
Text Mining

Fabrício J. Barth

Novembro de 2016

Objetivo

O objetivo desta aula é apresentar a área de Text Mining. Serão apresentados e discutidos conceitos e aplicações, além de técnicas e ferramentas para a implementação de soluções de Text Mining.

Sumário

- Revisar os conceitos: aprendizagem de máquina, *knowledge discovery in databases* e a linguagem de programação R.
- Pré-processamento em Text Mining: *Bag of words*, n-grams.
- Clustering e Classificação com dados não estruturados:
 - ★ Análise de mensagens dos twitter usando algoritmos de agrupamento.
 - ★ Desenvolvimento de algoritmos anti-spam.
 - ★ Desenvolvimento de Sistemas de Pergunta & Resposta.

-
- Reconhecimento de Entidades Nomeadas.
 - Considerações finais.
 - Referências.

Conceitos

Knowledge Discovery in Databases (KDD)

KDD é o processo não trivial de identificação de padrões em dados que sejam válidos, novos, potencialmente úteis e compreensíveis [Fayyad, 1996].

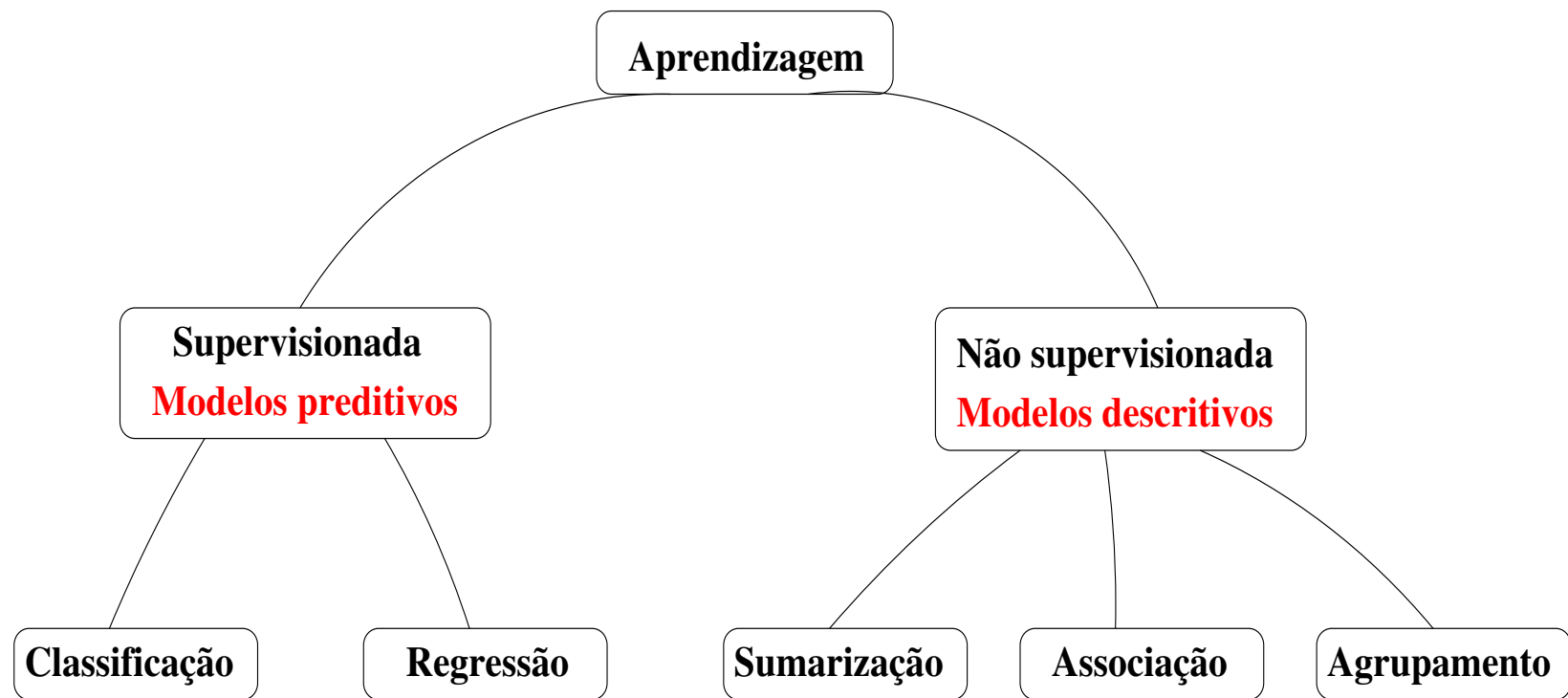
Descobrir conhecimento útil:

- Sintetizar informação:
 - ★ a partir de logs de servidores web, identificar qual é o caminho mais frequente de navegação dos usuários no site.
 - ★ a partir de notícias publicadas em veículos web, sumarizar os principais eventos do dia.
- Prescrever ações:
 - ★ a partir do histórico de candidaturas em vagas de um candidato, recomendar novas vagas para o mesmo.
 - ★ a partir de conteúdo previamente moderado, construir uma aplicação capaz de moderar conteúdo automaticamente.

Processo de KDD

1. Qual é a pergunta?
2. Aquisição e pré-processamento dos dados.
3. Análise Exploratória.
4. Modelagem: construção do modelo descritivo ou preditivo.
5. Avaliação do modelo.
6. Entrega: relatórios estáticos, dinâmicos, sistemas ou funcionalidade de sistemas.

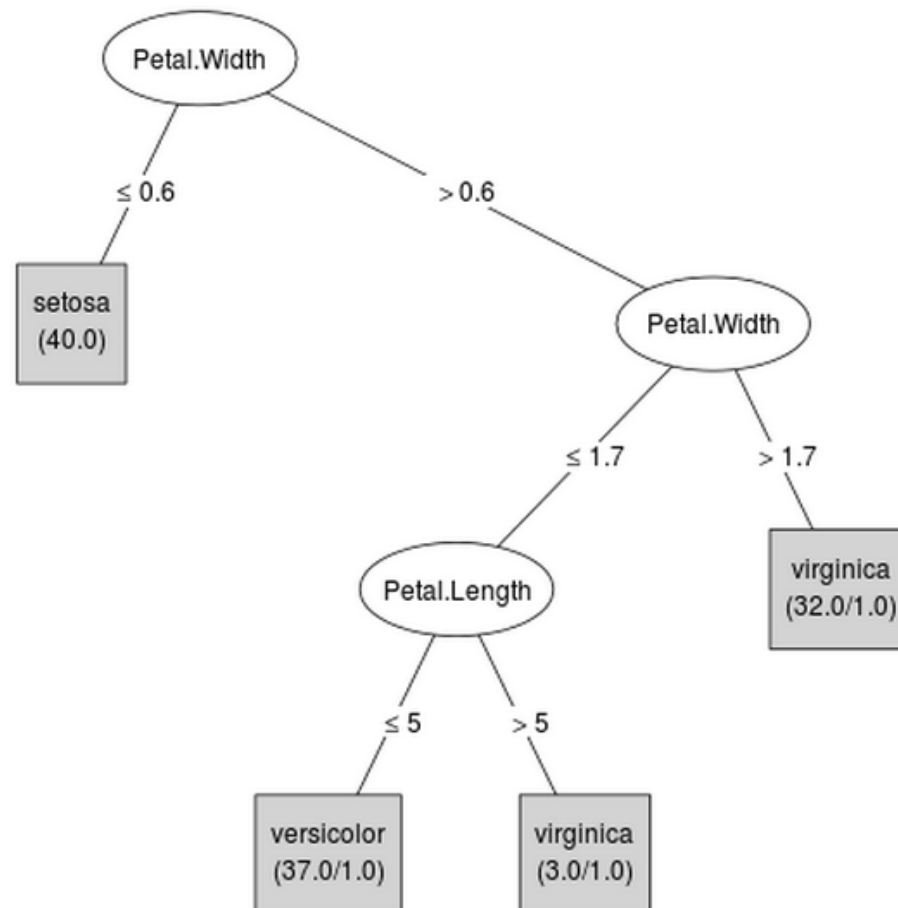
Aprendizagem de máquina



Exemplo de dataset com classe

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.10	3.50	1.40	0.20	setosa
2	4.90	3.00	1.40	0.20	setosa
3	4.70	3.20	1.30	0.20	setosa
4	4.60	3.10	1.50	0.20	setosa
5	5.00	3.60	1.40	0.20	setosa
52	6.40	3.20	4.50	1.50	versicolor
53	6.90	3.10	4.90	1.50	versicolor
54	5.50	2.30	4.00	1.30	versicolor
55	6.50	2.80	4.60	1.50	versicolor
56	5.70	2.80	4.50	1.30	versicolor
104	6.30	2.90	5.60	1.80	virginica
105	6.50	3.00	5.80	2.20	virginica
110	7.20	3.60	6.10	2.50	virginica
115	5.80	2.80	5.10	2.40	virginica

Exemplo de modelo preditivo

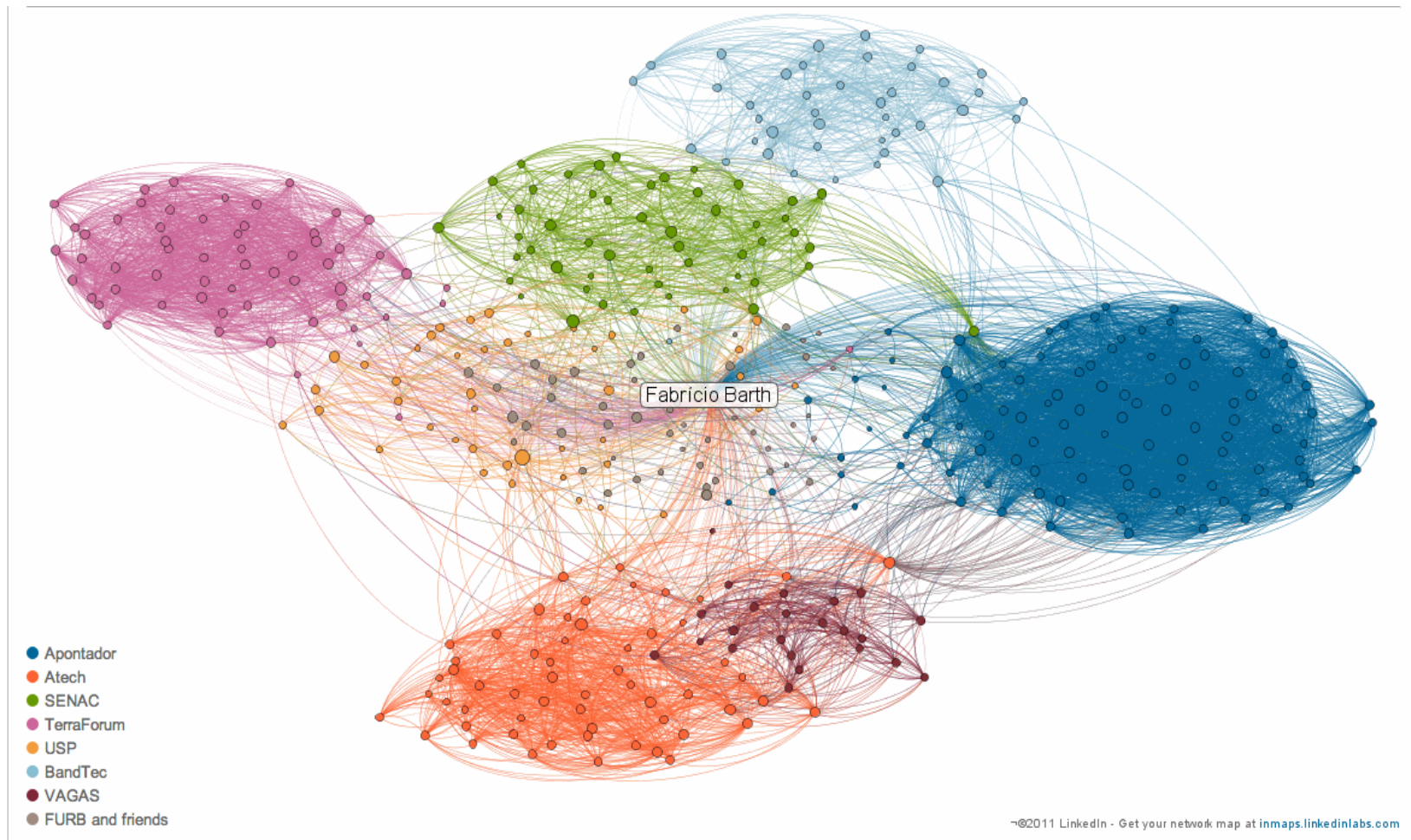


Exemplos de aprendizagem não supervisionada

Table 1: Exemplo de tabela com conexões entre usuários

usuário	$user_1$	$user_2$	$user_3$	\dots	$user_n$
$user_1$	1	1	0	\dots	1
$user_2$	1	1	0	\dots	0
$user_3$	1	0	1	\dots	0
$user_4$	0	1	0	\dots	0
\dots	\dots	\dots	\dots	\dots	\dots
$user_n$	1	1	0	\dots	1

Exemplo de identificação de grupos em redes sociais



Ferramentas que suportam o processo de KDD

O processo de KDD (pode/deve) ser suportado por ferramentas computacionais, tais como:

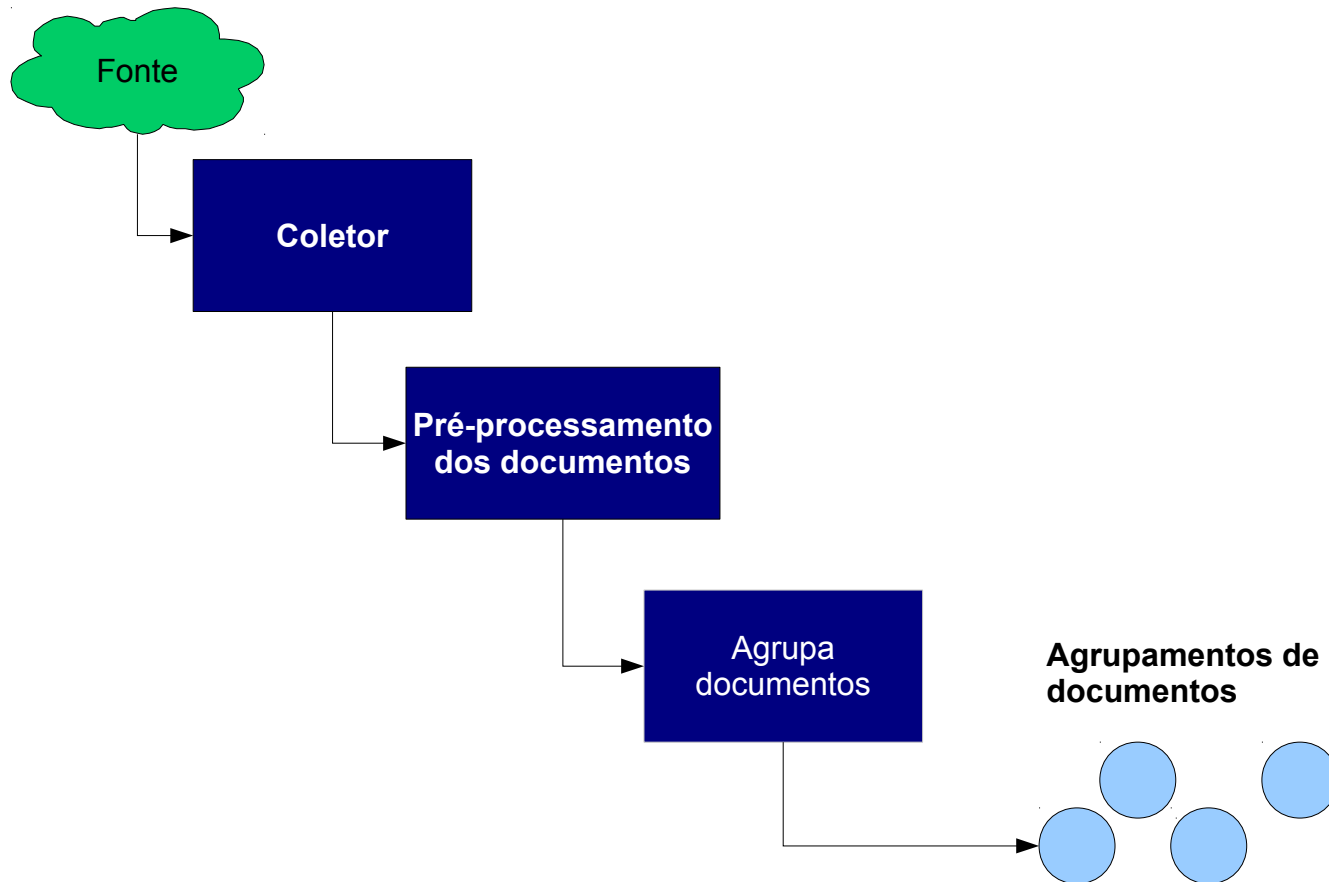
- R
- SPSS
- RapidMiner
- Weka
- Tableau
- Python, Julia, Octave, Matlab.

Projeto R

- <http://www.r-project.org/>
- R Studio - <http://www.rstudio.com/>
- É free
- É a linguagem de programação mais popular para análise de dados
- Script é melhor que clicar e arrastar:
 - ★ É mais fácil de comunicar.
 - ★ Reproduzível.
 - ★ É necessário pensar mais sobre o problema.
- Existe uma quantia grande de pacotes disponíveis

Análise de mensagens do twitter usando algoritmos de agrupamento

Componentes para uma solução...



Coletando dados do twitter com o R

```
library(twitterR)
cred <- OAuthFactory$new(
  consumerKey="XXXX",
  consumerSecret="XXXX",
  requestURL="https://api.twitter.com/oauth/request_token",
  accessURL="https://api.twitter.com/oauth/access_token",
  authURL="http://api.twitter.com/oauth/authorize")

cred$handshake()
registerTwitterOAuth(cred)

dados <- searchTwitter('economist_brasil', n=250)
df <- twListToDF(dados)
save(df, file=" ../data/20140424_economist_brasil.rda")
```

Pré-processamento dos dados

Acessar e fazer o download do projeto

<https://github.com/fbarth/mlr>

Formato de um documento

... Esta disciplina tem como objetivo apresentar os principais conceitos da área de Inteligência Artificial, caracterizar as principais técnicas e métodos, e implementar alguns problemas clássicos desta área sob um ponto de vista introdutório.

A estratégia de trabalho, o conteúdo ministrado e a forma dependerão dos projetos selecionados pelos alunos.

Inicialmente, os alunos deverão trazer os seus Projetos de Conclusão de Curso, identificar intersecções entre o projeto e a disciplina, e propor atividades para a disciplina. ...

Conjunto de Exemplos - Atributo/Valor

Doc.	apresent	form	tecnic	caracteriz	...
d_1	0.33	0.33	0.33	0.33	...
d_2	0	0.5	0.2	0.33	...
d_3	1	0.6	0	0	...
d_4	0.4	0.3	0.33	0.4	...
d_5	1	0.4	0.1	0.1	...
d_n

Atributo/Valor usando vetores

Como representar os documentos?

$$\vec{d}_i = (p_{i1}, p_{i2}, \dots, p_{in}) \quad (1)$$

- Os atributos são as palavras que aparecem nos documentos.
- As palavras do texto precisam ser normalizadas: caixa baixa, remover acentuação, remover stop-words, aplicar algoritmos de steaming.

Remover stop-words

- Em todos os idiomas existem átomos (palavras) que não significam muito. **Stop-words**

Esta disciplina **tem como** objetivo apresentar **os** principais conceitos **da** área **de** Inteligência Artificial, caracterizar **as** principais técnicas **e** métodos, **e** implementar alguns problemas clássicos **desta** área **sob um** ponto **de** vista introdutório.

...

Algoritmos de steaming

- Algumas palavras podem aparecer no texto de diversas maneiras: **técnica**, **técnicas**, **implementar**, **implementação**...
- **Stemming** - encontrar o radical da palavra e usar apenas o radical.

Radicalizador para o português:

- Regra de reducao de plurais (Regras N)
- Regra de reducao de femininos (Regras G)
- Regras de reducao de aumentativos e diminutivos (Regras T)
- Regras de reducao de grau (Regras S)
- Outras regras (Regras O)
- Regras para formas verbais (Regras V)

A sequência para as formas verbais reduz-se à aplicação da regra para redução ao infinito.

Para os nomes (substantivos, adjetivos e advérbios) aplica-se a seguinte sequência: $N \rightarrow G \rightarrow T \rightarrow S \rightarrow O$

Atributo/Valor usando vetores

- Já conhecemos os atributos.
- E os valores?
 - ★ **Booleana** - se a palavra aparece ou não no documento (1 ou 0)
 - ★ **Por frequência do termo** - a frequência com que a palavra aparece no documento (normalizada ou não)
 - ★ **Ponderação tf-idf** - o peso é proporcional ao número de ocorrências do termo no documento e inversamente proporcional ao número de documentos onde o termo aparece.

Por frequência do termo

(apresent,0.33) (form,0.33) (tecnic,0.33) (caracteriz,0.33)
(projet,1.0) (introdutori,0.33) (objet,0.33) (inteligente,0.33)
(conclusa,0.33) (selecion,0.33) (intersecco,0.33) (classic,0.33)
(identific,0.33) (conceit,0.33) (trabalh,0.33) (disciplin,1.0)
(traz,0.33)

Conjunto de Exemplos - Atributo/Valor

Doc.	apresent	form	tecnic	caracteriz	...
d_1	0.33	0.33	0.33	0.33	...
d_2	0	0.5	0.2	0.33	...
d_3	1	0.6	0	0	...
d_4	0.4	0.3	0.33	0.4	...
d_5	1	0.4	0.1	0.1	...
d_n

Conjunto de Exemplos com **Classe** - Atributo/Valor

Doc.	apresent	form	tecnic	caracteriz	...	Classe
d_1	0.33	0.33	0.33	0.33	...	AG
d_2	0	0.5	0.2	0.33	...	RC
d_3	1	0.6	0	0	...	AM
d_4	0.4	0.3	0.33	0.4	...	AG
d_5	1	0.4	0.1	0.1	...	AM
d_n

Representação Booleana

- O peso é calculado levando-se em consideração a existência ou não do termo no documento.

$$p_{i,j} = \begin{cases} 1 & \text{if } f_{i,j} > 0 \\ 0 & \text{if } f_{i,j} = 0 \end{cases} \quad (2)$$

Representação por Frequência do Termo

- O peso é calculado levando-se em consideração a frequência com que o termo acontece no documento. Esta frequência pode ser ponderada ou não.

$$p_{i,j} = f_{i,j} \quad (3)$$

$$p_{i,j} = \frac{f_{i,j}}{\max_z f_{z,j}} \quad (4)$$

Representação por Ponderação tf-idf

- O peso é proporcional ao número de ocorrências do termo no documento e inversamente proporcional ao número de documentos onde o termo aparece.

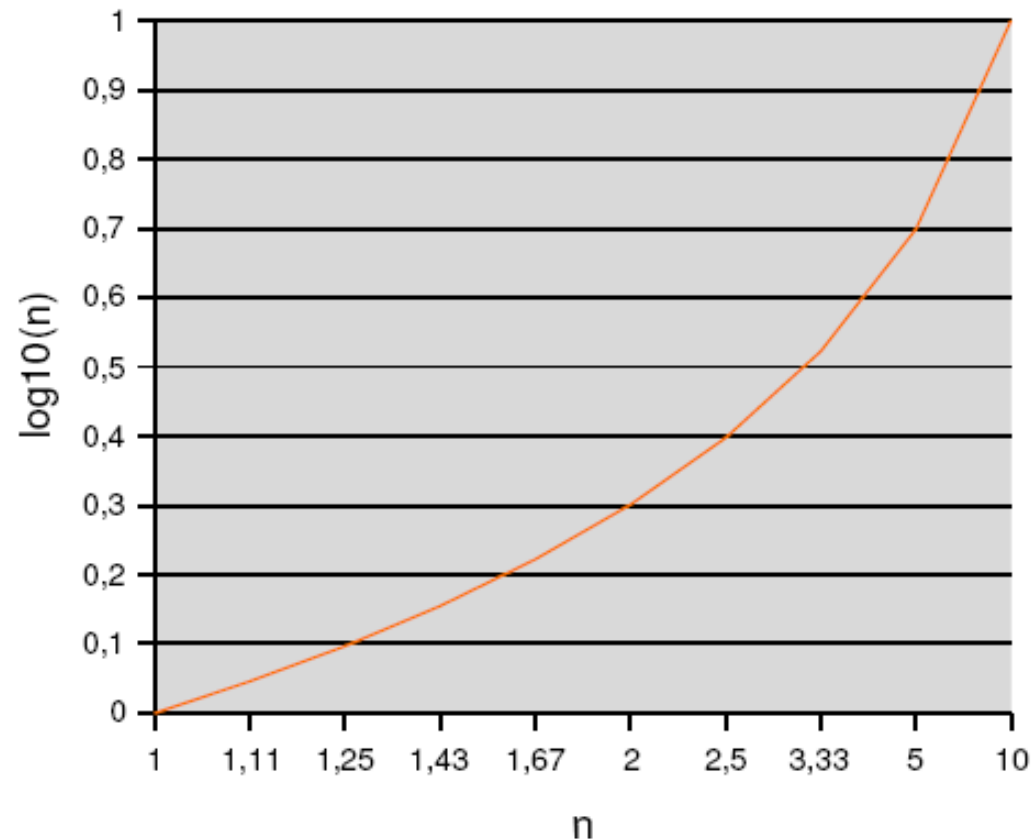
$$p_{i,j} = \frac{f_{i,j}}{\max_z f_{z,j}} \times \log \frac{N}{n_i} \quad (5)$$

onde,

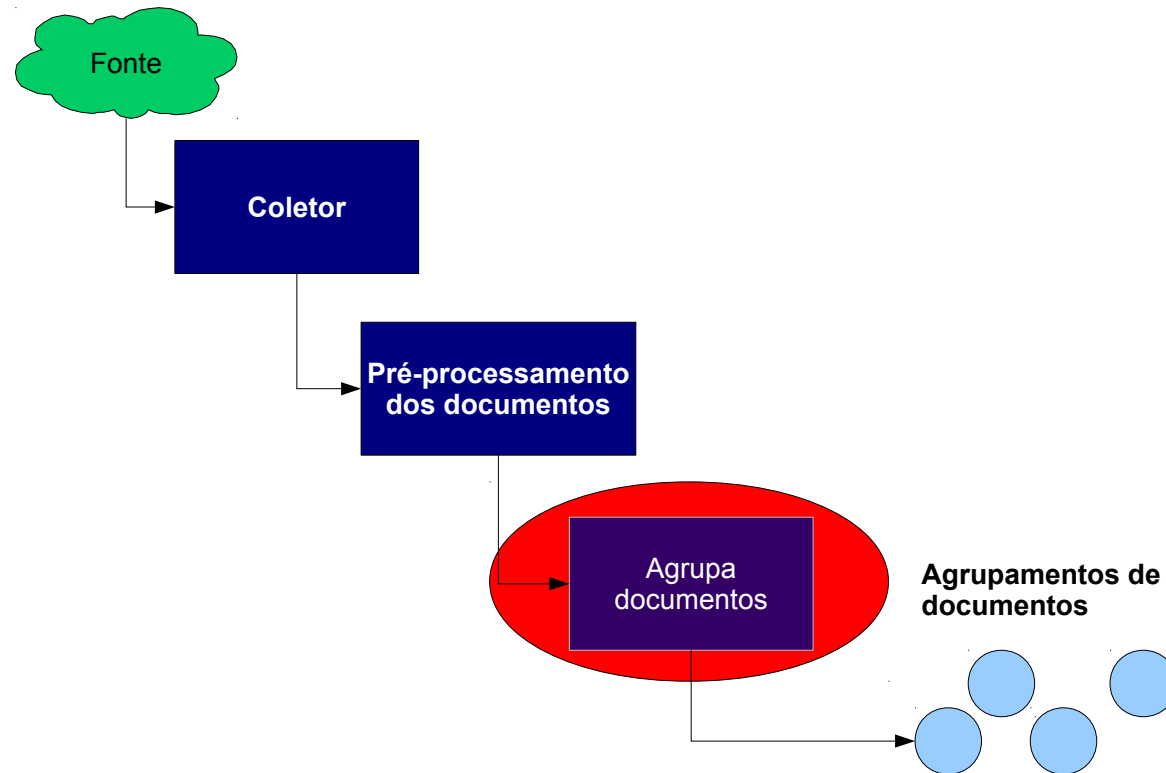
- N = número de documentos.
- n_i = número de documentos onde a palavra i aparece.
- $\frac{f_{i,j}}{\max_z f_{z,j}}$ = frequência normalizada da palavra i no documento j .

Relembrando as propriedades do log

N	ni	n	log10(n)
10	10	1	0
10	9	1,11	0,05
10	8	1,25	0,1
10	7	1,43	0,15
10	6	1,67	0,22
10	5	2	0,3
10	4	2,5	0,4
10	3	3,33	0,52
10	2	5	0,7
10	1	10	1



Componentes para uma solução...



Exemplo de projeto no R

Projeto MLR - Script `twitter/exemploAgrupamentoTexto.R`

Algoritmos para Agrupamento

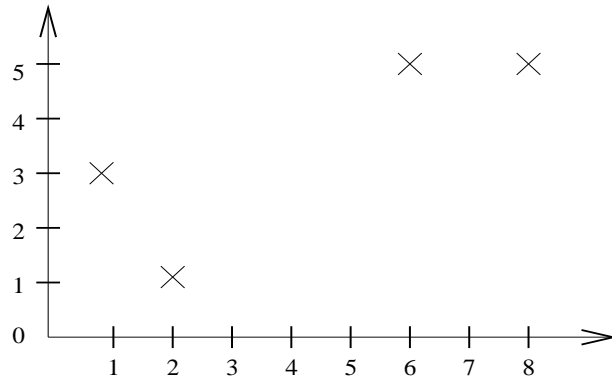
Definições de Algoritmos de Agrupamento

- O objetivo dos algoritmos de agrupamento é colocar os objetos **similares** em um **mesmo grupo** e objetos **não similares** em **grupos diferentes**.
- Normalmente, objetos são descritos e agrupados usando um conjunto de **atributos e valores**.
- Não existe nenhuma informação sobre a classe ou categoria dos objetos.

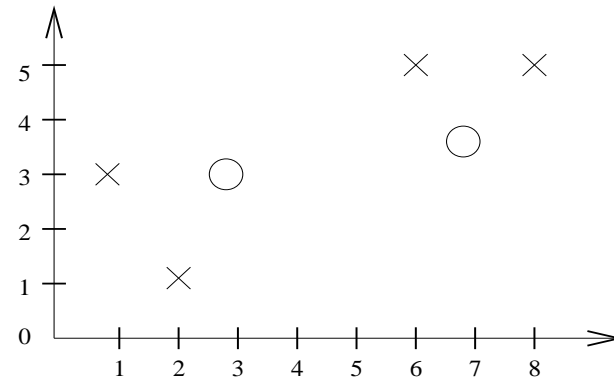
Algoritmos para Agrupamento - *K-means*

- **K** significa o número de agrupamentos (que deve ser informado à priori).
- Sequência de ações **iterativas**.
- A parada é baseada em algum critério de qualidade dos agrupamentos (por exemplo, similaridade média).

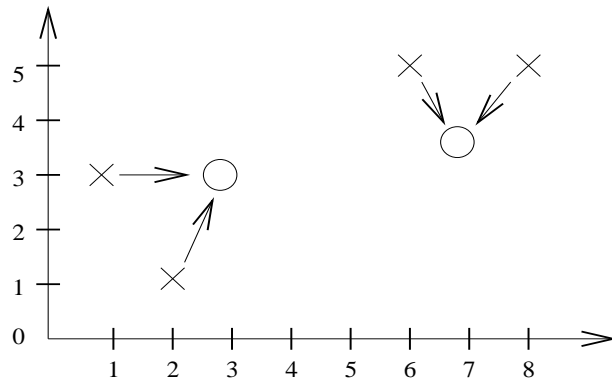
Algoritmo para Agrupamento - *K-means*



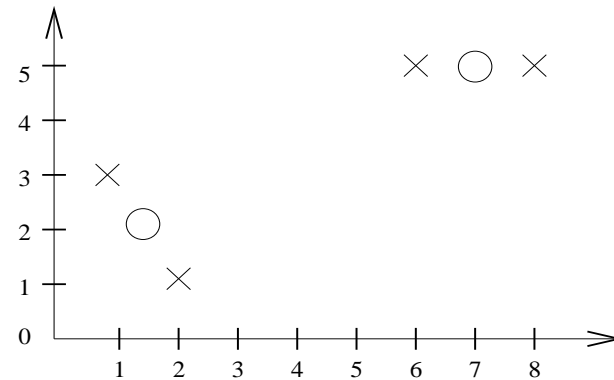
(1) Objetos que devem ser agrupados



(2) Sorteio dos pontos centrais dos agrupamentos



(3) Atribuição dos objetos aos agrupamentos



(4) Definição do centro do agrupamento

Algoritmo **K-means**

- A medida de distância pode ser a distância Euclidiana:

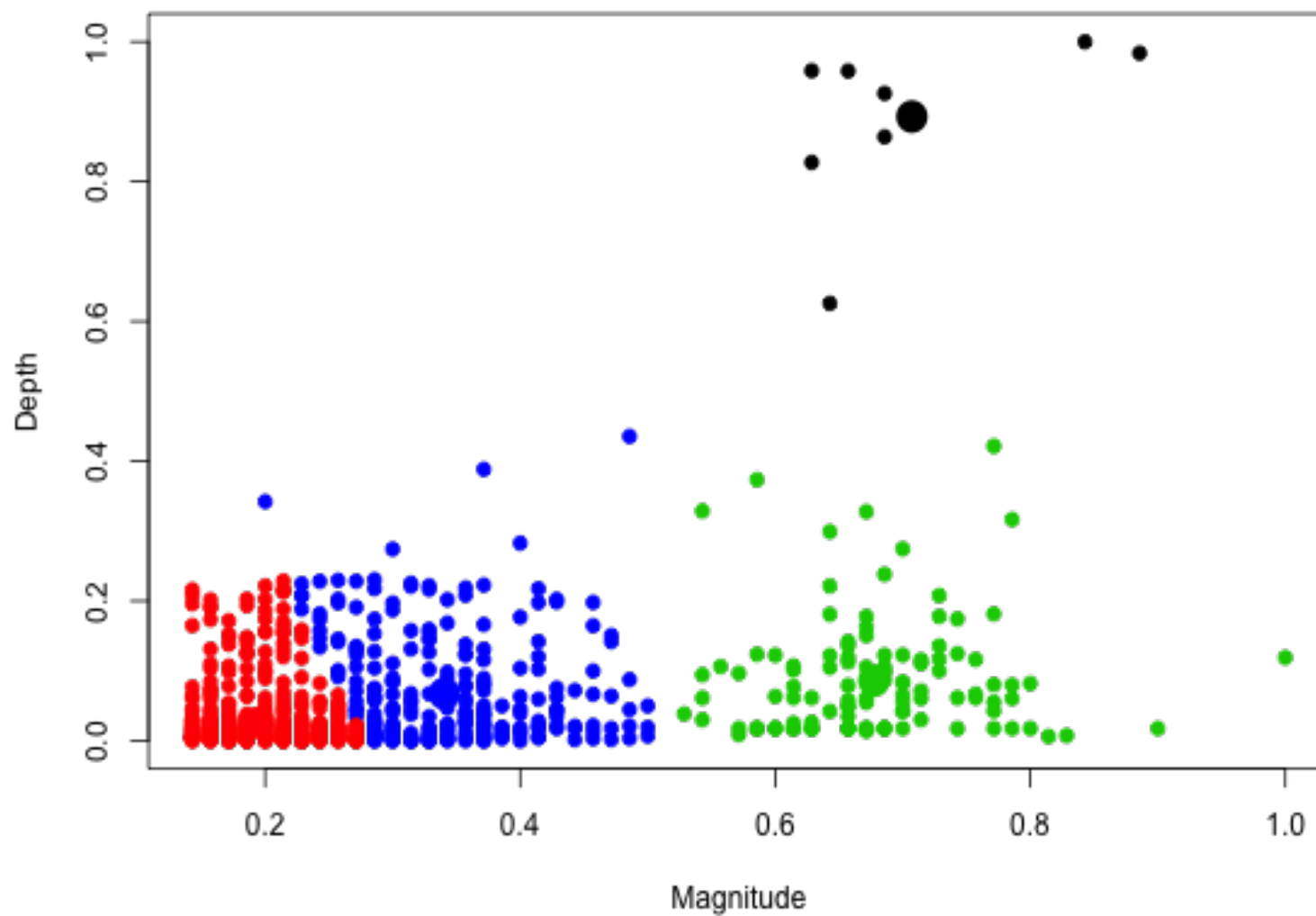
$$| \vec{x} - \vec{y} | = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (6)$$

- a função para computar o ponto central pode ser:

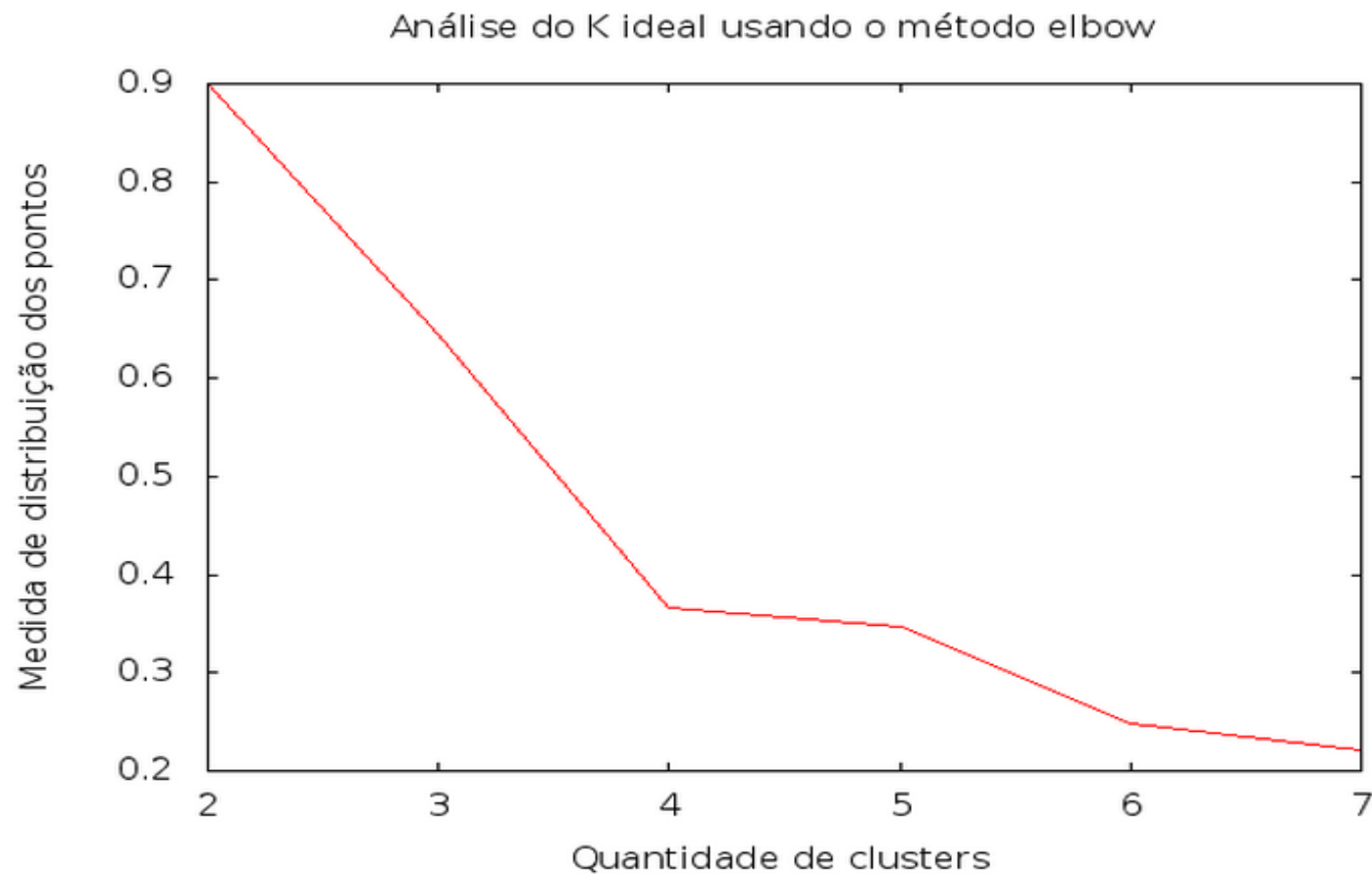
$$\vec{\mu} = \frac{1}{M} \sum_{\vec{x} \in C} \vec{x} \quad (7)$$

onde M é igual ao número de pontos no agrupamento C .

Clusters de abalos sísmicos (Wed Apr 10 22:50:58 2013)



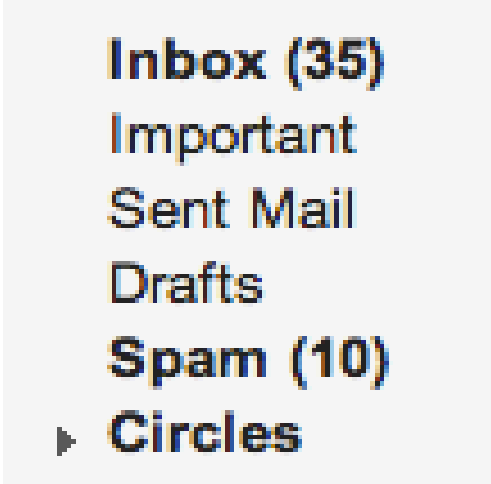
Como determinar o melhor k ?



A medida de distribuição dos pontos normalmente empregada é *sum of squared errors*.

Desenvolvimento de algoritmos anti-spam

Exemplos onde aplicar



A screenshot of an email client's folder list. The folders are listed vertically: 'Inbox (35)', 'Important', 'Sent Mail', 'Drafts', 'Spam (10)', and 'Circles'. The 'Circles' folder is highlighted with a right-pointing arrow.

- Inbox (35)**
- Important
- Sent Mail
- Drafts
- Spam (10)**
- ▶ **Circles**



Membro desde 26/05/2011

★ 6449
avaliações

👤 2193
seguidores

🏠 3
loais

📷 53
fotos

★★★★★ 15/01/2014 via [Apontador Android](#)

excelente churrascaria,tudo muito gostoso



Essa avaliação me ajudou (0)

[Reportar abuso](#)



Membro desde 11/07/2013

★ 12
avaliações

👤 1
seguidor

🏠 0
local

📷 0
foto

★★★★★ 11/07/2013 via [Apontador](#)

A comida é boa, porém o ambiente é um pouco cheio e causa demora no atendimento.



Essa avaliação me ajudou (2)

[Reportar abuso](#)



Membro desde 26/02/2011

★ 646
avaliações

👤 493
seguidores

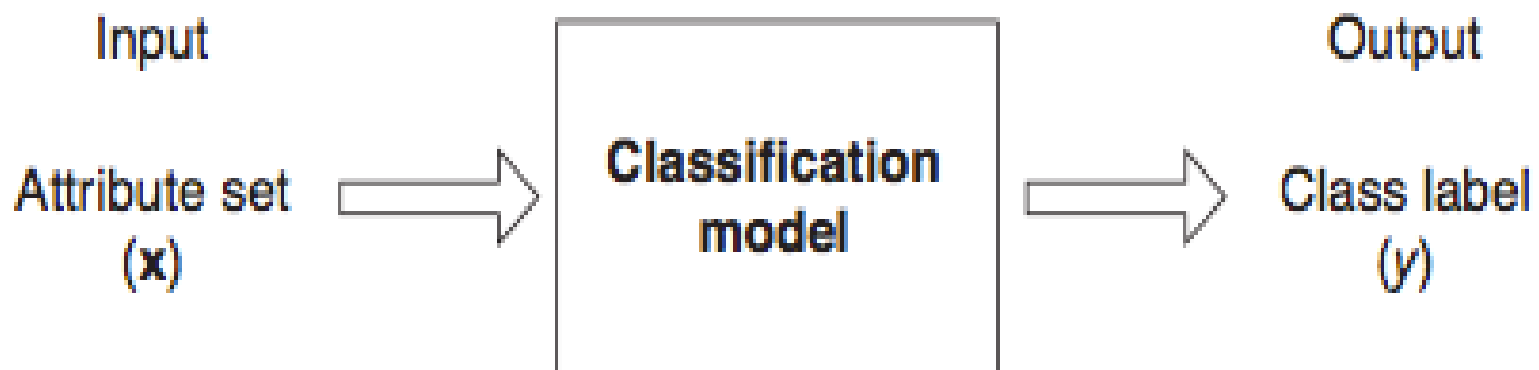
🏠 126
loais

📷 300
fotos

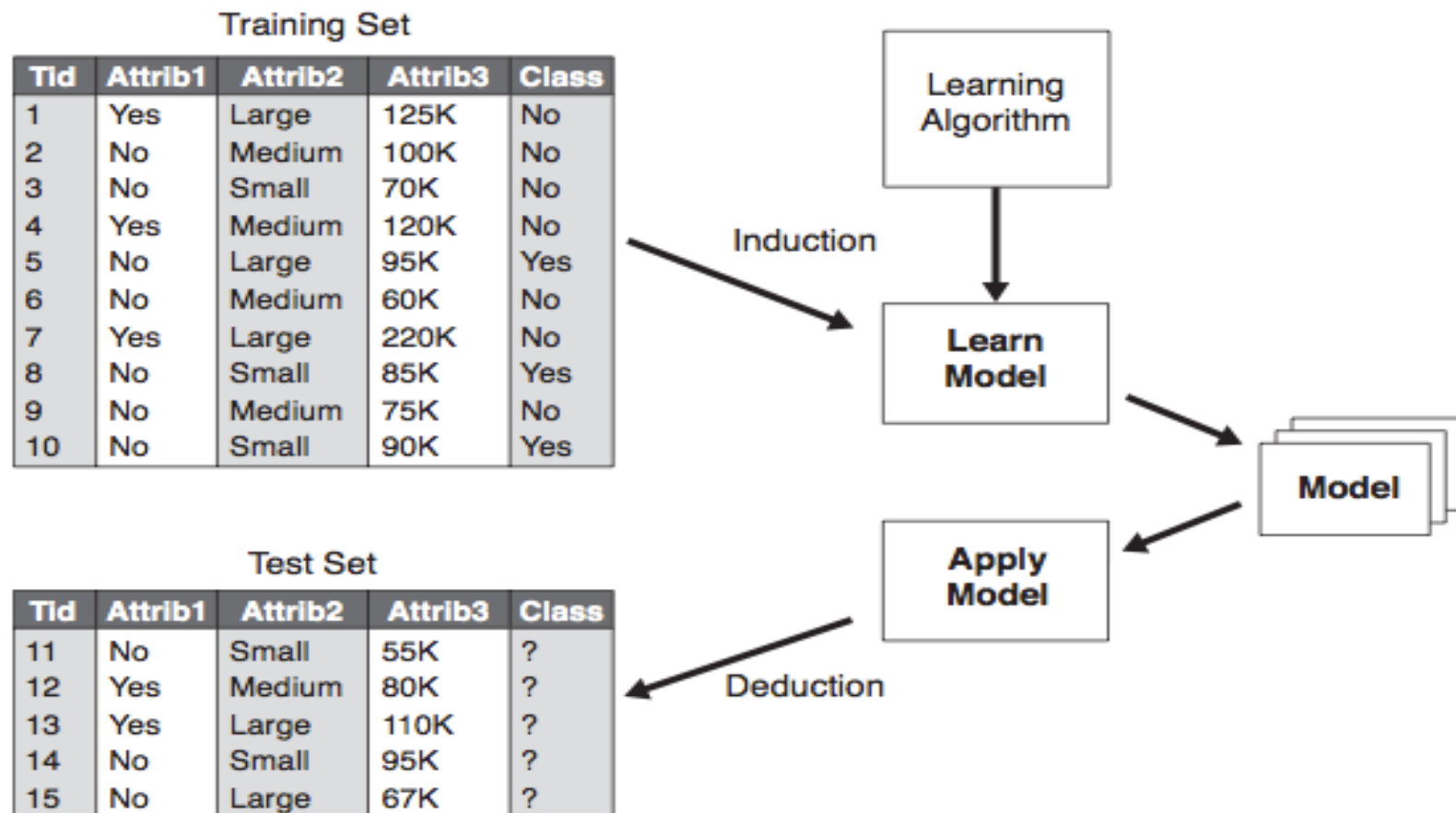
★★★★★ 24/06/2013 via [Apontador](#)

uma boa comida vc entra aqui, recomendo a todos!!

Modelos preditivos para classificação

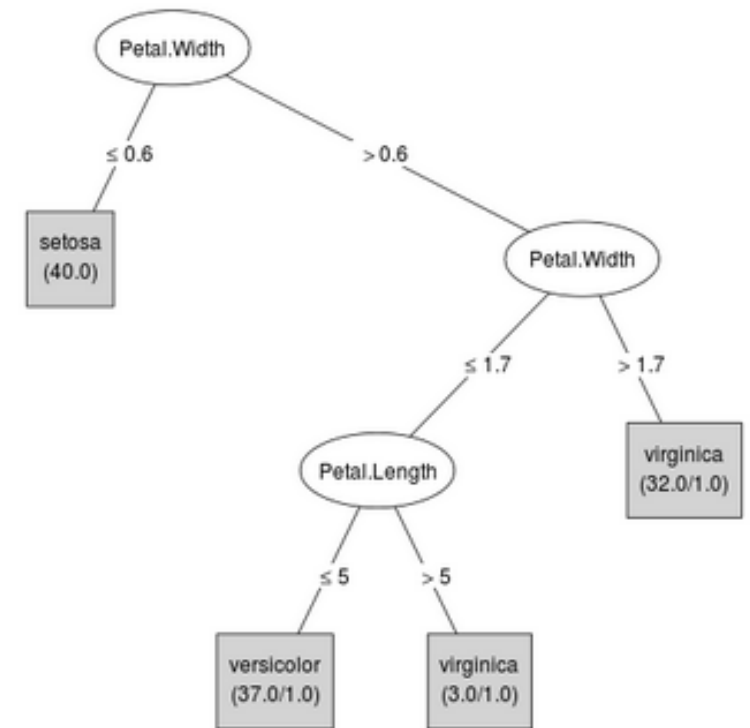
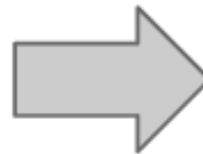


Desenvolvimento de modelos preditivos para classificação



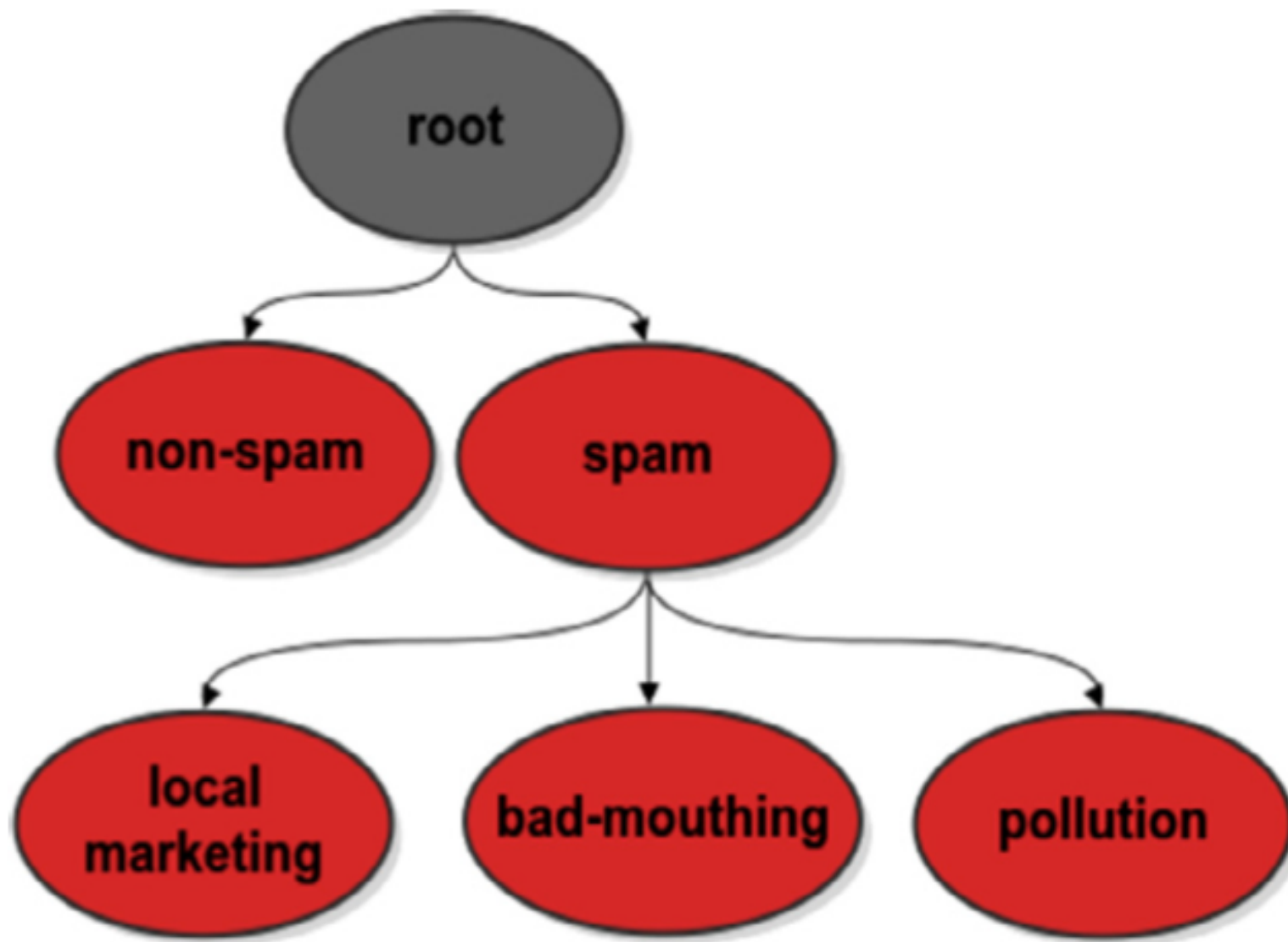
Aprendizado de árvores de decisão

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
15	5.8	4.0	1.2	0.2	setosa
16	5.7	4.4	1.5	0.4	setosa
19	5.7	3.8	1.7	0.3	setosa
30	4.7	3.2	1.6	0.2	setosa
32	5.4	3.4	1.5	0.4	setosa
50	5.0	3.3	1.4	0.2	setosa
73	6.3	2.5	4.9	1.5	versicolor
74	6.1	2.8	4.7	1.2	versicolor
81	5.5	2.4	3.8	1.1	versicolor
89	5.6	3.0	4.1	1.3	versicolor
90	5.5	2.5	4.0	1.3	versicolor
91	5.5	2.6	4.4	1.2	versicolor
104	6.3	2.9	5.6	1.8	virginica
112	6.4	2.7	5.3	1.9	virginica
122	5.6	2.8	4.9	2.0	virginica
126	7.2	3.2	6.0	1.8	virginica
146	6.7	3.0	5.2	2.3	virginica
148	6.5	3.0	5.2	2.0	virginica



Exemplo de classificação de Spam

- Objetivo: identificação de spam no www.apontador.com.br (*Location Based Social Network*).
- Tipos de atributos utilizados para caracterizar o conjunto de *posts* no site: **conteúdo, sobre o usuário, sobre o lugar e social.**



Questões...

- Q_1 : Será que é possível construir uma função de anti-spam com acurácia acima de 90 % apenas com atributos de conteúdo?

Método para construção do modelo: Florestas de árvores de decisão

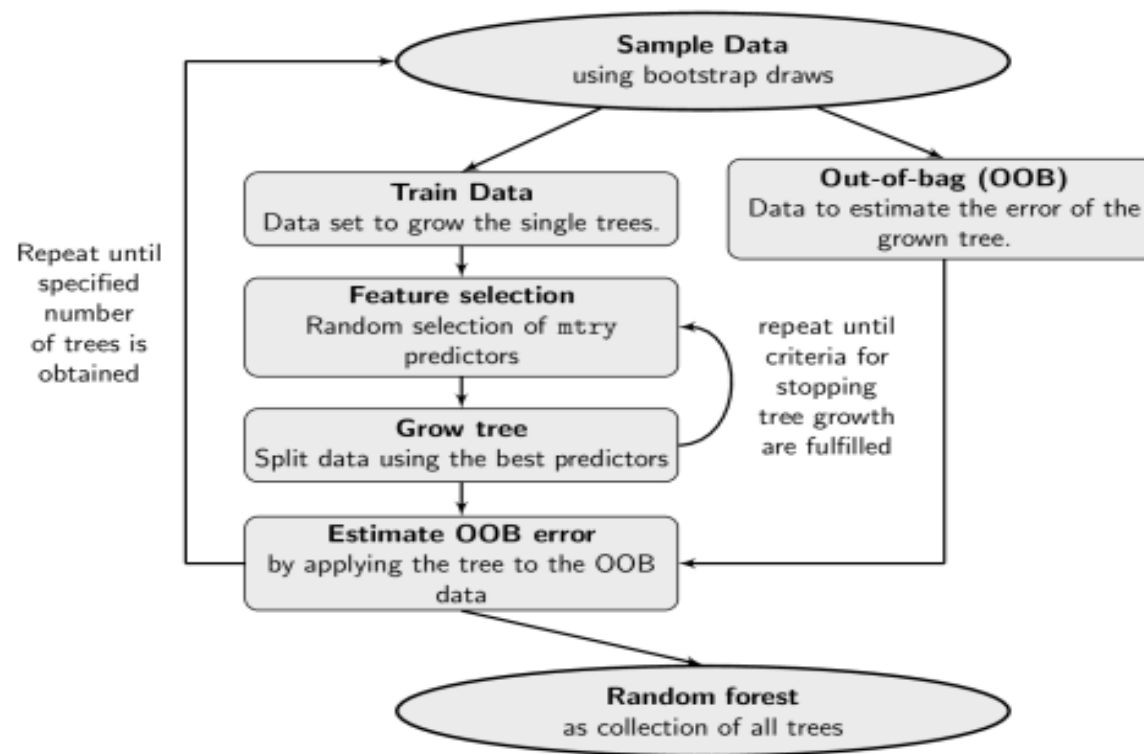


Figure 1: Random Forest Algorithm

Código

- Projeto: <https://github.com/fbarth/mlr>
- Arquivo: `scripts/antiSpam/attr_conteudos.R`

Questões...

- Q_1 : Será que é possível construir uma função de anti-spam com acurácia acima de 90 % apenas com atributos de conteúdo?
- Q_2 : **Será que é possível construir uma função de anti-spam com acurácia acima de 90 % utilizando todos os tipos de atributos coletados?**

Código

- Projeto: <https://github.com/fbarth/mlr>
- Arquivo: `scripts/antiSpam/attr_todos.R`

Considerações finais

- Análise de mensagens do twitter
 - ★ Transformação de informação não-estruturada em estruturada.
 - ★ Uso do algoritmo k-means
 - ★ Este mesmo processo pode ser aplicado para outros problemas similares: análise de notícias, análise de patentes e artigos científicos.

-
- Desenvolvimento de algoritmos anti-spam
 - ★ Uso do algoritmo random forest.
 - ★ Como desenvolver e avaliar um modelo preditivo.
 - ★ Este mesmo processo pode ser aplicado para outros problemas similares, inclusive problemas de recomendação de itens.

Referências

- Bing Liu. Web Data Mining: exploring hyperlinks, contents, and usage data, 2008.
- Tom Mitchell. Machine Learning, 1997.
- Ian H. Witten and Eibe Frank. Data Mining: Practical Machine Learning Tools and Techniques (Third Edition), 2011.
- Pang-Ning Tan, Michael Steinbach and Vipin Kumar. Introduction to Data Mining, 2006.
- Andrew Ng. <http://www.ml-class.org>

-
- Andy and Matthew. Classification and regression by randomForest. R News, vol. 3, number 3, pages 18-22, 2002.
 - Costa, H.; Merschmann, L. H. C.; Barth, F.; Benevenuto, F. Pollution, Bad-mouthing, and Local Marketing: The Underground of Location-based Social Networks. Information Sciences, 2014.
 - RDataMining.com: Text Mining.
<http://www.rdatamining.com/examples/text-mining>.
Acessado em 14 de junho de 2013.
 - Ingo Feinerer. Introduction to the tm Package: Text Mining in R. <http://cran.r-project.org/web/packages/tm/vignettes/tm.pdf>. Acessado em 14 de junho de 2013.

-
- Barth, F. J. Ferramentas para a detecção de grupos em Wikis. In: VII Simpósio Brasileiro de Sistemas Colaborativos, 2010, Belo Horizonte. Anais do VII Simpósio Brasileiro de Sistemas Colaborativos. IEEE Computer Society, 2010. v.II. p.8 - 11.
 - Barth, F. J. ; Belderrain, M. C. R. ; Quadros, N. L. P. ; Ferreira, L. L. ; Timoszczuk, A. P. . Recuperação e mineração de informações para a área criminal. In: VI Encontro Nacional de Inteligência Artificial, 2007, Rio de Janeiro. Anais do XXVII Congresso da SBC, 2007.