doing Data Science, O'Reilly Media.

Silveira, Debora Pricila (2016), O que é Data Science?, Oficinadanet, 20/07/2016. [Online]. Disponível em:https://www.oficinadanet.com.br/post/16919-o-que-e-data-science. Acessado em: 13/10/2018.

The SunTec India Blog, Clean Data in CRM: The Key to Generate Sales-Ready Leads and Boost Your Revenue Pool (2016). [Online]. Disponível em:https://www.suntecindia.com/blog/clean-data-in-crm-the-key-to-generate-sales-ready-leads-and-boost-your-revenue-pool/. Acessado em: 09/10/2018.

Thieme, Nick (2018), R Generation, Significance Magazine, Royal Ststistics Society, N. 14, August 2018. pp. 14-19.

Tukey, J. (1961) The Future of Data Analysis, Princeton University. [Online]. Disponível em: https://projecteuclid.org/download/pdf_1/euclid.aoms/1177704711. Acessado em: 09/10/2018.