WEB Scraping com R

Extrair, processar e produzir conhecimento de dados na WEB

Prof. Walmes Zeviani

Curso de Especialização em
Data Science & Big Data
Universidade Federal do Paraná

Data! Data! Data! *I can't make bricks without clay!*

Sir Arthur Conan Doyle

War is ninety percent ***information***.

Napoleon Bonaparte

Knowledge is power. *You can't begin a career, for that matter even a relationship, unless you know everything there is to know about it.*

Randeep Hooda

*The alchemists in their search for gold discovered many other things of greater **value**.*

Arthur Schopenhauer

*Without big **data analytics**, companies are blind and deaf, wandering out onto the Web like deer on a freeway.*

Geoffrey Moore

*The goal is to turn **data into information**, and **information into insight**.*

Carly Fiorina

Justificativa

- Dados públicos: conhecimento acionável.
  - Notícias: política, economia, etc.
  - Finanças: ações, cotação de moeda.
  - Opinião: projeção de vendas.
  - Avaliações de consumidor (customer review).
  - Condições/previsões climáticas.
  - Condições de tráfego.
  - Torneios de esporte e e-sports.

Justificativa

- Dados públicos: conhecimento acionável.
  - Notícias: política, economia, etc.
  - Finanças: ações, cotação de moeda.
  - Opinião: projeção de vendas.
  - Avaliações de consumidor (customer review).
  - Condições/previsões climáticas.
  - Condições de tráfego.
  - Torneios de esporte e e-sports.
- Dados gratuitamente disponíveis:
  - Na forma de WEB APIs: Twitter, LinkedIn, OMDB, apilist.fun, www.programmableweb.com.
  - Em sites governamentais: dados.gov.br, www.geoservicos.ibge.gov.br, www.ibge.gov.br, www.ipeadata.gov.br, www.tre-pr.jus.br, www.infraero.gov.br.
  - Em órgãos/instituições www.sine.com.br, www.ceasa.pr.gov.br, www.reclameaqui.com.br, www.atletismofap.org.
  - Em sites de empresas: comércio de imóveis/veículos, celulares, bolsa de valores.

O WS e a Engenharia de Caractetísticas

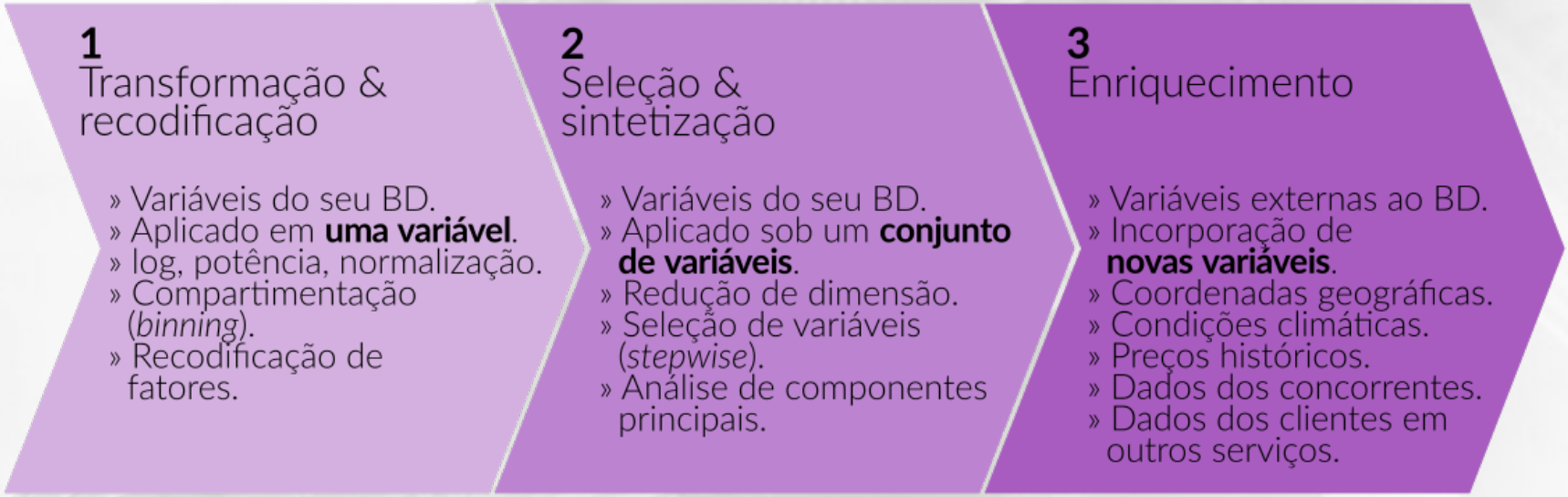


Figura 1: Diagrama de fluxo dos níveis de maturidade em engenharia de características para ciência de dados.

Enquete

- Quem sabe o que é e para que serve WS?
- Quem sabe fazer WS?
- Quem quer dominar WS?
- Quem conheçe HTML, CSS, JS e HTTP?

Enquete

- Quem sabe o que é e para que serve WS?
- Quem sabe fazer WS?
- Quem quer dominar WS?
- Quem conheçe HTML, CSS, JS e HTTP?

WS em 90 minutos…

Aconteceu o Big Bang… A vida surgiu de seres unicelulares… As civilizações de desenvolveram… Ciência… Gerras… Hoje temos a internet.

Enquete

- Quem sabe o que é e para que serve WS?
- Quem sabe fazer WS?
- Quem quer dominar WS?
- Quem conheçe HTML, CSS, JS e HTTP?

WS em 90 minutos...

Aconteceu o Big Bang... A vida surgiu de seres unicelulares... As civilizações de desenvolveram... Ciência... Gerras... Hoje temos a internet.

- CH mínima de curso introdutório: 20h.
- Valor de *workshop* de 8h: R$ 500.
- Domínio vem com a prática.
- Curso completo em `http://leg.ufpr.br/~walmes/ensino/web-scraping`.

Objetivo

- Consumo de dados da WEB é WEB Scraping (WS).
- O R possui muitas funcionalidades para WS.
- **Objetivos**
  - Apresentar o landscape de funcionalidades do R para WS.
  - Descrever a estrutura de documentos XML/HTML e JSON.
  - Introduzir a linguagem de consulta Xpath.
  - Fazer aplicações de WS com R.
  - Apontar os principais desafios em WS.

WEB API

- Disponibilização de dados via Web API é a forma padrão de
  - conferir acesso aos usuários.
  - transferir dados entre aplicações.
- API: *Application Program Interface* = Interface de Programação de Aplicativos.

WEB API

- Disponibilização de dados via Web API é a forma padrão de
  - conferir acesso aos usuários.
  - transferir dados entre aplicações.
- API: *Application Program Interface* = Interface de Programação de Aplicativos.

Uma web API é uma interface programática consistente de um ou mais endpoints publicamente expostos para um sistema definido de mensagens requisição-resposta, tipicamente expresso em JSON ou XML, que é acessado via internet – mais comumente por meio de um servidor baseado em HTTP
`https://pt.wikipedia.org/wiki/Web_API`.

Algumas WEB API

- The Open Movie Database.
  - A página documenta os parâmetros de consulta.
  - É necessário criar um token de acesso.
  - Titanic, Batman.
  - Temporada 1 de Game of Thrones.

Algumas WEB API

- The Open Movie Database.
  - A página documenta os parâmetros de consulta.
  - É necessário criar um token de acesso.
  - Titanic, Batman.
  - Temporada 1 de Game of Thrones.
- Earthquake Catalog API.
  - Dados sobre eventos sísmicos.
  - Não requer conta ou token.
  - Documentação da API.
  - Eventos sísmicos hoje (04/05/2018 das 00h às 07h).

Algumas WEB API

- The Open Movie Database.
  - A página documenta os parâmetros de consulta.
  - É necessário criar um token de acesso.
  - Titanic, Batman.
  - Temporada 1 de Game of Thrones.
- Earthquake Catalog API.
  - Dados sobre eventos sísmicos.
  - Não requer conta ou token.
  - Documentação da API.
  - Eventos sísmicos hoje (04/05/2018 das 00h às 07h).
- Open Weather Map API:
  - Usado por aplicações de smartphones e serviços baseados em previsões metereológicas.
  - Requer criar conta mas possui plano free e outros.
  - Documentação das várias APIs.
  - Requer ID da cidade para busca. Consulte a ID.
  - Condição climática de Curitiba agora.

Anatomia de documentos XML

```
<breakfast_menu>
  <food>
    <name>Belgian Waffles</name>
    <price>$5.95</price>
    <description>
      Two of our famous Belgian Waffles with plenty of real
      maple syrup
    </description>
    <calories>650</calories>
  </food>
  <food>
    <name>Berry-Berry Belgian Waffles</name>
    <price>$8.95</price>
    <description>
      Light Belgian waffles covered with an assortment of
      fresh berries and whipped cream
    </description>
    <calories>900</calories>
  </food>
</breakfast_menu>
```

https://www.w3schools.com/xml/simple.xml

Documentos XML

- XML: *eXtensible Markup Language*.
- Usado para representar dados em diversos formatos.
- Tabelas, planilhas, documentos de texto, imagens, mapas, desenhos vetoriais, webpages, redes sociais, estilos de formatação de referências bibliográficas.
- É tão genérico que pode representar qualquer tipo de estrutura de dados.
- Comum para dados de estrutura hierárquica e/ou com metadados.

Tipos de arquivos XML (dialetos)

- HTML (HiperText Markup Language): páginas de internet.
- KML (Keyhole Markup Language): informação geográfica tri-dimensional.
- CSL (Citation Style Language): referências bibliográficas.
- ODF (Open Document Format): documentos de texto, planilha e slides, etc.
- SVG (Scalable Vector Graphics): formato de imagens vetoriais.
- Epub: publicação/livro eletrônico.

Mais exemplos

- Arquivo XML: https://msdn.microsoft.com/en-us/library/ms762271(v=vs.85).aspx.
- Polígonos dos Estados BR: http://www.gmapas.com/poligonos-ibge.
- Estilo de referência ABNT: http://dl.dropbox.com/u/9905692/links%20do%20site/estilos%20abnt/ABNT_UFPR_2011-Mendeley.csl.
- Imagem vetorial: https://upload.wikimedia.org/wikipedia/en/2/22/Heckert_GNU_white.svg.

O essencial

- A unidade básica é o **elemento** ou nó.
- O elemento é começa e termina com a **tag nomeada**.
- O par de tags delimita o **conteúdo do elemento**.
- Elementos podem conter elementos = estrutura hierárquica.
- Um elemento pode ter atributos do tipo `campo = "valor"`.
- A estrutura se assemelha a uma **árvore**.
- Análogo às listas do R e dicionários do Python.
- Embora muito verboso, a taxa de compressão é boa.
- WEB APIs usam XML para expor dados.

Anatomia de documentos JSON


```
{
   "coord":{
      "lon":-49.27,
      "lat":-25.43
   },
   "main":{
      "temp":299.64,
      "pressure":1020,
      "humidity":39,
      "temp_min":299.15,
      "temp_max":300.15
   },
   "wind":{
      "speed":3.1,
      "deg":340
   },
   "clouds":{
      "all":40
   },
   "dt":1525453200,
   "id":6322752,
   "name":"Curitiba",
   "cod":200
}
```


Open Weather Map - Curitiba.

Documentos JSON

- JSON: *JavaScript Oriented Notation*.
- Originou de um ramo da sintaxe JavaScript.
- É simples, leve e não verboso.
- Empregado em: WEB API e SGBD NoSQL.
- Tipos primitivos de dados: lógico, númerico, string e nulo.
- Containers para coleção de dados: array e lista.
- Tem flexibilidade para representar estruturas complexas de dados.
- Não tem campos de atributos para metadados.

Documentos HTML

- HTML: *HyperText Markup Language*.
- É baseado em XML.
- Função: exibição de conteúdo na WEB.
- É uma linguagem de marcação.
- As marcações instruem o navegador sobre como exibir a página.
- O navegador mostra a renderização do HTML.
- Os navegadores têm recursos para inspeção do código fonte.
- Pressione F12 no seu navegador para iniciar a inspeção.
- Parte do sucesso em WS está na habilidade em inspecionar o código-fonte.

Até aqui...

- Dados na WEB estão em:
  - API: formatos XML e JSON.
  - Páginas: formato HTML.
- Estrutura de dados hierárquica.
- Consulta baseada na inspeção do código fonte.
- Importante: aspectos da requisição cliente-servidor.

Protocolos de comunicação

- Comunição na WEB usa protocolos nos bastidores baseados em
  - TCP: *transmission control protocol*.
  - IP: *internet protocol*.
  - TCP/IP: cuidam da transferência de dados entre computadores pela rede.
- Existem protocolos TCP/IP específicos.
  - HTTP: *hypertext transfer protocol*.
  - FTP: *file transfer protocol*.
  - POP: *post office protocol*.
  - SMTP: *simple mail transfer protocol* .
  - IMAP: *internet message access protocol*.
- Eles definem padrões de comunicação cliente-servidor sobre tarefas específicas.

HTTP essencial

- HTTP parece simples mas na realidade é muito flexível e amplamente utilizado.
- E capaz de transferir, reter ou enviar praticamente qualquer tipo de informação.
- A comunicação mais simples é o cliente fazer requisições (por URL) e o servidor atendê-las (enviando a página/arquivo).
- As ferramentas de desenvolvimento do navegador (tecla F12, aba *Network*) serão úteis para monitorar e detalhar os processos gerados pelas requisições.

Fluxo da comunicação

1. O nagegador é o cliente HTTP (faz requisições).
2. Uma requisição por URL solicita ao servidor de DNS (*domain name service*) quem é o IP que responde pelo domínio da URL (a URI).
3. Ao saber o IP, o cliente envia requisões por HTTP para o servidor.
4. O servidor responde as requições HTTP enviado o conteúdo solicitado (página, imagem, arquivo, etc).

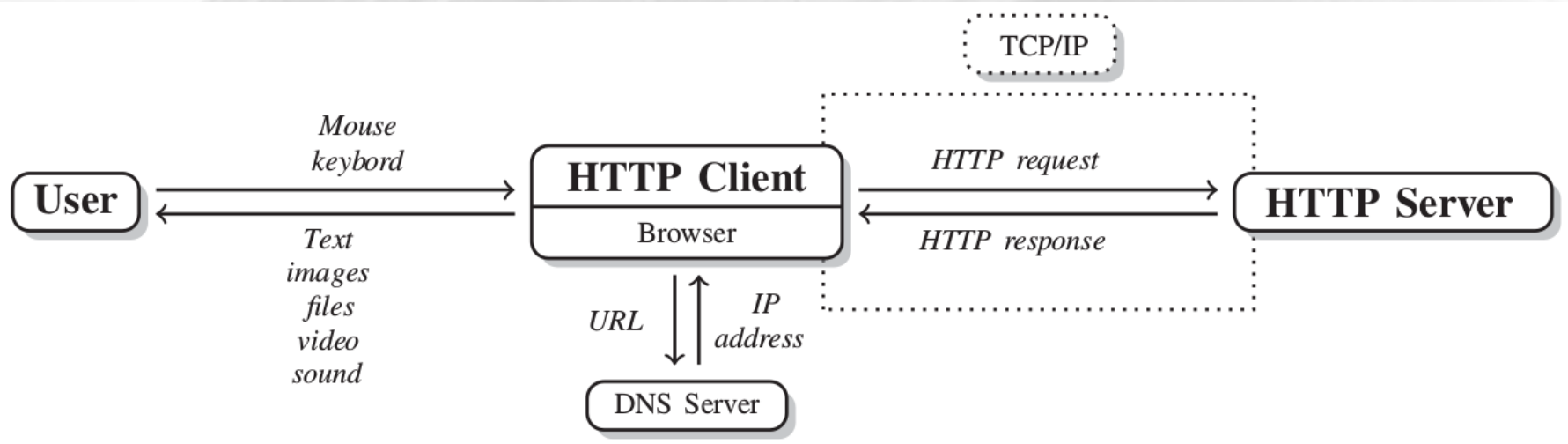


Figura 2: Ilustração da comunicação cliente-servidor por HTTP.

WS e protocolos

- Lei 80/20: conhecendo 20% de HTTP se faz 80% de WS.
- O HTTP é fundamental quando:
  - Impõe limite no número de requisições.
  - Reduzir a latência (tempo de comunicação/transferência).
  - Site usa cookies.
  - É necessário fazer autenticação.
  - É necessário usar *proxy*.
  - O site tem CAPTCHA.
- Mais detalhes disso em Infraestrutura Computacional.

Landscape de funcionades do R para WS

- Linguagens de baixo nível (em C).
  - `libxml2`: parse de XML/HTML.
  - `libcurl`: requisições HTTP, etc.
- Interfaces primárias no R.
  - `XML`: processamento de arquivos XML/HTML. Interface para `libxml2`.
  - `RCurl`: interface para `libcurl`.
  - `jsonlite`: parse de documentos JSON (funções internas em C).
  - `RSelenium`: interface R para usar o Selenium Web Driver.
- Interfaces mais consistentes em R (Hadley Wickham).
  - `xml2`: interface mais leve e consistente para a `libxml2`.
  - `httr`: interface mais leve e consistente para a `libcurl`.
  - `rvest`: escrita sobre a `httr` e `xml2` para facilitar tarefas de WS.

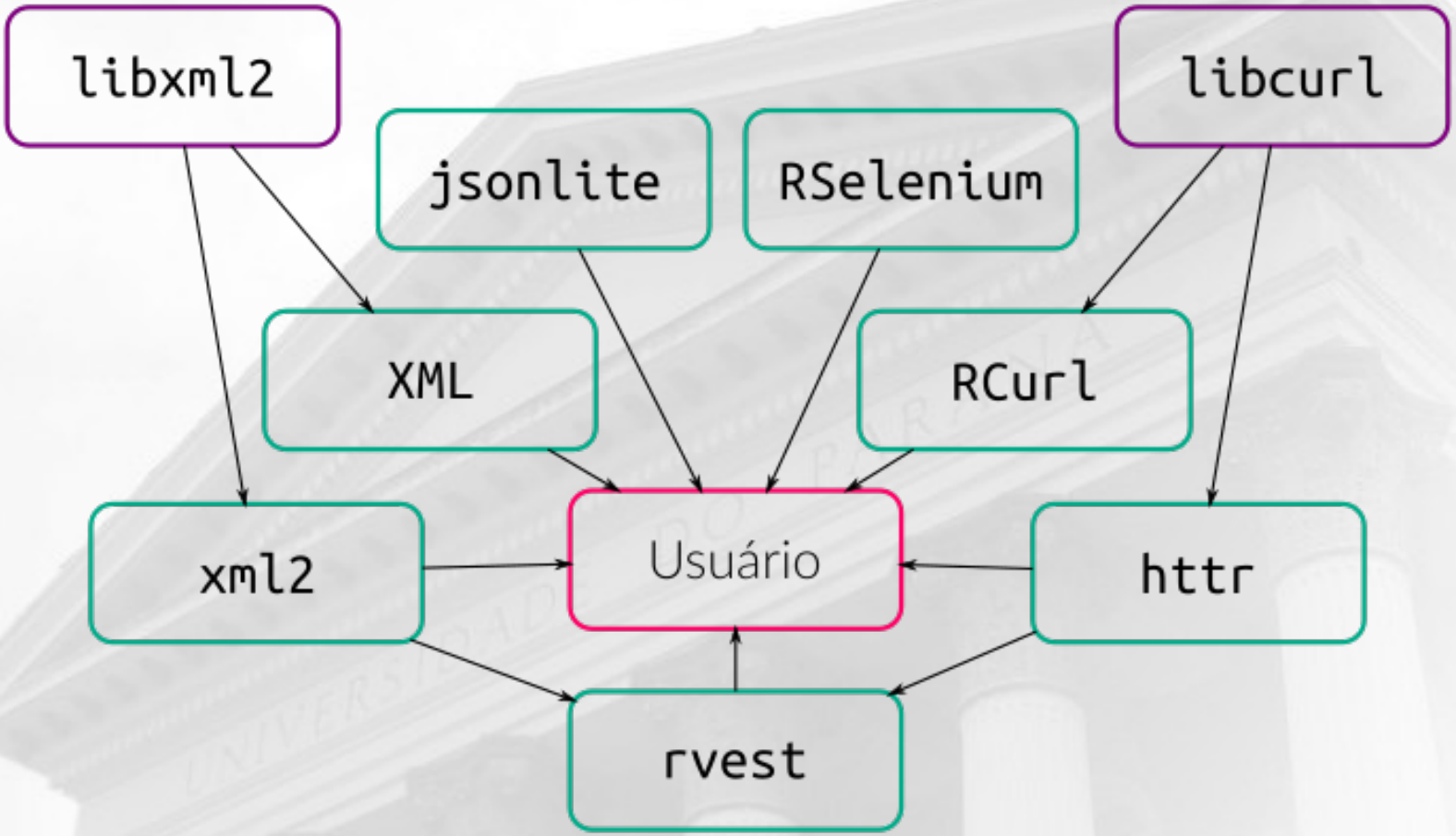


Figura 3: Landscape de funcionalidade no R para WEB Scraping.

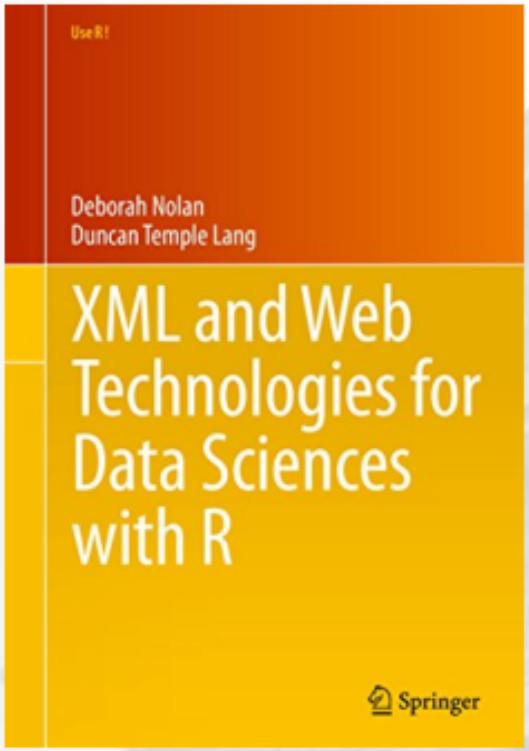
Instalação dos pacotes

```
# Coloque os pacotes desejados em um vetor e execute.
pkgs <- c("XML", "RCurl", "jsonlite")
install.packages(pkgs = pkgs,
                 dependencies = TRUE,
                 repos = "http://cran-r.c3sl.ufpr.br/")


# Carregar os pacotes (um por um).
library(XML)
library(RCurl)
library(jsonlite)


# NINJA: Carregando todos os pacotes do vetor em série.
sapply(pkgs, FUN = library, character.only = TRUE)
```

Referências



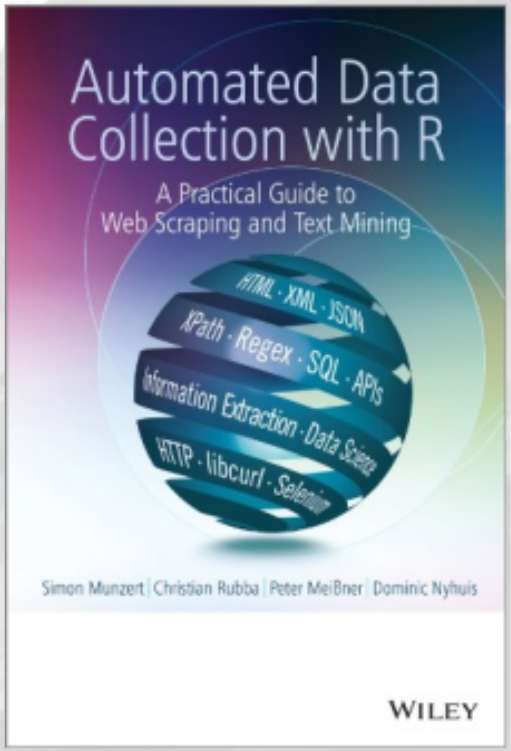


Figura 4: Principais referências sobre WS.

Até aqui...

- Documentos envolvidos em exposição de dados na WEB.
- O mínimo sobre comunição cliente-servidor.
- Landscape de pacotes no R para WS.
- Principais referências para WS.

Xpath

- É uma linguagem de consulta (aponta/extrai) para documentos XML/HTML.
- É um padrão do *World Wide Web Consortium* (W3C).
- Especifica *caminhos* pelos nós e ramos da árvore (DOM)
- Uma expressão Xpath retorna o conteúdo que bate com o caminho descrito.
- Em alguns aspectos, se assemelha a Expressões Regulares.
- Folhas de cola (cheat sheet)
  - `http://ricostacruz.com/cheatsheets/xpath.html`.
  - `http://scraping.pro/5-best-xpath-cheat-sheets-and-quick-references/`.
  - `http://xpath.alephzarro.com/content/cheatsheet.html`.
  - `http://www.mulberrytech.com/quickref/XSLT_1quickref-v2.pdf`.

Predicados Xpath

- Predicados permitem criar filtros baseados em propriedades dos elementos.
- Predicados são descritos nos colchetes após o nome do elemento.
  - `//div[<predicado>]`
- Os predicados podem usar: índices, testes no conteúdo e relacionamento entre elementos (parentesco).

Selenium Webdriver

- Selenium Webdriver (SWD).
- Envia instruções para o navegador e retém os resultados.
- Basicamente é encontrar e manipular elementos na página.
- Instruções podem ser:
  - Um clique de mouse sobre um botão ou hyperlink.
  - Rolar a página.
  - Pressionar F5 (atualizar página).
  - Prencher login e senha (autenticar).
  - Preencher e submeter um formulário.
  - Fazer download de um arquivo.
  - Selecionar e recortar um texto.
- O SWD abre o navegador e guia/controla ele como se fosse um usuário.
- Projeto: `http://www.seleniumhq.org/projects/webdriver/`.
- Documentação: `http://www.seleniumhq.org/docs/`.
- Tutorial: `http://toolsqa.com/selenium-tutorial/`.

Aplicações

- O básico de manipulação em XML.
- WS em página HTML.
- Parse de JSON.
- Usando o RSelenium.

Conhecimento útil para ES

- Execução agendade de tarefas (`crontab`).
- Conhecimento em Expressões Regulares (REGEX).
- Conhecimento em representação de datas.
- Conhecimento de funcionamento de formulários HTML.
- JavaScript.

Considerações finais

- 90 minutos é pouco para falar de WS.
- Interessados devem conseguir acesso a literatura recomendada.
- Visitem: `http://leg.ufpr.br/~walmes/ensino/web-scraping`.
- Reunir interessados e fazer um Workshop/oficina, etc.

Considerações finais

- 90 minutos é pouco para falar de WS.
- Interessados devem conseguir acesso a literatura recomendada.
- Visitem: `http://leg.ufpr.br/~walmes/ensino/web-scraping`.
- Reunir interessados e fazer um Workshop/oficina, etc.

Obrigado