
Instituto Nacional de Pesquisas Espaciais

<http://www.inpe.br>

Coordenação dos Laboratórios Associados – CTE

Laboratório Associado de Computação e Matemática Aplicada – LAC

<http://www.lac.inpe.br>



Missão: Produzir ciência e tecnologia nas áreas espacial e do ambiente terrestre e oferecer produtos e serviços singulares em benefício do Brasil.



- Atuação Inter- e Multidisciplinar no Desenvolvimento Científico e Tecnológico para a Inovação na Área Espacial.
- Formação de Recursos Humanos (Pós-Graduação).



- Modelagem Computacional.
- Engenharia e Segurança de Sistemas.
- **Análise, Processamento e Extração da Informação.**

- Sistemas inteligentes em aplicações espaciais e ambientais.
 - Classificação de padrões (culturas agrícolas).
 - Planejamento inteligente (controle de satélites).
 - **Mineração de Dados.**
- Tratamento de incerteza e processos de tomada de decisão.
 - Controle inteligente - navegação autônoma.
 - Classificação de imagens de satélite.
 - Processamento de sinais e imagens.
 - Previsão de eventos extremos.
- Processamento Inteligente de Imagens.
 - Operadores inteligentes adaptáveis para processamento de imagens.

Conceitos de Mineração de Dados Multimídia

Rafael Santos

- Apresentar conceitos, técnicas e exemplos de aplicação de mineração de dados.
- Descrever alguns dos algoritmos mais utilizados.
- Mostrar aplicações destes algoritmos para dados multimídia, incluindo problemas e possíveis soluções.

- ***Math-Lite!***

- Introdução e motivação: o *tsunami* de dados.
- Relação entre dados, informação e conhecimento.
- Definição de mineração de dados e descoberta de conhecimento em bases de dados.
- Exemplos de aplicação de técnicas de mineração de dados.

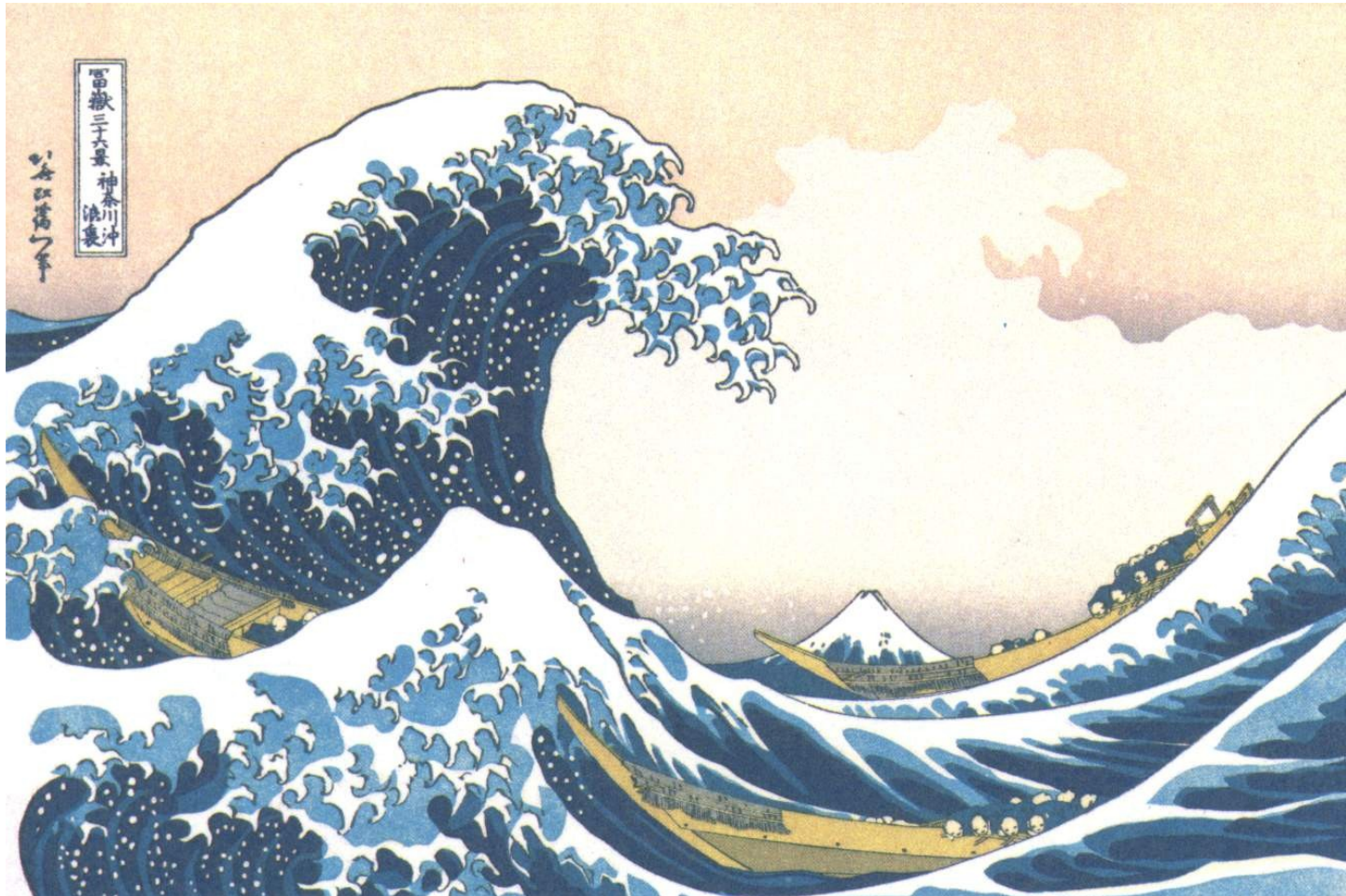
- Conceitos básicos.
- Espaço de atributos e similaridade.
- Pré-processamento.
- Técnicas e algoritmos com aplicações.
- Visualização.
- Outras técnicas associadas à mineração de dados.

- Pré-processamento para dados multimídia e o problema do *gap* semântico.
- Mineração de Dados Espaciais.
- Mineração de Dados Temporais.
- Mineração de Imagens.
- Mineração de Áudio.
- Mineração de Vídeo.
- Mineração de Texto.
- Mineração da WWW.

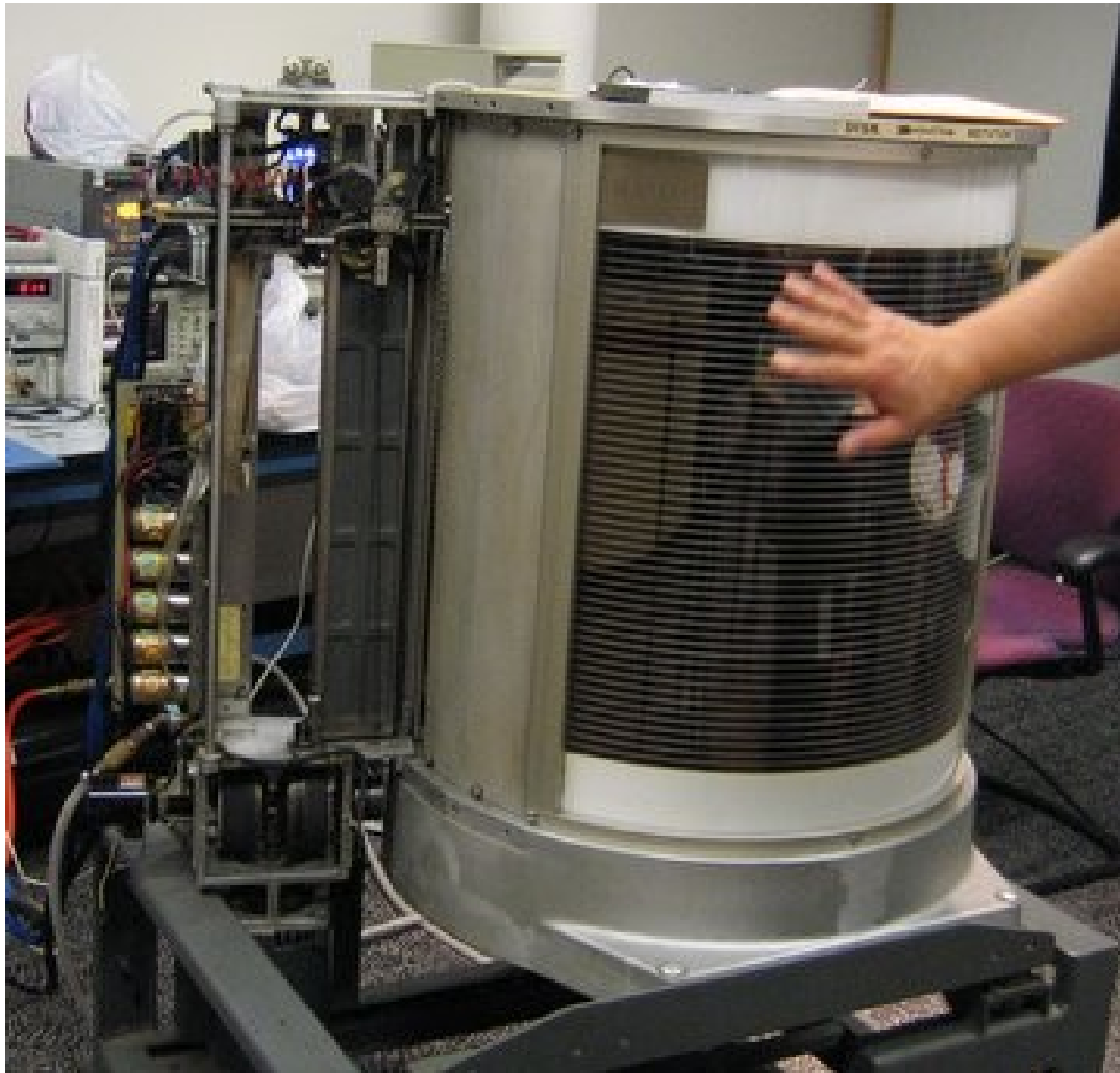
Introdução e Motivação

O Tsunami de Dados

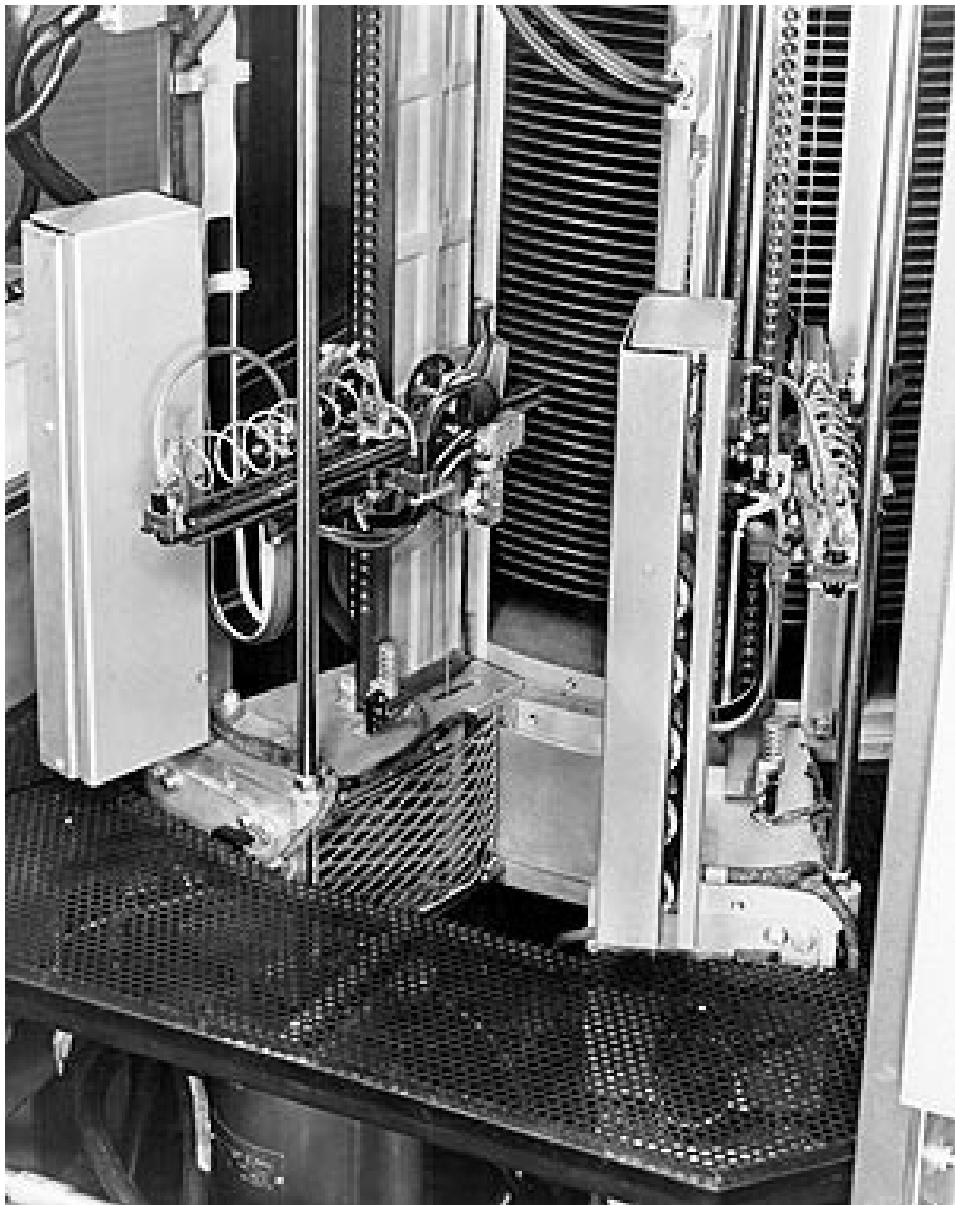
“We are drowning in information but starved for knowledge.” – John Naisbitt, Megatrends (1984).



O que é e como nos afeta?



Introdução e Motivação

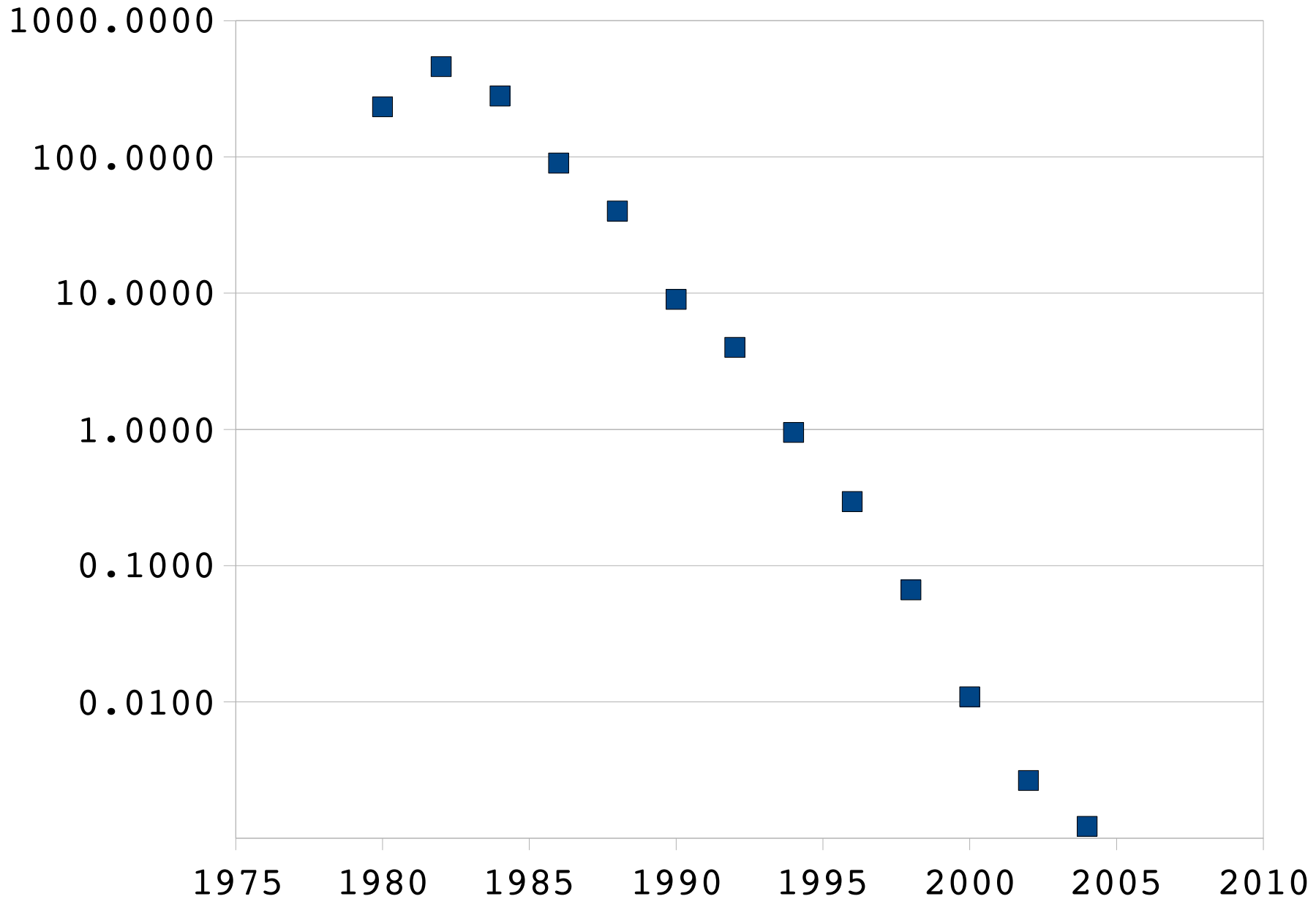




- Armazenamento do **RAMAC** (*Random Access Method of Accounting and Control*), IBM, 1956.
- 50 discos de 24 pol. de diâmetro.
- Quase 5 megabytes.
- Custo: Us\$ 160.000

Leia mais em <http://en.wikipedia.org/wiki/RAMAC> e http://www-03.ibm.com/ibm/history/exhibits/650/650_album.html

Introdução e Motivação



<http://www.littletechshoppe.com/ns1625/winchest.html>

1981

First compare quality. Then compare cost.

Morrow Designs' 10 megabyte hard disk system: \$3,695.

MORE MEMORY, LESS MONEY
Compare Morrow Designs' DISCUS™ 1000" hard disk systems to any system available for S-100 or Cromemco machines. First, compare features. Then, compare cost per megabyte. The M26 works out to under \$300 a megabyte. And the M10 is about half the cost of competing systems.

COMPLETE SUBSYSTEMS.
Both the M10 (9") and the M26 (11"), are delivered complete with disk-controller cables, fan, power supply cabinet and CP/M™ operating system. It's your choice: 10 Mbs 8" at \$3,695 or 20 Mbs 14" at \$4,995. That's single unit. Quantity prices are available.

BUILD TO FOUR DRIVES.
104 Megabytes with the M26. 40+ megabytes with the M10. Formatted. Additional drives: M26: \$4,499. M10: \$3,195. Quantity discounts available.

S-100, CROMEMCO AND NORTH STAR™
The M26 and M10 are sealed-media hard disk drives. Both S-100 controllers incorporate intelligence to supervise all data transfers through four I/O ports (command, 2 status and data). Transfers between drives and controllers are transparent to the CPU. The controller can also generate interrupts at the completion of each command... internally increasing system throughput. Sectors are individually write-protectable for multi-user environments. North Star or Cromemco? Call Mike Miller, Amarillo, TX, (806) 373-9533, for the software package that allows the M26/M10 to run on North Star DOS. MICAH of

Morrow Designs' 26 megabyte hard disk system: \$4,995.

MORROW DESIGNS

San Jose, CA, (415) 332-4443, offers a CP/M expanded to full Cromemco CDD5 compatibility

AND NOW, MULTI-IO.™
Multi-IO is an I/O controller that allows multi-terminal and multi-purpose use of S-100 and Cromemco computers. Three serial and two parallel output ports. Real time clock. Fully programmable interrupt controller. Designed with daisy-wheel printers in mind. Price: \$292 (incl. \$349 assembled and tested).

MAKE HARD COMPARISONS.
You'll find that Morrow Designs' hard disk systems offer the best price/performance ratio available for S-100, Cromemco and North Star computers. See the M26 and M10 hard disk subsystems at your computer dealer. Or, write Morrow Designs. Need information fast? Call us at (415) 524-2101.

Look to Morrow for answers.

CP/M is a trademark of Digital Research Inc. Copyright © a trademark of Cromemco, Inc. Multi-IO is a trademark of Morrow-Design Computers, Inc.

2008



Us\$ 320.

Us\$ 370/M → Us\$ 0.0032/M

www.vintagecomputing.com

- Crescimento explosivo na capacidade de gerar, coletar e armazenar dados:
 - Científicos: imagens, sinais.
 - Sociais: censos, pesquisas.
 - Econômicos e comerciais: transações bancárias e comerciais, compras, ligações telefônicas, acessos à web, transações com código de barras e RFID.
 - Segurança: acessos à sistemas em rede (*logs*), e-mails corporativos, registro de atividades.
- Justificativas para este aumento:
 - Barateamento de componentes e ambientes computacionais.
 - Exigências científicas/sociais.
 - Mudança de paradigmas!

- *SLAC (Stanford Linear Accelerator Center)*:
 - 200 terabytes/ano, 10 megabytes/segundo por 10 anos.
 - 2 petabytes = 2.097.152 gigabytes = ~440.000 DVDs = pilha de ~4.4km de altura.
 - LHC: 27 terabytes/dia, 1 ano = ~ 9.6 petabytes = pilha de 21km.
- *Digital Palomar Observatory Sky Survey (POSS-II)*:
 - 3 terabytes de imagens, estimados 2 bilhões de objetos em 1995.
- *Large Synoptic Survey Telescope*: 3.2 gigapixels, 20 a 30 terabytes/noite.
- *Pan-STARRS*: 4x1.4 gigapixels, 4 petabytes/ano, “filme” de 10 terapixels.

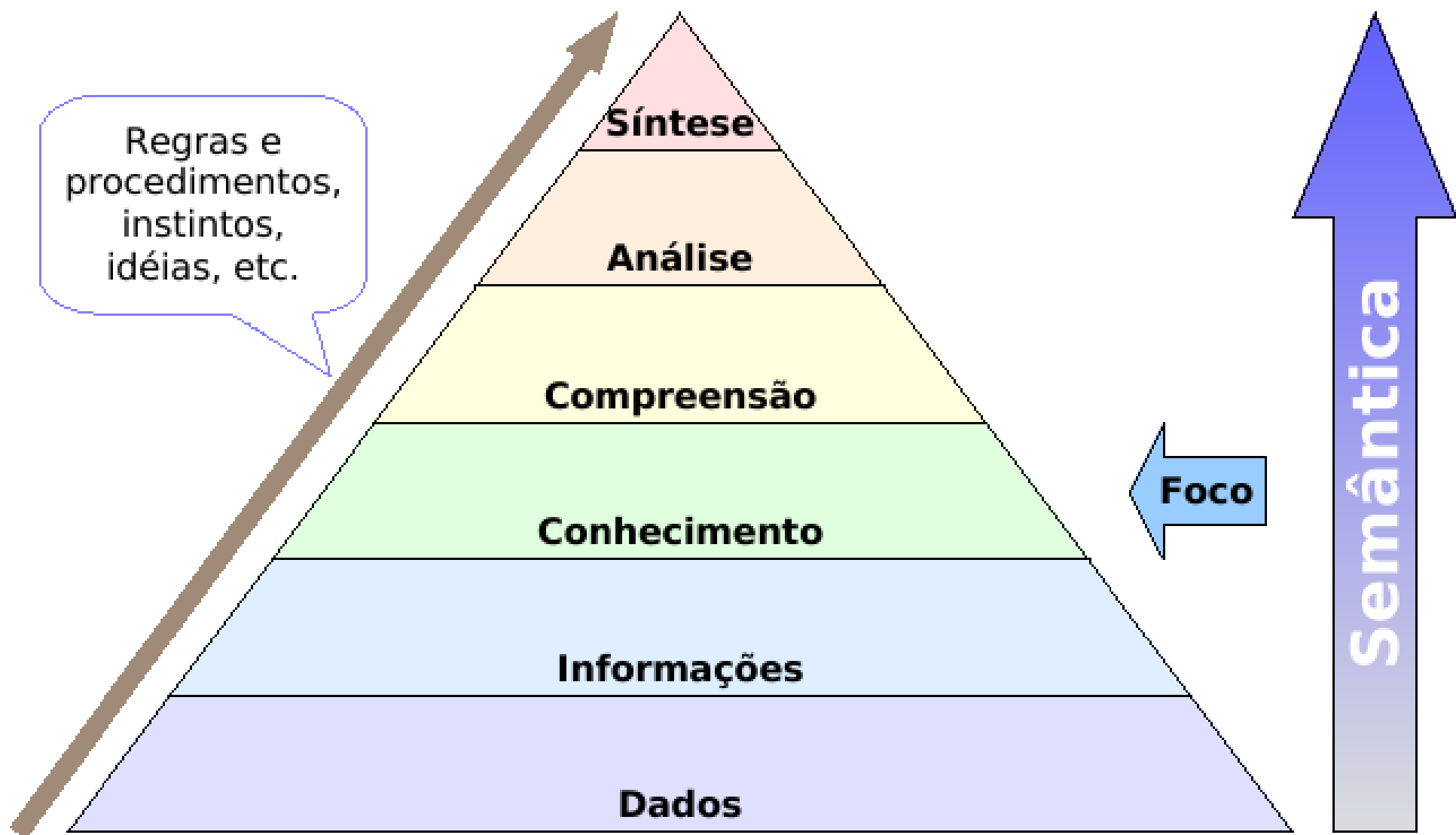
- *Surveys* associado a observatórios virtuais tem o potencial de usar terabytes de imagens e gerar gigabytes de tabelas.
- *Wayback machine*: 2 petabyte, 20 terabytes/mês (85 bilhões de páginas, 130.000 vídeos, 330.000 arquivos de áudio, 500.000 documentos em julho de 2008).
- *YouTube*: 45 terabytes de vídeos em 2006.
- *Flickr*: 2.2 bilhões de imagens.
- *GenBank*: coleções de sequências, mais de 100.000 organismos, 360 gigabytes.

- 1 terabyte = 200 dólares: 1 ano de armazenamento do LHC = 2 milhões de dólares.
- Transmitir 1 petabyte em uma rede de 100Mb/s: 86 milhões de segundos = 2 anos e 9 meses.
- 1 petabyte = pilha de 2.2 km de altura em DVDs. 100 computadores criando DVDs, cada DVD em meia hora: 46 dias para copiar um petabyte.

- Mídia impressa, filmes, mídia magnética e ótica produziram aproximadamente 5 exabytes de novos dados em 2002.
 - 1 exabyte = 1.024 petabytes = 1.048.576 terabytes.
- Consumidor americano típico gera 100G de dados em sua vida:
 - = ~ 26 exabytes para a população presente.
- Quantos registros de ligações telefônicas?
- Quantas transações de cartões por dia?
- Quantos acessos a diversos servidores de informação?

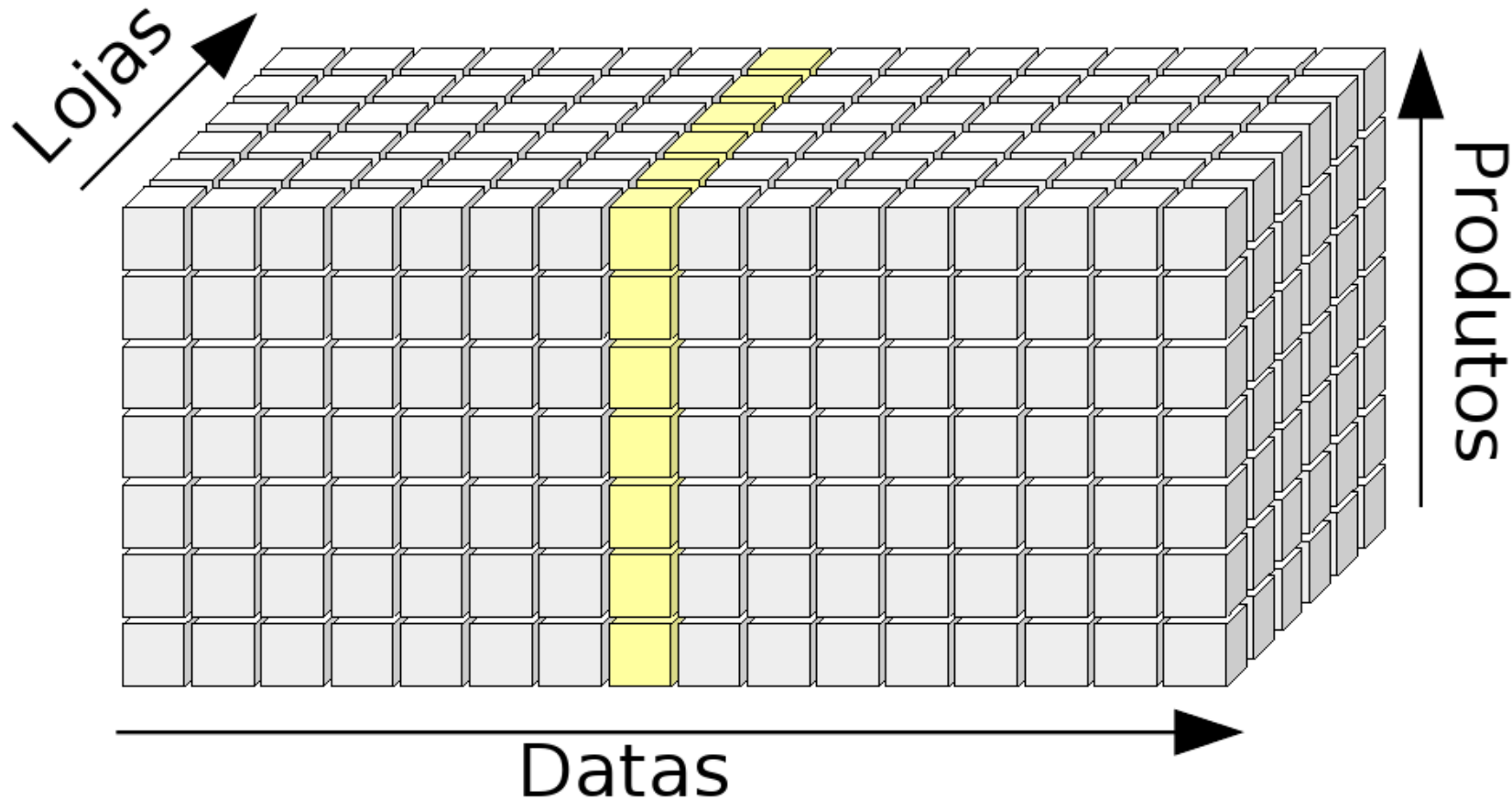
- *O que você tem no seu disco rígido?*

- Mas o que é feito destes dados?
 - Localizar, filtrar é relativamente simples...
 - Indexar pode ser mais complicado.
- Como identificar..
 - Padrões (“X” acontece se...)
 - Exceções (isto é diferente de... por causa de...)
 - Tendências (ao longo do tempo, “Y” deve acontecer...)
 - Correlações (se “M” acontece, “N” também deve acontecer.)
- O que existe de interessante nestes dados? Como definir “interessante”?
- **Informação**, e não dados, valem dinheiro / tempo / conhecimento!

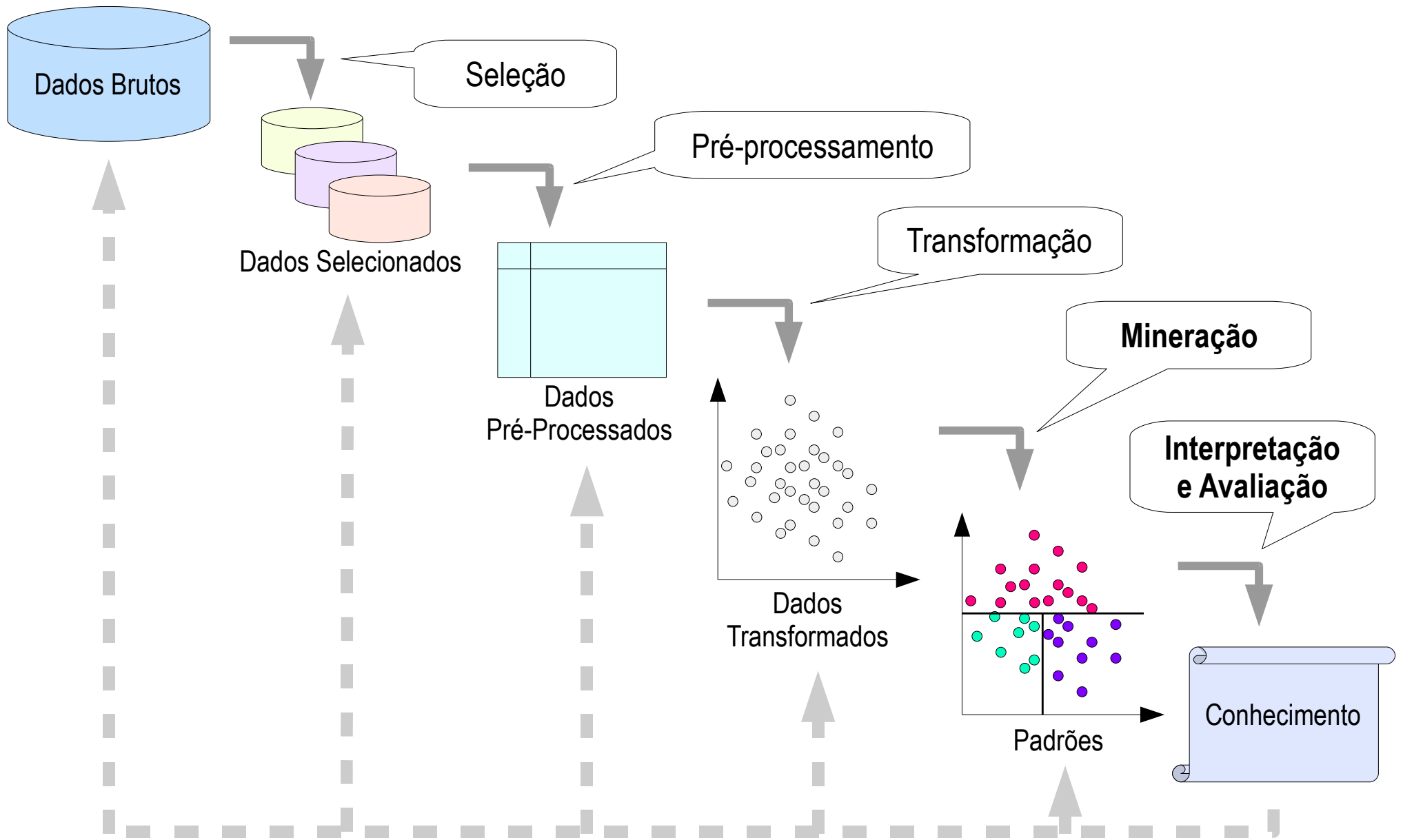


- Parte do processo de descoberta de conhecimentos em bancos de dados (*Knowledge Discovery in Databases, KDD*).
- **KDD**: Processo geral de descoberta de conhecimentos **úteis** **previamente desconhecidos** a partir de **grandes** bancos de **dados** (adaptado de Fayyad *et al*).

- Não é SQL nem OLAP, embora estas técnicas possam ser parte do processo.



Knowledge Discovery in Databases



- De acordo com Fayyad et. al.
 1. Compreensão do domínio da aplicação.
 2. Criação de conjunto de dados para descoberta.
 3. Limpeza e pré-processamento dos dados.
 4. Redução e reprojeção.
 5. Escolha da tarefa de mineração de dados.
 6. Escolha dos algoritmos de mineração e de seus parâmetros.
 7. **Mineração de dados.**
 8. Interpretação.
 9. Consolidação e avaliação.

- *Data mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner (Hand, Mannila and Smyth, Principles of Data Mining).*
- *Data mining is an interdisciplinary field bringing together techniques from machine learning, pattern recognition, statistics, databases, and visualization to address the issue of information extraction from large data bases (Evangelos Simoudis, citado em Daniel T. Larose, Discovering Knowledge in Data – An Introduction to Data Mining).*

- Interseção com outras técnicas e ciências.
- **Não é a “nova estatística!”**
- Usa muitos conceitos e técnicas de estatística, reconhecimento de padrões, aprendizado por máquina, inteligência artificial, bancos de dados, processamento de alto desempenho, visualização, etc.
- Tem caráter exploratório e prático.
- **Não dispensa interação e supervisão humanas!**

- **Amazon.com:** melhoria da customização da interface com o usuário (melhoria de vendas por indicação), eliminação de fraudes.
- **1-800-FLOWERS.com:** compreensão e antecipação de comportamento de clientes, descoberta de tendências e explicação de observações (CRM).
- **U.S. Census Bureau:** análise de dados espaciais (com SAS e software da ESRI) de ensino público para determinar políticas para melhoria na educação.
- **Japan Credit Bureau:** melhoria da resposta a campanhas de marketing, retenção de clientes, identificação de novos segmentos de mercado.

SAS Success stories: <http://www.sas.com/success/technology.html>

- **Columbia Interactive/Columbia University:** Análise de visitas a sites, coletando “trilhas” de usuários (como usam o site, que páginas são mais atraentes para usuários, quando usuários deixam o site) para melhorar interatividade e planejar conteúdo.
- **Casino:** cadeia com 115 hipermercados, 400 supermercados, mais de 4000 lojas e 260 lanchonetes. Criou programa de cartões de fidelidade e tem coletado dados dos cartões e hábitos de consumo.
- **TIM (Telecom Italia Mobile):** redução de *churn*, análise de comportamento do usuário e segmentação do banco de dados de usuários.

SAS Success stories: <http://www.sas.com/success/technology.html>

- **IMS America:** Empresa de pesquisa de mercado farmacêutico, mantém um banco de dados de 1.5 bilhões de prescrições de 600.000 médicos, usadas em 33.000 farmácias. Usa o banco para verificar que médicos mudaram seu padrão de prescrições para informar à companhias farmacêuticas, que podem decidir por campanhas de marketing dirigido aos médicos.
- **Harrah's Entertainment Inc.:** Cassino, dobrou lucros usando informações de cartões de “jogadores freqüentes”, identificando que um grupo de jogadores que gastavam entre 100 e 499 dólares (30% dos jogadores) geravam a maior parte do lucro do cassino. Testou diferentes promoções para este grupo, obtendo melhor fidelidade com menor custo e aumentando a resposta a campanhas de marketing.

Miriam Wasserman, *Mining data*. <http://www.bos.frb.org/economic/nerr/rr2000/q3/mining.htm>

- Muitos artigos nas áreas:
 - Mineração de dados espaciais/espaço-temporais, Análise de objetos móveis e trajetórias.
 - Mineração de imagens e sinais de diversos tipos.
 - Segurança, detecção de intrusão, análise de *logs*, análise de *malware*, *spam* e *worms*.
 - Tráfego e roteamento de redes.
 - Análise de grafos / redes de conexões (ex. redes sociais).
 - Análise de documentos (XML, HTML).
 - Bioinformática.

- Evidentemente raros e não anunciados...
 - ***Total Information Awareness***: forte rejeição pela ACLU, outras entidades.
 - **Gazelle.com**: caso-teste, investimento não seria recuperado.
 - Bebidas dietéticas levam à obesidade.

- ***Data Mining* é automático:** é um processo, é iterativo, requer supervisão.
- **Investimentos são recuperados rapidamente:** depende de muitos fatores!
- ***Software* são intuitivos e simples:** é mais importante conhecer os conceitos dos algoritmos e o negócio em si!
- ***Data Mining* pode identificar problemas no negócio:** DM pode encontrar padrões e fenômenos, identificar causa deve ser feito por especialistas.

Adaptado de Daniel T. Larose, *Discovering Knowledge in Data – An Introduction to Data Mining*

Analogia



- Falamos sobre terabytes e petabytes, mas não podemos mostrar exemplos práticos nesta escala.
- Falamos sobre dezenas ou centenas de atributos de diversos tipos, mas não é simples demonstrar algoritmos usando-os.
- Ficamos limitados a *toy problems*, geralmente em duas dimensões numéricas, focando mais em características do algoritmo do que em performance e escalabilidade.

Conceitos Básicos

- Um exemplo (quase) prático.
- Categorias de algoritmos de mineração de dados.
- Representação de dados para mineração de dados.
 - Tipos de atributos.
- Espaço de Atributos.
- Pré-processamento.

Exemplo (quase) prático



Saída	Rota	Demora Sinal A	Demora Sinal B	Tempo	Chegada
07:15	A	70	90	Sol	07:37
07:15	A	50	50	Nublado	07:46
07:15	A	20	100	Nublado	07:45
07:15	A	70	90	Sol	07:37
07:15	A	30	60	Chuva	07:56
07:15	B	70	90	Sol	07:42
07:15	B	70	70	Chuva	07:59
07:15	B	60	60	Nublado	07:51
07:15	B	90	100	Sol	07:41
07:15	B	50	50	Chuva	08:01
07:15	C	70	100	Chuva	08:03
07:15	C	40	90	Sol	07:46
07:15	C	40	70	Chuva	08:03
07:15	C	80	100	Chuva	08:04
07:15	C	100	70	Sol	07:46

Exemplo (quase) prático



Saída	Rota	Demora Sinal A	Demora Sinal B	Tempo	Chegada
07:15	A	70	90	Sol	07:37
07:15	A	50	50	Nublado	07:46
07:15	A	20	100	Nublado	07:45
07:15	A	70	90	Sol	07:37
07:15	A	30	60	Chuva	07:56
07:15	B	70	90	Sol	07:42
07:15	B	70	70	Chuva	07:59
07:15	B	60	60	Nublado	07:51
07:15	B	90	100	Sol	07:41
07:15	B	50	50	Chuva	08:01
07:15	C	70	100	Chuva	08:03
07:15	C	40	90	Sol	07:46
07:15	C	40	70	Chuva	08:03
07:15	C	80	100	Chuva	08:04
07:15	C	100	70	Sol	07:46

- Existe algum padrão nas diferenças de tempo? Existe algo fora de um padrão?
- Podemos caracterizar as situações de trânsito em função do tempo e das rotas?
 - Podemos escolher a melhor das rotas em função do tempo?
- Podemos prever o horário da chegada a partir dos outros dados?

- **Classificação:** aprendizado de uma função que pode ser usada para mapear dados em uma de várias classes discretas definidas previamente.
 - Se sair na hora **X** e o tempo estiver **Y** vai chegar atrasado.
- **Regressão ou Predição:** aprendizado de uma função que pode ser usada para mapear os valores associados aos dados em um ou mais valores reais.
 - Se sair na hora **X** e o tempo estiver **Y** vai chegar **N** minutos depois.

- **Agrupamento (ou *clustering*)**: identificação de grupos de dados onde os dados tem características semelhantes aos do mesmo grupo e onde os grupos tenham características diferentes entre si.
 - As rotas **B** e **C** tem características semelhantes.
- **Sumarização**: descrição do que caracteriza um conjunto de dados (ex. conjunto de regras que descreve o comportamento e relação entre os valores dos dados).
 - A rota **A** tem sinais com duração menor.

- **Detecção de desvios ou *outliers*:** identificação de dados que deveriam seguir um padrão esperado mas não o fazem.
 - As rotas **B** e **C** tem características semelhantes.
- **Identificação de associações:** identificação de grupos de dados que apresentam co-ocorrência entre si (ex. cesta de compras).
 - A rota **C** com chuva causa atraso de mais de **N** minutos.
- Técnicas podem ser usadas em mais de uma fase do processo de KDD.

Saída	Rota	Demora Sinal A	Demora Sinal B	Tempo	Chegada
07:15	A	80	90	Sol	07:37
07:15	A	20	90	Nublado	07:45
07:15	A	70	40	Nublado	07:45
07:15	A	60	80	Sol	07:37
07:15	A	70	50	Chuva	07:55
07:15	B	80	30	Sol	07:41
07:15	B	40	90	Chuva	08:00
07:15	B	90	100	Nublado	07:52

- Para facilitar...

- Dados em uma única tabela.
- Cada linha na tabela é uma **instância** ou **amostra** (registros).
- Cada coluna na tabela é um **atributo** (campos).
- Cada instância da base de dados tem os mesmos campos e que cada campo tem o mesmo tipo de valor.
- Eventualmente um atributo para uma instância pode ser desconhecido ou estar faltando.

- Tipos de atributos
 - Atributos **nominais** são rótulos, nomes, basicamente servem para identificar uma amostra e diferenciá-la de outra.
 - Atributos **categóricos** são semelhantes aos nominais mas são escolhidos de um conjunto definido.
 - Atributos **numéricos** expressam algo medido (com instrumentos, por exemplo).
 - Atributos **ordinais** são valores discretos mas que apresentam uma ordem imposta ou implícita.
- Podemos transformar alguns tipos em outros.
- Entender a diferença e limitações é muito importante!

- Pré-Processamento
 - Atributos com representação inadequada para tarefa e algoritmo.
 - Atributos cujos valores não tenham informações adequadas.
 - Excesso de atributos (podem ser redundantes ou desnecessários).
 - Atributos insuficientes.
 - Excesso de instâncias (afetam tempo de processamento).
 - Instâncias insuficientes.
 - Instâncias incompletas (sem valores para alguns atributos).
- Assim como a mineração de dados em si, requer conhecimento sobre os dados e algoritmo que será usado!

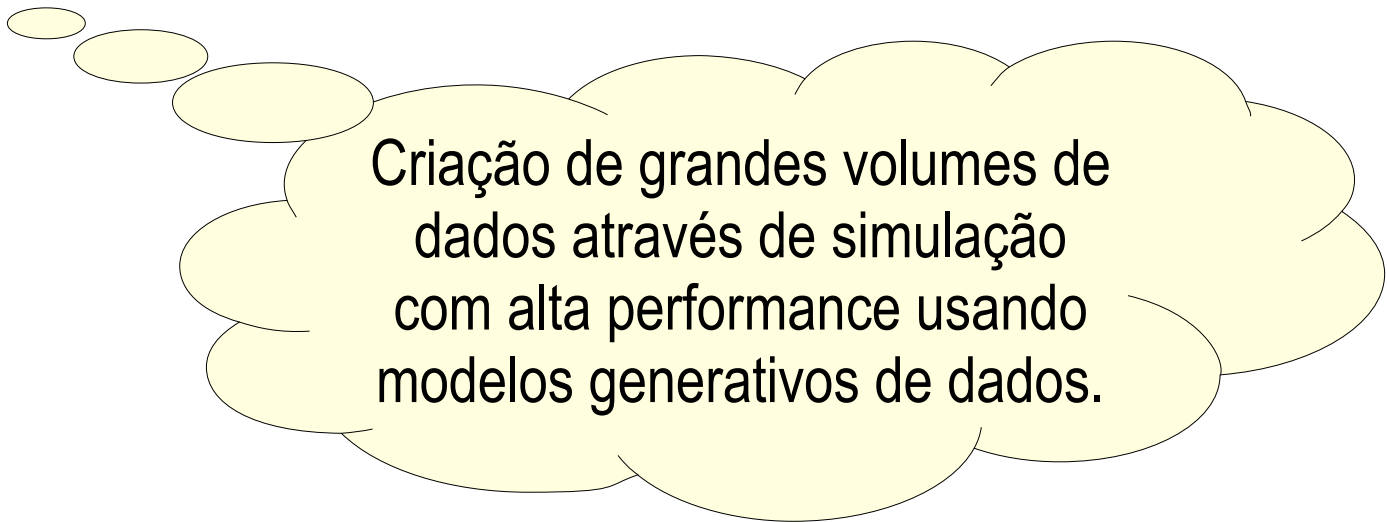


- Problemas:
 - Redes Neurais *Back-propagation* só operam com valores numéricos.
 - Alguns algoritmos de busca de associações só operam com valores simbólicos/discretos.
- Soluções:
 - Conversão de tipos de atributos (quando aplicável!)
 - Remoção dos atributos inadequados.
 - Separação em subtarefas usando os valores discretos dos atributos.

- Problemas:
 - Atributos com baixíssima variabilidade nos valores.
 - Atributos redundantes ou altamente correlacionados com outros.
- Soluções:
 - Remoção dos atributos inadequados.
 - Unificação de atributos ou derivação de novos atributos.

- Problemas:
 - Muitos atributos → complexidade de processamento.
 - Correlações irrelevantes podem complicar o processo de mineração (a não ser que seja necessário descobri-las!)
- Soluções:
 - Remoção dos atributos irrelevantes (possivelmente depois de alguma análise).
 - Mudança de representação ou projeção (usando, por exemplo, *PCA* ou Mapas de Kohonen).

- Problemas:
 - Poucos atributos podem não possibilitar mineração adequada (para identificar classes, por exemplo).
- Soluções:
 - Enriquecimento com dados complementares (se puderem ser obtidos!)
 - Enriquecimento com combinações não lineares.
 - *Data Farming*.

A yellow thought bubble with a black outline, containing text. It has several smaller bubbles leading to it from the left.

Criação de grandes volumes de dados através de simulação com alta performance usando modelos generativos de dados.

- Problemas:
 - Muitas instâncias podem tornar o processamento inviável: alguns algoritmos requerem várias iterações com os dados.
 - Problema relacionado: desbalanceamento de instâncias para classificação.
- Soluções:
 - Redução por amostragem.
 - Redução por prototipagem.
 - Particionamento do conjunto de dados.

- Problemas:
 - Poucas instâncias podem comprometer o resultado (que será pouco genérico ou confiável).
 - Casos raros podem não ser representados.
- Soluções:
 - Coleta de mais instâncias.
 - *Data Farming*.

- Problemas:
 - Dados coletados podem ter valores de atributos faltando.
 - Por que estão faltando? Rever modelagem do processo e coleta!
- Soluções:
 - Eliminação de dados/atributos com muitos valores faltando.
 - Completar através de proximidade/similaridade com dados completos.
 - Separar em conjuntos para processamento independente ou associado.

- Restrições dos algoritmos (para aplicabilidade, para garantir completeza e para reduzir complexidade).
 - É possível/viável?
- Devemos também considerar...
 - Atributos e dados podem/devem ser representados de outra forma?
 - Algumas conversões de tipos podem ser destrutivas: cuidado com discretização!

Saída	Rota	Demora Sinal A	Demora Sinal B	Tempo	Chegada
07:15	A	80	90	Sol	07:37
07:15	A	20	90	Nublado	07:45
07:15	A	70	40	Nublado	07:45
07:15	A	60	80	Sol	07:37
07:15	A	70	50	Chuva	07:55
07:15	B	80	30	Sol	07:41
07:15	B	40	90	Chuva	08:00
07:15	B	90	100	Nublado	07:52

- Instâncias são vetores de dados em um espaço N -dimensional.
 - Que “aparência” tem a distribuição das instâncias no espaço de atributos?
 - Existe correlação entre atributos?
 - Existe possibilidade de classificação simples?
 - Existem desvios ou *outliers* comprometedores?
 - As classes implícitas nos dados são separáveis?
- Conceito de *proximidade no espaço N -dimensional* (= **semelhança** de atributos) essencial!

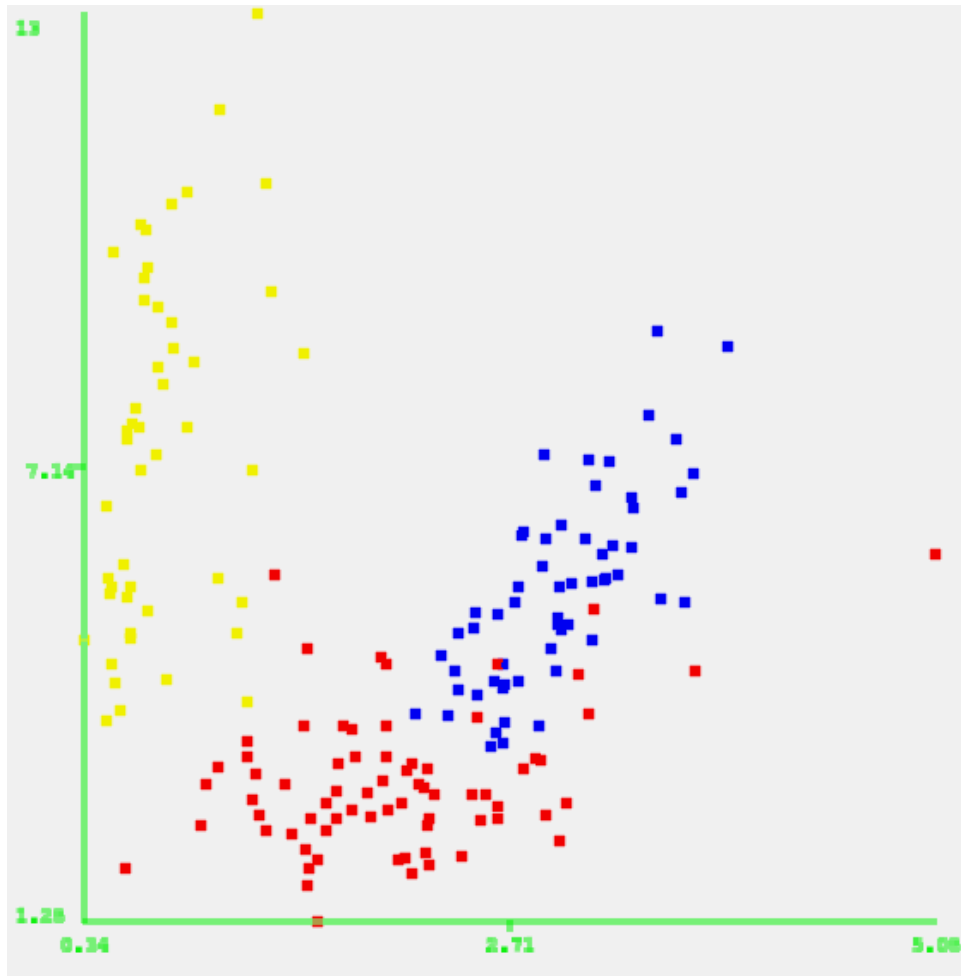
Conceitos Básicos: Espaço de Atributos



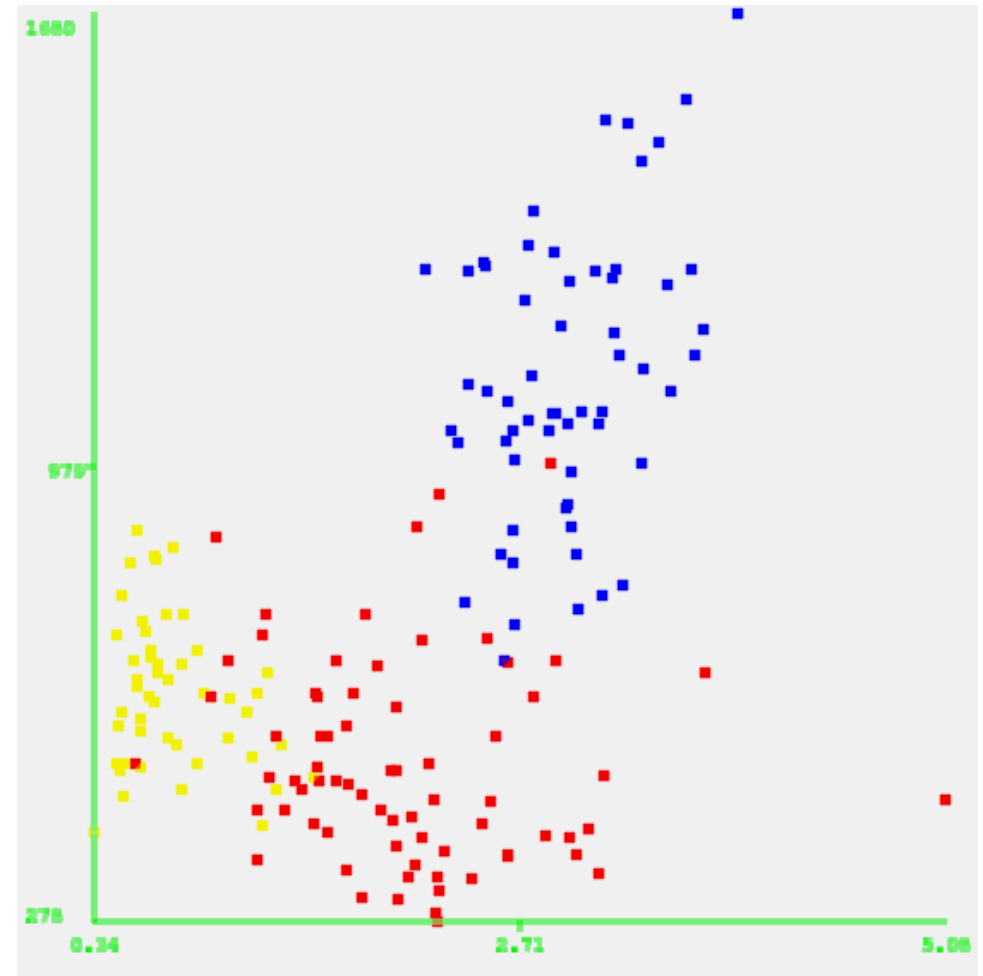
- Origem do vinho a partir de conteúdo físico-químico (13 atributos)
<http://archive.ics.uci.edu/ml/datasets/Wine> (nomes de atributos originais)

No.	Alcohol Numeric	MalicAcid Numeric	Ash Numeric	AlcalinityOfAsh Numeric	Magnesium Numeric	TotalPhenols Numeric	Flavanoids Numeric	NonflavanoidPhenols Numeric	Proanthocyanins Numeric	ColorIntensity Numeric	Hue Numeric	OD280_OD315OfDilutedWines Numeric	Proline Numeric	ORIGIN Nominal
1	14.23	1.71	2.43	15.6	127.0	2.8	3.06	0.28	2.29	5.64	1.04	3.92	106...	1
2	13.2	1.78	2.14	11.2	100.0	2.65	2.76	0.26	1.28	4.38	1.05	3.4	105...	1
3	13.16	2.36	2.67	18.6	101.0	2.8	3.24	0.3	2.81	5.68	1.03	3.17	118...	1
4	14.37	1.95	2.5	16.8	113.0	3.85	3.49	0.24	2.18	7.8	0.86	3.45	148...	1
5	13.24	2.59	2.87	21.0	118.0	2.8	2.69	0.39	1.82	4.32	1.04	2.93	735.0	1
6	14.2	1.76	2.45	15.2	112.0	3.27	3.39	0.34	1.97	6.75	1.05	2.85	145...	1
7	14.39	1.87	2.45	14.6	96.0	2.5	2.52	0.3	1.98	5.25	1.02	3.58	129...	1
8	14.06	2.15	2.61	17.6	121.0	2.6	2.51	0.31	1.25	5.05	1.06	3.58	129...	1
9	14.83	1.64	2.17	14.0	97.0	2.8	2.98	0.29	1.98	5.2	1.08	2.85	104...	1
10	13.86	1.35	2.27	16.0	98.0	2.98	3.15	0.22	1.85	7.22	1.01	3.55	104...	1
11	14.1	2.16	2.3	18.0	105.0	2.95	3.32	0.22	2.38	5.75	1.25	3.17	151...	1
12	14.12	1.48	2.32	16.8	95.0	2.2	2.43	0.26	1.57	5.0	1.17	2.82	128...	1
13	13.75	1.73	2.41	16.0	89.0	2.6	2.76	0.29	1.81	5.6	1.15	2.9	132...	1
14	14.75	1.73	2.39	11.4	91.0	3.1	3.69	0.43	2.81	5.4	1.25	2.73	115...	1
15	14.38	1.87	2.38	12.0	102.0	3.3	3.64	0.29	2.96	7.5	1.2	3.0	154...	1
16	13.63	1.81	2.7	17.2	112.0	2.85	2.91	0.3	1.46	7.3	1.28	2.88	131...	1
17	14.3	1.92	2.72	20.0	120.0	2.8	3.14	0.33	1.97	6.2	1.07	2.65	128...	1
18	13.83	1.57	2.62	20.0	115.0	2.95	3.4	0.4	1.72	6.6	1.13	2.57	113...	1
19	14.19	1.59	2.48	16.5	108.0	3.3	3.93	0.32	1.86	8.7	1.23	2.82	168...	1
20	13.64	3.1	2.56	15.2	116.0	2.7	3.03	0.17	1.66	5.1	0.96	3.36	845.0	1
21	14.06	1.63	2.28	16.0	126.0	3.0	3.17	0.24	2.1	5.65	1.09	3.71	780.0	1
22	12.93	3.8	2.65	18.6	102.0	2.41	2.41	0.25	1.98	4.5	1.03	3.52	770.0	1
23	13.71	1.86	2.36	16.6	101.0	2.61	2.88	0.27	1.69	3.8	1.11	4.0	103...	1
24	12.85	1.6	2.52	17.8	95.0	2.48	2.37	0.26	1.46	3.93	1.09	3.63	101...	1
25	13.5	1.81	2.61	20.0	96.0	2.53	2.61	0.28	1.66	3.52	1.12	3.82	845.0	1
26	13.05	2.05	3.22	25.0	124.0	2.63	2.68	0.47	1.92	3.58	1.13	3.2	830.0	1
27	13.39	1.77	2.62	16.1	93.0	2.85	2.94	0.34	1.45	4.8	0.92	3.22	119...	1
28	13.3	1.72	2.14	17.0	94.0	2.4	2.19	0.27	1.35	3.95	1.02	2.77	128...	1
29	13.87	1.9	2.8	19.4	107.0	2.95	2.97	0.37	1.76	4.5	1.25	3.4	915.0	1
30	14.02	1.68	2.21	16.0	96.0	2.65	2.33	0.26	1.98	4.7	1.04	3.59	103...	1
31	13.73	1.5	2.7	22.5	101.0	3.0	3.25	0.29	2.38	5.7	1.19	2.71	128...	1
32	13.58	1.66	2.36	19.1	106.0	2.86	3.19	0.22	1.95	6.9	1.09	2.88	151...	1
33	13.68	1.83	2.36	17.2	104.0	2.42	2.69	0.42	1.97	3.84	1.23	2.87	990.0	1
34	13.76	1.53	2.7	19.5	132.0	2.95	2.74	0.5	1.35	5.4	1.25	3.0	123...	1
35	13.51	1.8	2.65	19.0	110.0	2.35	2.53	0.29	1.54	4.2	1.1	2.87	109...	1
36	13.48	1.81	2.41	20.5	100.0	2.7	2.98	0.26	1.86	5.1	1.04	3.47	920.0	1
37	13.28	1.64	2.84	15.5	110.0	2.6	2.68	0.34	1.36	4.6	1.09	2.78	880.0	1
38	13.05	1.65	2.55	18.0	98.0	2.45	2.43	0.29	1.44	4.25	1.12	2.51	110...	1
39	13.07	1.5	2.1	15.5	98.0	2.4	2.64	0.28	1.37	3.7	1.18	2.69	102...	1

Conceitos Básicos: Espaço de Atributos

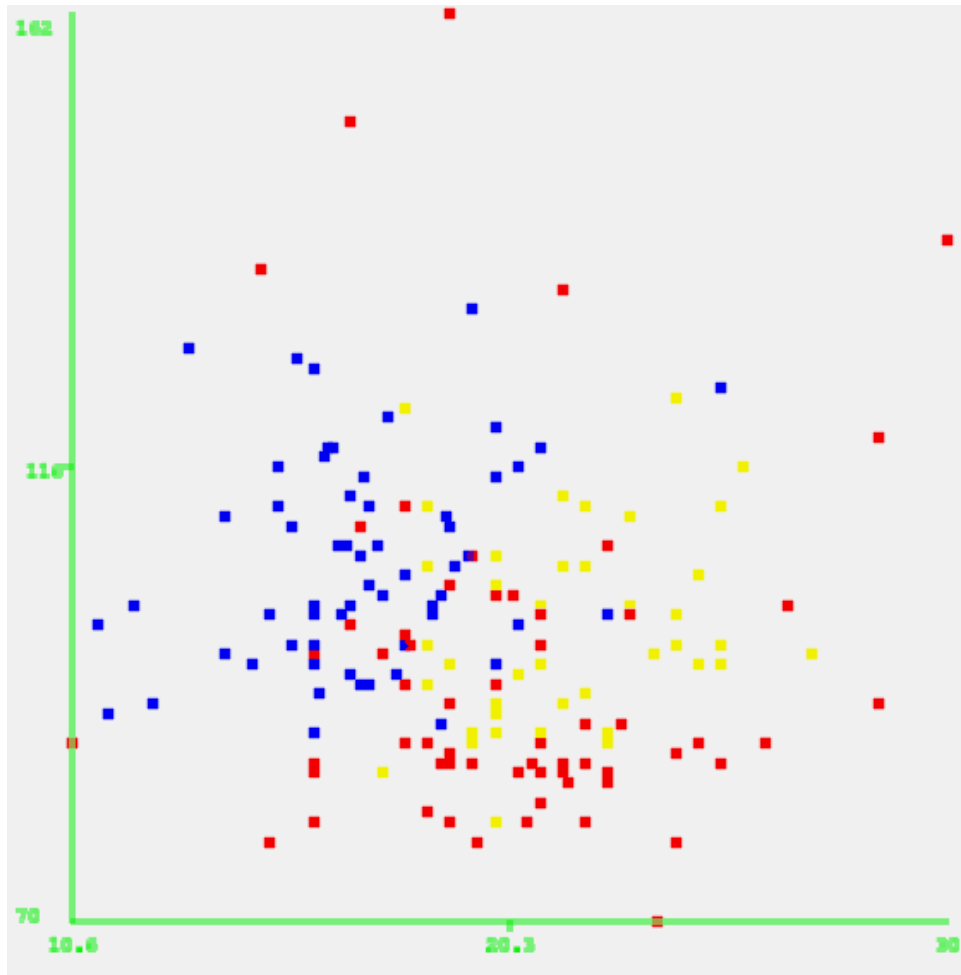


X: Flavonoids, Y: Color Intensity

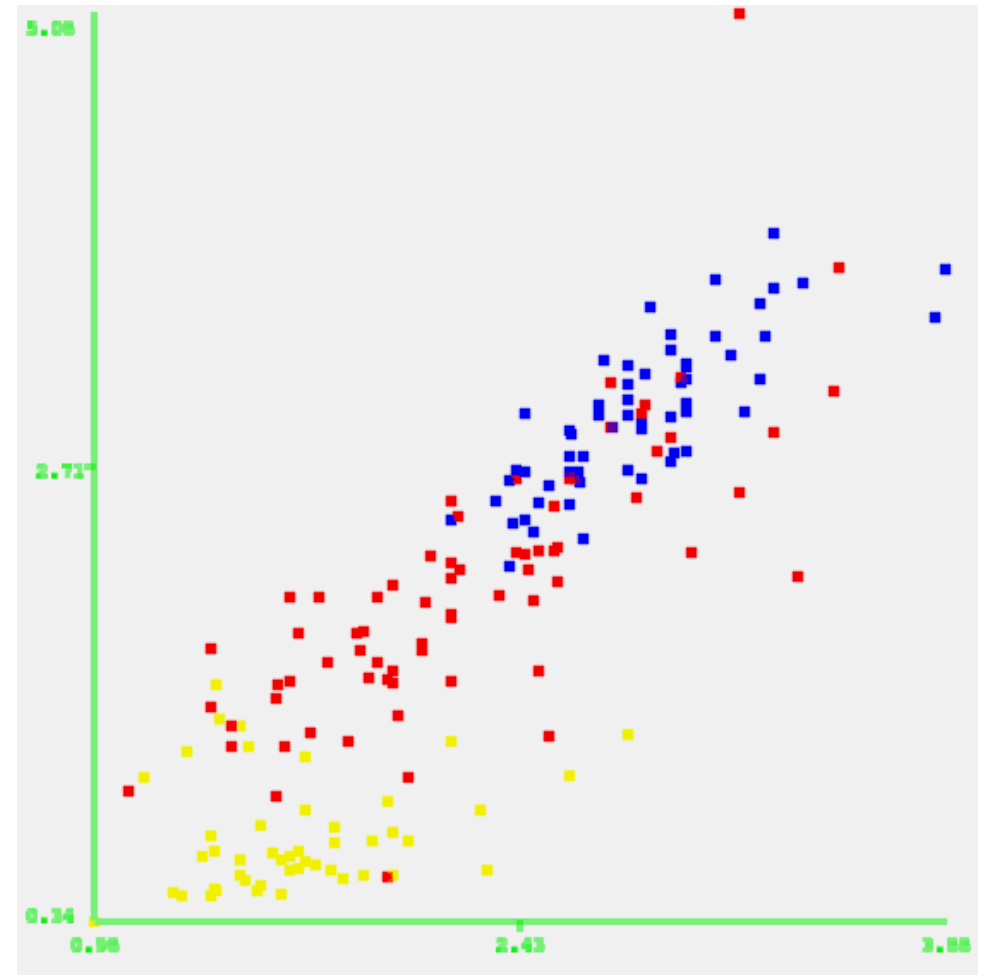


X: Flavonoids, Y: Proline

Conceitos Básicos: Espaço de Atributos

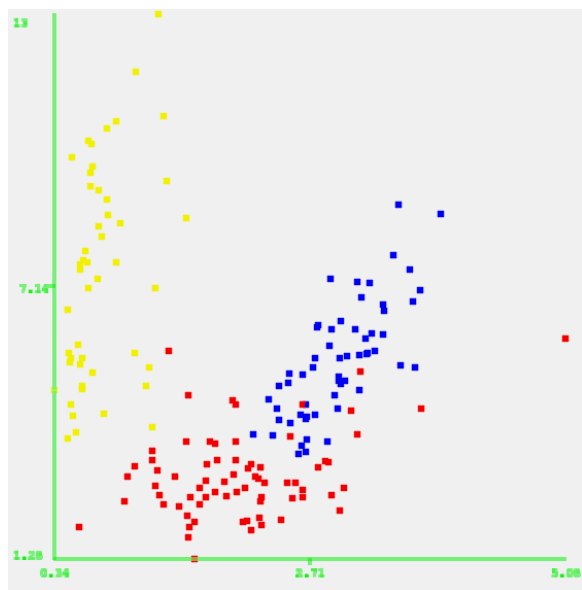


X: Alkalinity of Ash, Y: Magnesium

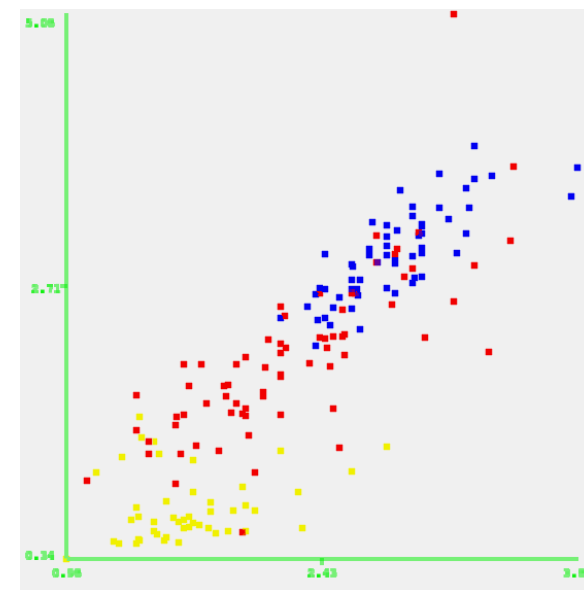


X: Total Phenols, Y: Flavonoids

- Visualização pode mostrar várias informações sobre os dados!
 - Quais atributos permitem separação em classes?
 - Quais atributos são correlacionados?
 - Como é a distribuição das classes (se houver)?
 - Existem estruturas interessantes?

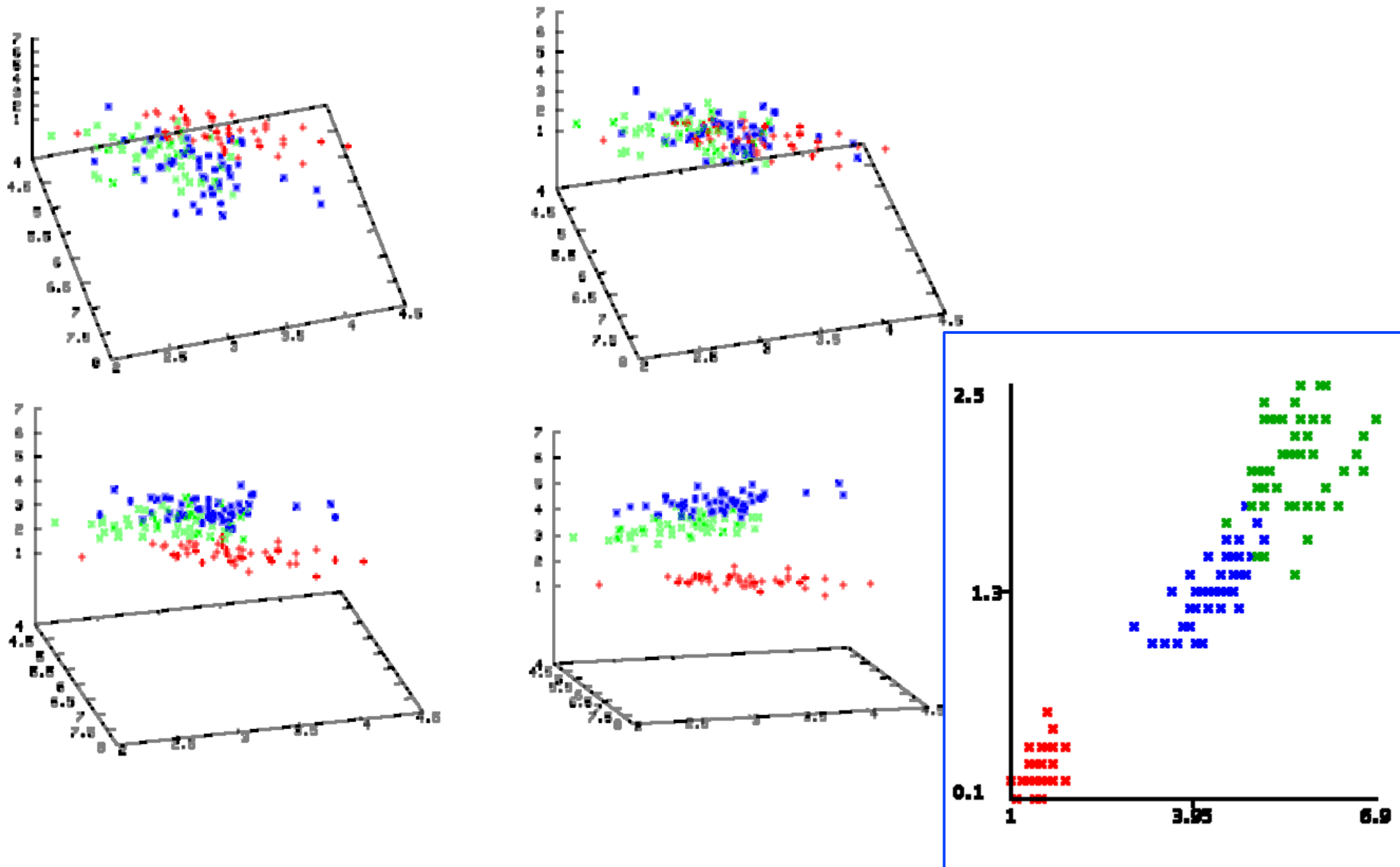


X: Flavonoids, Y: Color Intensity



X: Total Phenols, Y: Flavonoids

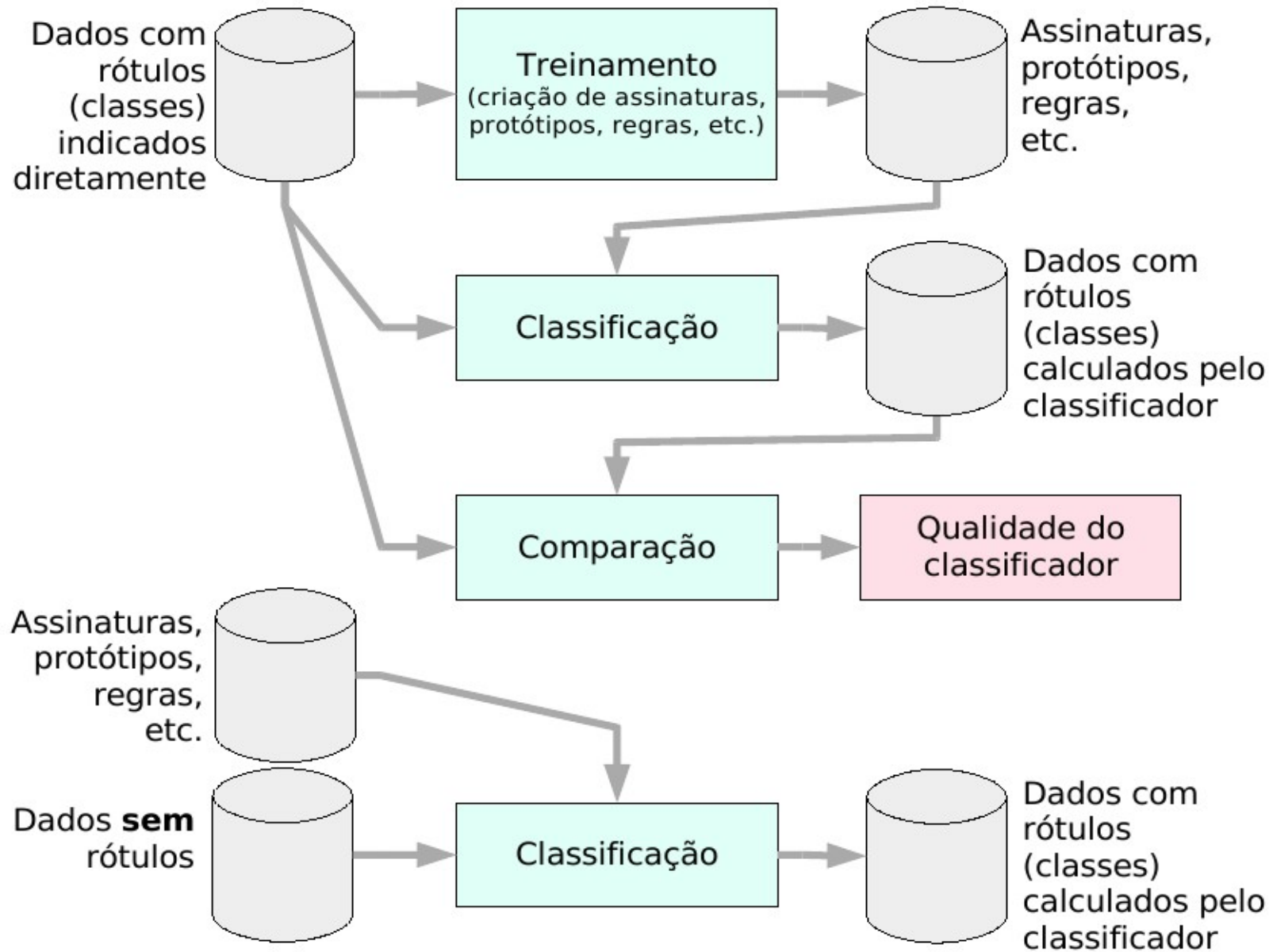
Conceitos Básicos: Espaço de Atributos

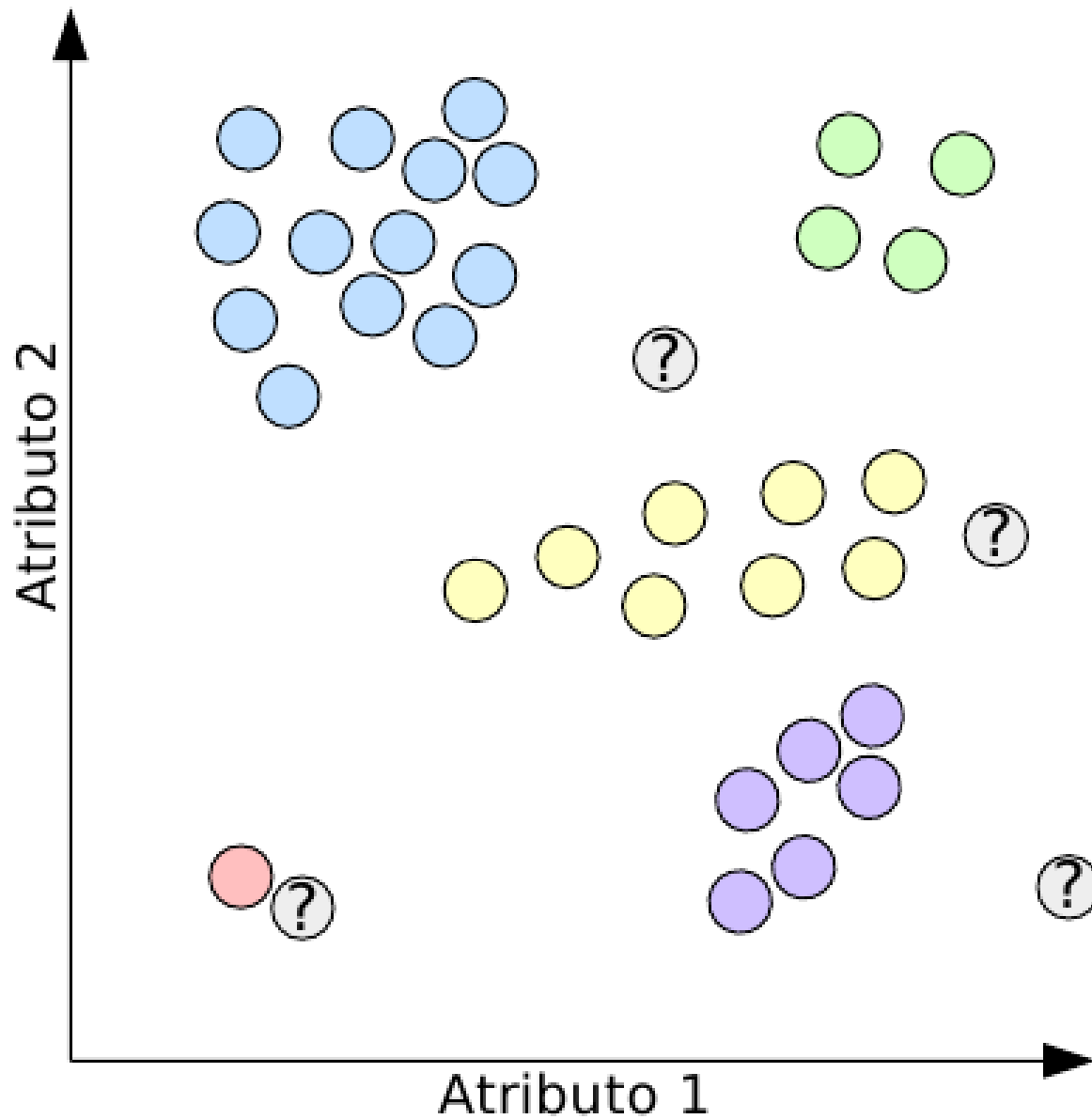


Classificação

- **Predição de uma categoria ou classe discreta.**
- Como entrada: instâncias para as quais as classes são conhecidas.
 - Com isso criamos um **classificador** ou **modelo** (fase de treinamento).
- Como entrada em uma segunda fase, temos vários dados para os quais as classes não são conhecidas.
 - Usamos o classificador para indicar classes para estes dados.
 - Podemos avaliar o modelo classificando instâncias com classes conhecidas.
- **Se temos como rotular instâncias, para que classificar?**

Classificação

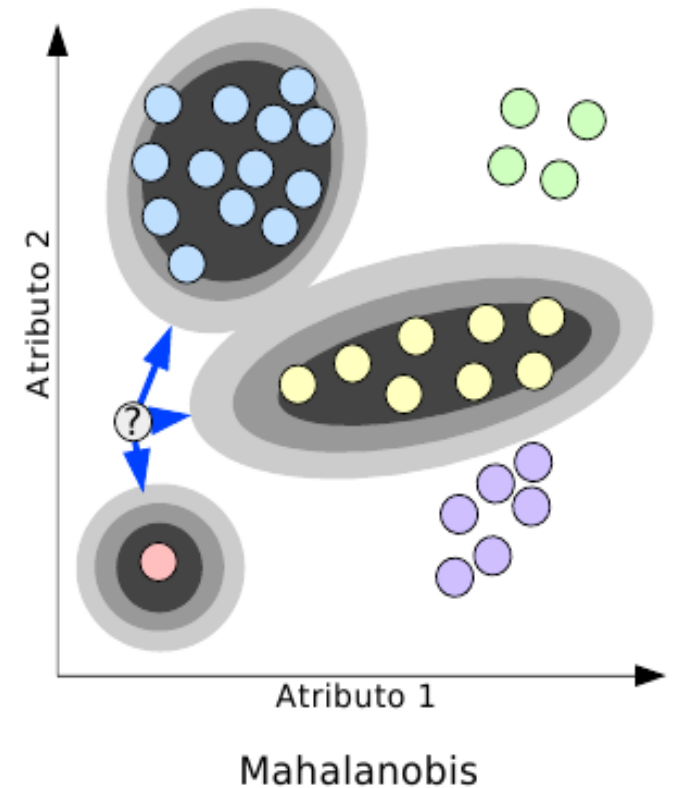
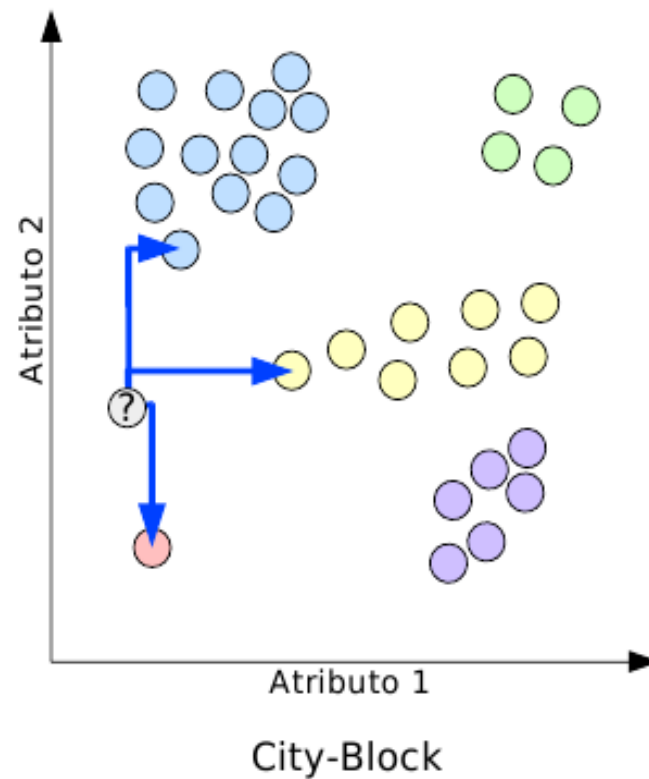
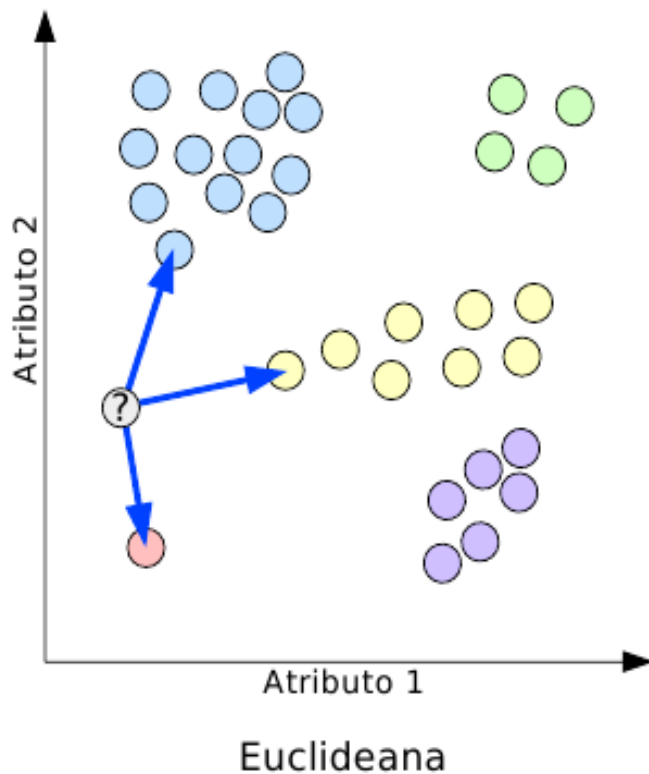




Devemos considerar a possibilidade de empate e/ou rejeição.

- Métodos de classificação supervisionada:
 - Baseados em distâncias e diferenças, usando protótipos ou assinaturas: mínima distância euclideana e variantes.
 - Baseados em separabilidade (entropia): hiperparalelepípedo regular, árvores de decisão e variantes.
 - Baseados em particionamento: redes neurais (*back-propagation*), SVM (*support vector machines*).
 - Baseados diretamente nos dados: vizinhos mais próximos e similares.
- Existe superposição nesta taxonomia...

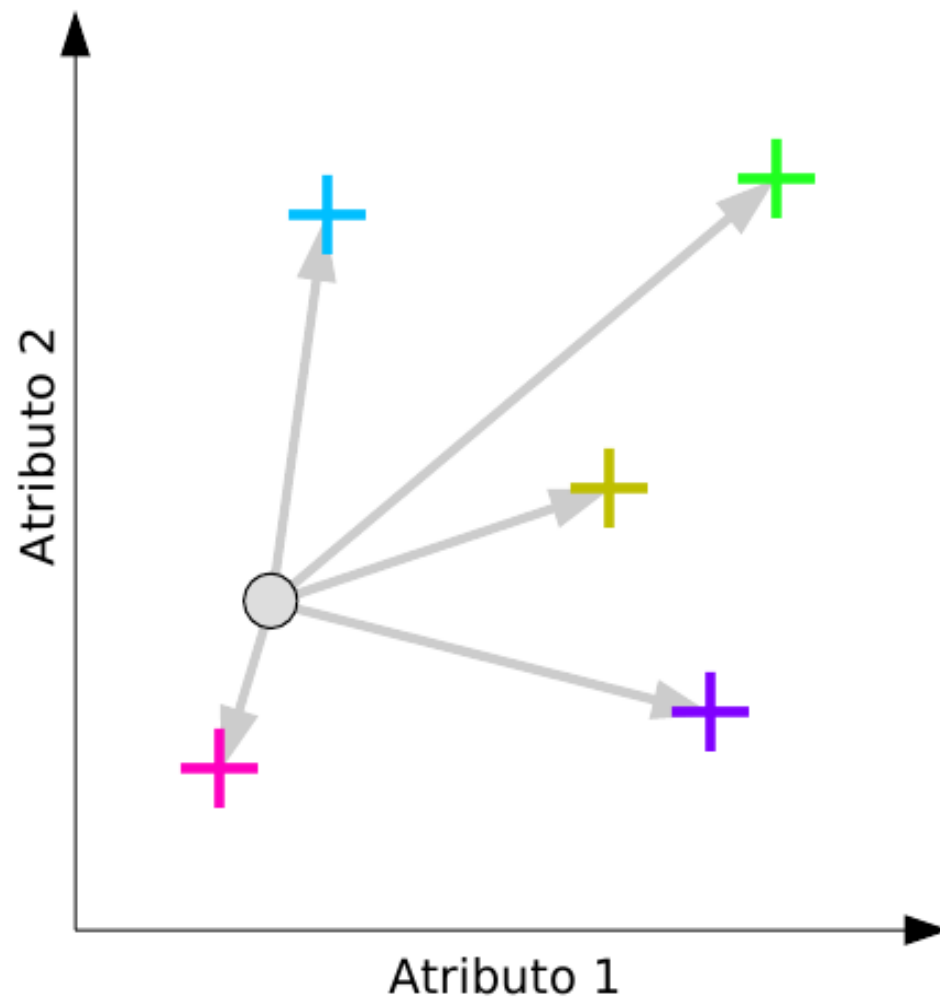
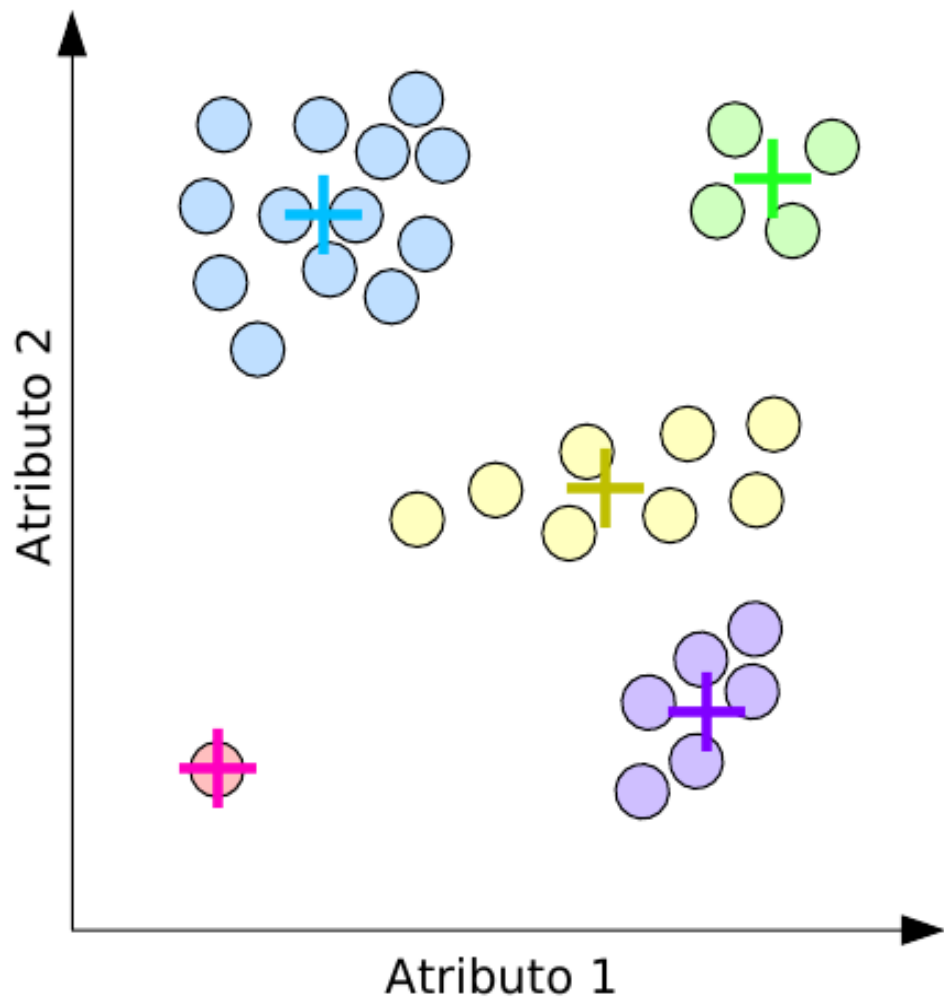
- Medidas de distância



- Como usar atributos não-numéricos?

- Menor distância a protótipo.
 - Mais exatamente: Mínima Distância Euclideana.
- Usa protótipo de uma classe como assinatura.
- Compara atributos de uma instância com os protótipos → o protótipo mais próximo (considerando a distância Euclideana) indica a classe.
- Raramente mostra empate, requer parâmetro adicional para rejeição.

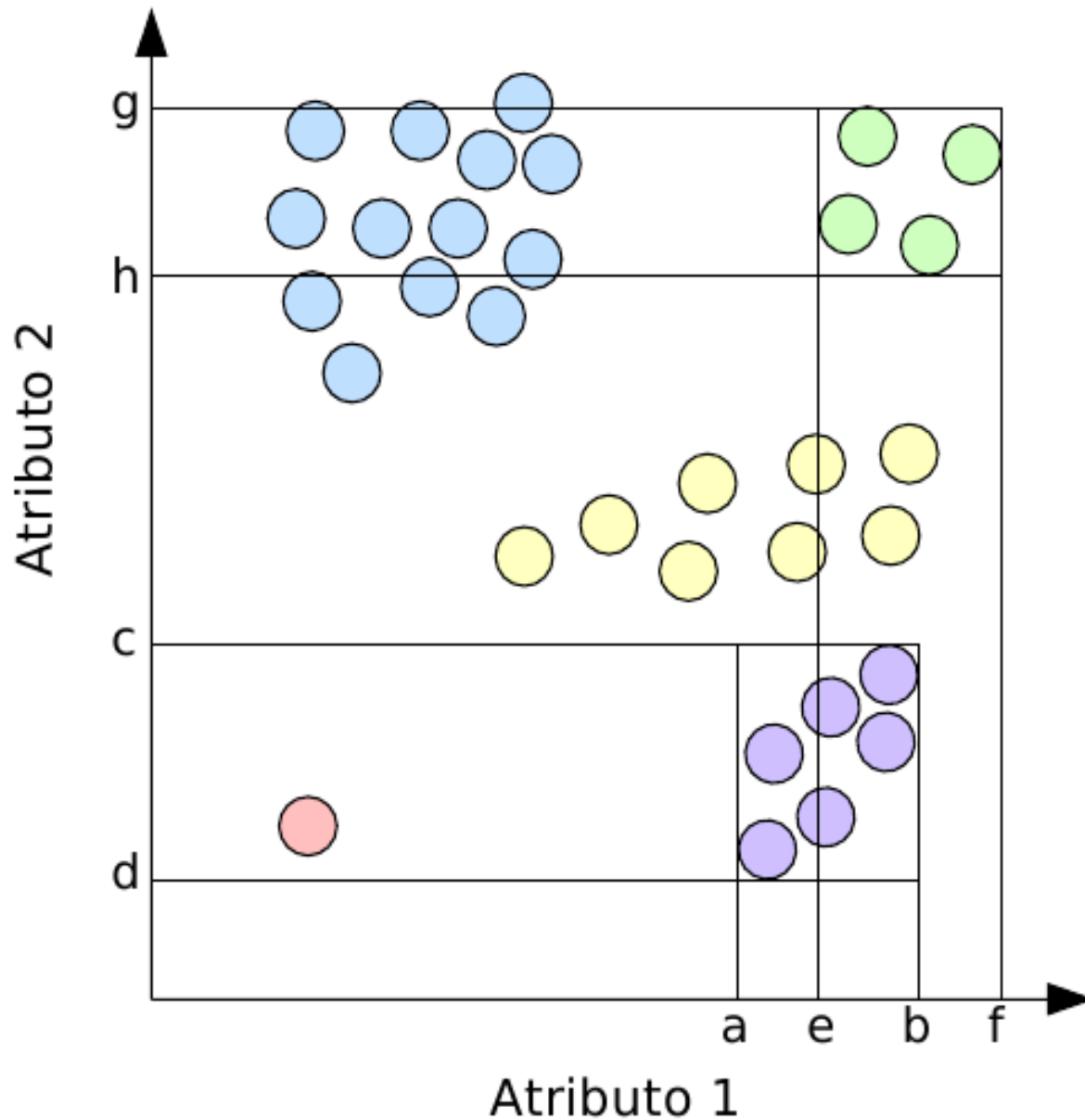
Classificação: Mínima Distância Euclideana



- Vantagens:
 - Simples de implementar.
 - Modelo de simples interpretação.
- Problemas:
 - Distribuição das classes nem sempre (quase nunca?) é hiperesférica.
 - Somente para atributos numéricos.
- Soluções:
 - Modelagem aproximada com mais de um protótipo.
 - Medidas de distância para outros tipos de atributos podem ser usadas...
 - ...mas como criar estas medidas?

- Método do hiperparalelepípedo regular:
- Usa limiares ou extremos de cada classe como assinaturas.
- Classificação simples, permite rejeição.
- Atributos nominais e ordinais podem ser usados diretamente.
- Interoperabilidade com sistemas especialistas e árvores de decisão.
- **Pode haver empate!**

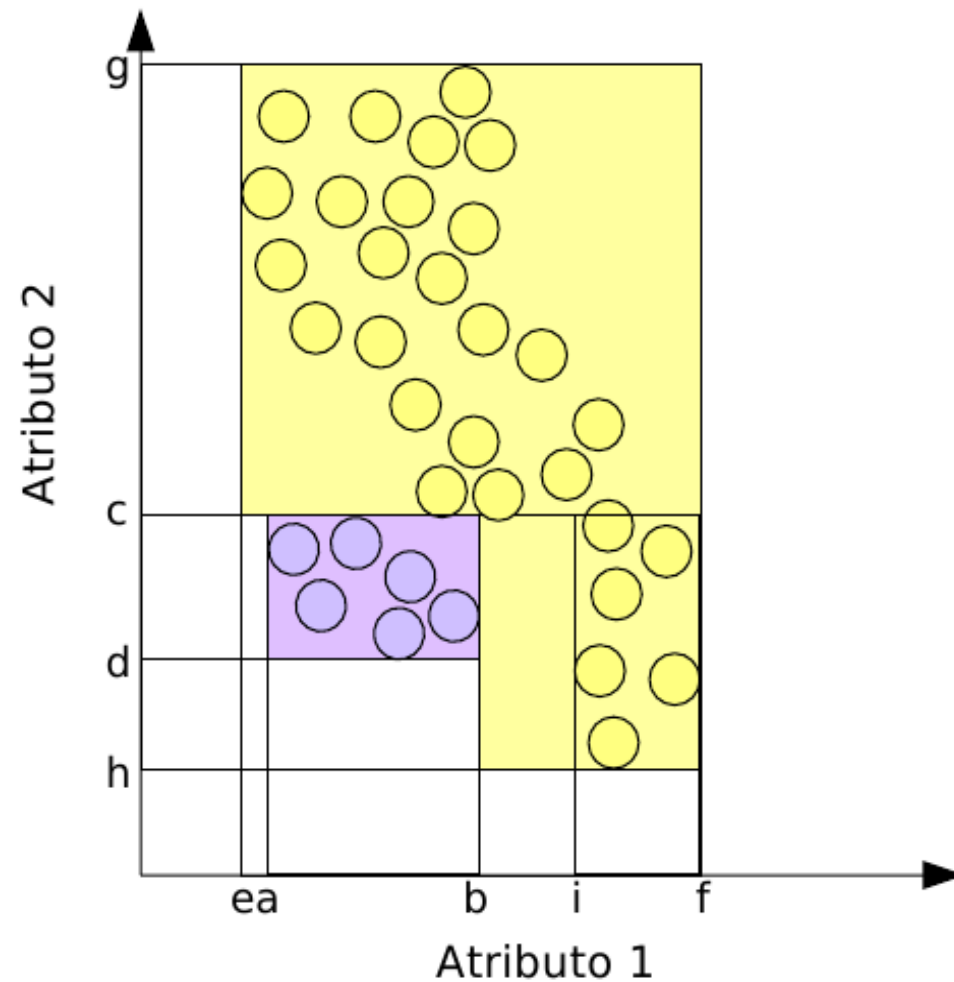
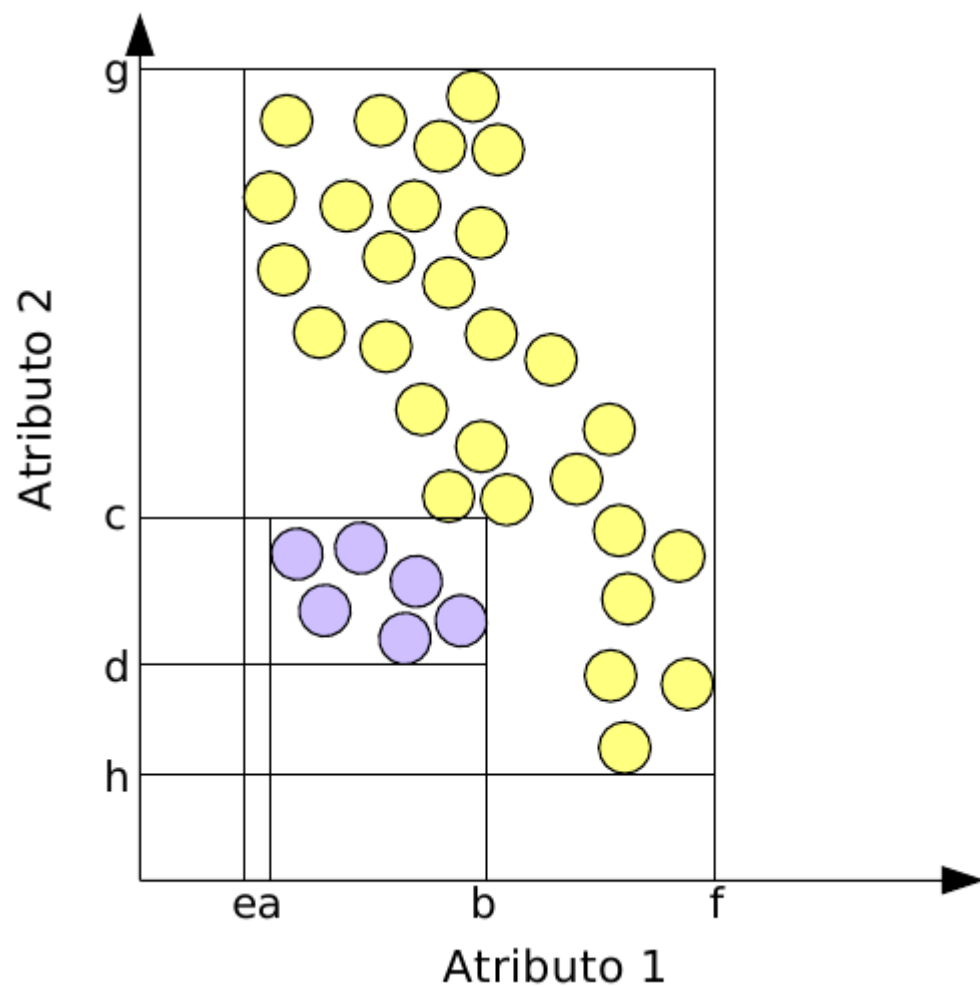
Classificação: Hiperparalelepípedo regular



Se atributo 1 está entre a e b e atributo 2 está entre c e d **então** classe é lilás.

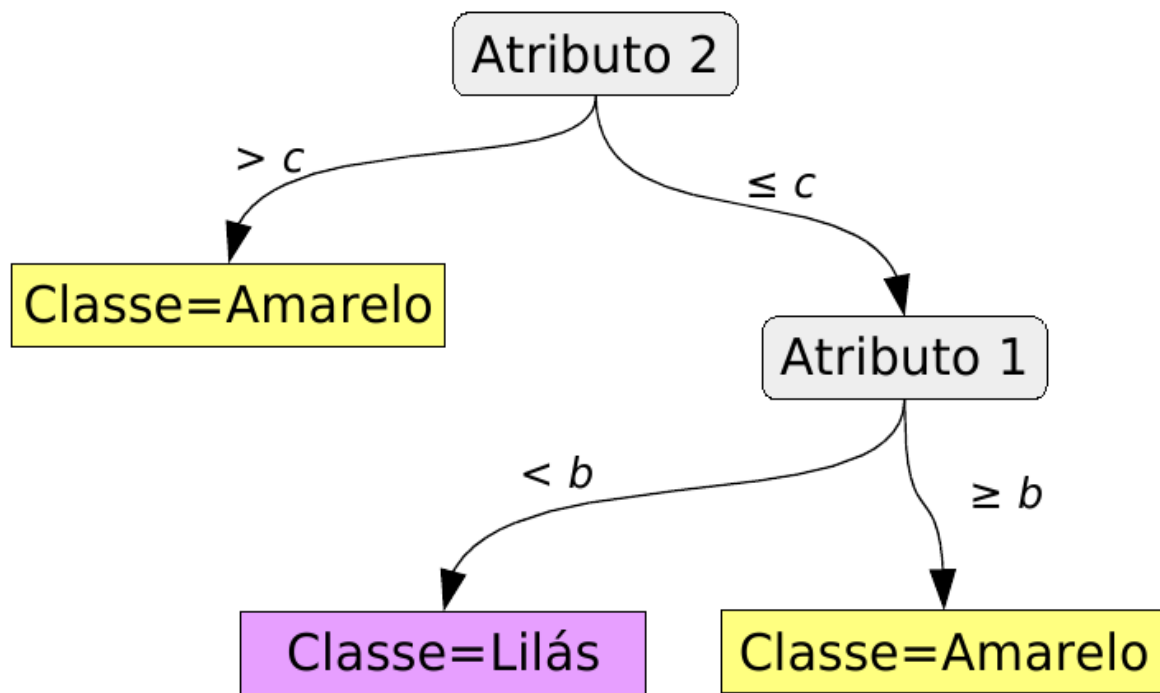
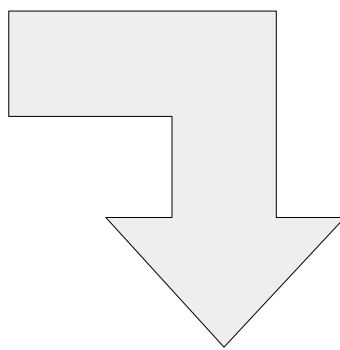
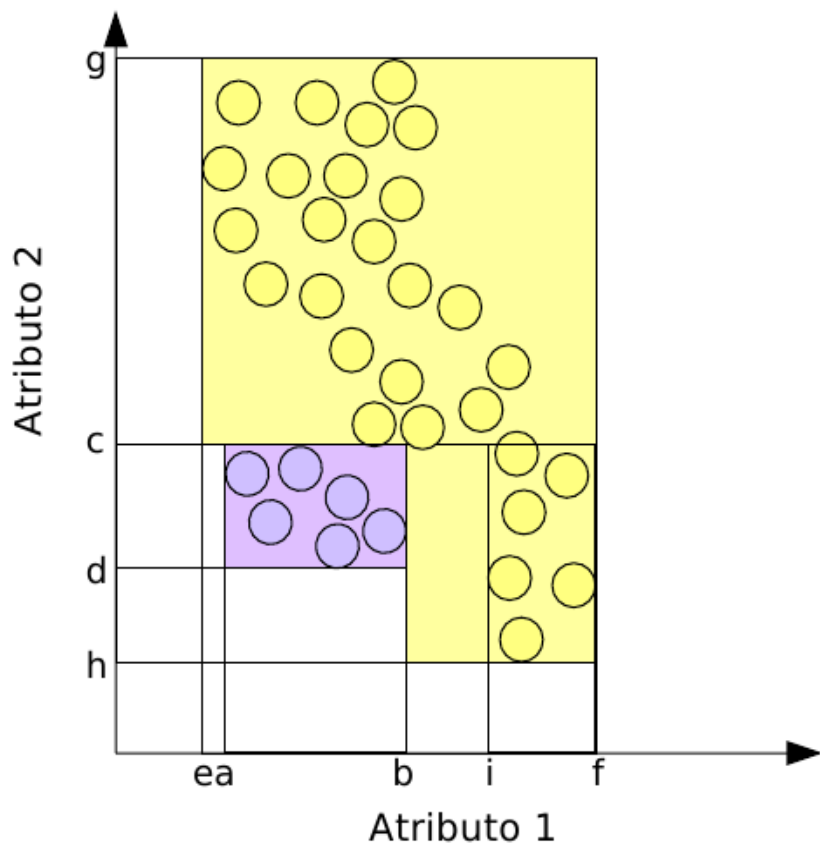
Se atributo 1 está entre e e f e atributo 2 está entre g e h **então** classe é verde.

- Empate



- Vantagens:
 - Simples de implementar.
 - Modelo de simples interpretação.
- Problemas:
 - Influenciável por casos extremos.
 - Cortes ortogonais nos valores dos atributos.
- Soluções:
 - Filtragem de casos extremos é simples.
 - Cortes não-ortogonais possíveis (PCA, SOM, etc.)...
 - ...mas interpretação complexa!

Classificação: Árvores de decisão



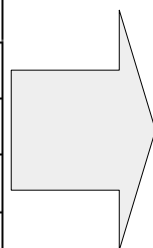
- Para determinar que nós serão criados temos que ter instâncias com classes definidas.
 - Devemos saber também qual é o atributo a ser usado como classe.
- São um conjunto de testes sobre uma base de dados que indica a classe a partir dos valores dos atributos de entrada.
 - Nós em uma árvore de decisão: testes sobre os atributos.
 - Folhas: determinação das classes.
- Muito utilizada por ser facilmente interpretável.
- Semelhança com sistemas especialistas.

Classificação: Árvores de decisão

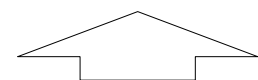


- Criação:
 - Exemplo usando força bruta.

Curso	Esporte	Tipo de comida
computação	futebol	japonesa
computação	natação	fastfood
computação	natação	fastfood
computação	natação	fastfood
matemática	voleibol	italiana
matemática	natação	vegetariana
matemática	voleibol	fastfood
biologia	futebol	fastfood
biologia	futebol	italiana
biologia	futebol	vegetariana
biologia	futebol	italiana
biologia	natação	fastfood



Voleibol		Italiana Fastfood	
Natação	Fastfood Fastfood Fastfood	Vegetariana	Fastfood
Futebol	Japonesa		Fastfood Italiana Vegetariana Italiana
	Computação	Matemática	Biologia



Consequente

- Criação com força bruta:
 - Tabela de Decisão com cada combinação e consequentes.
 - 3 cursos, 3 esportes: 9 células.
 - 8 cursos, 6 esportes, 4 preferências musicais, 3 preferências por filmes: 576 células em quatro dimensões.
 - Cada célula contém mistura de consequentes: como generalizar?

Voleibol		Italiana Fastfood	
Natação	Fastfood Fastfood Fastfood	Vegetariana	Fastfood
Futebol	Japonesa		Fastfood Italiana Vegetariana Italiana
	Computação	Matemática	Biologia

- Criação mais inteligente:
- Tenta minimizar erros de classificação/agrupamento e número de nós e folhas através do *ganho de informação*.
 - Existe maior ganho de informação → correlação/dependência entre **esporte** e **comida** do que entre **curso** e **comida**.
 - Conseguiremos classificar com menos perda de informação → de forma mais compacta comida a partir de **esporte** do que comida a partir de **curso**.
- Implementada nos algoritmos ID3, C4.5 (J4.8 Weka), C5.0

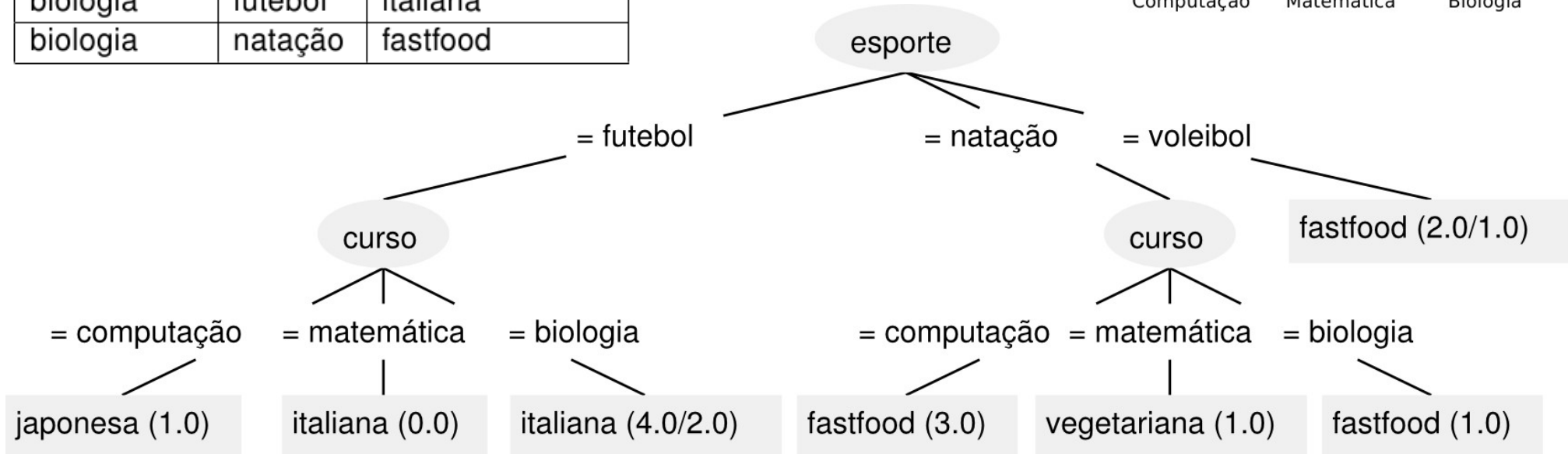
Veja <http://www.lac.inpe.br/~rafael.santos/cap-359.jsp> para uma explicação mais detalhada.

Classificação: Árvores de decisão



Curso	Esporte	Tipo de comida
computação	futebol	japonesa
computação	natação	fastfood
computação	natação	fastfood
computação	natação	fastfood
matemática	voleibol	italiana
matemática	natação	vegetariana
matemática	voleibol	fastfood
biologia	futebol	fastfood
biologia	futebol	italiana
biologia	futebol	vegetariana
biologia	futebol	italiana
biologia	natação	fastfood

Voleibol		Italiana Fastfood	
	Natação	Fastfood Fastfood Fastfood	Vegetariana
Futebol		Japonesa	
	Computação	Matemática	Biologia

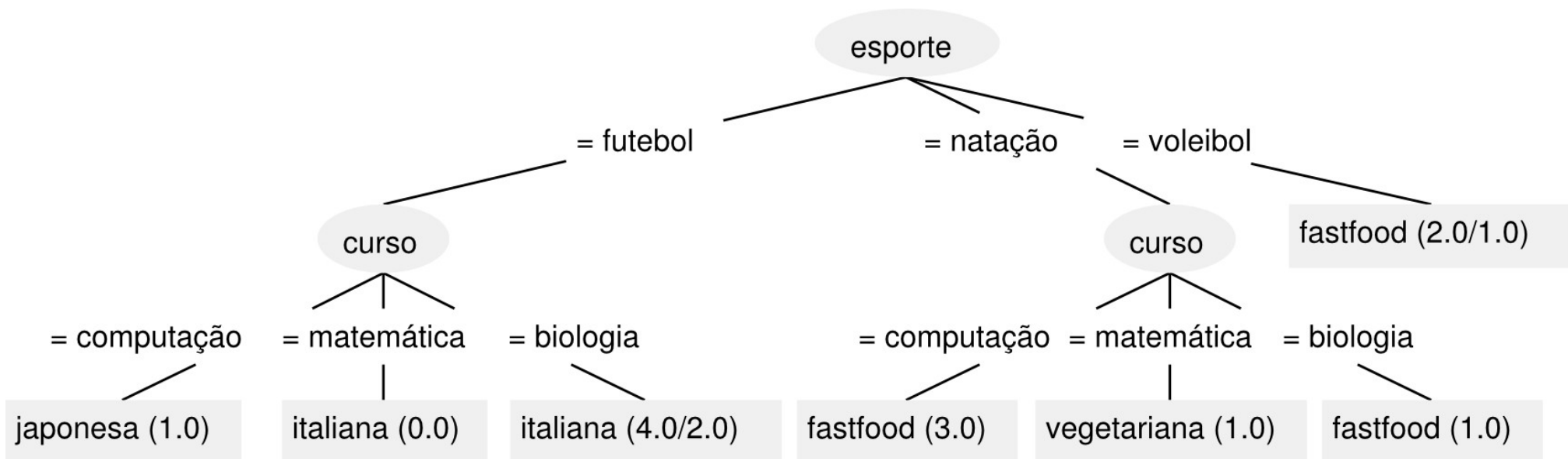


Classificação: Árvores de decisão



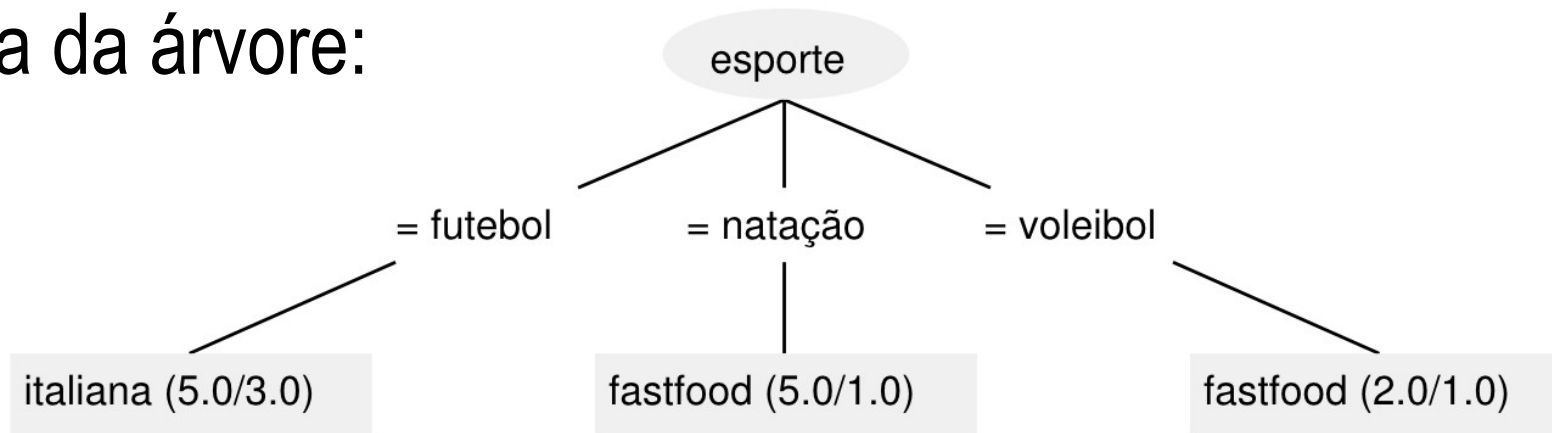
```
esporte = futebol
|  curso = computação: japonesa (1.0)
|  curso = matemática: italiana (0.0)
|  curso = biologia: italiana (4.0/2.0)
esporte = natação
|  curso = computação: fastfood (3.0)
|  curso = matemática: vegetariana (1.0)
|  curso = biologia: fastfood (1.0)
esporte = voleibol: fastfood (2.0/1.0)
```

```
=== Confusion Matrix ===
 a b c d  <-- classified as
 1 0 0 0  | a = japonesa
 0 5 1 0  | b = fastfood
 0 1 2 0  | c = italiana
 0 0 1 1  | d = vegetariana
```



weka.classifiers.trees.J48 -U -M 1

- Poda da árvore:



```

esporte = futebol: italiana (5.0/3.0)
esporte = natação: fastfood (5.0/1.0)
esporte = voleibol: fastfood (2.0/1.0)
    
```

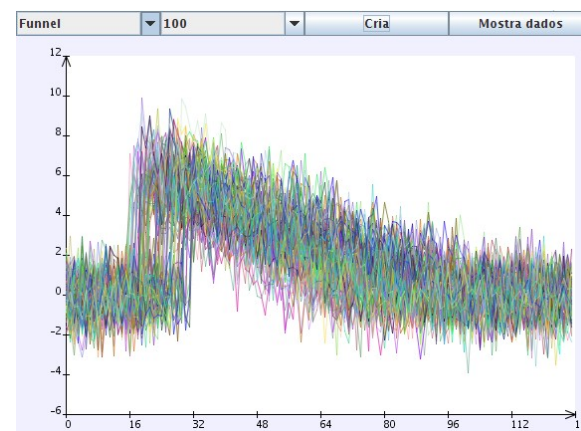
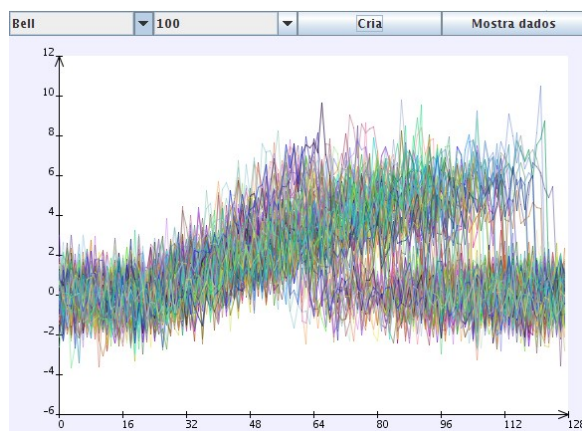
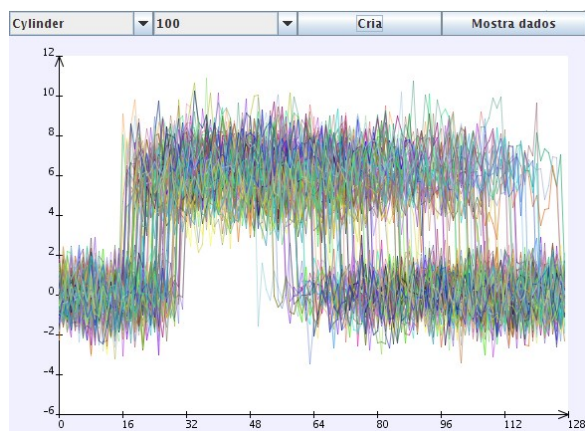
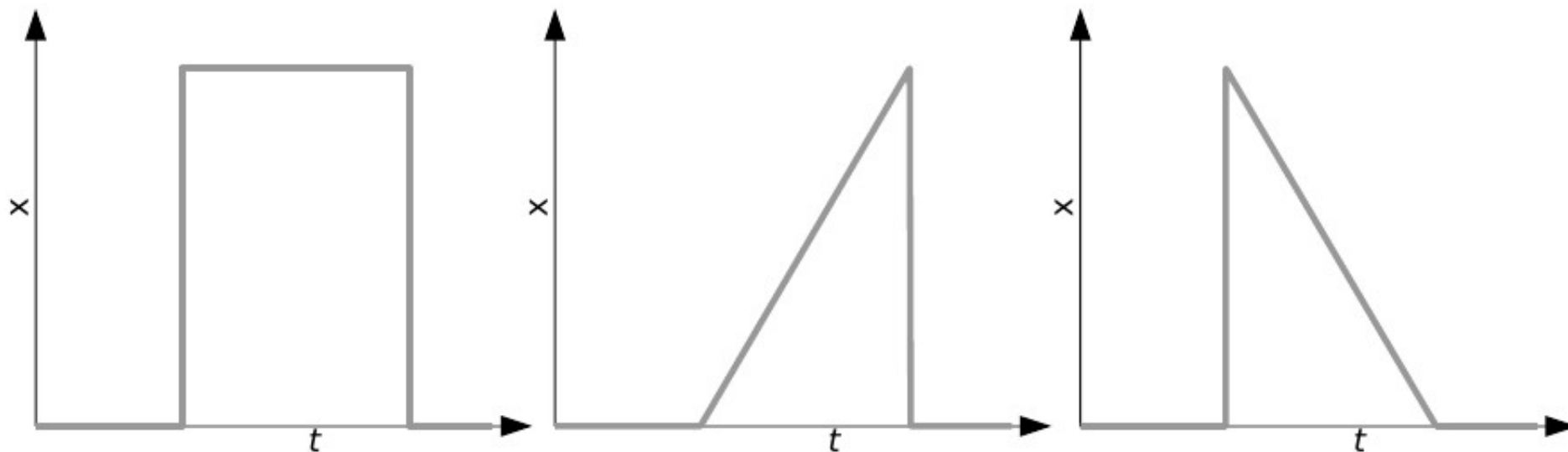
```

=== Confusion Matrix ===
 a b c d  <-- classified as
 0 0 1 0  | a = japonesa
 0 5 1 0  | b = fastfood
 0 1 2 0  | c = italiana
 0 1 1 0  | d = vegetariana
    
```

weka.classifiers.trees.J48 -C 0.7 -M 2

Voleibol		Italiana Fastfood	
Natação	Fastfood Fastfood Fastfood	Vegetariana	Fastfood
Futebol	Japonesa		Fastfood Italiana Vegetariana Italiana
	Computação	Matemática	Biologia

- Outro exemplo: *Cylinder*, *Bell*, *Funnel*

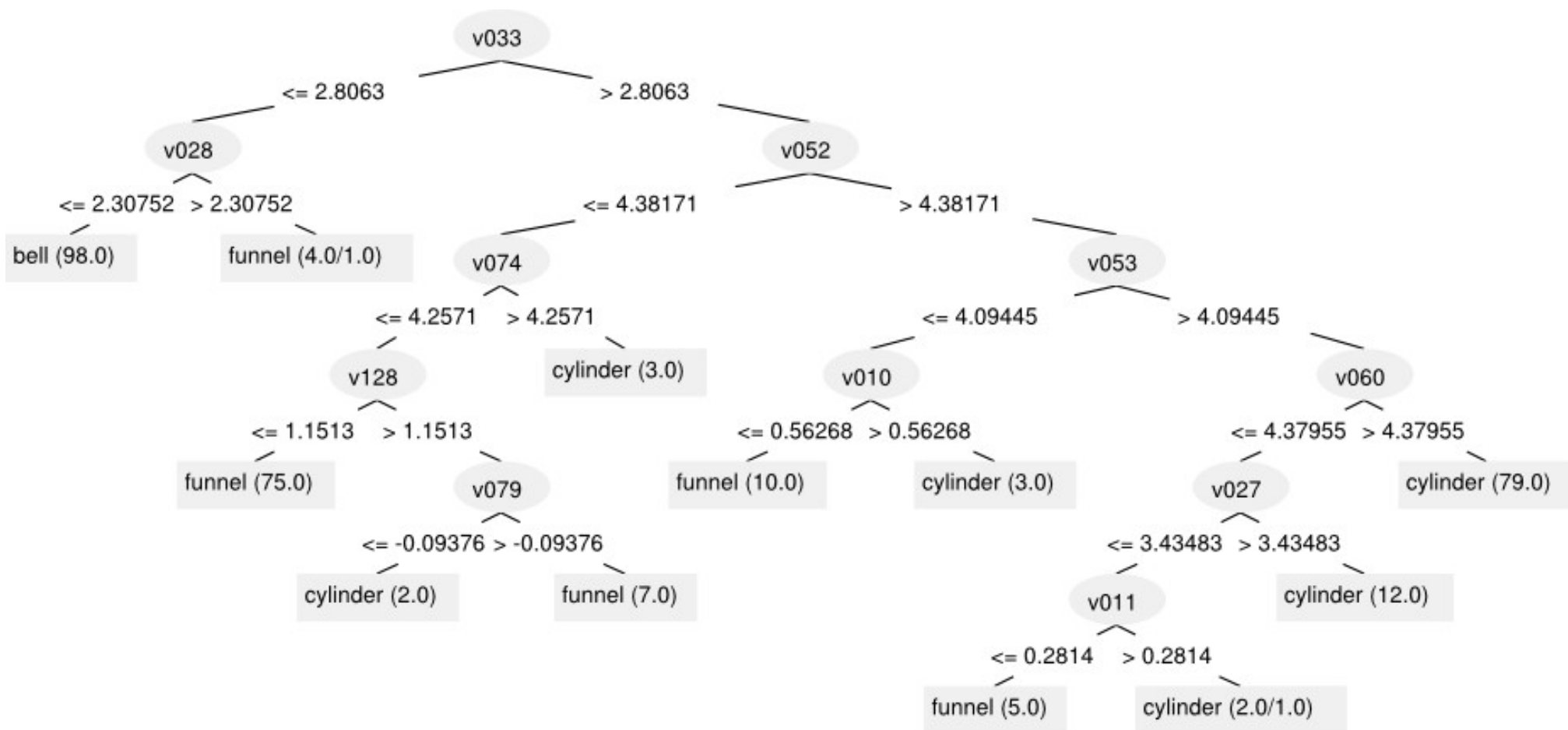


- Outro exemplo: *Cylinder, Bell, Funnel*

```
v033 <= 2.8063
|   v028 <= 2.30752: bell (98.0)
|   v028 > 2.30752: funnel (4.0/1.0)
v033 > 2.8063
|   v052 <= 4.38171
|   |   v074 <= 4.2571
|   |   |   v128 <= 1.1513: funnel (75.0)
|   |   |   v128 > 1.1513
|   |   |   |   v079 <= -0.09376: cylinder (2.0)
|   |   |   |   v079 > -0.09376: funnel (7.0)
|   |   |   v074 > 4.2571: cylinder (3.0)
|   |   v052 > 4.38171
|   |   |   v053 <= 4.09445
|   |   |   |   v010 <= 0.56268: funnel (10.0)
|   |   |   |   v010 > 0.56268: cylinder (3.0)
|   |   |   v053 > 4.09445
|   |   |   |   v060 <= 4.37955
|   |   |   |   |   v027 <= 3.43483
|   |   |   |   |   |   v011 <= 0.2814: funnel (5.0)
|   |   |   |   |   |   v011 > 0.2814: cylinder (2.0/1.0)
|   |   |   |   |   v027 > 3.43483: cylinder (12.0)
|   |   |   |   v060 > 4.37955: cylinder (79.0)
```

```
=== Confusion Matrix ===
  a  b  c  <-- classified as
87  1 12 |  a = cylinder
 1 97  2 |  b = bell
14  2 84 |  c = funnel
```

- Outro exemplo: *Cylinder, Bell, Funnel*

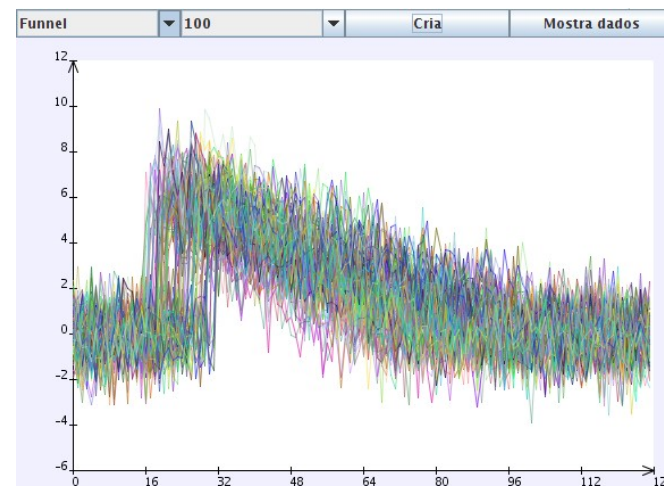
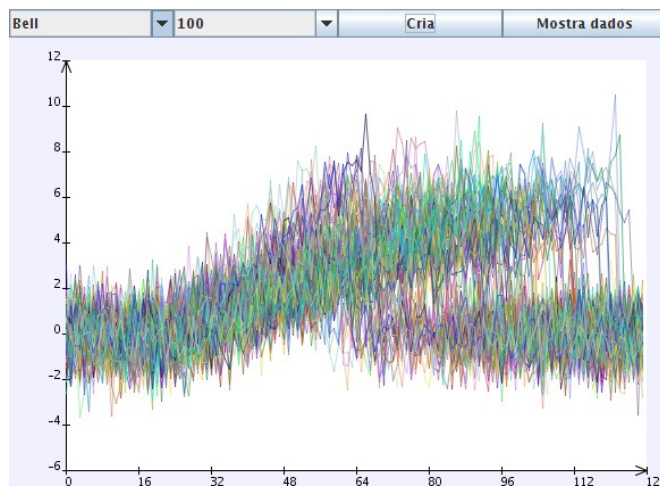
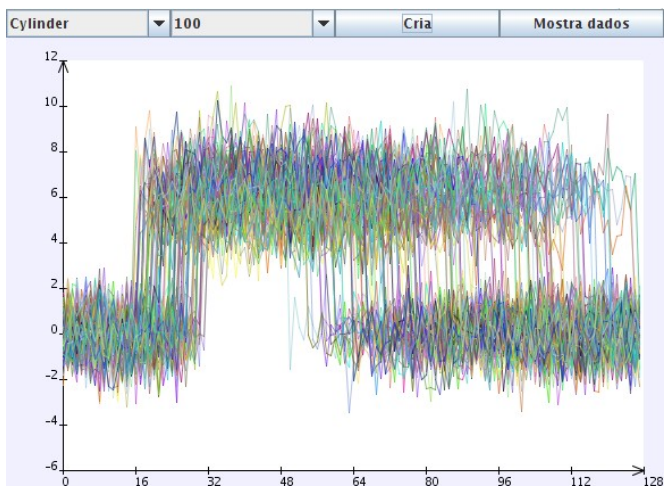


Classificação: Árvores de decisão



```
v033 <= 2.8063: bell (102.0/3.0)
v033 > 2.8063
|   v052 <= 4.38171: funnel (87.0/5.0)
|   v052 > 4.38171
|   |   v053 <= 4.09445: funnel (13.0/3.0)
|   |   v053 > 4.09445: cylinder (98.0/6.0)
```

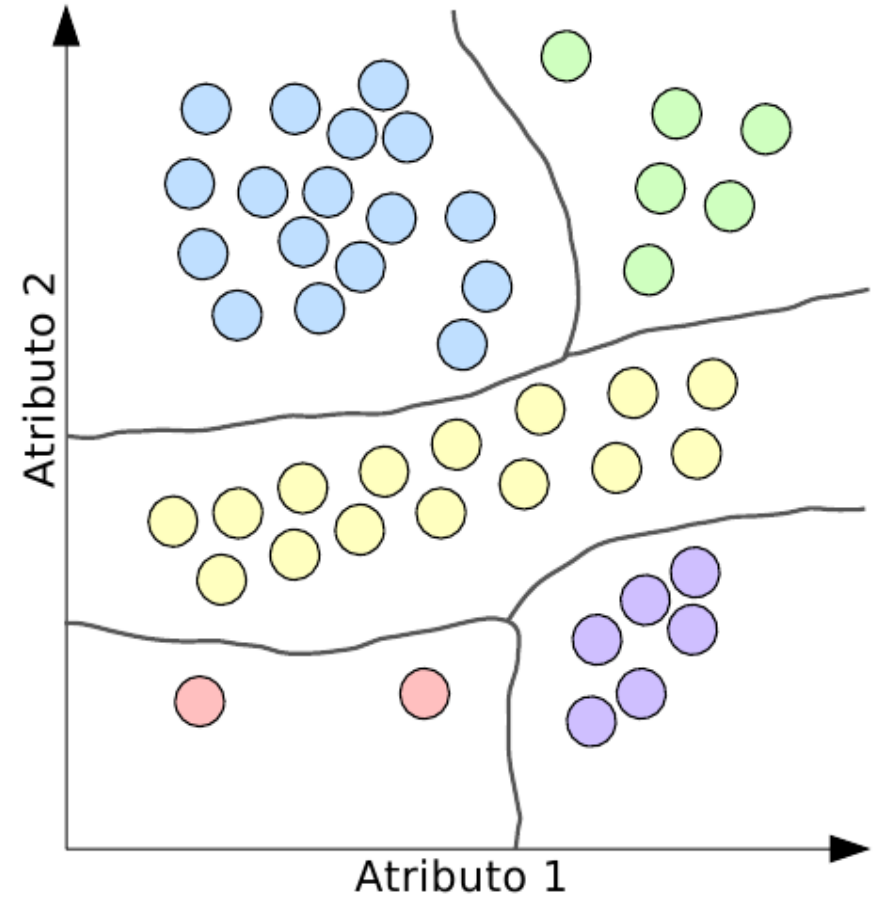
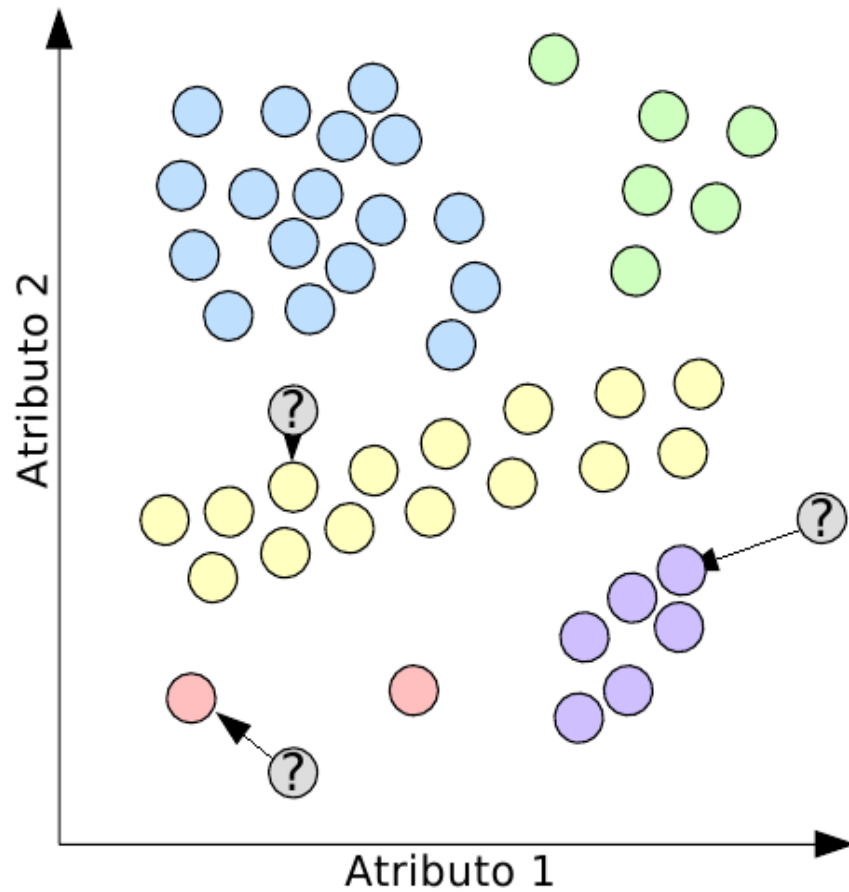
```
=== Confusion Matrix ===
  a  b  c  <-- classified as
86  0 14 |  a = cylinder
 0 98  2 |  b = bell
 7  3 90 |  c = funnel
```



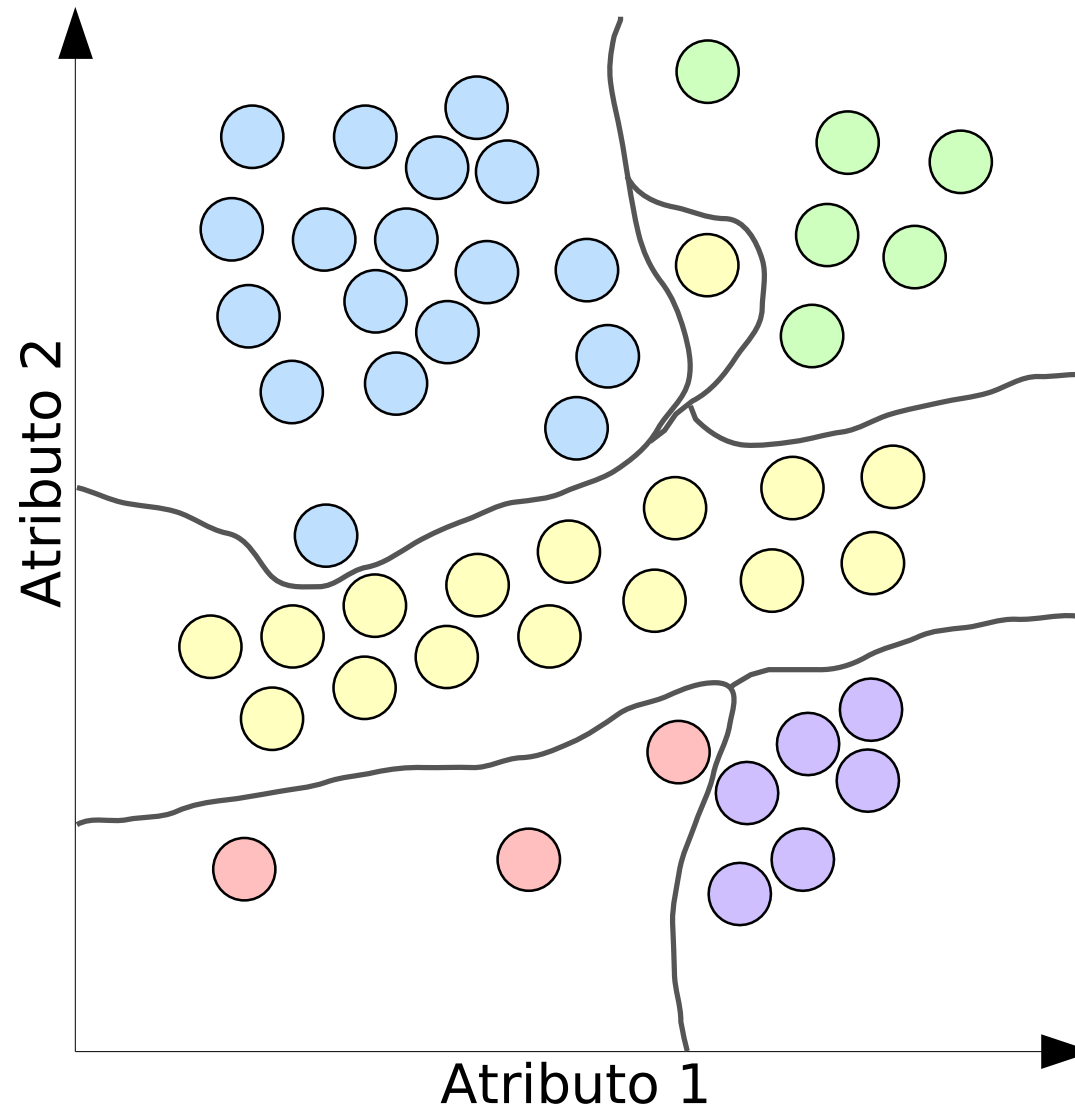
- Vantagens:
 - Modelo de simples interpretação.
 - Possível variar precisão x concisão.
- Problemas:
 - Estrutura (*ordem*) da árvore depende dos dados.
 - Em muitos casos a árvore pode ser extensa!
- Soluções:
 - Análise da árvore é passo de mineração de dados (avaliação do modelo).
 - Como/quando/onde podar?

- Bastante intuitivo: se uma instância de classe desconhecida estiver bem próxima de uma de classe conhecida, as classes devem ser as mesmas.
 - Proximidade **sempre** no espaço de atributos!
- Não criamos protótipos ou assinaturas: usamos as próprias instâncias.
- Cria hipersuperfícies de separação (conjunto de hiperplanos).

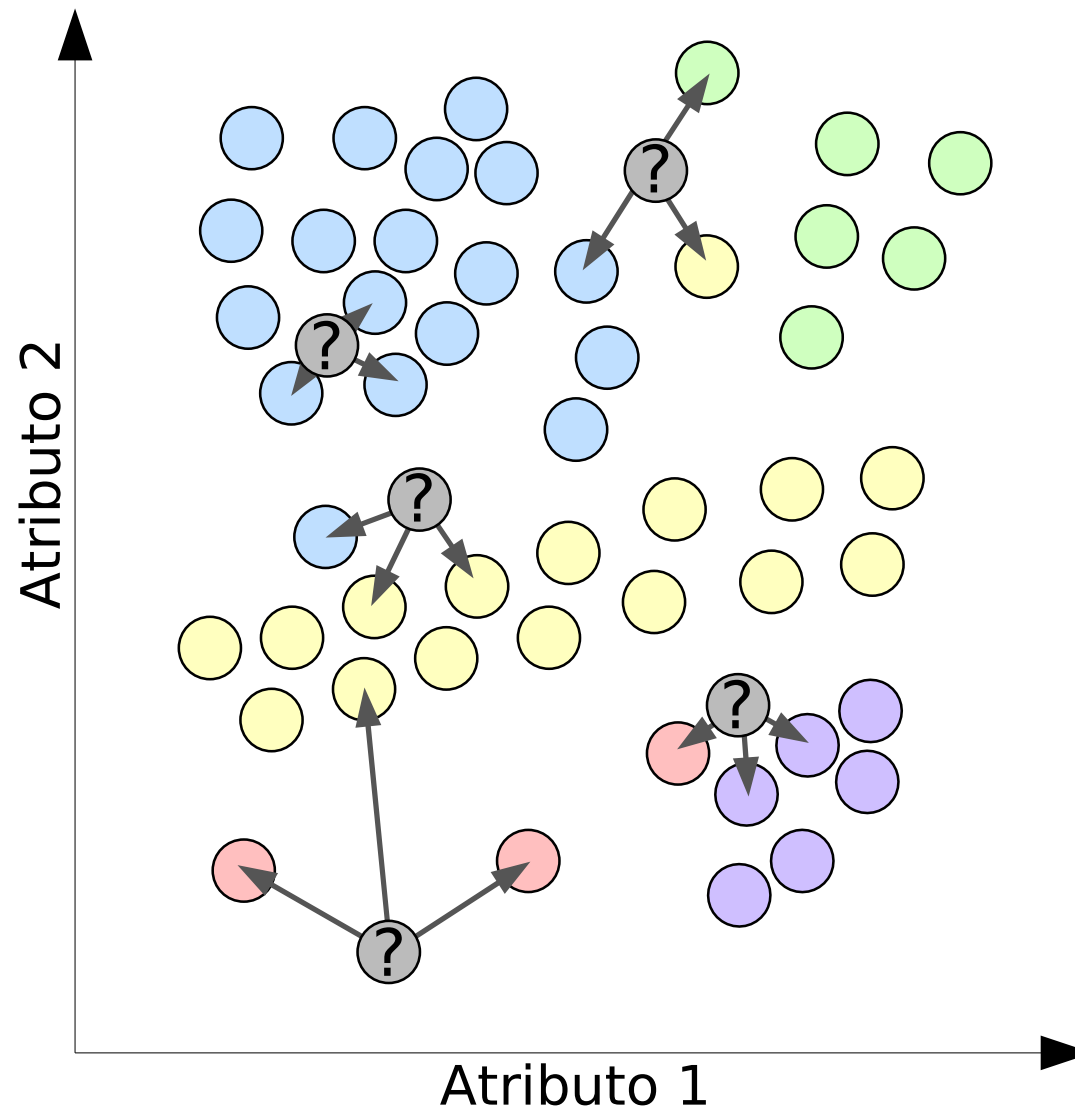
Classificação: Vizinhos mais Próximos



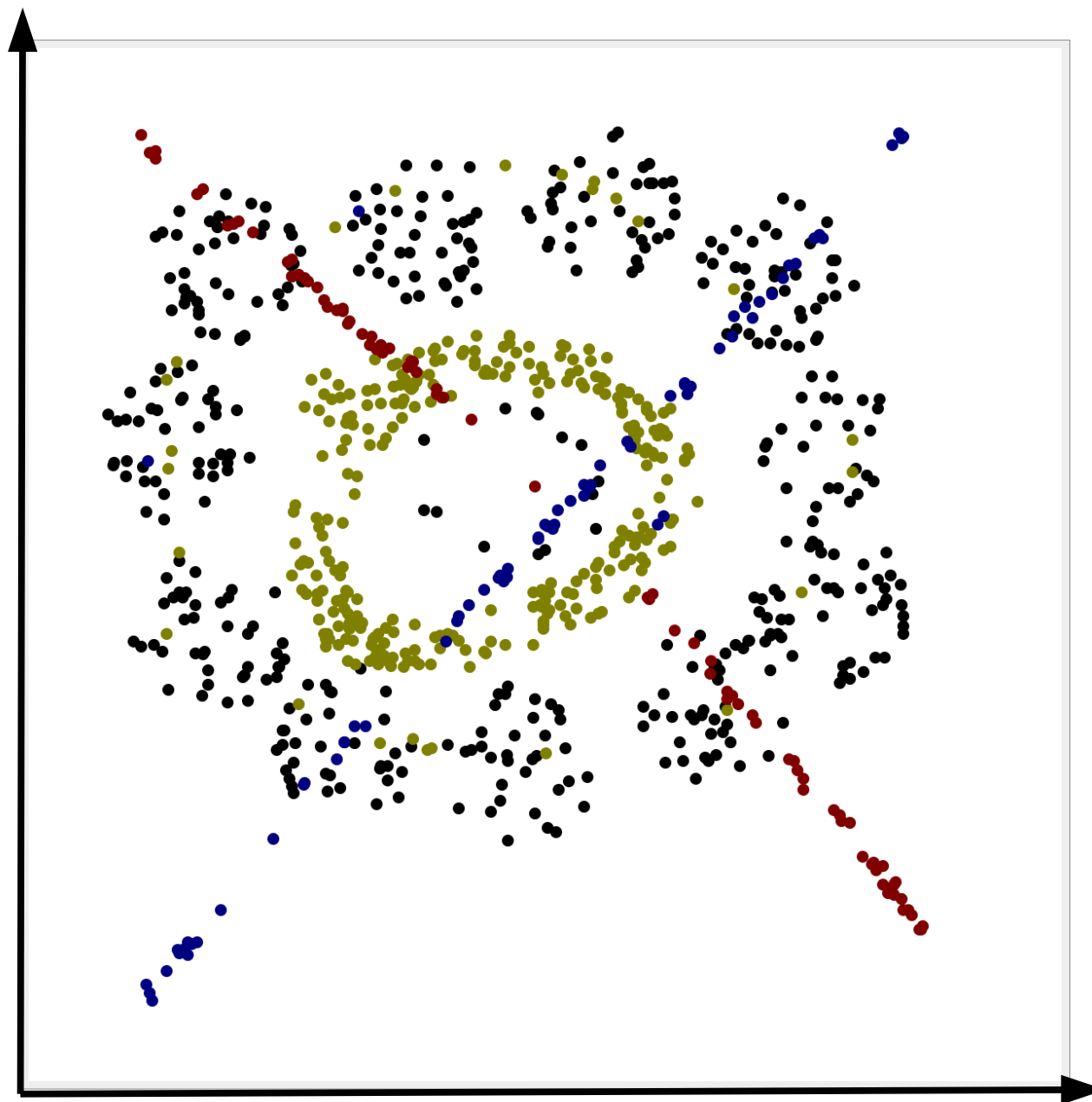
- Problema (?) potencial: *outliers*.



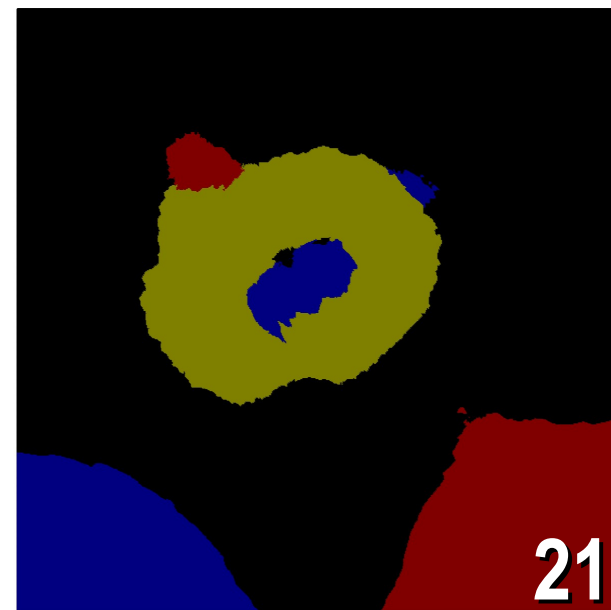
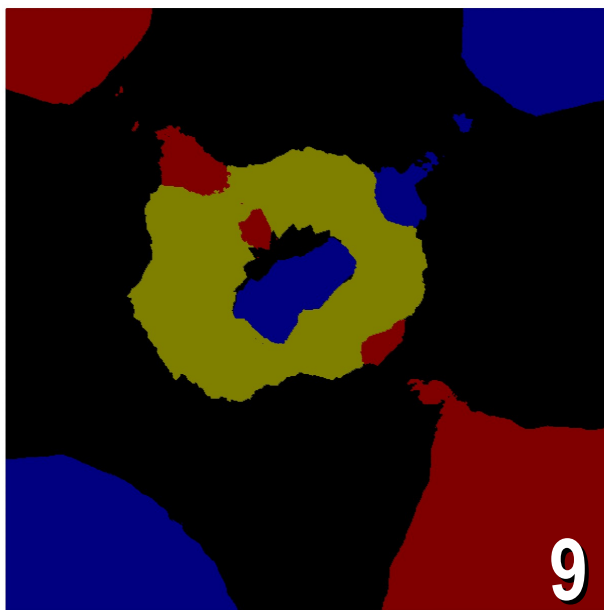
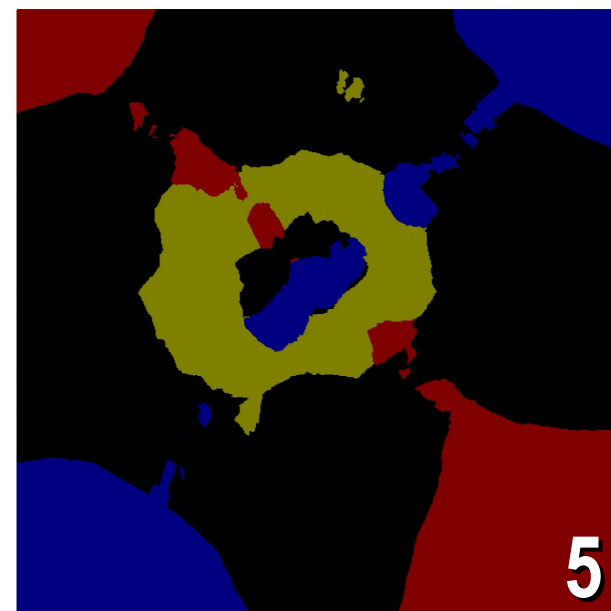
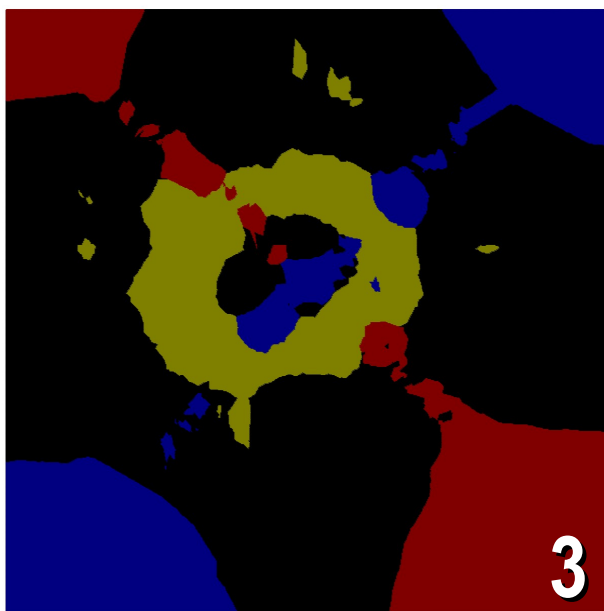
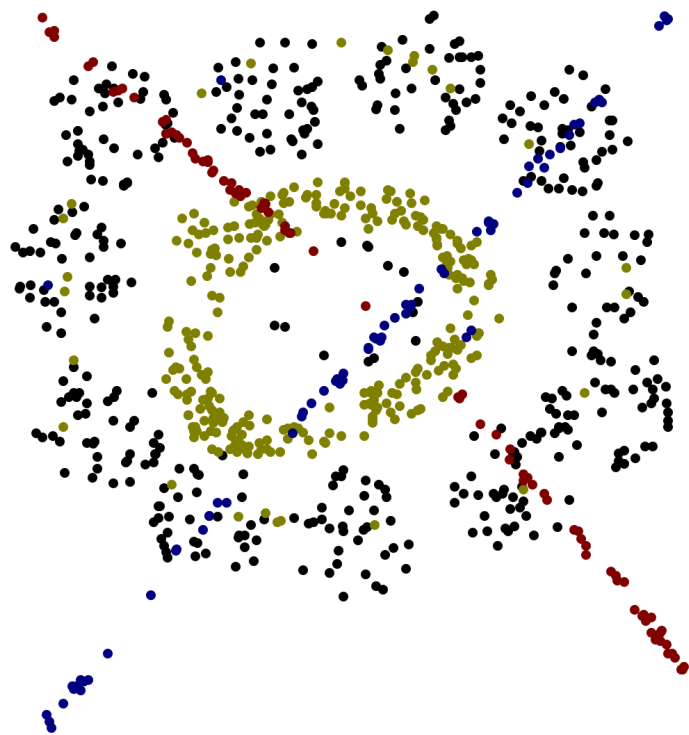
- Solução: usar K vizinhos mais próximos.



Classificação: Vizinhos mais Próximos

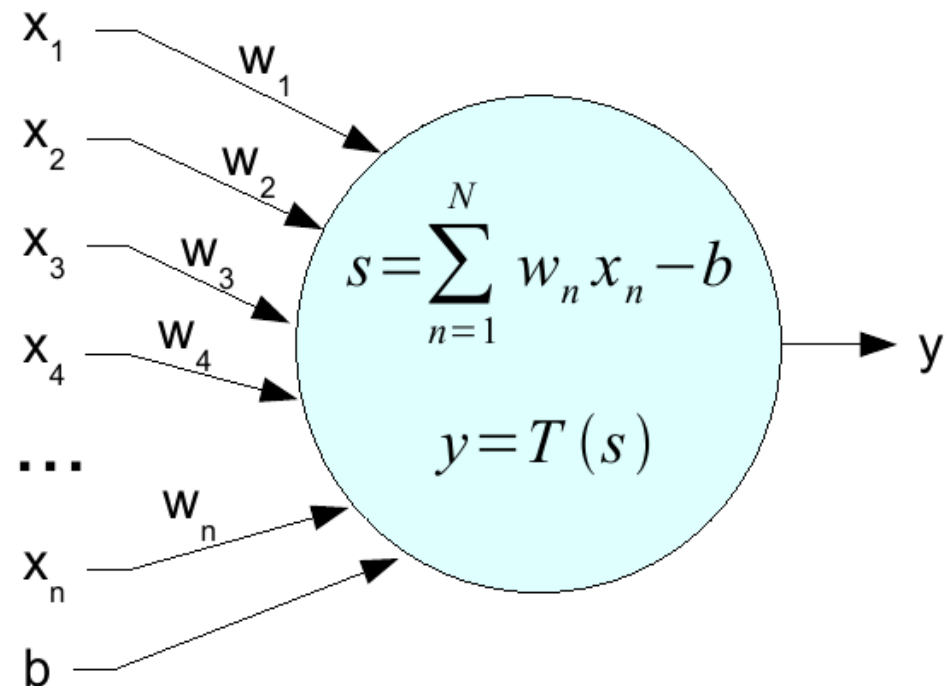


Classificação: Vizinhos mais Próximos

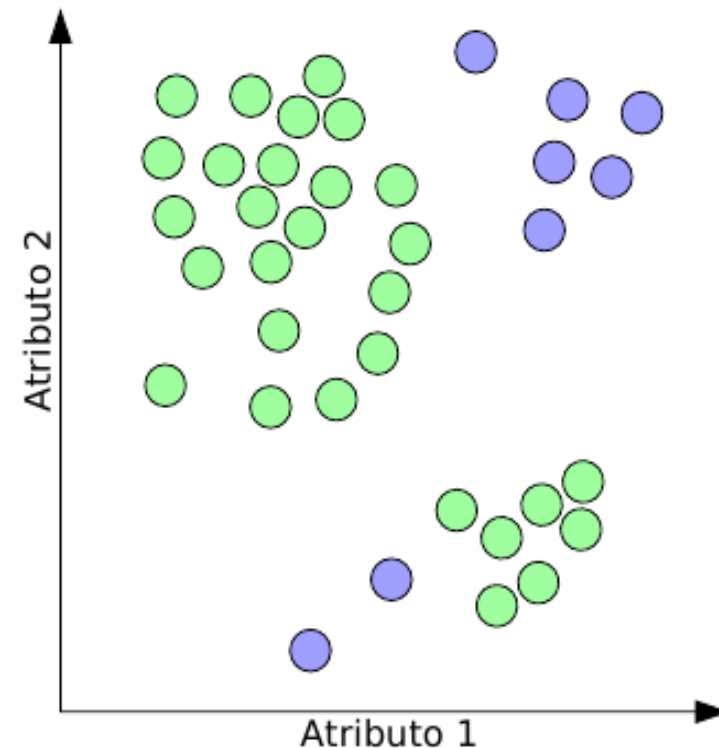
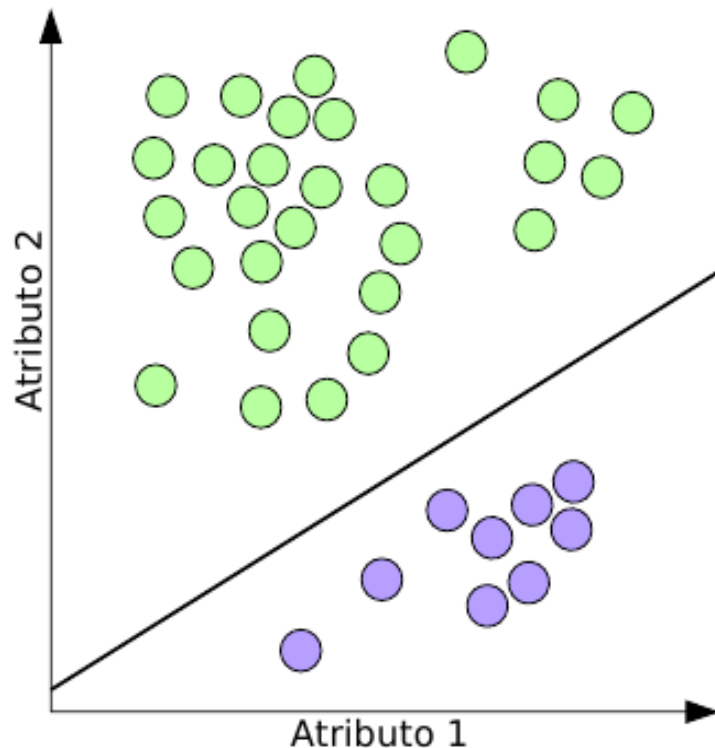


- Vantagens:
 - Dispensa fase de treinamento.
 - Aplicável para classes com qualquer tipo de distribuição (até disjuntas!)
- Problemas:
 - Difícil explicar/interpretar o “modelo”.
 - Complexidade computacional.
 - Influência de *outliers* e *clumps*.
- Soluções:
 - Algoritmos híbridos: redução do número de instâncias para comparação.

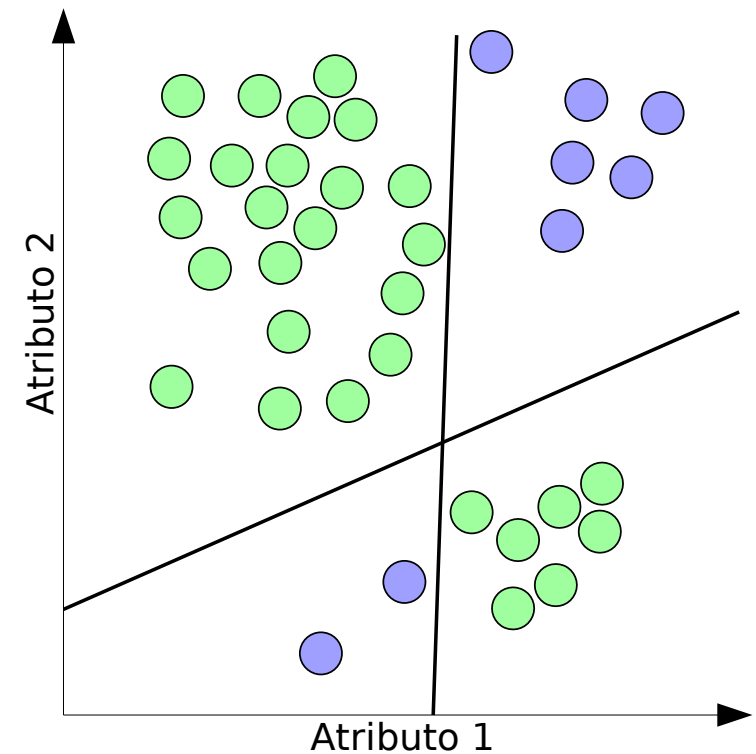
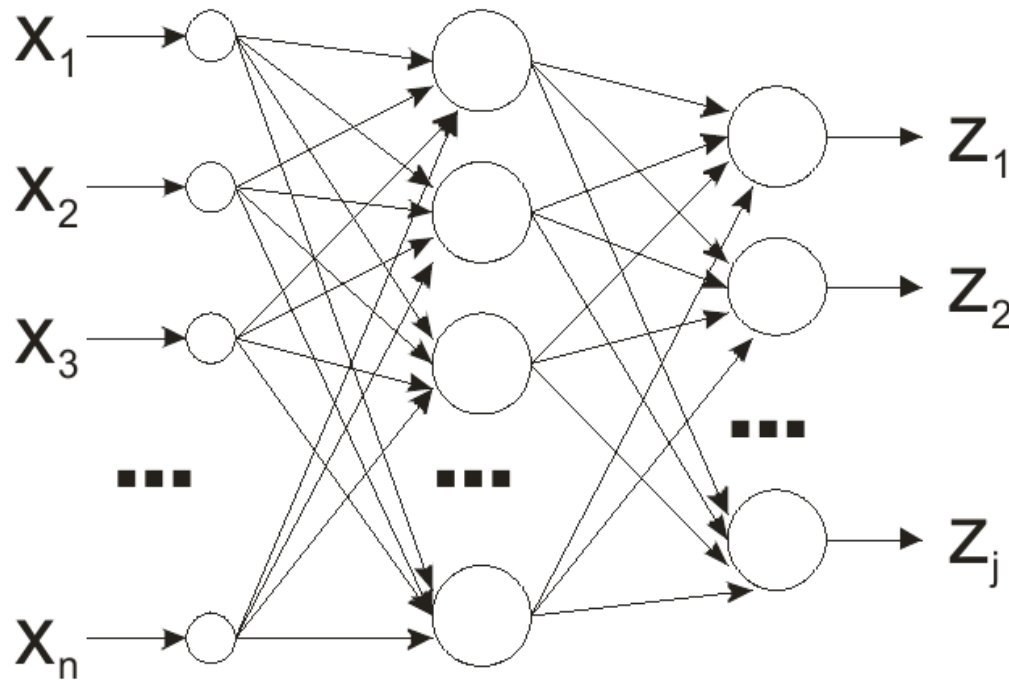
- Este exemplo: Perceptrons em múltiplas camadas.
 - Existem outros modelos e variantes.
- Redes Neurais Artificiais (RNAs ou NNs): algoritmos baseados em simulações simplificadas de neurônios reais.
 - Neurônios processam valores de entrada e apresentam um de saída.
 - Vários neurônios artificiais conectados → rede neural.



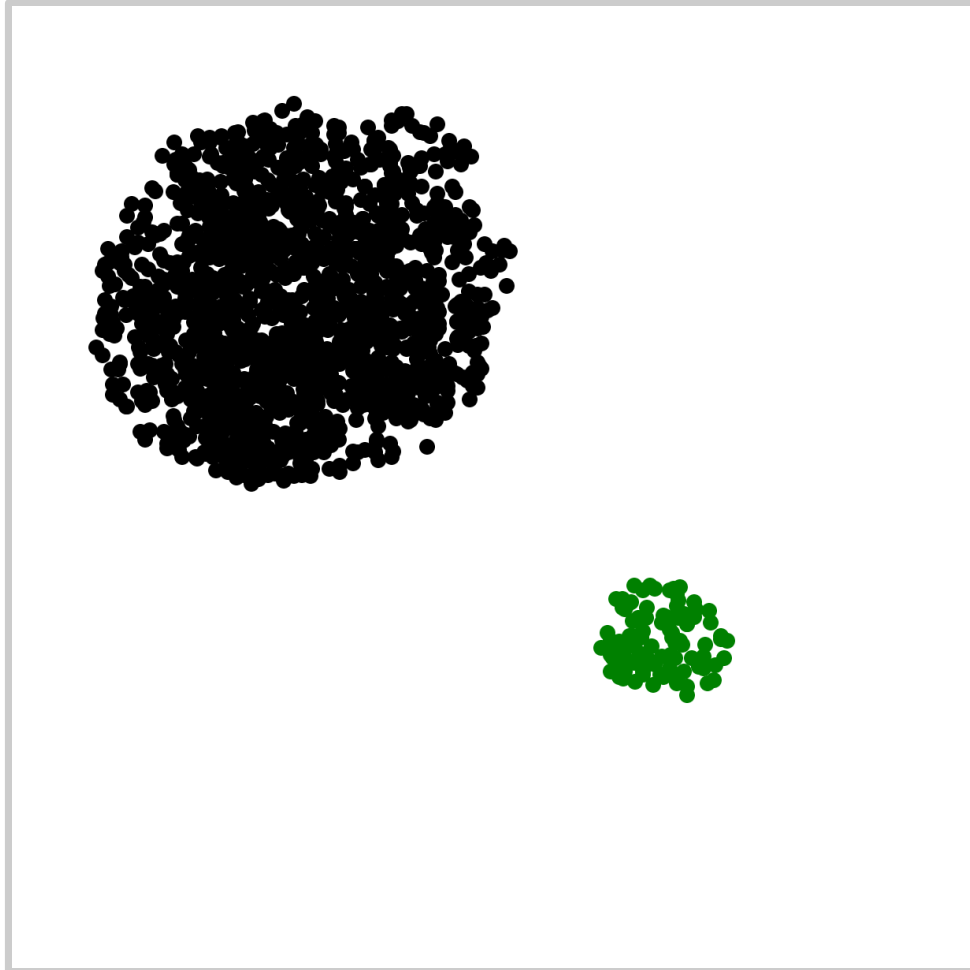
- Um perceptron corresponde a um hiperplano no espaço de atributos:
 - Separa duas classes linearmente separáveis,
 - Não separa mais que duas classes ou faz separações não-lineares.



- Solução: perceptrons podem ser combinados em camadas.
 - Entrada: distribui valores para perceptrons na próxima camada.
 - Camada(s) escondida(s) (*hidden layer*): criam hiperplanos e combinações.
 - Saída: apresenta resultados.

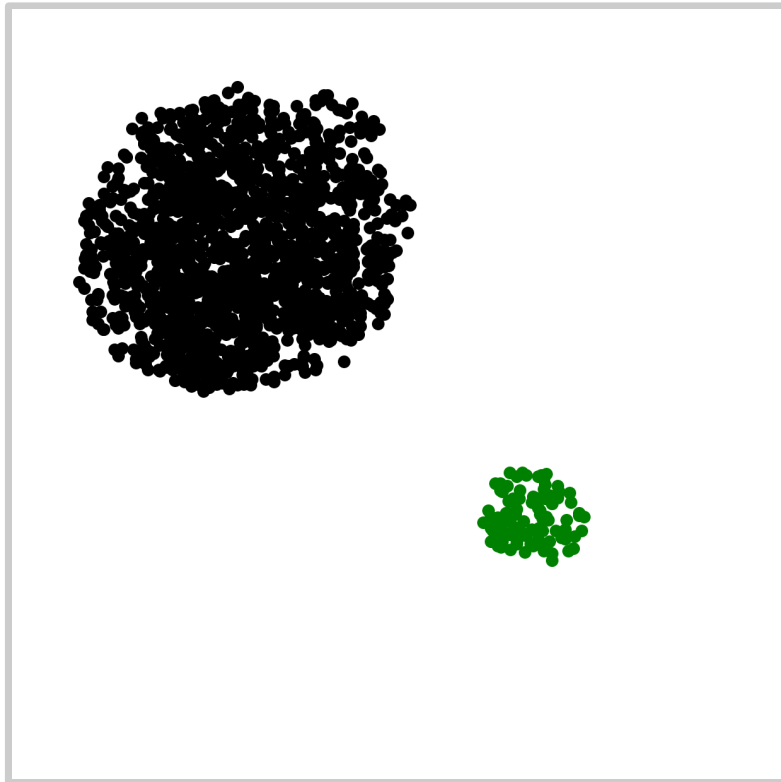


- Vantagens:
 - Capacidade de separar bem classes não linearmente separáveis (com múltiplas camadas).
- Problemas:
 - Difícil explicar/interpretar o “modelo” (caixa preta).
 - Treinamento pode ser complexo (computacionalmente caro), definição da arquitetura também.
- Soluções:
 - Múltiplas arquiteturas e avaliação da qualidade de classificação.



- Primeiro exemplo simples:
 - Duas classes que podem facilmente ser linearmente separadas.
 - Arquitetura 2x2x2:
 - 2 neurônios na camada de entrada (2 atributos).
 - 1 camada escondida com 2 neurônios.
 - 2 neurônios na camada de saída (2 classes).

Classificação: Redes Neurais



Sigmoid Node 0

Inputs	Weights
Threshold	-5.142297502584935
Node 2	6.063964228629336
Node 3	6.148552185386907

Sigmoid Node 1

Inputs	Weights
Threshold	5.1422827635337285
Node 2	-6.111259386964719
Node 3	-6.101248971633012

Sigmoid Node 2

Inputs	Weights
Threshold	2.267842125362997
Attrib atributo1	-4.008925758147538
Attrib atributo2	4.035102089969922

Sigmoid Node 3

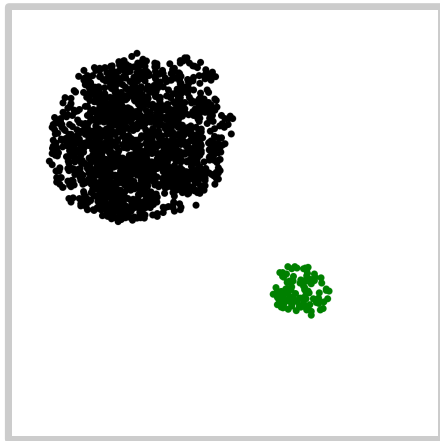
Inputs	Weights
Threshold	2.2790468422497985
Attrib atributo1	-4.031913175764998
Attrib atributo2	4.038136743308941

Class 0

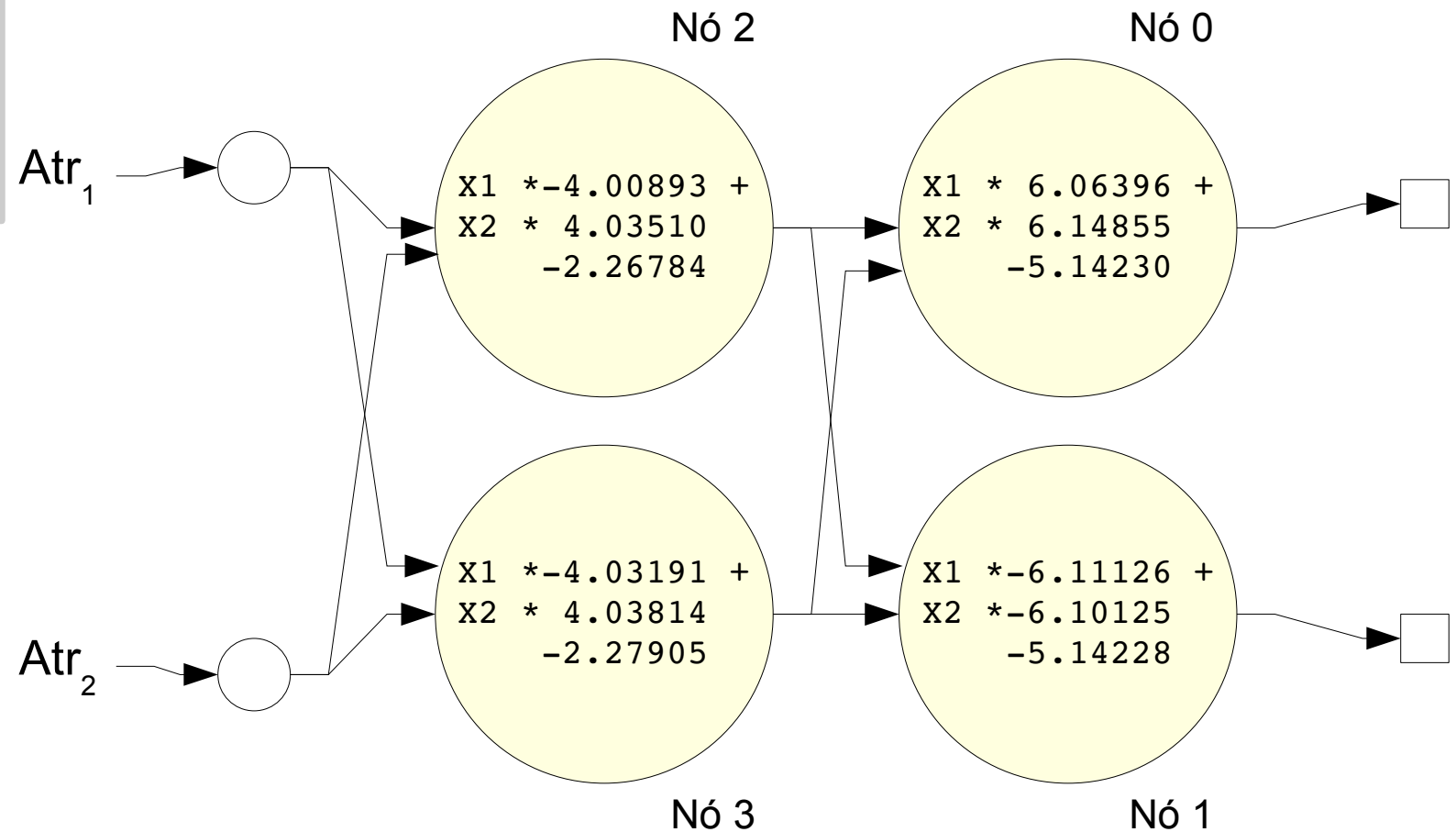
Input
Node 0

Class 8

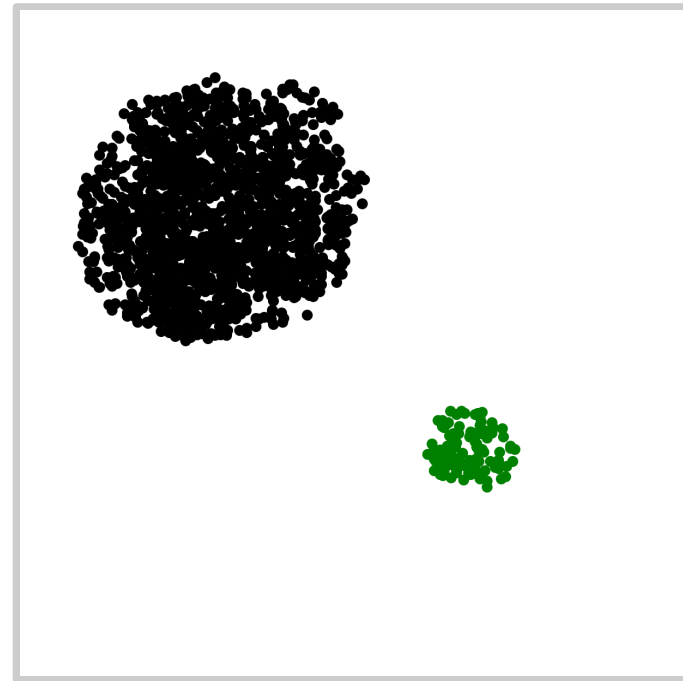
Input
Node 1



- Interpretação dos pesos

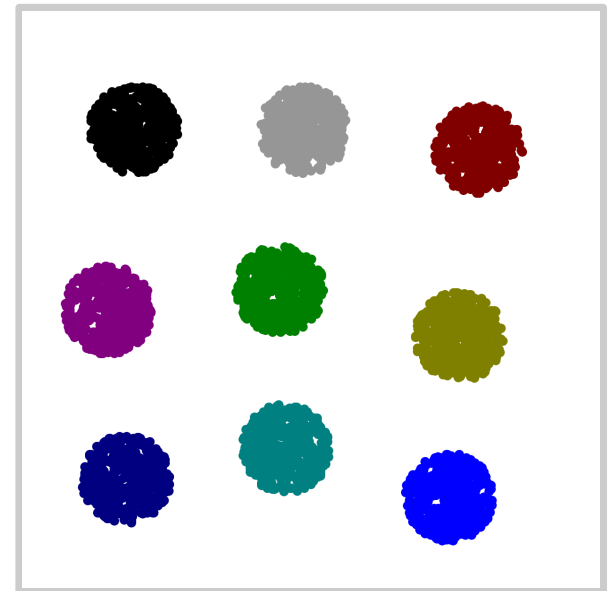
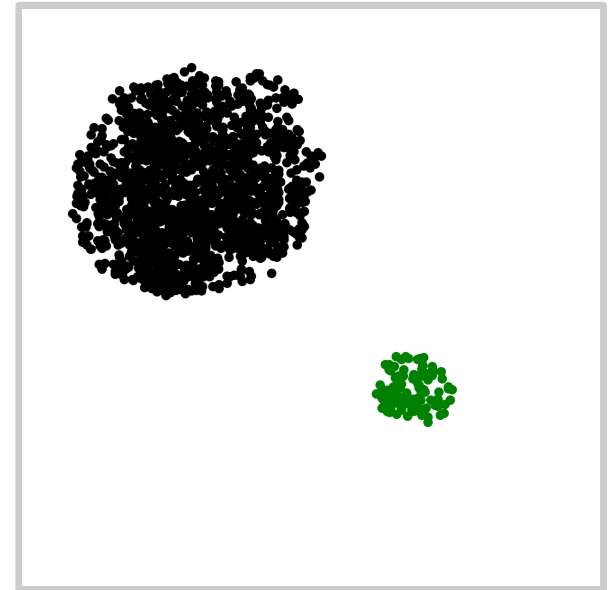


- Classificação com o modelo



Atr. 1	Atr. 2	Saída Nó 0	Saída Nó 1	Classe	Observações
0	0	-22.62	22.62	8	Canto Inf. Esq.
0	640	31527.72	-31527.61	0	Canto Sup. Esq.
640	0	-31446.89	31446.17	8	Canto Inf. Dir.
640	640	103.46	-104.06	0	Canto Sup. Dir.
320	320	40.42	-40.72	0	Ponto central
191	540	17219.8	-17219.92	0	Ex. Classe 0
458	237	-10827.12	10826.65	8	Ex. Classe 8

- Arquitetura 2x1x2 também classifica corretamente 100% das amostras no primeiro exemplo.
- Segundo exemplo:
 - Arquitetura 2x1x9: muitos erros, não adequada para este problema.
 - 2 neurônios na camada escondida são suficientes para classificar com 100% de acerto.





- Três arquiteturas para classificação:
 - 1 camada escondida, 5 neurônios.
 - 1 camada escondida, 25 neurônios.
 - 1 camada escondida, 250 neurônios.
- Camada de entrada: sempre 2 neurônios (x,y) .
- Camada de saída: sempre 2 neurônios (k,b) .

5

```
Correctly Classified Instances    3050  88.4314 %  
Incorrectly Classified Instances  399  11.5686 %  
=== Confusion Matrix ===  
   a    b  <-- classified as  
2689   18 |      a = 0  
 381  361 |      b = 13
```

25s

25

```
Correctly Classified Instances    3446  99.913 %  
Incorrectly Classified Instances    3   0.087 %  
=== Confusion Matrix ===  
   a    b  <-- classified as  
2705    2 |      a = 0  
  1  741 |      b = 13
```

96s

250

```
Correctly Classified Instances    3448  99.971 %  
Incorrectly Classified Instances    1   0.029 %  
=== Confusion Matrix ===  
   a    b  <-- classified as  
2707    0 |      a = 0  
  1  741 |      b = 13
```

786s

- Vantagens:
 - Capacidade de separar bem classes não linearmente separáveis (com múltiplas camadas).
- Problemas:
 - Difícil explicar/interpretar o “modelo” (caixa preta).
 - Treinamento pode ser complexo (computacionalmente caro), definição da arquitetura também.
- Soluções:
 - Múltiplas arquiteturas e avaliação da qualidade de classificação.
 - Avaliação de treinamento e teste.

Agrupamento (Clusterização)

- Algoritmos para criação de grupos de instâncias
 - Similares entre si,
 - Diferentes de instâncias em outros grupos.
 - Não-supervisionado (?)
- Também conhecidos como algoritmos de aprendizado auto-organizado.
- Diferença entre instâncias e (protótipos de) grupos é dada por um valor: medidas de distância ou similaridade / dissimilaridade.

- Duas abordagens gerais:
 - Particionais:
 - Criam grupos de forma iterativa.
 - Reparticiona/reorganiza até atingir um limiar (tempo, erro quadrático, etc).
 - Ao terminar fornece pertinência final de instâncias a grupos.
 - Hierárquicos:
 - *Bottom-up*: cria pequenos grupos juntando as instâncias, repetindo até atingir um critério.
 - *Top-down*: considera todas as instâncias como pertencentes a um grande grupo, subdivide recursivamente este grupo.
 - Podem criar *dendogramas*: agrupamentos hierárquicos com números alternativos de grupos.

- Particional.
- Entrada: instâncias, medida de distância, número de grupos (K).
- Saída: centróides dos grupos, pertinência das instâncias aos grupos, métricas.
- O algoritmo tenta minimizar o erro quadrático calculado entre as instâncias e os centróides dos grupos.

K-Médias: Passos

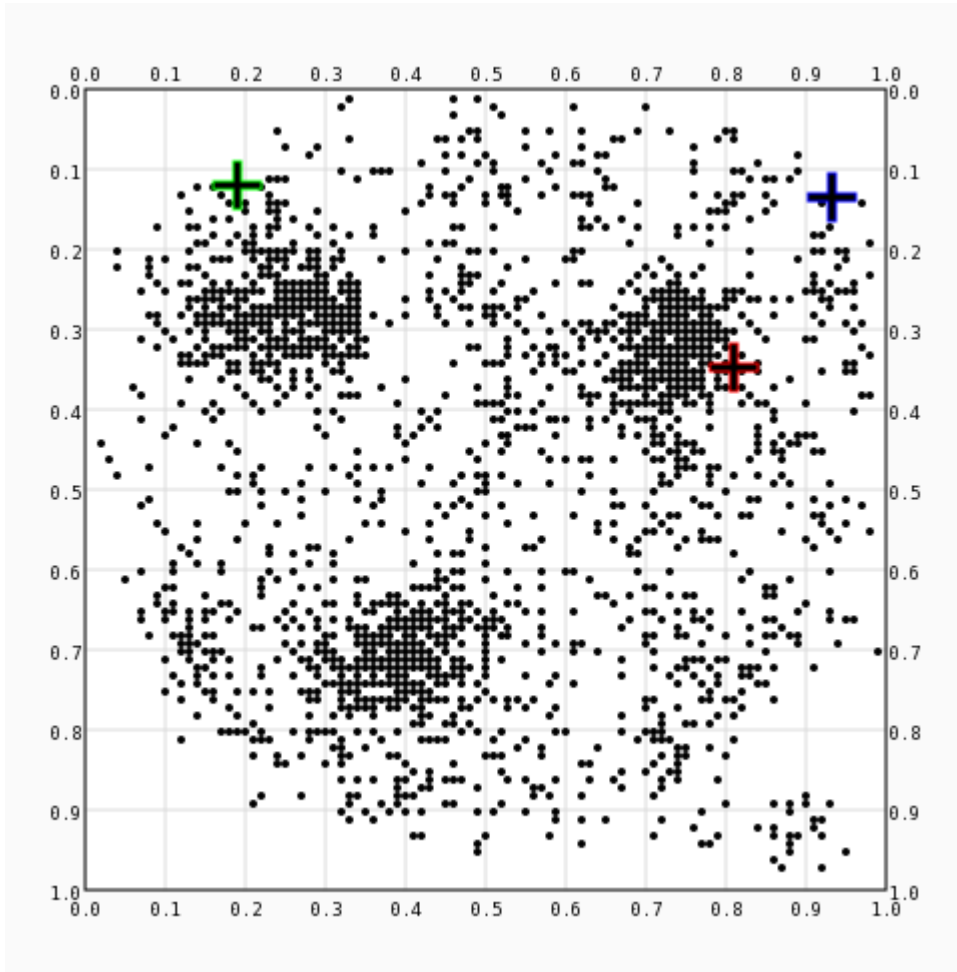
1. Inicializamos os centróides dos K grupos.
2. Marcamos cada instância como pertencente ao grupo (centróide) mais próximo.
3. Recalculamos os centróides dos grupos considerando as pertinências.

$$v_i = \frac{1}{n_i} \sum_{x_k \in C_i} x_k$$

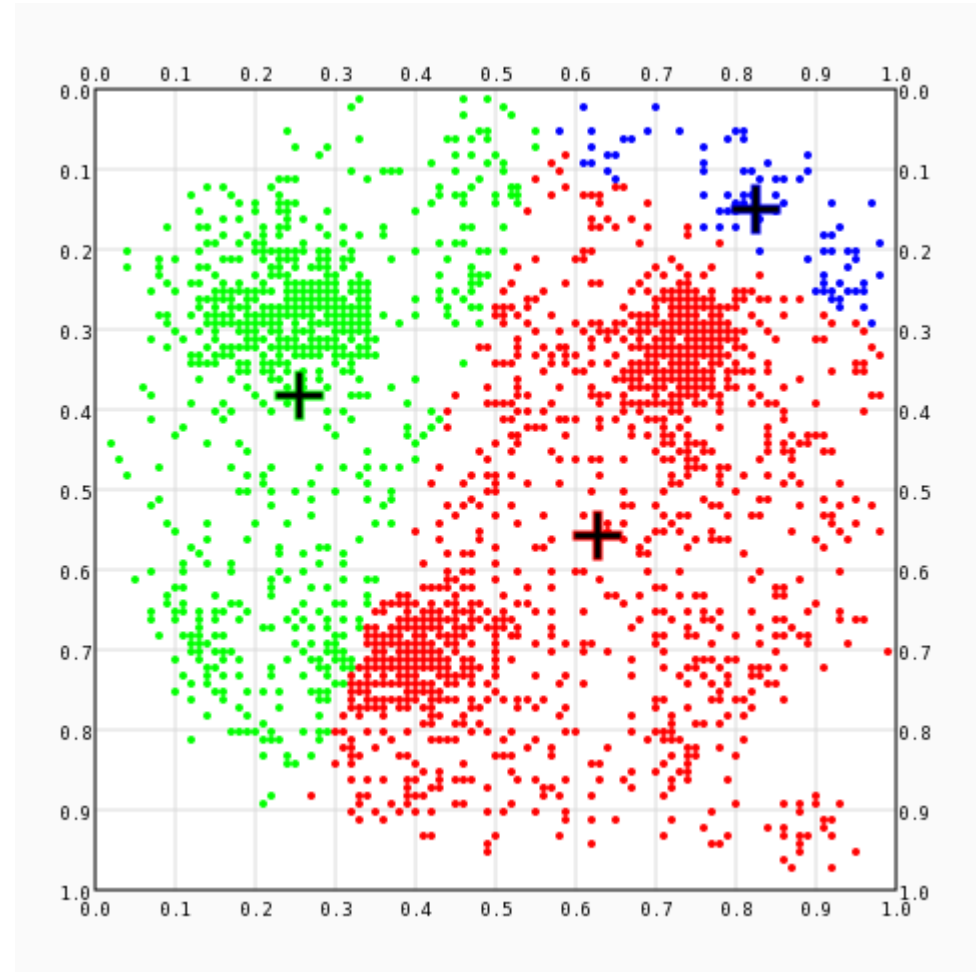
4. Recalculamos o erro quadrático total.

$$J = \sum_{k=1}^n \sum_{x_k \in C_i} |x_k - v_i|^2$$

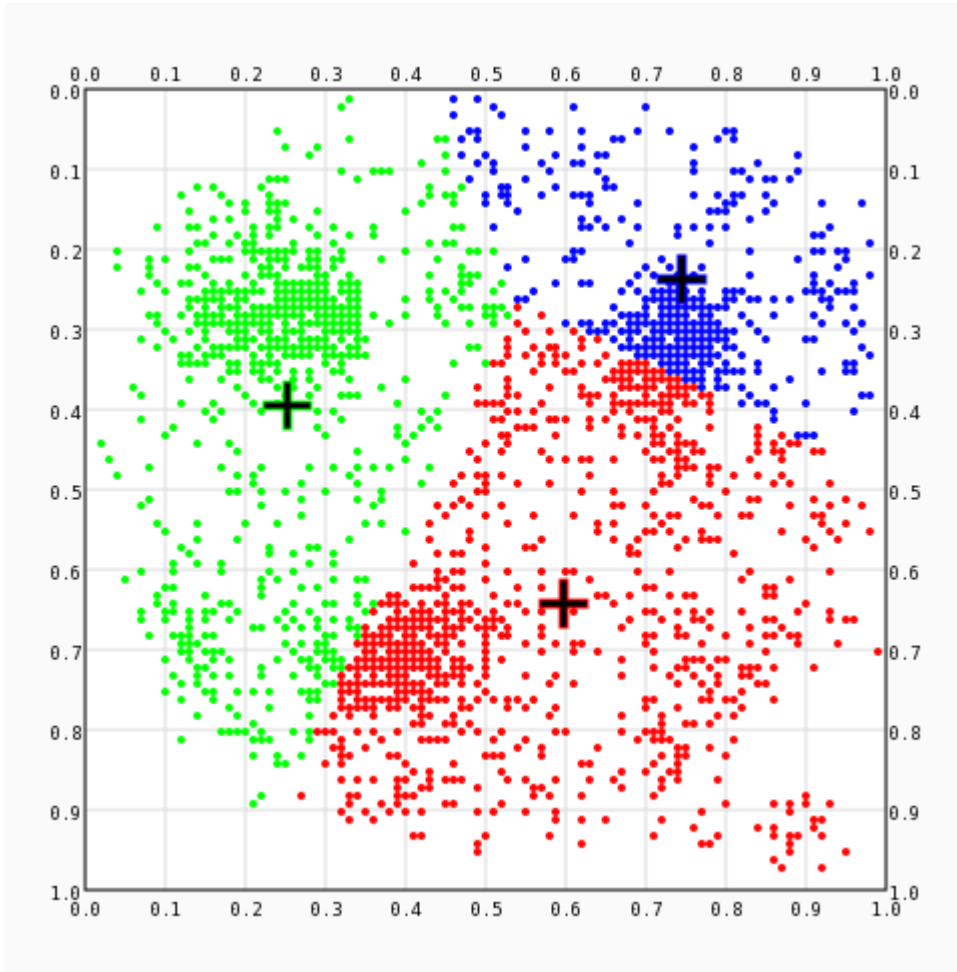
5. Verificamos condições de parada e repetimos a partir do passo 2.



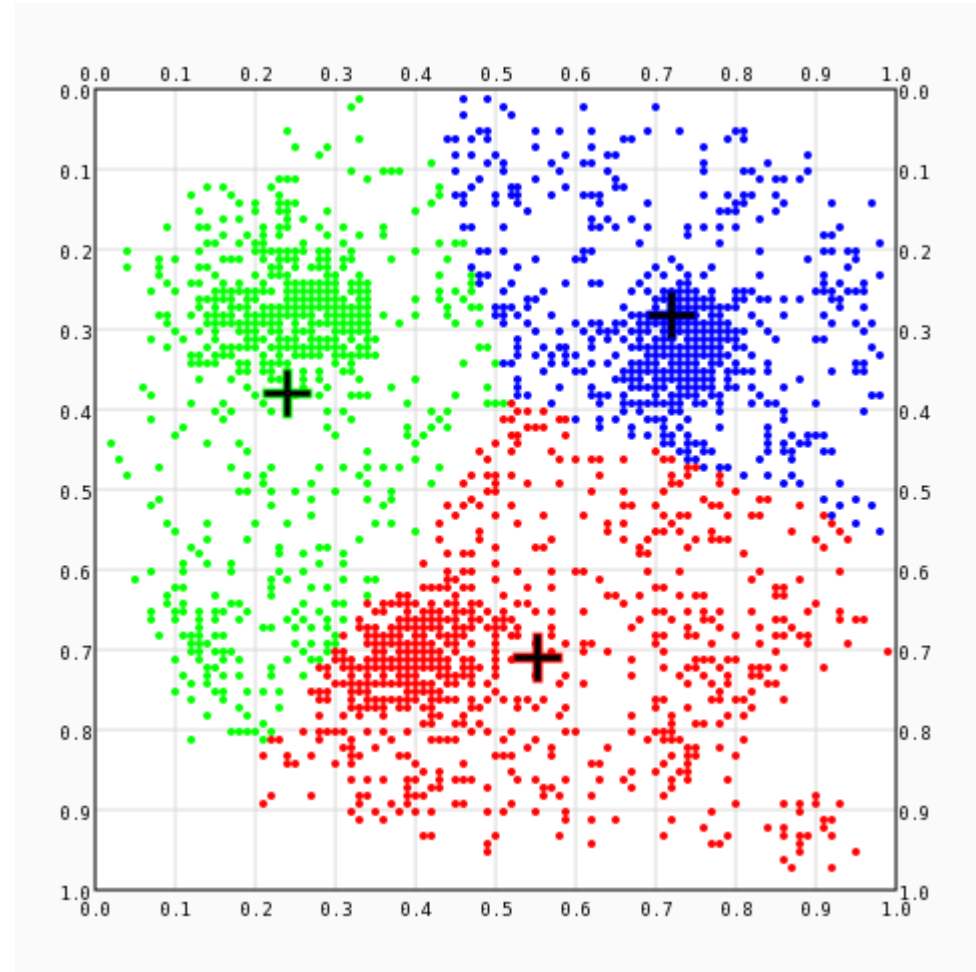
0



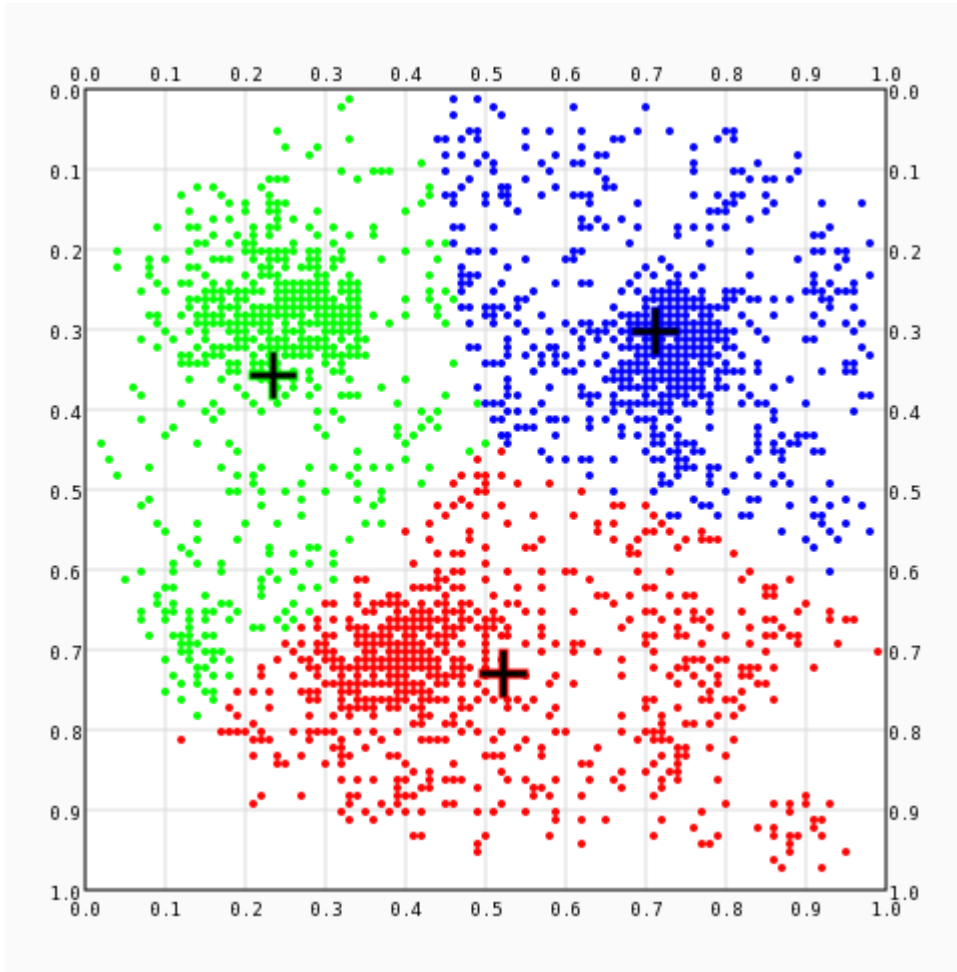
1



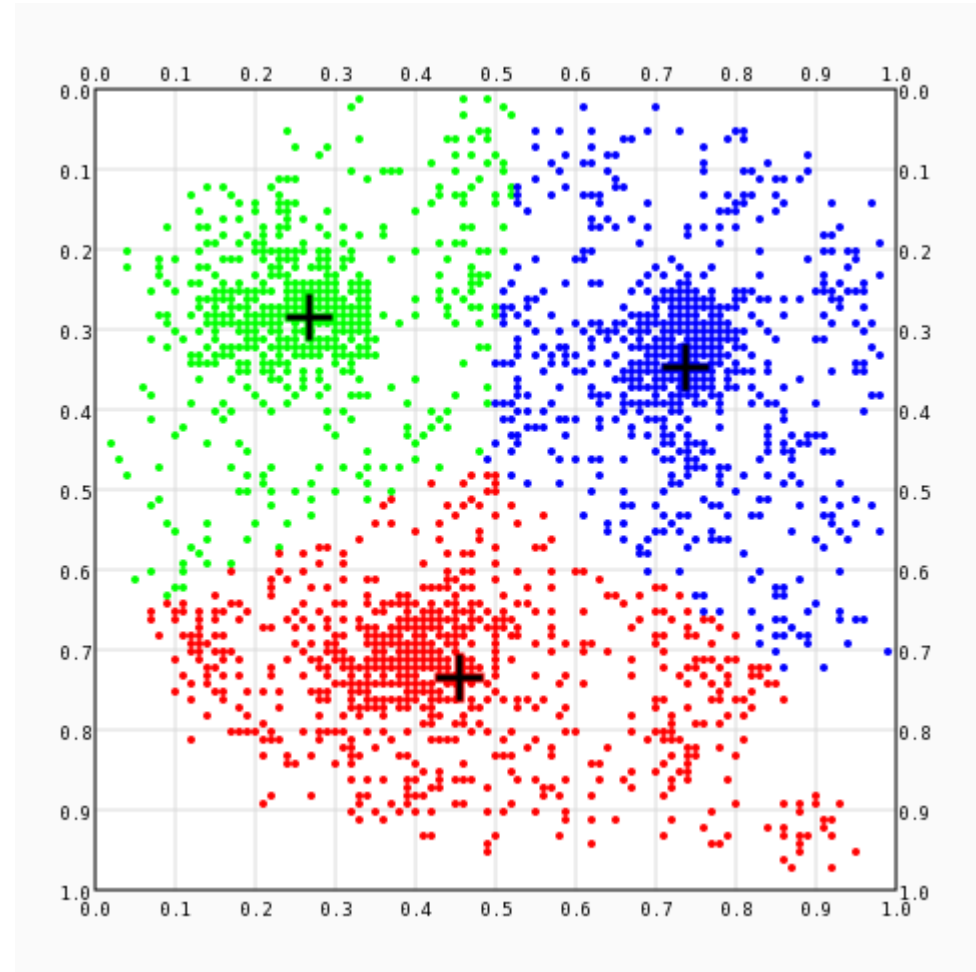
2



3



4

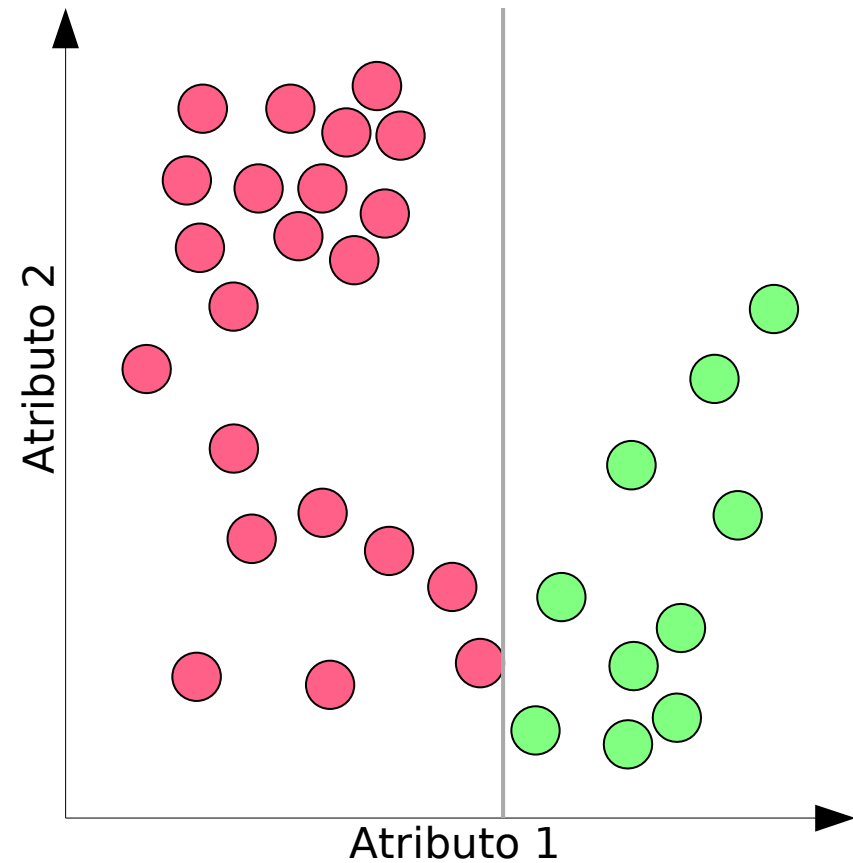
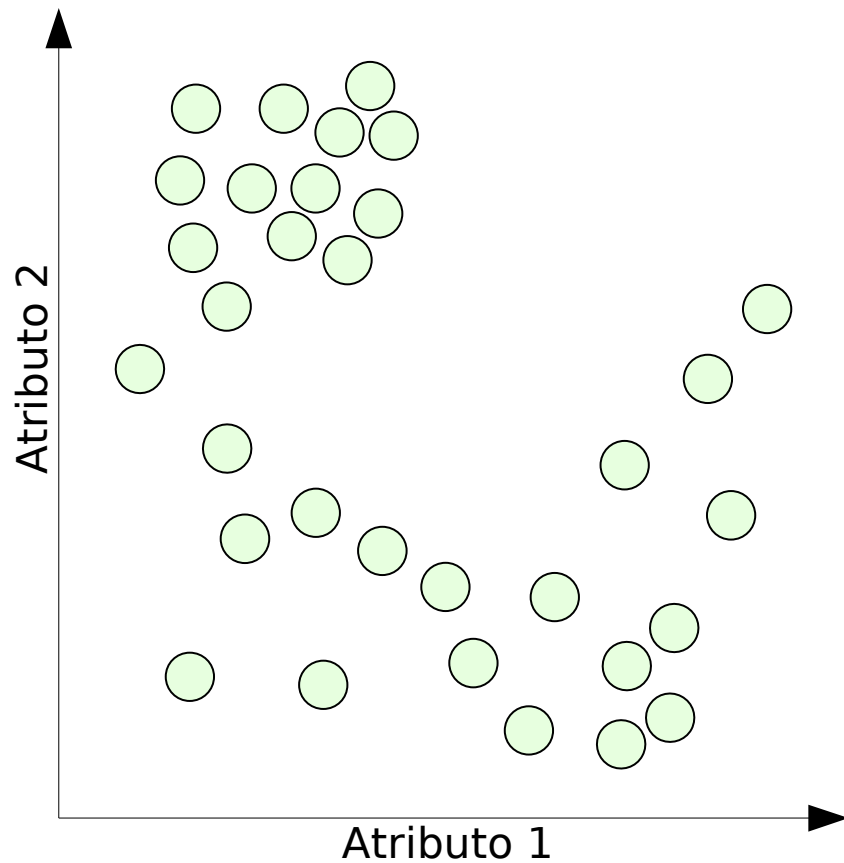


10

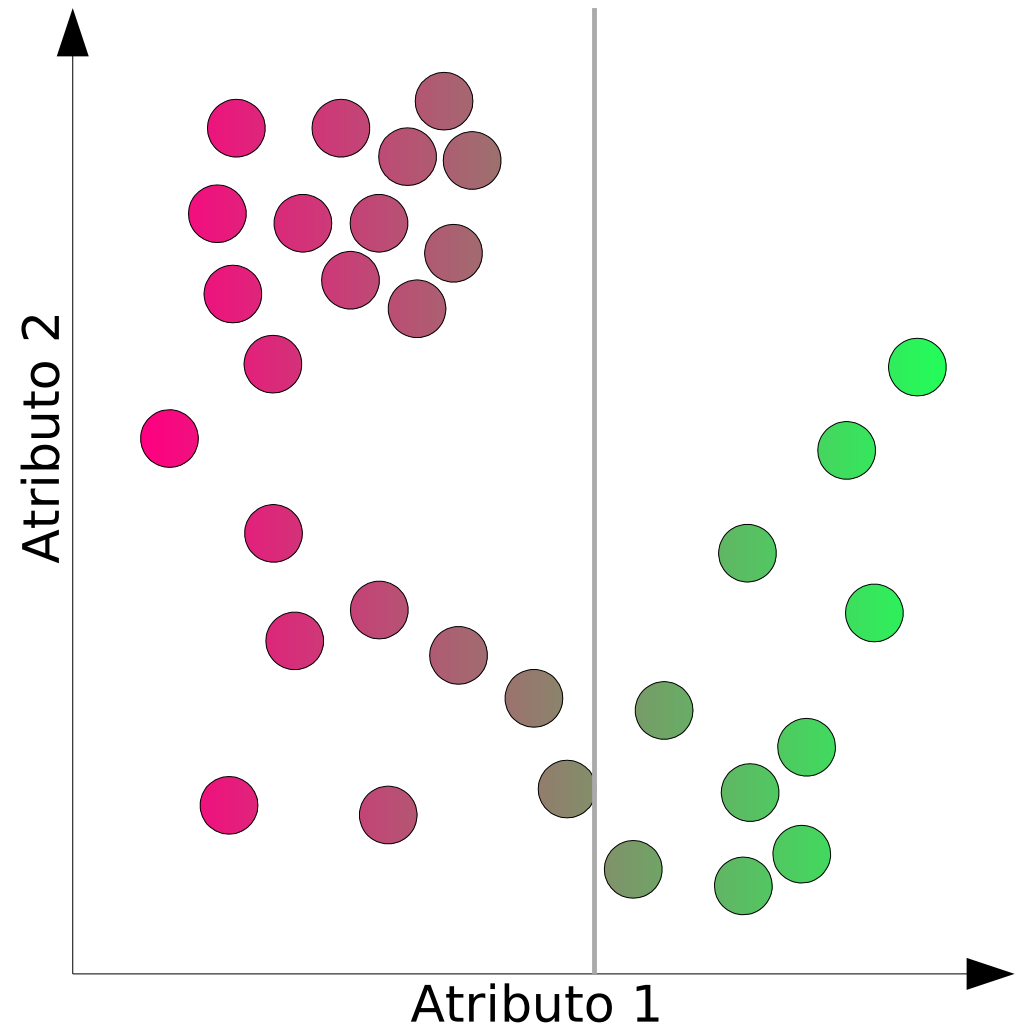
- Problemas:
 - Múltiplas iterações considerando todos os dados: problemas de performance.
 - Inicialização: como escolher centróides iniciais (impacto na convergência).
 - Converge para um mínimo local.
 - Singularidades: grupos sem instâncias relacionadas.
 - Não podemos calcular seus centróides.
 - Escolha de K ?
 - Existe um K' melhor do que o K ?

- K-Médias mais heurísticas: nada de pequenos grupos, quebraremos grupos com grande variância.
- Mais complexo, demorado do que simples K-Médias.
- Mais parâmetros devem ser especificados, mas por se tratar de uma heurística, estes parâmetros podem ser aproximados.
- Descrição no livro do Carl Looney: 12 passos em 3 páginas.

- Consideremos pertinência a classe ou grupo...



- ... não precisa ser estritamente booleana!
 - Cada instância pode pertencer a mais de uma categoria com pertinências *entre* 0 e 1.



- ... não precisa ser estritamente booleana!
 - Cada instância pode pertencer a mais de uma categoria com pertinências *entre* 0 e 1.

- Exemplo:

Instância	Classe A	Classe B	Classe C	Classe D
1	0.31	0.19	0.50	0.00
2	0.08	0.01	0.74	0.17
3	0.25	0.24	0.26	0.25
4	0.99	0.00	0.00	0.01
5	0.50	0.50	0.00	0.00

- Similar ao K-Médias, com mesmas características gerais.
- Cria uma *tabela de pertinência* de cada instância em cada grupo.
 - Tabela provê informações interessantes!

Instância	Classe A	Classe B	Classe C	Classe D
1	0.31	0.19	0.50	0.00
2	0.08	0.01	0.74	0.17
3	0.25	0.24	0.26	0.25
4	0.99	0.00	0.00	0.01
5	0.50	0.50	0.00	0.00

Fuzzy C-Médias: Passos

1. Inicializamos a tabela de pertinência.

2. Calculamos os centróides a partir das pertinências com

$$v_i = \frac{\sum_{k=1}^n \mu_{ik}^m x_{ik}}{\sum_{k=1}^n \mu_{ik}^m}$$

3. Calculamos a tabela de pertinências a partir dos centróides valores das instâncias com

$$\mu_{ik} = \frac{\left[\frac{1}{|x_k - v_i|^2} \right]^{1/(m-1)}}{\sum_{j=1}^c \left[\frac{1}{|x_k - v_j|^2} \right]^{1/(m-1)}}$$

4. Recalculamos a função objetivo

$$J = \sum_{k=1}^n \sum_{i=1}^c \mu_{ik}^m |x_k - v_i|^2$$

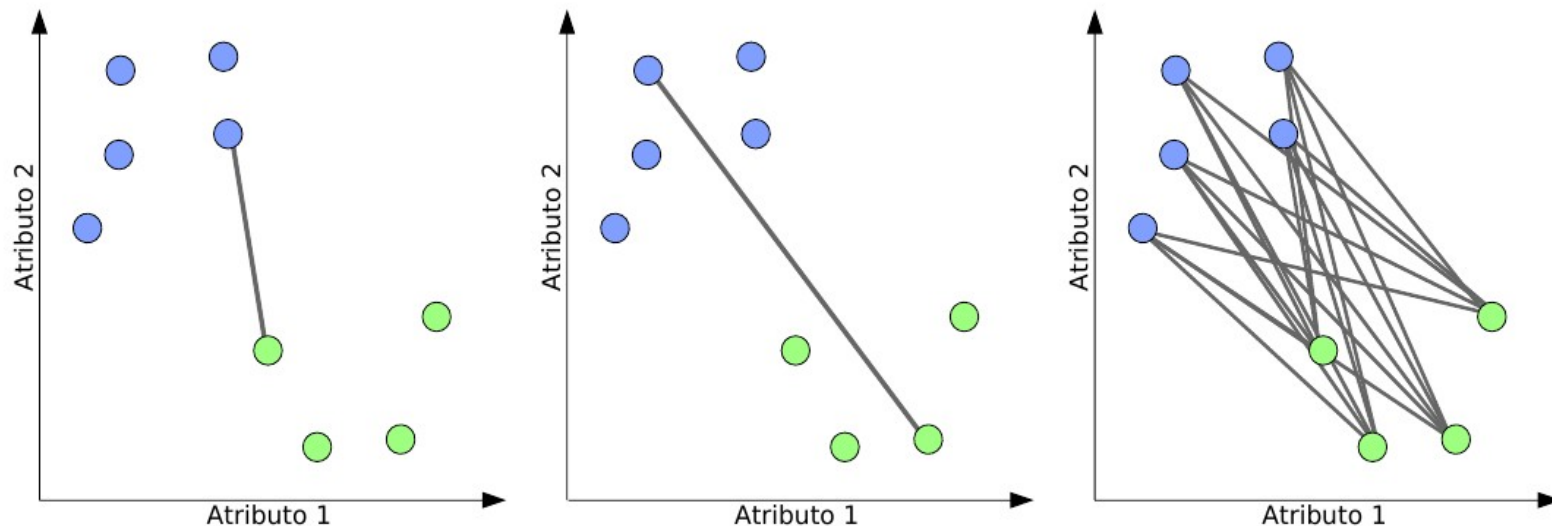
5. Verificamos condições de parada e repetimos a partir do passo 2.

- Exemplo com $C=6$ e imagem Ikonos.

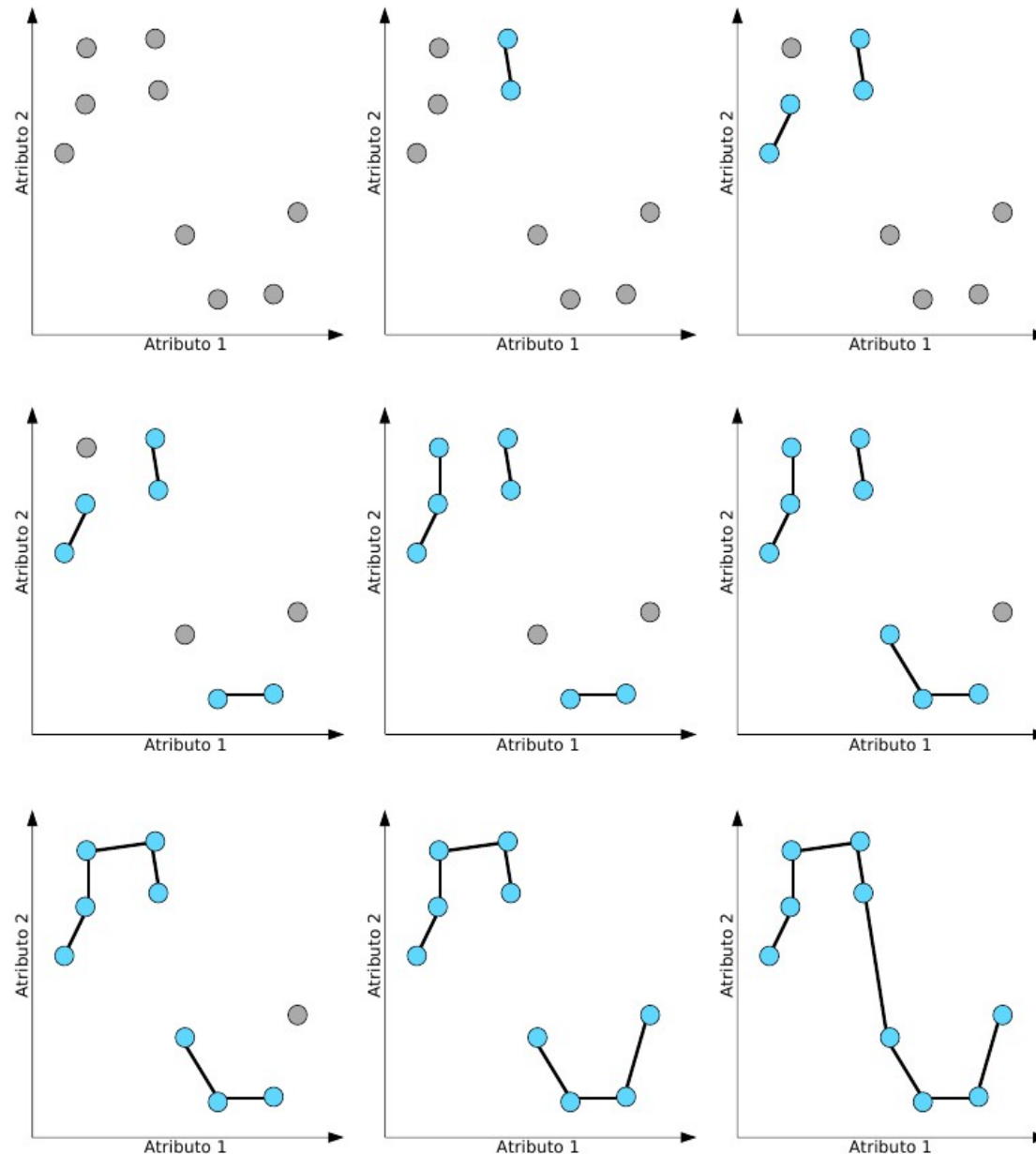


- *Bottom-up*:

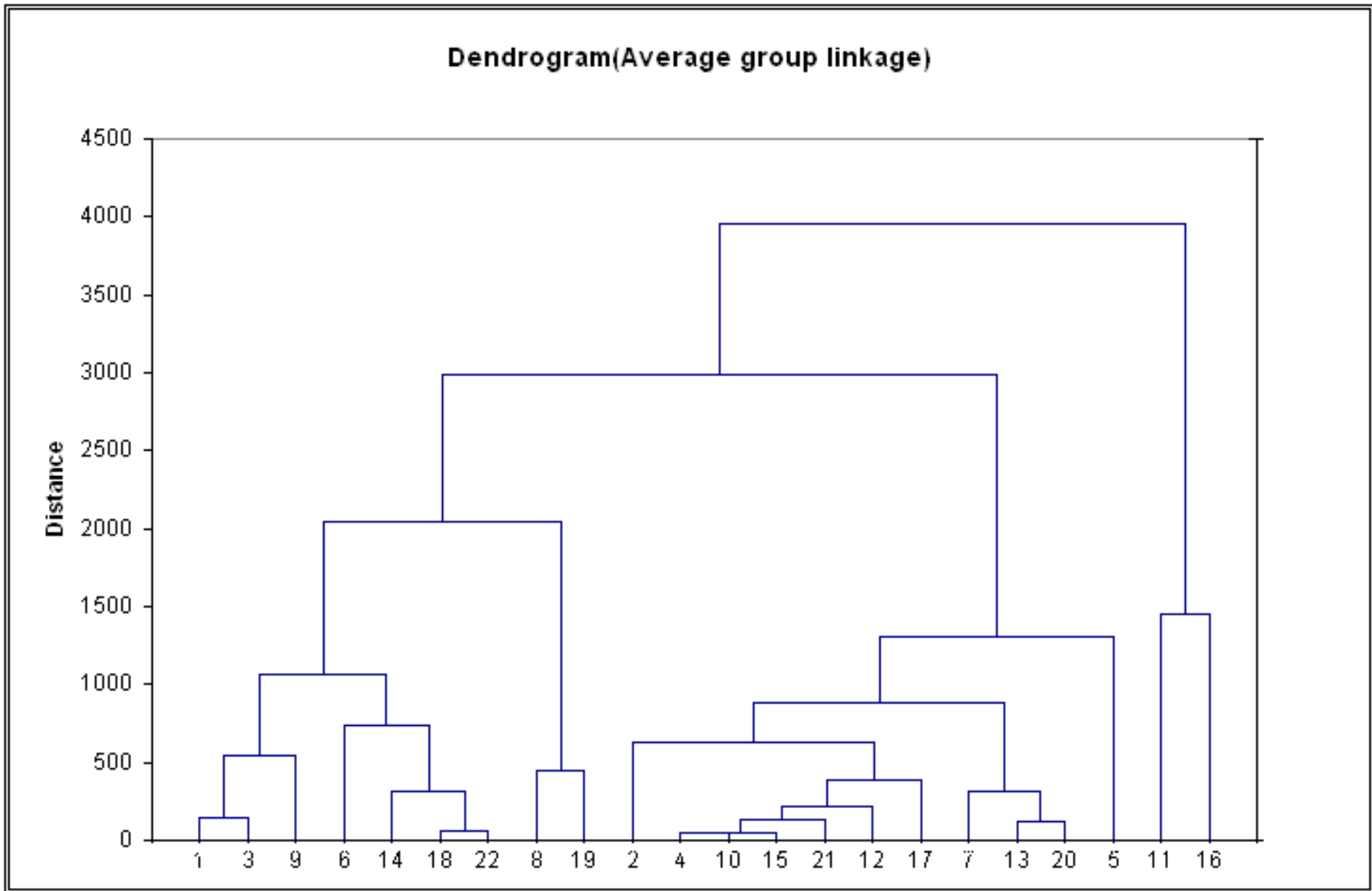
1. Considere todas as instâncias como grupos (centros são os valores da própria instância).
2. Crie uma matriz de distâncias que indique a distância de cada grupo a cada outro grupo.
3. Localize, nesta matriz, os dois grupos com menor **distância** entre eles, e efetue a união destes grupos.
4. Se ainda houver dois ou mais grupos, volte ao passo 2.



Agrupamento Hierárquico: Simulação



Agrupamento Hierárquico: Dendograma

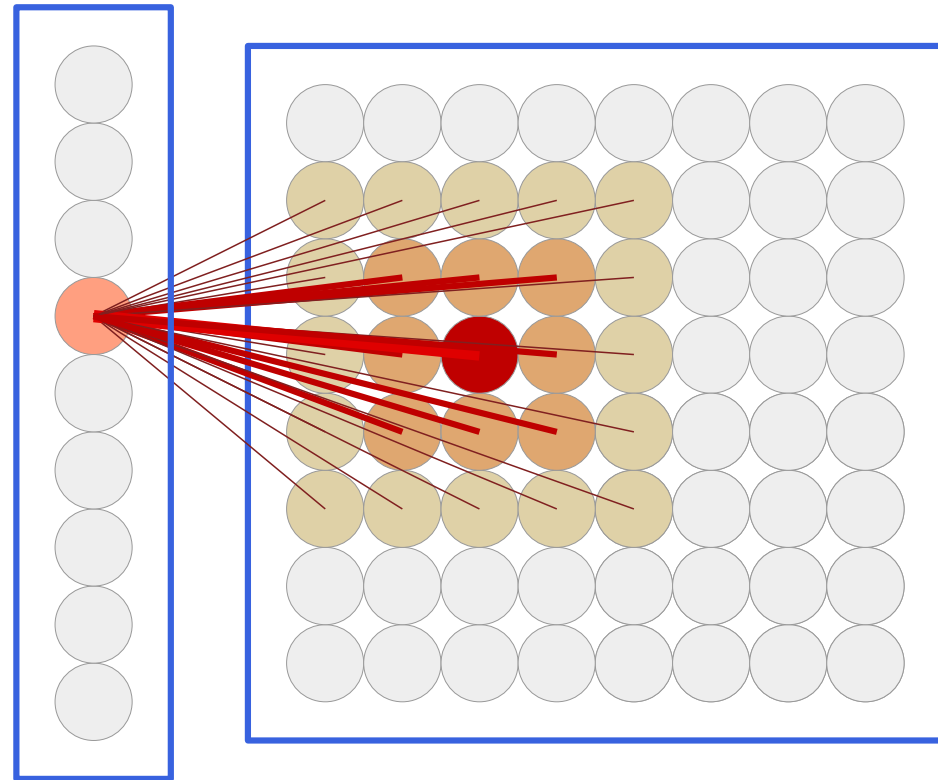


Fonte: XLMiner <http://www.resample.com/xlminer/>

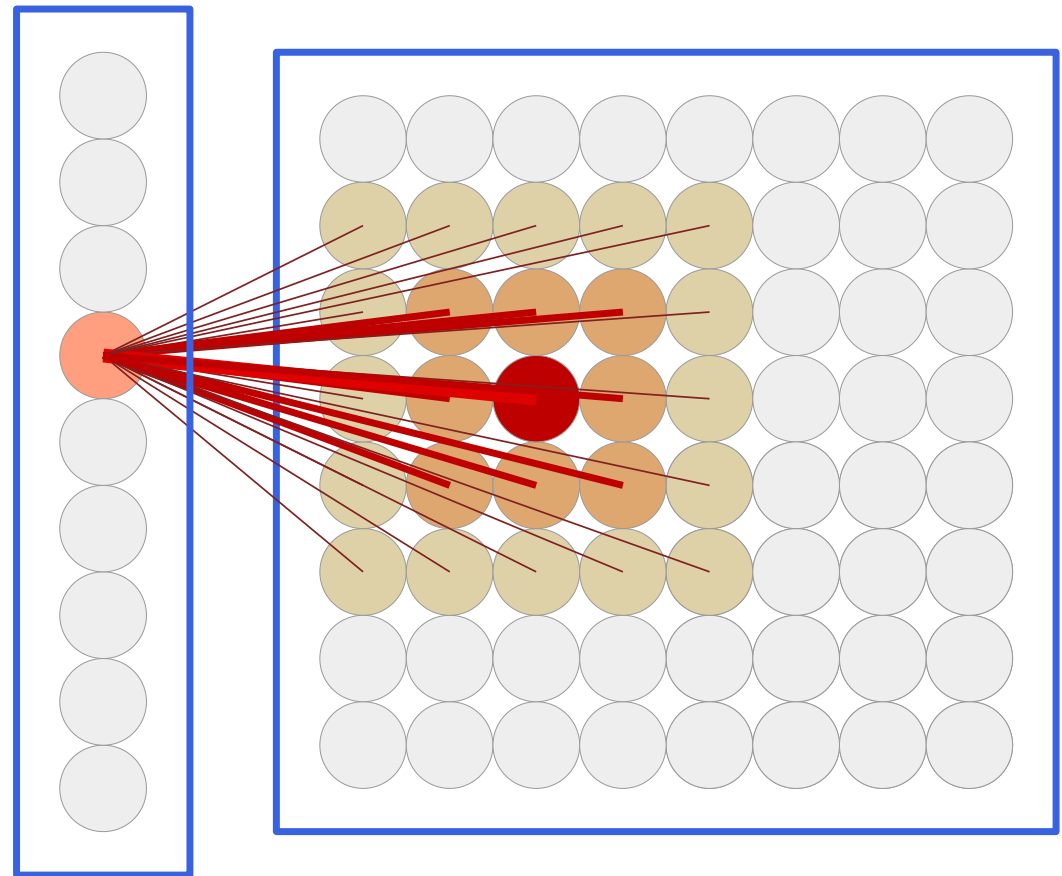
- Vantagens:
 - Número de agrupamentos pode ser determinado experimentalmente ou de forma exploratória.
 - Análise do resultado usando dendograma, que indica a estrutura hierárquica dos agrupamentos.
 - Resultado independe da ordem de apresentação dos dados.
- Problemas:
 - Matriz de distância pode consumir muita memória e seu recálculo é custoso.
 - Nem todos os elementos precisam ser recalculados.
 - Somente diagonal da matriz precisa ser armazenada.
 - O de sempre: como calcular distância não-numérica?

- Também conhecidos como redes de Kohonen.
- Mapeiam vetores em N dimensões para 2 ou 3 dimensões, preservando topologia.
- Por extensão, usados para fazer agrupamento e classificação em fase posterior.
- Usados também para redução de dimensionalidade com manutenção de topologia.

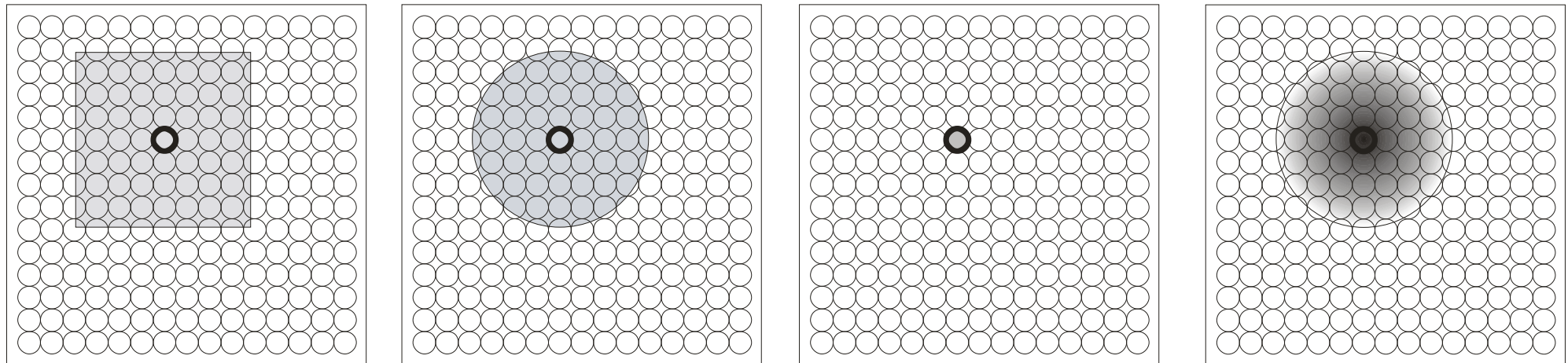
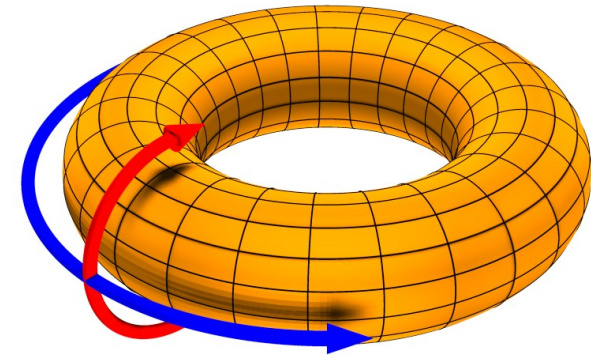
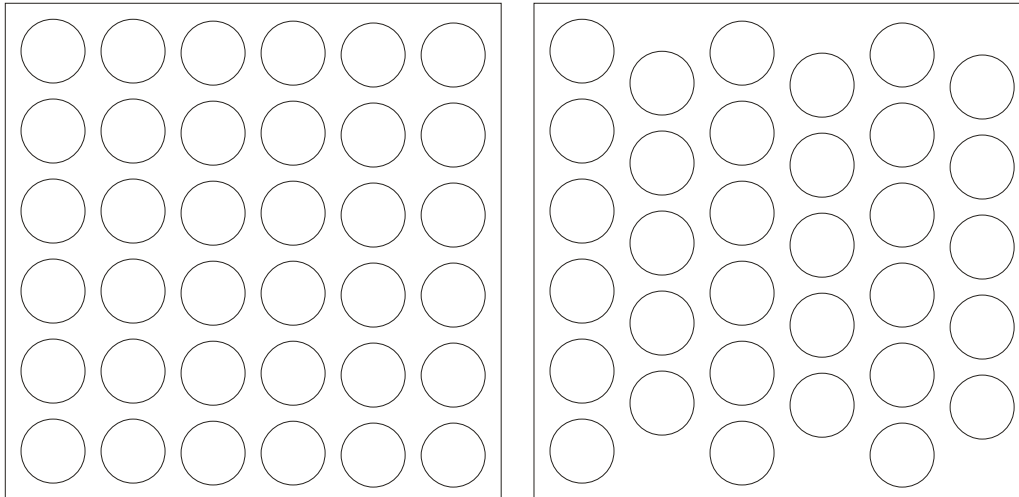
- Uma camada de entrada, contendo os dados que serão usados para treinamento.
- Uma camada de neurônios para mapeamento.
- Cada neurônio é um vetor com as mesmas dimensões da entrada.



- Entrada: Vetores de dados, rede (considerar arquitetura), parâmetros de treinamento.
- Saída: rede treinada, neurônios se assemelham a vetores apresentados.



- Topologia e vizinhança



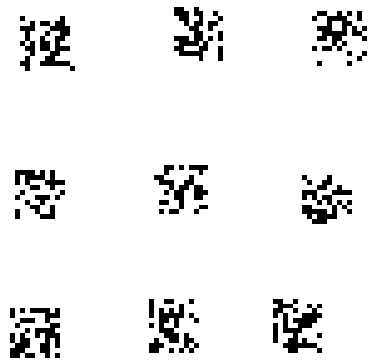
1. Inicializar vetores da rede (neurônios) com valores aleatórios.
2. Escolher uma amostra (vetor) de dados.
3. Encontrar o neurônio mais semelhante:
Aquele cuja distância no espaço de atributos seja a menor para o vetor de dados = o “mais parecido” ou vencedor (*Best Matching Unit*).
4. Atualizar os valores do neurônio vencedor e de seus vizinhos para que fiquem mais similares aos do vetor de entrada.

$$W_{t+1} = W_t + L_t R_t |W_t - D|$$

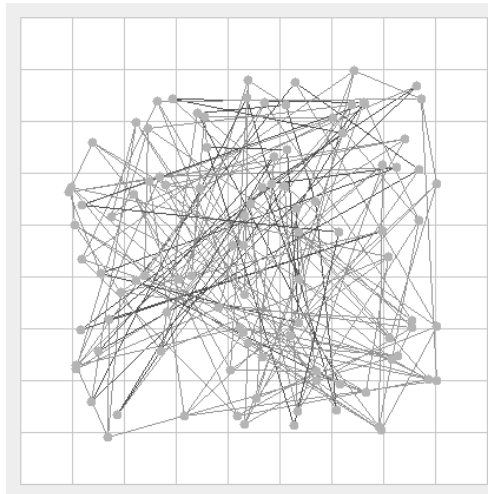
5. Verificar critérios de parada, retornar ao passo 2 se for o caso, atualizar valores para treinamento.

- Taxa de aprendizado (*learning rate* L):
 - Valor multiplicador que indica o quanto os valores de um neurônio serão aproximados do dado de entrada.
 - Deve decrescer à medida em que a rede é treinada até um valor mínimo.
- Raio da vizinhança (R).
 - Limiar/valor que indica se um neurônio próximo ao vencedor será considerado vizinho do mesmo.
 - Deve decrescer à medida em que a rede é treinada até um valor mínimo.
 - Aplicável somente à algumas vizinhanças.

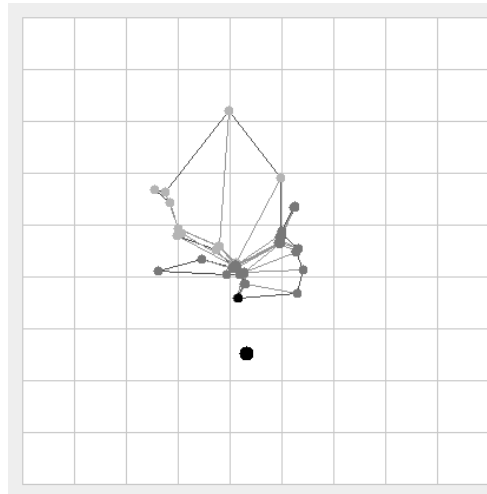
Self-Organizing Maps (SOMs): Exemplos



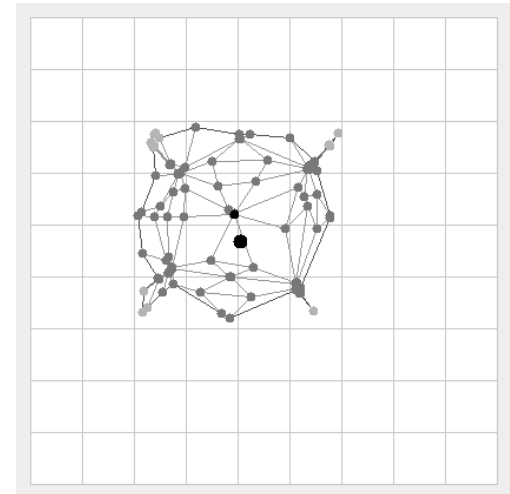
Dados Originais



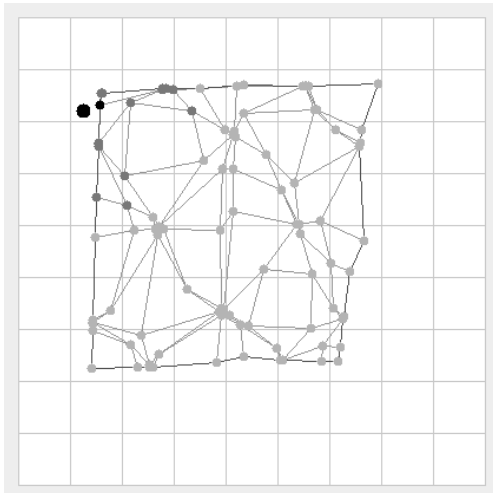
0 iterações



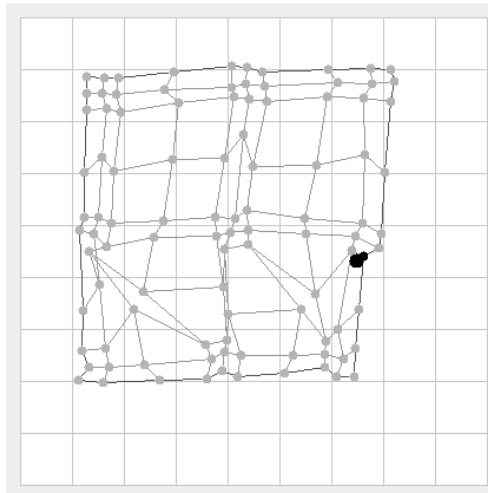
50000 iterações



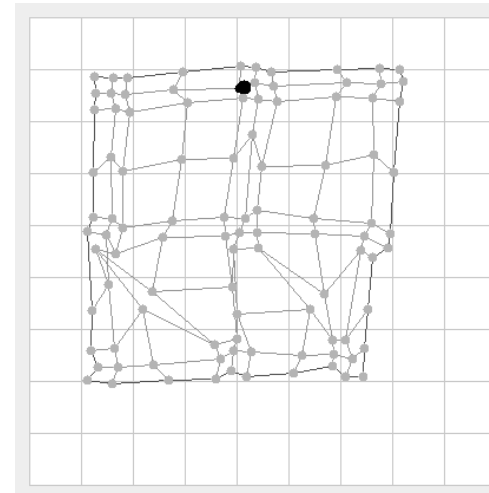
100000 iterações



200000 iterações



300000 iterações

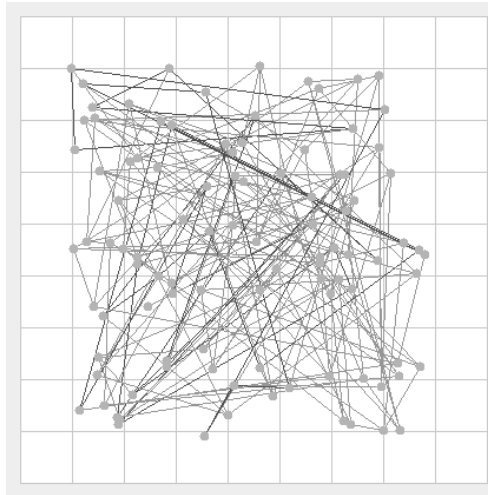


400000 iterações

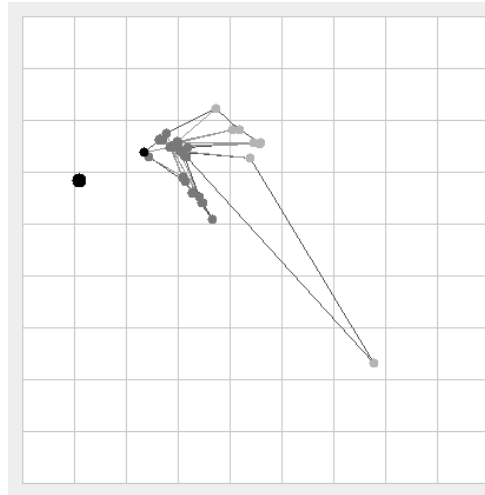
Self-Organizing Maps (SOMs): Exemplos



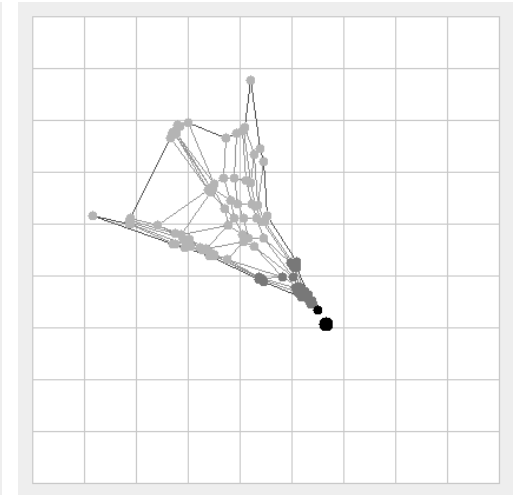
Dados Originais



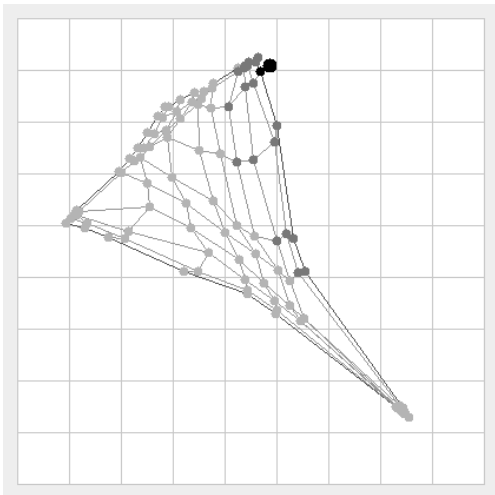
0 iterações



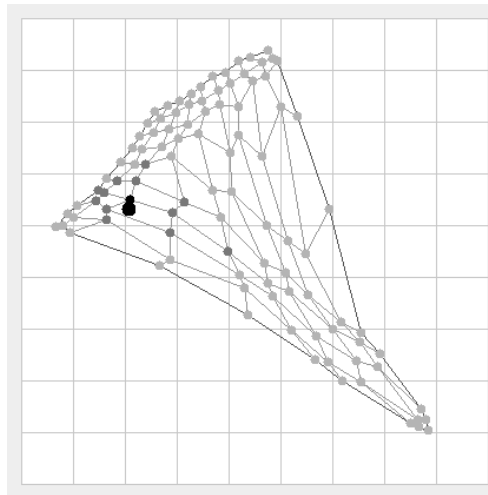
50000 iterações



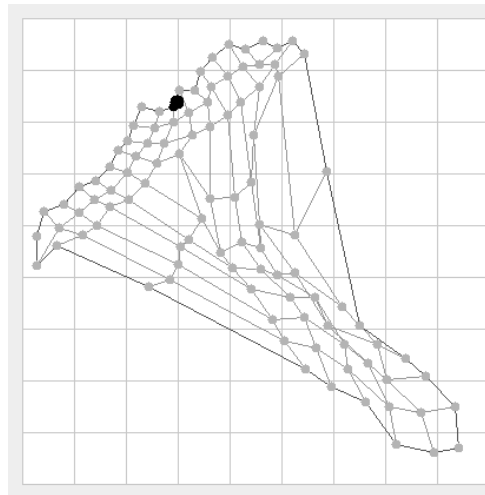
100000 iterações



150000 iterações

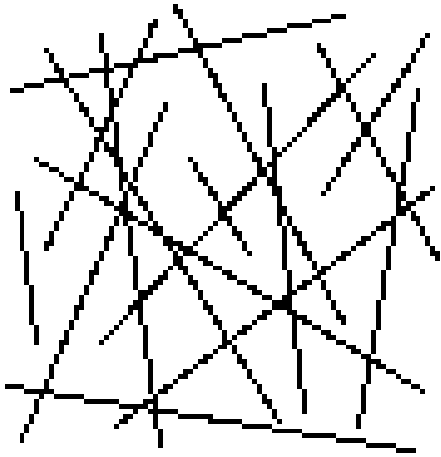


200000 iterações

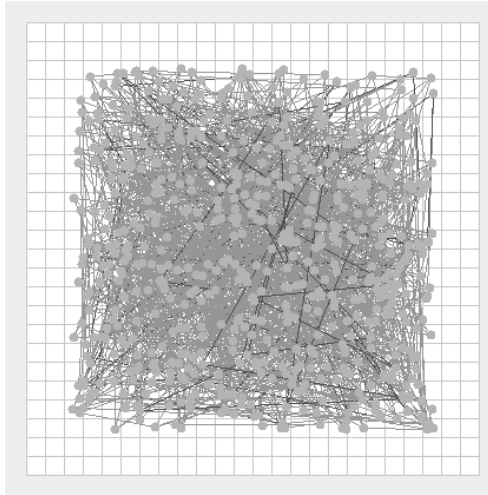


300000 iterações

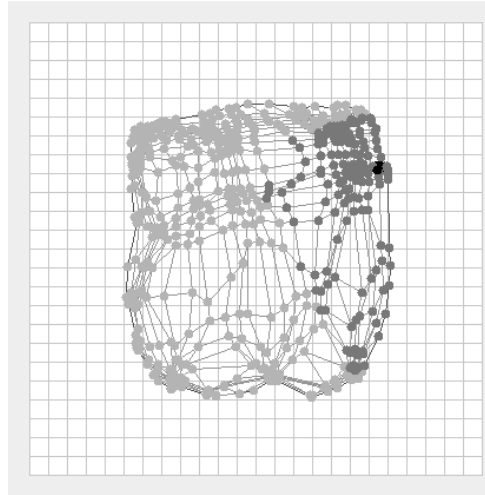
Self-Organizing Maps (SOMs): Exemplos



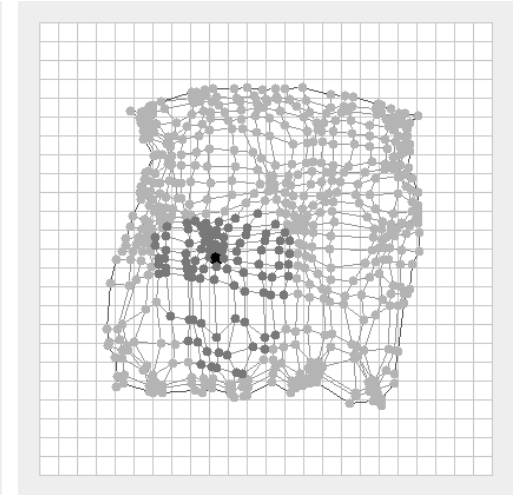
Dados Originais



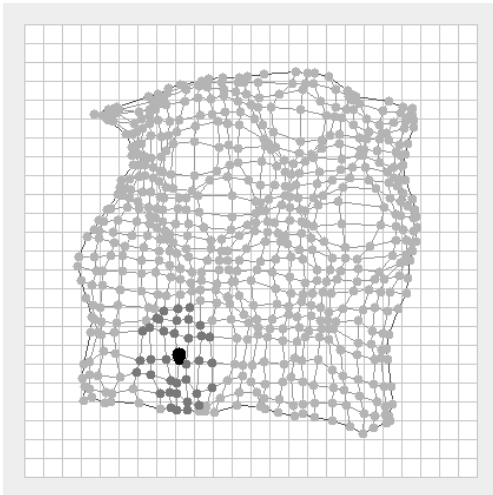
0 iterações
(25x25)



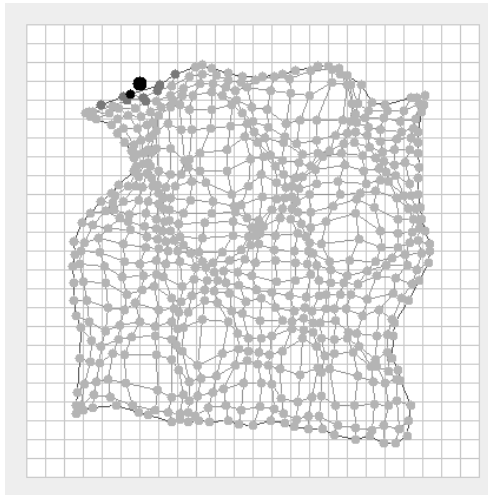
50000 iterações



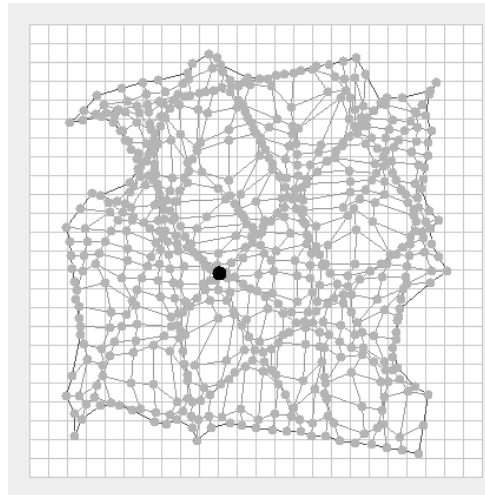
100000 iterações



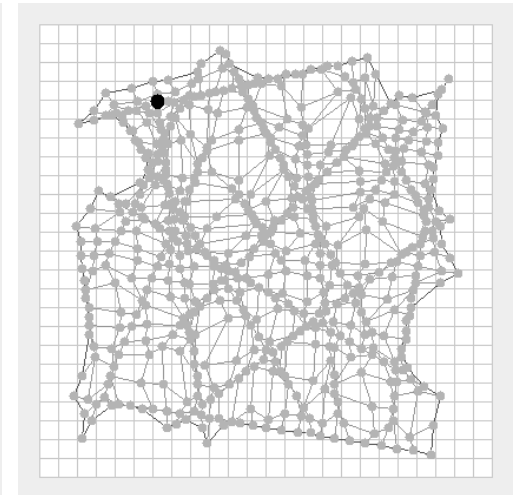
150000 iterações



200000 iterações



300000 iterações



400000 iterações

Regras de Associação

- Regras sobre relações e co-ocorrências em bases de dados:
- Se X ocorre na base de dados, então Y também ocorre (com alguma relação a X).
- Co-ocorrência: se X , Y e Z ocorrerem na base de dados então A também ocorre (com alguma relação a X , Y e Z).
 - X , Y e Z são os antecedentes da associação; A é o conseqüente.
 - Ocorrências consideradas em escopo limitado: não queremos dizer que se X ocorre em qualquer “local” da base de dados, Y também ocorrerá em qualquer “local”.
- Muito usado para verificar associações em tabelas de transações (“carrinhos de compra”)

- Exemplo simples:

Transação	Itens
1	leite, ovos, café, açúcar, fraldas, manteiga
2	leite, café, farinha
3	leite, ovos, açúcar
4	café, açúcar
5	fraldas
6	manteiga, ovos, leite
7	café, açúcar, leite, ovos
8	farinha, manteiga, ovos
9	manteiga, ovos, leite, café, açúcar
10	fraldas, café, cerveja

- Conclusões simples sobre a base de dados da tabela:
 - Quem compra leite quase sempre compra ovos.
 - Como definir “quase sempre”? Quantas vezes isso ocorre na base de dados?
 - Quem compra ovos e açúcar sempre compra leite.
 - Mas quantas compras contém ovos e açúcar? O que causa a compra de leite?
 - Quem compra cerveja sempre compra fraldas.
 - Quantas vezes isso ocorre na base de dados?
Isso é relevante?

Transação	Itens
1	leite, ovos, café, açúcar, fraldas, manteiga
2	leite, café, farinha
3	leite, ovos, açúcar
4	café, açúcar
5	fraldas
6	manteiga, ovos, leite
7	café, açúcar, leite, ovos
8	farinha, manteiga, ovos
9	manteiga, ovos, leite, café, açúcar
10	fraldas, café, cerveja

- Muitos que compram café também compram açúcar.
- Ninguém compra só leite.
 - Muitas outras associações negativas existem: quem compra fraldas não compra farinha, quem compra farinha não compra cerveja.
- Quais associações negativas são significativas?

Transação	Itens
1	leite, ovos, café, açúcar, fraldas, manteiga
2	leite, café, farinha
3	leite, ovos, açúcar
4	café, açúcar
5	fraldas
6	manteiga, ovos, leite
7	café, açúcar, leite, ovos
8	farinha, manteiga, ovos
9	manteiga, ovos, leite, café, açúcar
10	fraldas, café, cerveja

- Métricas:
- Significância em uma associação: ela pode existir mas ser muito rara em uma base de dados (ex. cerveja \rightarrow fraldas).
 - **Suporte $X \rightarrow Y$** : número de casos que contém X e Y dividido pelo número total de registros.
- Confiança em uma associação: o antecedente pode ocorrer várias vezes na base de dados mas nem sempre com o mesmo conseqüente associado.
 - **Confiança $X \rightarrow Y$** : número de registros que contém X e Y dividido pelo número de registros que contém X .

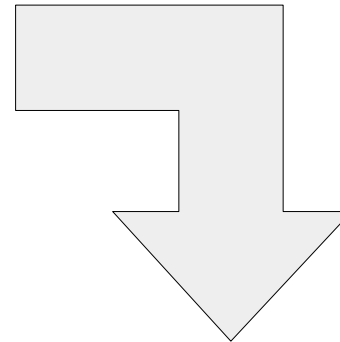
- Algoritmo Apriori:
 1. Entrada: coleção de dados associados, suporte mínimo, confiança mínima.
 2. Considerar $K = 1$ para criação de K -itemsets
 3. Analisar os dados associados e criar uma tabela de K -itemsets com suporte acima do suporte mínimo.
 4. Criar com os *itemsets* filtrados um conjunto de candidatos a $(K + 1)$ *itemsets*.
 5. Usar propriedades do Apriori para eliminar *itemsets* infreqüentes.
 6. Repetir desde o passo 3 até que o conjunto gerado seja vazio.
 7. Listar regras de associação (com permutações) e aplicar limite de confiança.

Veja <http://www.lac.inpe.br/~rafael.santos/cap-359.jsp> para uma explicação mais detalhada.

- Simulação do Apriori com suporte mínimo 25% e confiança 75%:

Transação	Itens
1	leite, ovos, café, açúcar, fraldas, manteiga
2	leite, café, farinha
3	leite, ovos, açúcar
4	café, açúcar
5	fraldas
6	manteiga, ovos, leite
7	café, açúcar, leite, ovos
8	farinha, manteiga, ovos
9	manteiga, ovos, leite, café, açúcar
10	fraldas, café, cerveja

Passos detalhados em
<http://www.lac.inpe.br/~rafael.santos/cap-359.jsp>

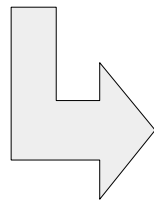


Transação	leite	ovos	café	açúcar	fraldas	manteiga	farinha	cerveja
1	1	1	1	1	1	1	0	0
2	1	0	1	0	0	0	1	0
3	1	1	0	1	0	0	0	0
4	0	0	1	1	0	0	0	0
5	0	0	0	0	1	0	0	0
6	1	1	0	0	0	1	0	0
7	1	1	1	1	0	0	0	0
8	0	1	0	0	0	1	1	0
9	1	1	1	1	0	1	0	0
10	0	0	1	0	1	0	0	1

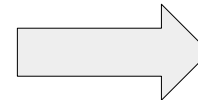
Regras de Associação

- Simulação do Apriori com suporte mínimo 25% e confiança 75%:

Transação	leite	ovos	café	açúcar	fraldas	manteiga	farinha	cerveja
1	1	1	1	1	1	1	0	0
2	1	0	1	0	0	0	1	0
3	1	1	0	1	0	0	0	0
4	0	0	1	1	0	0	0	0
5	0	0	0	0	1	0	0	0
6	1	1	0	0	0	1	0	0
7	1	1	1	1	0	0	0	0
8	0	1	0	0	0	1	1	0
9	1	1	1	1	0	1	0	0
10	0	0	1	0	1	0	0	1



1-itemsets	Suporte
leite	60%
ovos	60%
café	60%
açúcar	50%
fraldas	30%
manteiga	40%
farinha	20%
cerveja	10%



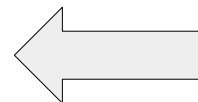
2-itemsets	Suporte
[leite,ovos]	50%
[leite,café]	40%
[leite,açúcar]	40%
[leite,fraldas]	10%
[leite,manteiga]	30%
[ovos,café]	30%
[ovos,açúcar]	40%
[ovos,fraldas]	10%
[ovos,manteiga]	40%
[café,açúcar]	40%
[café,fraldas]	20%
[café,manteiga]	20%
[açúcar,fraldas]	10%
[açúcar,manteiga]	20%
[fraldas,manteiga]	10%

Regras de Associação

- Simulação do Apriori com suporte mínimo 25% e confiança 75%:

Transação	leite	ovos	café	açúcar	fraldas	manteiga	farinha	cerveja
1	1	1	1	1	1	1	0	0
2	1	0	1	0	0	0	1	0
3	1	1	0	1	0	0	0	0
4	0	0	1	1	0	0	0	0
5	0	0	0	0	1	0	0	0
6	1	1	0	0	0	1	0	0
7	1	1	1	1	0	0	0	0
8	0	1	0	0	0	1	1	0
9	1	1	1	1	0	1	0	0
10	0	0	1	0	1	0	0	1

<i>3-itemsets</i>	Suporte
[leite,ovos,café]	30%
[leite,ovos,açúcar]	40%
[leite,ovos,manteiga]	30%
[leite,café,açúcar]	30%
[leite,café,manteiga]	20%
[leite,açúcar,manteiga]	20%
[ovos,café,açúcar]	30%
[ovos,café,manteiga]	20%
[ovos,açúcar,manteiga]	20%
[café,açúcar,manteiga]	20%



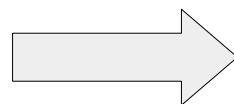
<i>2-itemsets</i>	Suporte
[leite,ovos]	50%
[leite,café]	40%
[leite,açúcar]	40%
[leite,fraldas]	10%
[leite,manteiga]	30%
[ovos,café]	30%
[ovos,açúcar]	40%
[ovos,fraldas]	10%
[ovos,manteiga]	40%
[café,açúcar]	40%
[café,fraldas]	20%
[café,manteiga]	20%
[açúcar,fraldas]	10%
[açúcar,manteiga]	20%
[fraldas,manteiga]	10%

Regras de Associação

- Simulação do Apriori com suporte mínimo 25% e confiança 75%:

Transação	leite	ovos	café	açúcar	fraldas	manteiga	farinha	cerveja
1	1	1	1	1	1	1	0	0
2	1	0	1	0	0	0	1	0
3	1	1	0	1	0	0	0	0
4	0	0	1	1	0	0	0	0
5	0	0	0	0	1	0	0	0
6	1	1	0	0	0	1	0	0
7	1	1	1	1	0	0	0	0
8	0	1	0	0	0	1	1	0
9	1	1	1	1	0	1	0	0
10	0	0	1	0	1	0	0	1

3-itemsets	Suporte
[leite,ovos,café]	30%
[leite,ovos,açúcar]	40%
[leite,ovos,manteiga]	30%
[leite,café,açúcar]	30%
[leite,café,manteiga]	20%
[leite,açúcar,manteiga]	20%
[ovos,café,açúcar]	30%
[ovos,café,manteiga]	20%
[ovos,açúcar,manteiga]	20%
[café,açúcar,manteiga]	20%



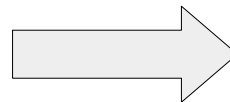
4-itemsets	Suporte
[leite,ovos,café,açúcar]	30%
[leite,ovos,café,manteiga]	20%
[leite,ovos,açúcar,manteiga]	20%
[leite,café,açúcar,manteiga]	20%
[ovos,café,açúcar,manteiga]	20%

Regras de Associação

- Simulação do Apriori com suporte mínimo 25% e confiança 75%:

Transação	leite	ovos	café	açúcar	fraldas	manteiga	farinha	cerveja
1	1	1	1	1	1	1	0	0
2	1	0	1	0	0	0	1	0
3	1	1	0	1	0	0	0	0
4	0	0	1	1	0	0	0	0
5	0	0	0	0	1	0	0	0
6	1	1	0	0	0	1	0	0
7	1	1	1	1	0	0	0	0
8	0	1	0	0	0	1	1	0
9	1	1	1	1	0	1	0	0
10	0	0	1	0	1	0	0	1

2-itemsets	Suporte
[leite,ovos]	50%
[leite,café]	40%
[leite,açúcar]	40%
[leite,fraldas]	10%
[leite,manteiga]	30%
[ovos,café]	30%
[ovos,açúcar]	40%
[ovos,fraldas]	10%
[ovos,manteiga]	40%
[café,açúcar]	40%
[café,fraldas]	20%
[café,manteiga]	20%
[açúcar,fraldas]	10%
[açúcar,manteiga]	20%
[fraldas,manteiga]	10%

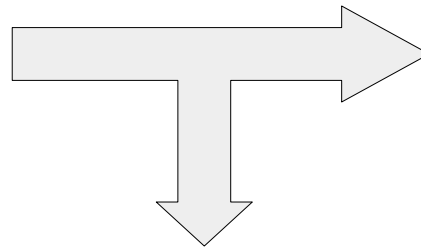


Regra	Suporte	Confiança
[ovos → leite]	50%	83%
[leite → ovos]	50%	83%
[café → leite]	40%	66%
[leite → café]	40%	66%
[açúcar → leite]	40%	80%
[leite → açúcar]	40%	66%
[manteiga → leite]	30%	75%
[leite → manteiga]	30%	50%
[café → ovos]	30%	50%
[ovos → café]	30%	50%
[açúcar → ovos]	40%	80%
[ovos → açúcar]	40%	66%
[manteiga → ovos]	40%	100%
[ovos → manteiga]	40%	66%
[açúcar → café]	40%	80%
[café → açúcar]	40%	66%

Regras de Associação

- Simulação do Apriori com suporte mínimo 25% e confiança 75%:

3-itemsets	Suporte
[leite,ovos,café]	30%
[leite,ovos,açúcar]	40%
[leite,ovos,manteiga]	30%
[leite,café,açúcar]	30%
[leite,café,manteiga]	20%
[leite,açúcar,manteiga]	20%
[ovos,café,açúcar]	30%
[ovos,café,manteiga]	20%
[ovos,açúcar,manteiga]	20%
[café,açúcar,manteiga]	20%



Regra	Suporte	Confiança
[café, ovos → leite]	30%	100%
[ovos, café → leite]	30%	100%
[ovos, leite → café]	30%	60%
[café, leite → ovos]	30%	75%
[leite, café → ovos]	30%	75%
[leite, ovos → café]	30%	60%
[açúcar, ovos → leite]	40%	100%
[ovos, açúcar → leite]	40%	100%
[ovos, leite → açúcar]	40%	80%
[açúcar, leite → ovos]	40%	100%
[leite, açúcar → ovos]	40%	100%
[leite, ovos → açúcar]	40%	80%

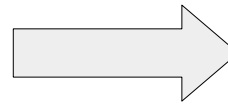
Regra	Suporte	Confiança
[manteiga, ovos → leite]	30%	75%
[ovos, manteiga → leite]	30%	75%
[ovos, leite → manteiga]	30%	60%
[manteiga, leite → ovos]	30%	100%
[leite, manteiga → ovos]	30%	100%
[leite, ovos → manteiga]	30%	60%
[açúcar, café → leite]	30%	75%
[café, açúcar → leite]	30%	75%
[café, leite → açúcar]	30%	75%
[açúcar, leite → café]	30%	75%
[leite, açúcar → café]	30%	75%
[leite, café → açúcar]	30%	75%
[açúcar, café → ovos]	30%	75%
[café, açúcar → ovos]	30%	75%
[café, ovos → açúcar]	30%	100%
[açúcar, ovos → café]	30%	75%
[ovos, açúcar → café]	30%	75%
[ovos, café → açúcar]	30%	100%

Regras de Associação

- Simulação do Apriori com suporte mínimo 25% e confiança 75%:

Transação	leite	ovos	café	açúcar	fraldas	manteiga	farinha	cerveja
1	1	1	1	1	1	1	0	0
2	1	0	1	0	0	0	1	0
3	1	1	0	1	0	0	0	0
4	0	0	1	1	0	0	0	0
5	0	0	0	0	1	0	0	0
6	1	1	0	0	0	1	0	0
7	1	1	1	1	0	0	0	0
8	0	1	0	0	0	1	1	0
9	1	1	1	1	0	1	0	0
10	0	0	1	0	1	0	0	1

4-itemsets	Suporte
[leite,ovos,café,açúcar]	30%
[leite,ovos,café,manteiga]	20%
[leite,ovos,açúcar,manteiga]	20%
[leite,café,açúcar,manteiga]	20%
[ovos,café,açúcar,manteiga]	20%



Regra	Suporte	Confiança
[açúcar, café, ovos → leite]	30%	100%
[café, açúcar, ovos → leite]	30%	100%
[café, ovos, açúcar → leite]	30%	100%
[café, ovos, leite → açúcar]	30%	100%
[açúcar, ovos, café → leite]	30%	100%
[ovos, açúcar, café → leite]	30%	100%
[ovos, café, açúcar → leite]	30%	100%
[ovos, café, leite → açúcar]	30%	100%
[açúcar, ovos, leite → café]	30%	75%
[ovos, açúcar, leite → café]	30%	75%
[ovos, leite, açúcar → café]	30%	75%
[ovos, leite, café → açúcar]	30%	100%
[açúcar, café, leite → ovos]	30%	100%
[café, açúcar, leite → ovos]	30%	100%
[café, leite, açúcar → ovos]	30%	100%
[café, leite, ovos → açúcar]	30%	100%
[açúcar, leite, café → ovos]	30%	100%
[leite, açúcar, café → ovos]	30%	100%
[leite, café, açúcar → ovos]	30%	100%
[leite, café, ovos → açúcar]	30%	100%
[açúcar, leite, ovos → café]	30%	75%
[leite, açúcar, ovos → café]	30%	75%
[leite, ovos, açúcar → café]	30%	75%
[leite, ovos, café → açúcar]	30%	100%

Não vimos casos de conseqüentes múltiplos (ex. [ovos, leite → café, açúcar] tem 60% de confiança).

Não calculamos associações negativas (ex. [açúcar → não cerveja], com suporte 50% e confiança 100%).

- Muitos problemas podem ser representados em matrizes binárias (ou variantes): enorme aplicabilidade.
- Associações negativas podem ser tão importantes quanto positivas.
- **Cuidado!** Na vida real as combinações e permutações podem ser muitas, e as regras quase redundantes!
 - Muitas regras geradas: **mineração de regras.**

- Muitas outras técnicas podem ser usadas:
- Pesquisa Operacional, Inteligência Artificial e outras.
- Outros modelos de redes neurais, *Rough Sets*, *Support Vector Machines*, etc.
- Técnicas de algoritmos genéticos, *Particle Swarm Optimization*, etc.
- Técnicas baseadas em sistemas imunes artificiais, biologia/vida artificial, etc.

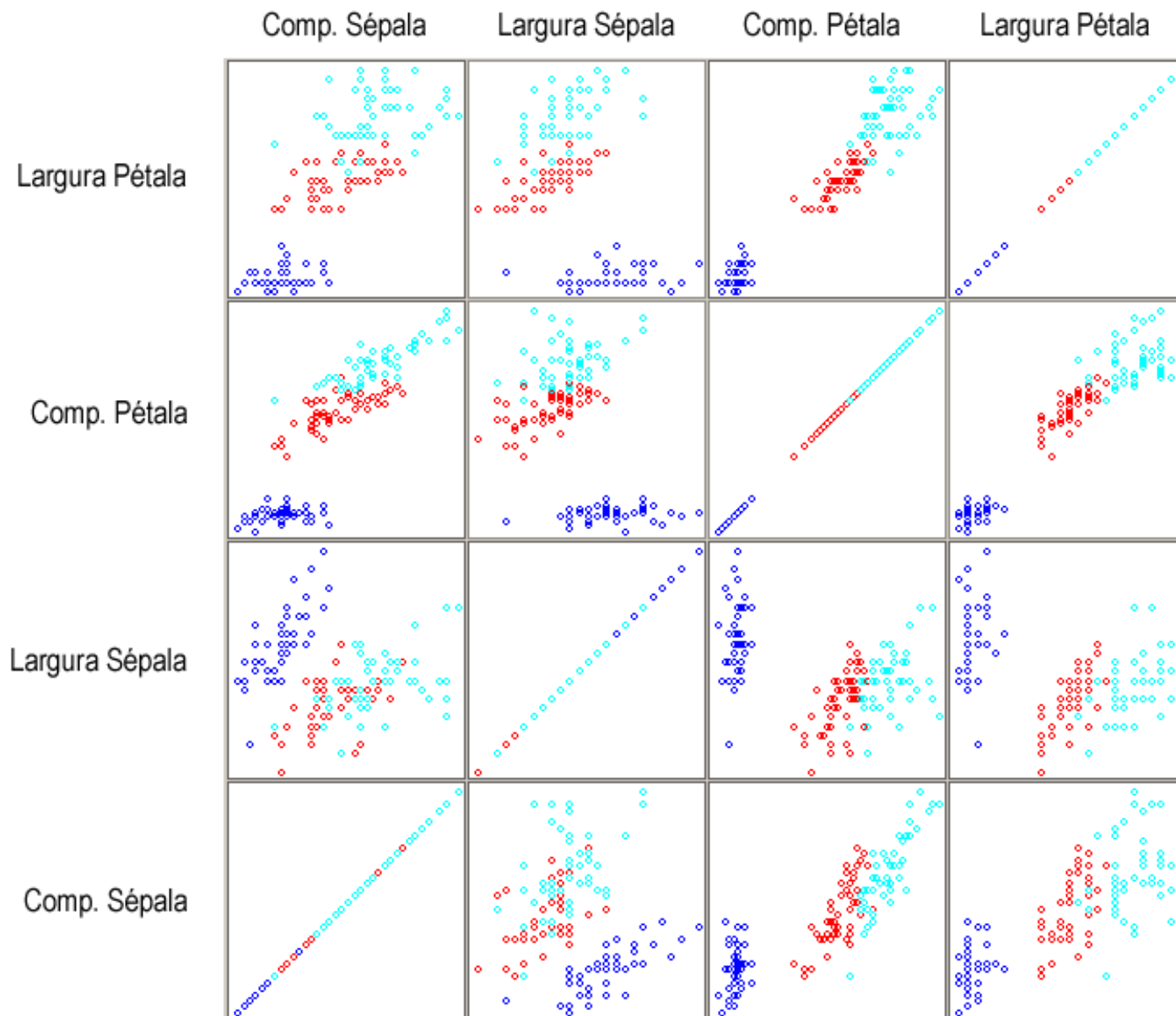
Visualização

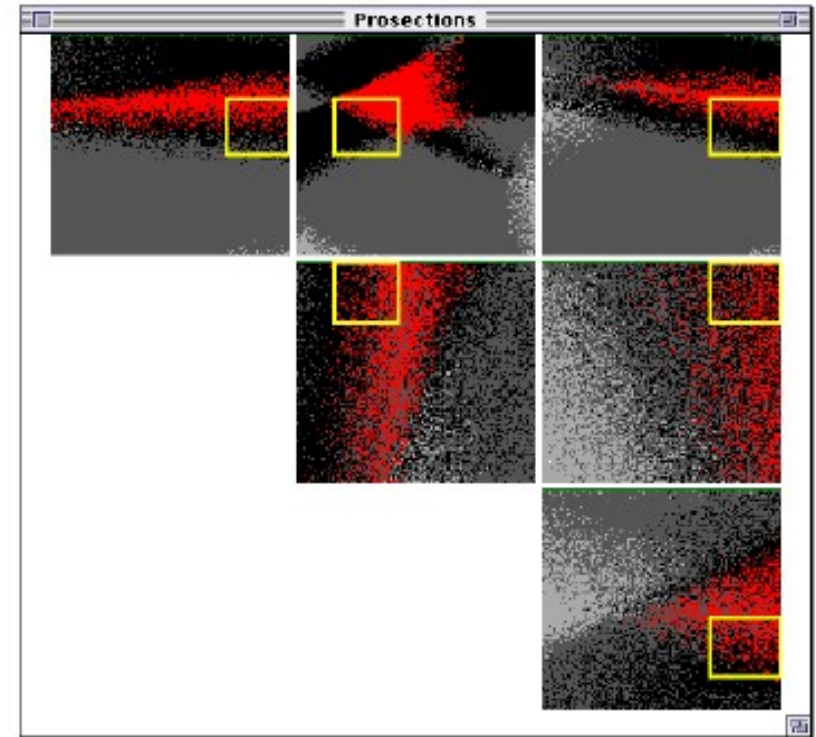
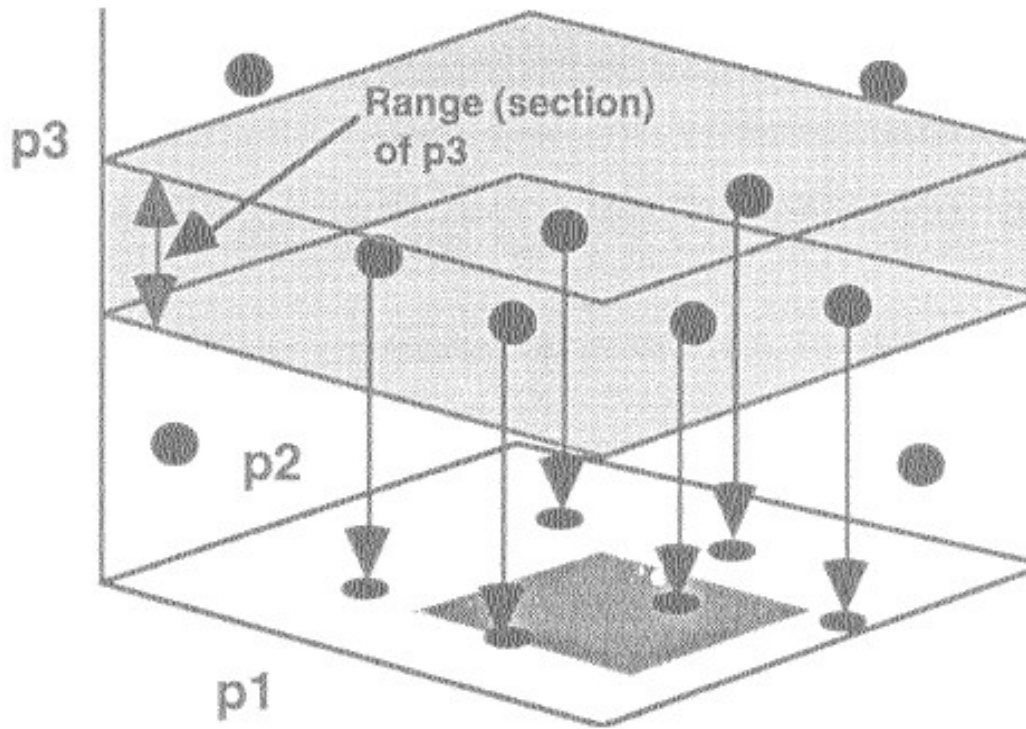
- Pode ser usada no início do processo de mineração...
 - Para ter uma idéia da distribuição dos dados ou de relações entre os dados para formulação de hipóteses.
 - Para selecionar atributos ou regiões de dados.
 - Para ter uma idéia de que tipos de algoritmos podem trazer resultados para estes dados.
- Pode ser usada no final do processo de mineração...
 - Para ver as informações/regras/grupos/etc. obtidos: sumarização do conhecimento.
 - Para ver distribuições contextualizadas (isto é, com conhecimento adicional adquirido integrado).
 - Análise Explorativa / Análise Confirmativa / Apresentação

- Desafios:
 - Métodos e técnicas específicos.
 - Limitações de hardware (humano e máquina!)
 - Número de dimensões (atributos) dos dados.
 - Número de instâncias para visualização.
 - “Empilhamento” e ordenação.
- Vantagens:
 - Inerentemente exploratório.
 - Padrões detectados mesmo que não sejam explicáveis!

- Idéia básica: transformações e projeções usando arranjos em um número menor de dimensões.
 - *Scatterplot Matrices*: K atributos em grade $K \times K$.
 - *Prosection Views*: *Scatterplot Matrices* com mecanismos de seleção (*drill-down*).
 - *Parallel Coordinates*: muito bom para dados mistos, requer exploração e rearranjos.
 - Visualização com Mapas de Kohonen (SOMs).

Visualização: Scatterplot Matrices



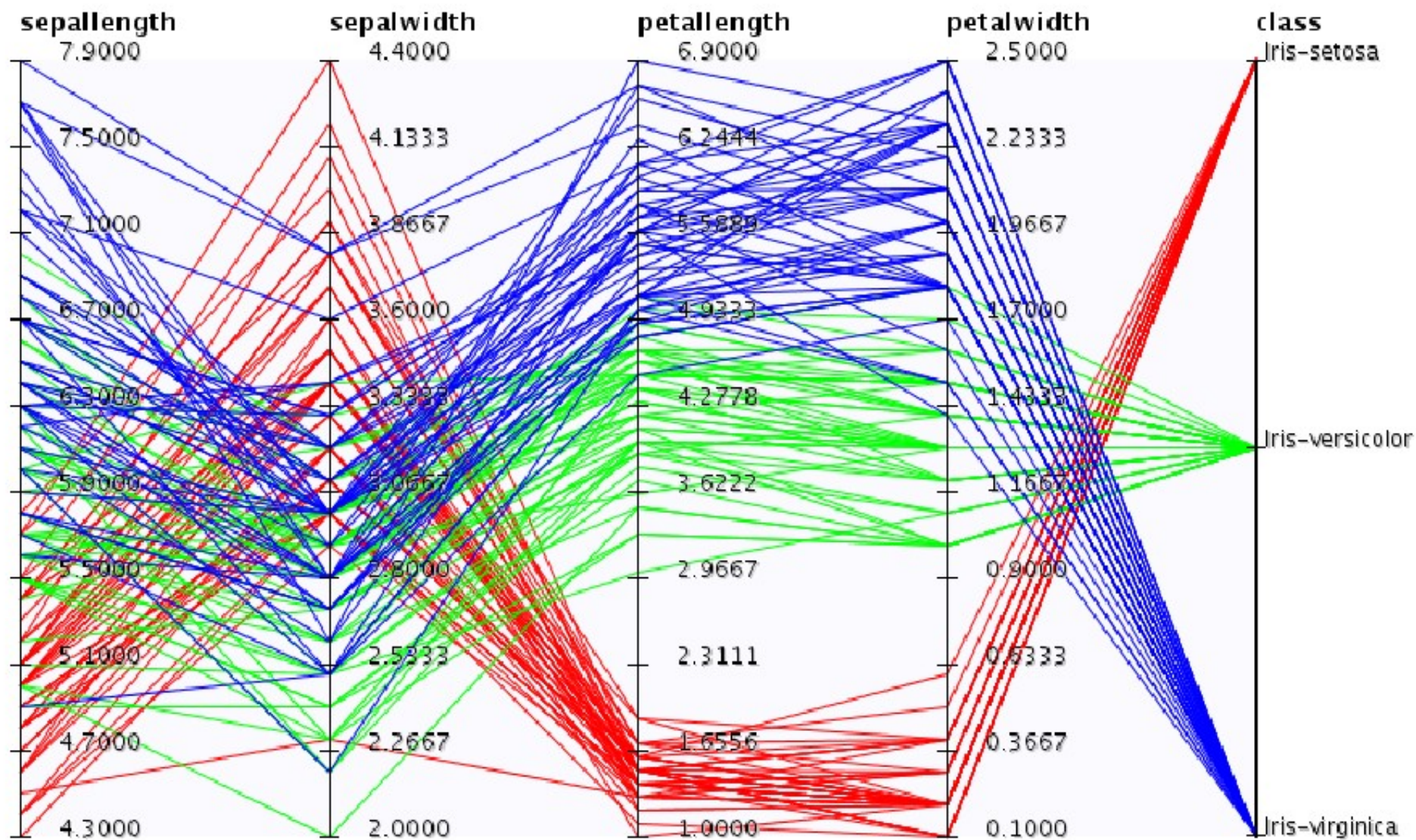


Exemplo de R. Spence, ilustrado no tutorial de Daniel Keim.

Visualização: *Parallel Coordinates*



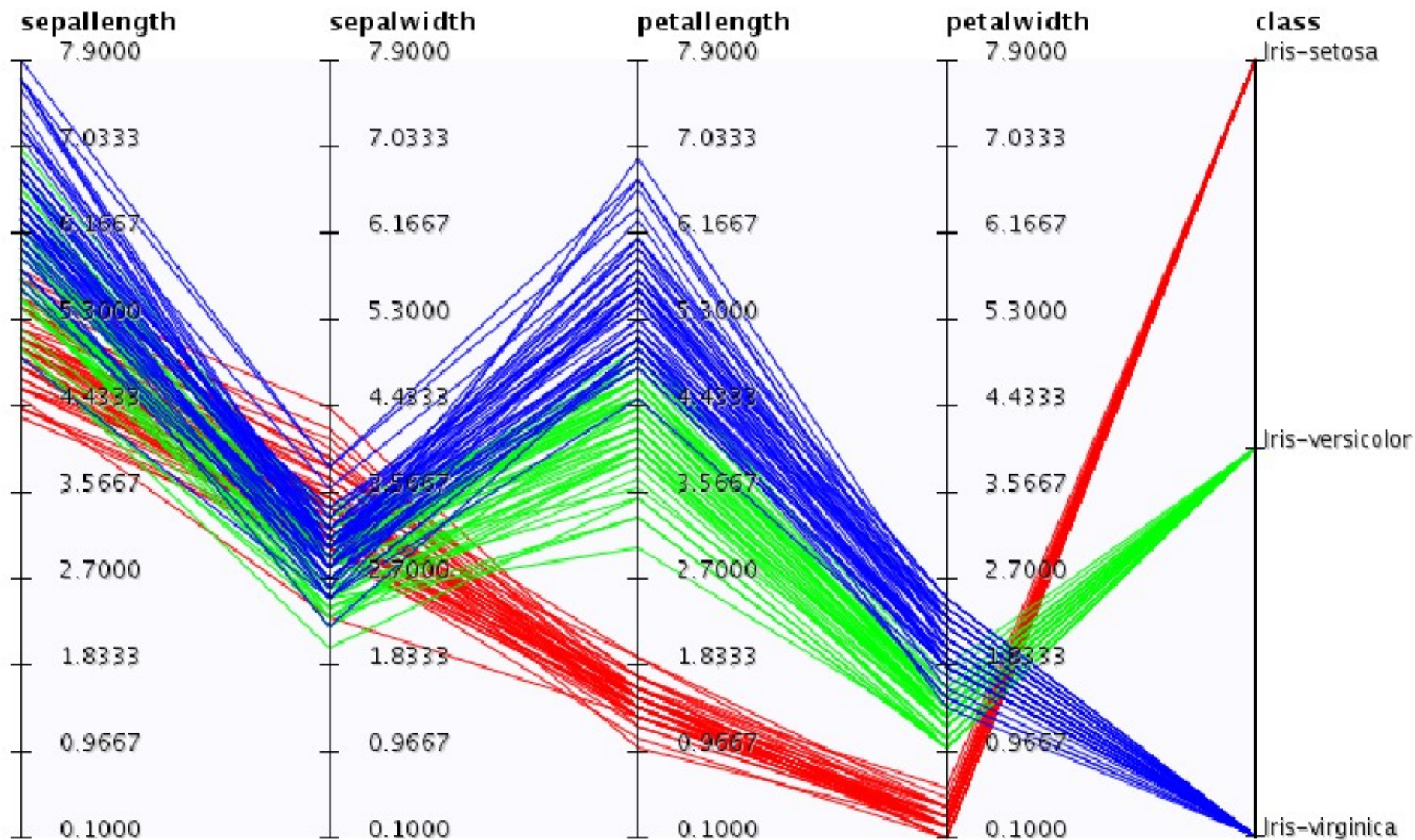
iris



Visualização: *Parallel Coordinates*



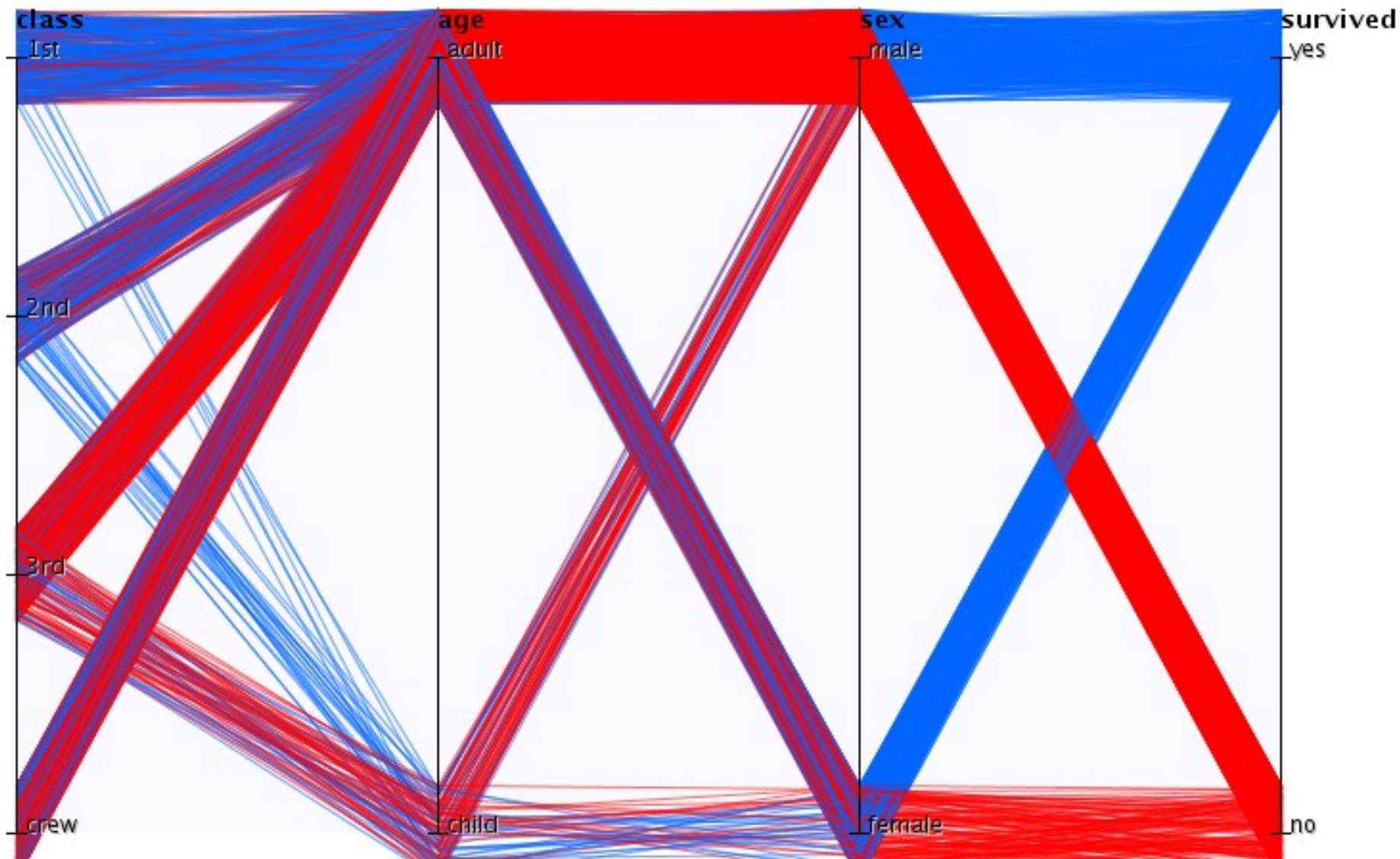
iris



Visualização: *Parallel Coordinates*



Titanic_survivors



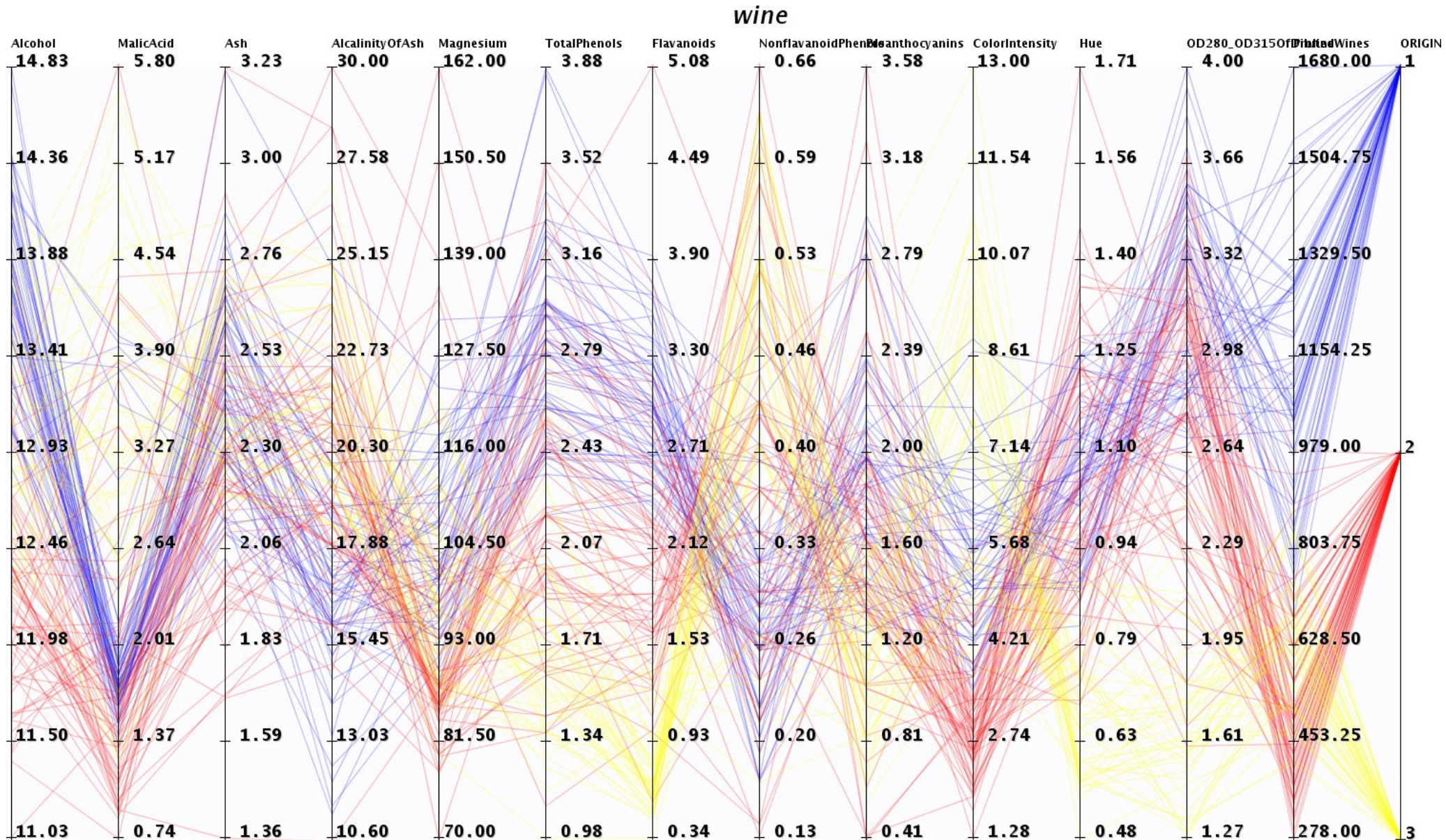
Visualização: *Parallel Coordinates*



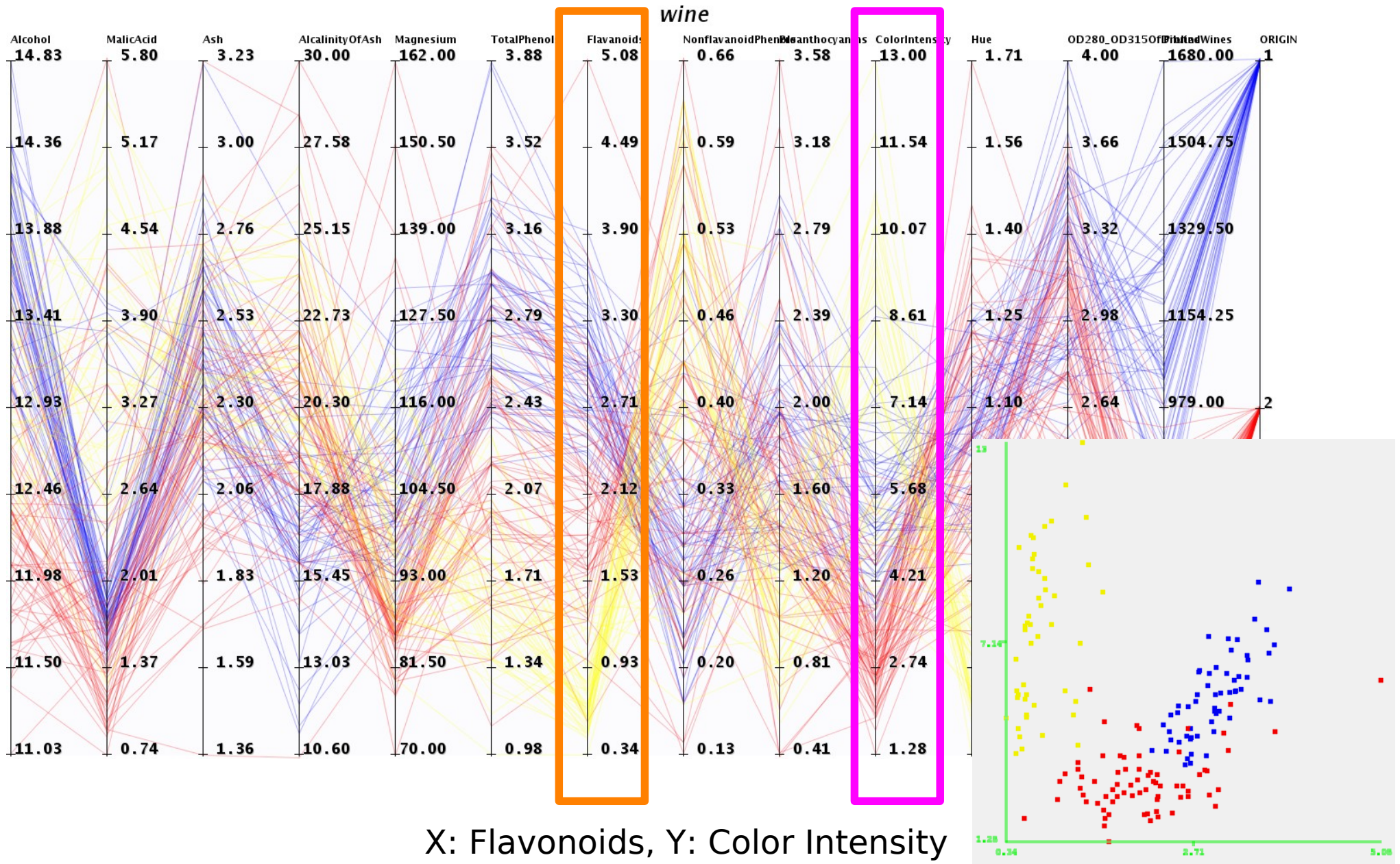
- Origem do vinho a partir de conteúdo físico-químico (13 atributos)
<http://archive.ics.uci.edu/ml/datasets/Wine> (nomes de atributos originais)

No.	Alcohol Numeric	MalicAcid Numeric	Ash Numeric	AlcalinityOfAsh Numeric	Magnesium Numeric	TotalPhenols Numeric	Flavanoids Numeric	NonflavanoidPhenols Numeric	Proanthocyanins Numeric	ColorIntensity Numeric	Hue Numeric	OD280_OD315OfDilutedWines Numeric	Proline Numeric	ORIGIN Nominal
1	14.23	1.71	2.43	15.6	127.0	2.8	3.06	0.28	2.29	5.64	1.04	3.92	106...	1
2	13.2	1.78	2.14	11.2	100.0	2.65	2.76	0.26	1.28	4.38	1.05	3.4	105...	1
3	13.16	2.36	2.67	18.6	101.0	2.8	3.24	0.3	2.81	5.68	1.03	3.17	118...	1
4	14.37	1.95	2.5	16.8	113.0	3.85	3.49	0.24	2.18	7.8	0.86	3.45	148...	1
5	13.24	2.59	2.87	21.0	118.0	2.8	2.69	0.39	1.82	4.32	1.04	2.93	735.0	1
6	14.2	1.76	2.45	15.2	112.0	3.27	3.39	0.34	1.97	6.75	1.05	2.95	145...	1
7	14.39	1.87	2.45	14.6	96.0	2.5	2.52	0.3	1.98	5.25	1.02	3.58	129...	1
8	14.06	2.15	2.61	17.6	121.0	2.6	2.51	0.31	1.25	5.05	1.06	3.58	129...	1
9	14.83	1.64	2.17	14.0	97.0	2.8	2.98	0.29	1.98	5.2	1.08	2.85	104...	1
10	13.86	1.35	2.27	16.0	98.0	2.98	3.15	0.22	1.85	7.22	1.01	3.55	104...	1
11	14.1	2.16	2.3	18.0	105.0	2.95	3.32	0.22	2.38	5.75	1.25	3.17	151...	1
12	14.12	1.48	2.32	16.8	95.0	2.2	2.43	0.26	1.57	5.0	1.17	2.82	128...	1
13	13.75	1.73	2.41	16.0	89.0	2.6	2.76	0.29	1.81	5.6	1.15	2.9	132...	1
14	14.75	1.73	2.39	11.4	91.0	3.1	3.69	0.43	2.81	5.4	1.25	2.73	115...	1
15	14.38	1.87	2.38	12.0	102.0	3.3	3.64	0.29	2.96	7.5	1.2	3.0	154...	1
16	13.63	1.81	2.7	17.2	112.0	2.85	2.91	0.3	1.46	7.3	1.28	2.88	131...	1
17	14.3	1.92	2.72	20.0	120.0	2.8	3.14	0.33	1.97	6.2	1.07	2.65	128...	1
18	13.83	1.57	2.62	20.0	115.0	2.95	3.4	0.4	1.72	6.6	1.13	2.57	113...	1
19	14.19	1.59	2.48	16.5	108.0	3.3	3.93	0.32	1.86	8.7	1.23	2.82	168...	1
20	13.64	3.1	2.56	15.2	116.0	2.7	3.03	0.17	1.66	5.1	0.96	3.36	845.0	1
21	14.06	1.63	2.28	16.0	126.0	3.0	3.17	0.24	2.1	5.65	1.09	3.71	780.0	1
22	12.93	3.8	2.65	18.6	102.0	2.41	2.41	0.25	1.98	4.5	1.03	3.52	770.0	1
23	13.71	1.86	2.36	16.6	101.0	2.61	2.88	0.27	1.69	3.8	1.11	4.0	103...	1
24	12.85	1.6	2.52	17.8	95.0	2.48	2.37	0.26	1.46	3.93	1.09	3.63	101...	1
25	13.5	1.81	2.61	20.0	96.0	2.53	2.61	0.28	1.66	3.52	1.12	3.82	845.0	1
26	13.05	2.05	3.22	25.0	124.0	2.63	2.68	0.47	1.92	3.58	1.13	3.2	830.0	1
27	13.39	1.77	2.62	16.1	93.0	2.85	2.94	0.34	1.45	4.8	0.92	3.22	119...	1
28	13.3	1.72	2.14	17.0	94.0	2.4	2.19	0.27	1.35	3.95	1.02	2.77	128...	1
29	13.87	1.9	2.8	19.4	107.0	2.95	2.97	0.37	1.76	4.5	1.25	3.4	915.0	1
30	14.02	1.68	2.21	16.0	96.0	2.65	2.33	0.26	1.98	4.7	1.04	3.59	103...	1
31	13.73	1.5	2.7	22.5	101.0	3.0	3.25	0.29	2.38	5.7	1.19	2.71	128...	1
32	13.58	1.66	2.36	19.1	106.0	2.86	3.19	0.22	1.95	6.9	1.09	2.88	151...	1
33	13.68	1.83	2.36	17.2	104.0	2.42	2.69	0.42	1.97	3.84	1.23	2.87	990.0	1
34	13.76	1.53	2.7	19.5	132.0	2.95	2.74	0.5	1.35	5.4	1.25	3.0	123...	1
35	13.51	1.8	2.65	19.0	110.0	2.35	2.53	0.29	1.54	4.2	1.1	2.87	109...	1
36	13.48	1.81	2.41	20.5	100.0	2.7	2.98	0.26	1.86	5.1	1.04	3.47	920.0	1
37	13.28	1.64	2.84	15.5	110.0	2.6	2.68	0.34	1.36	4.6	1.09	2.78	880.0	1
38	13.05	1.65	2.55	18.0	98.0	2.45	2.43	0.29	1.44	4.25	1.12	2.51	110...	1
39	13.07	1.5	2.1	15.5	98.0	2.4	2.64	0.28	1.37	3.7	1.18	2.69	102...	1

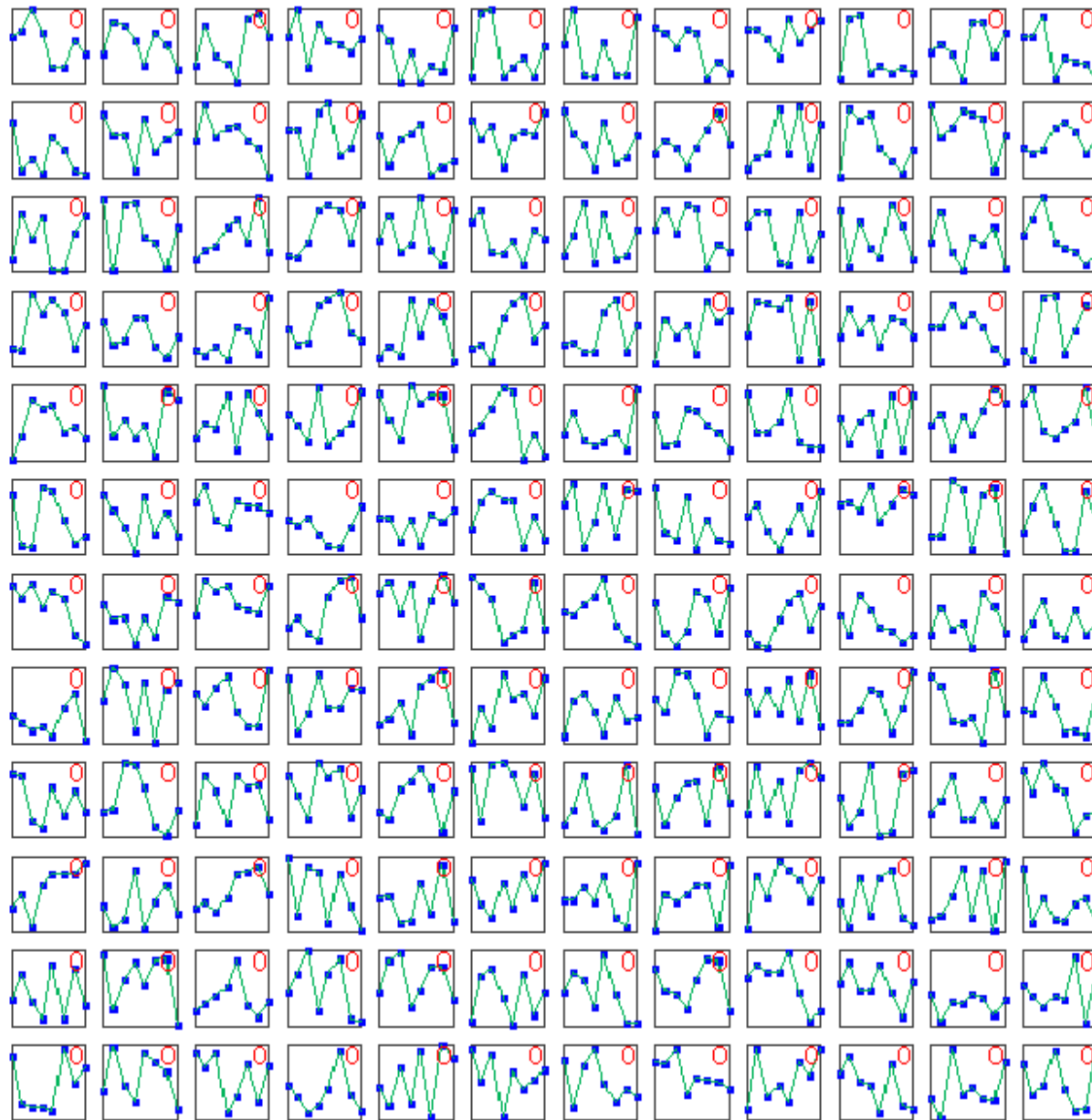
Visualização: *Parallel Coordinates*



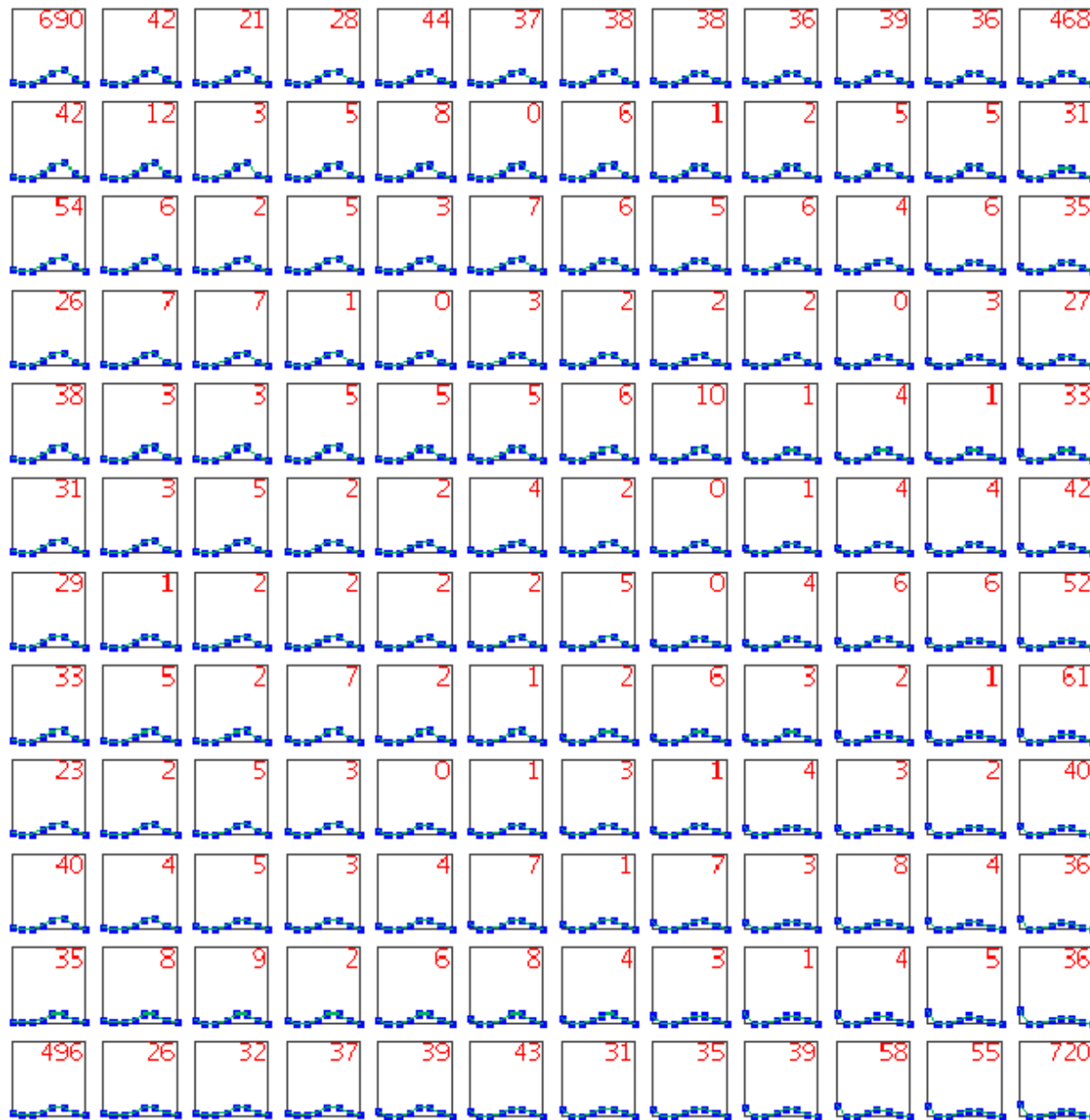
Visualização: *Parallel Coordinates*



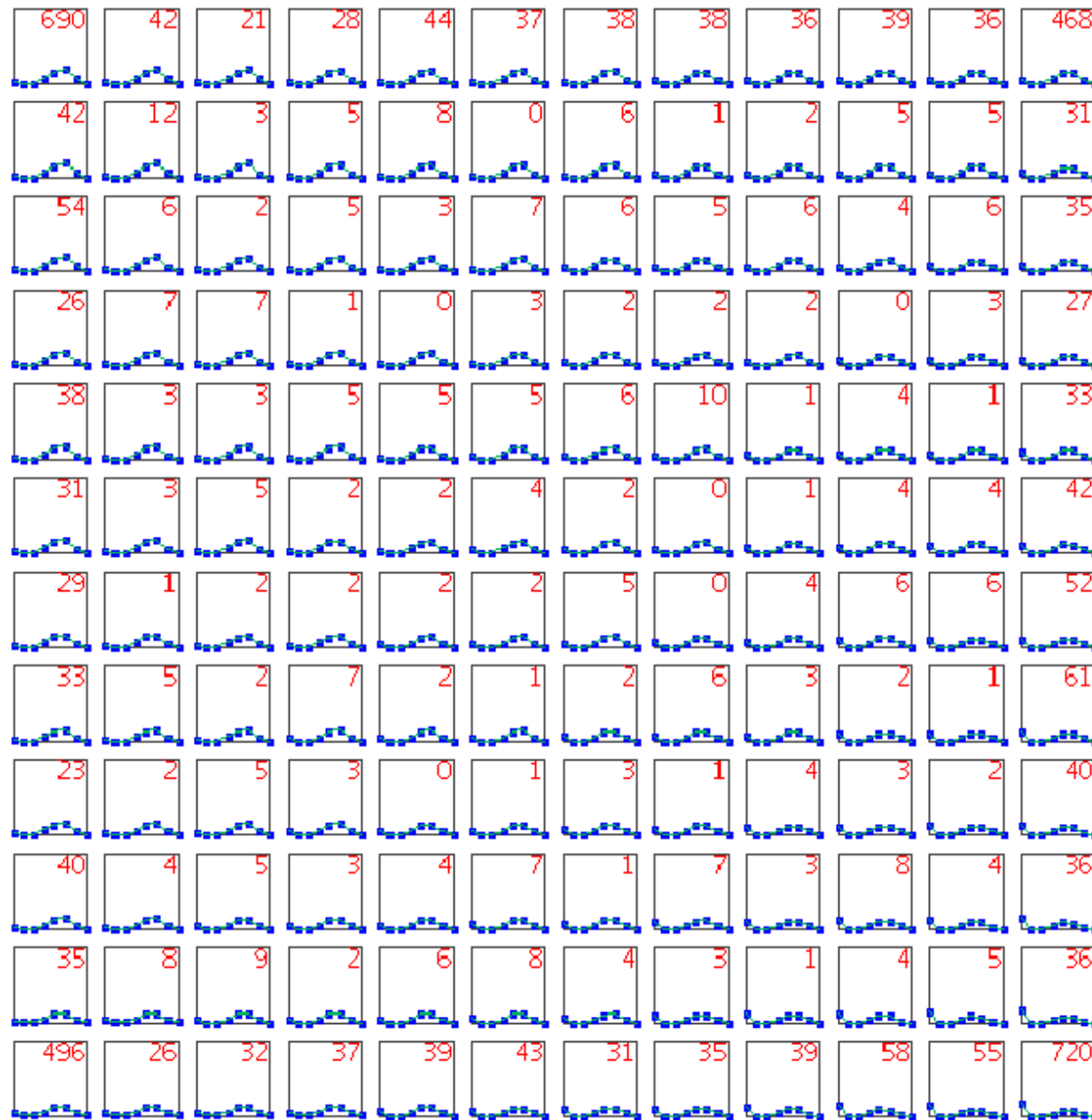
Visualização: *Self-Organizing Maps*



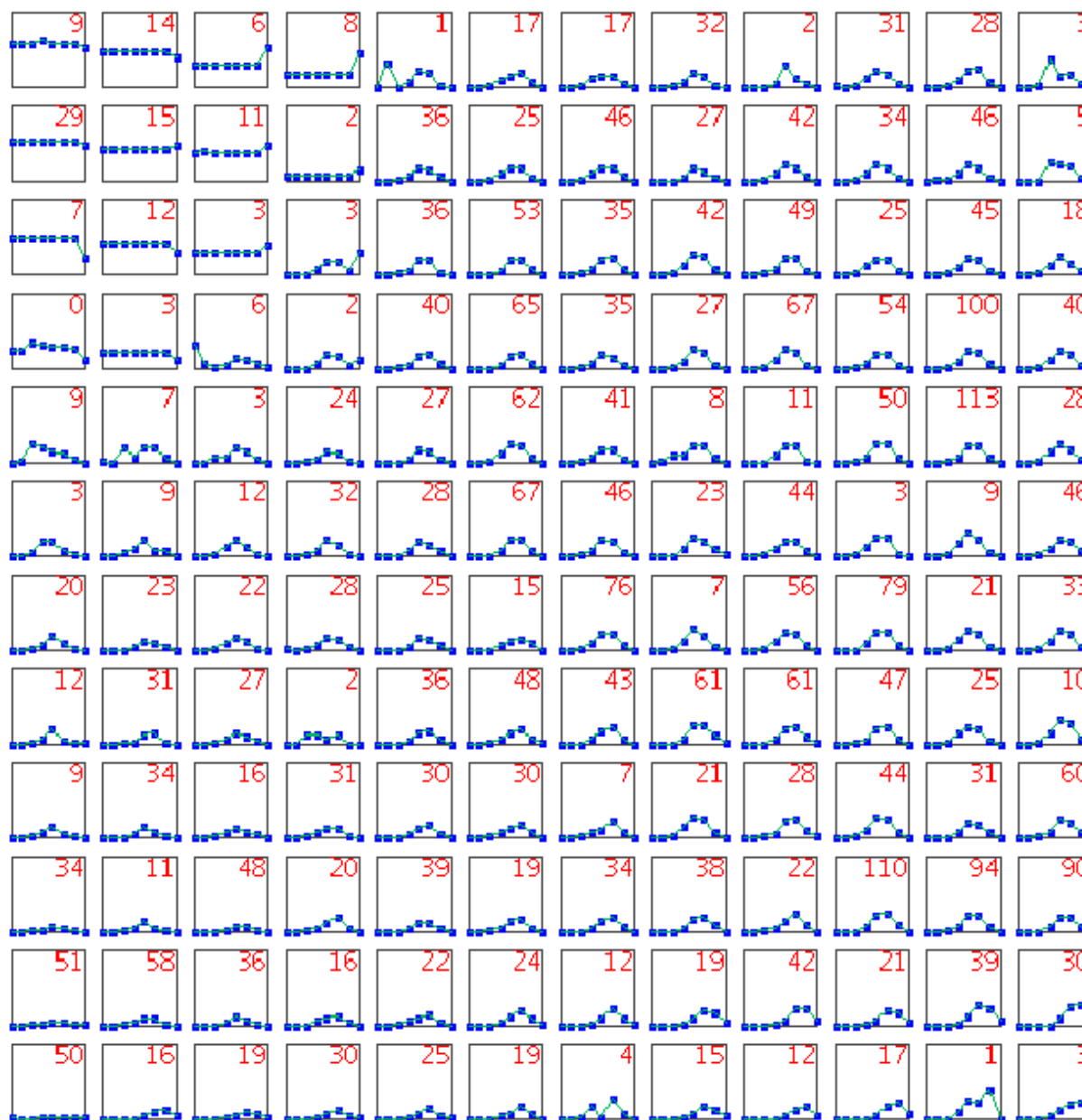
Visualização: Self-Organizing Maps



Visualização: Self-Organizing Maps

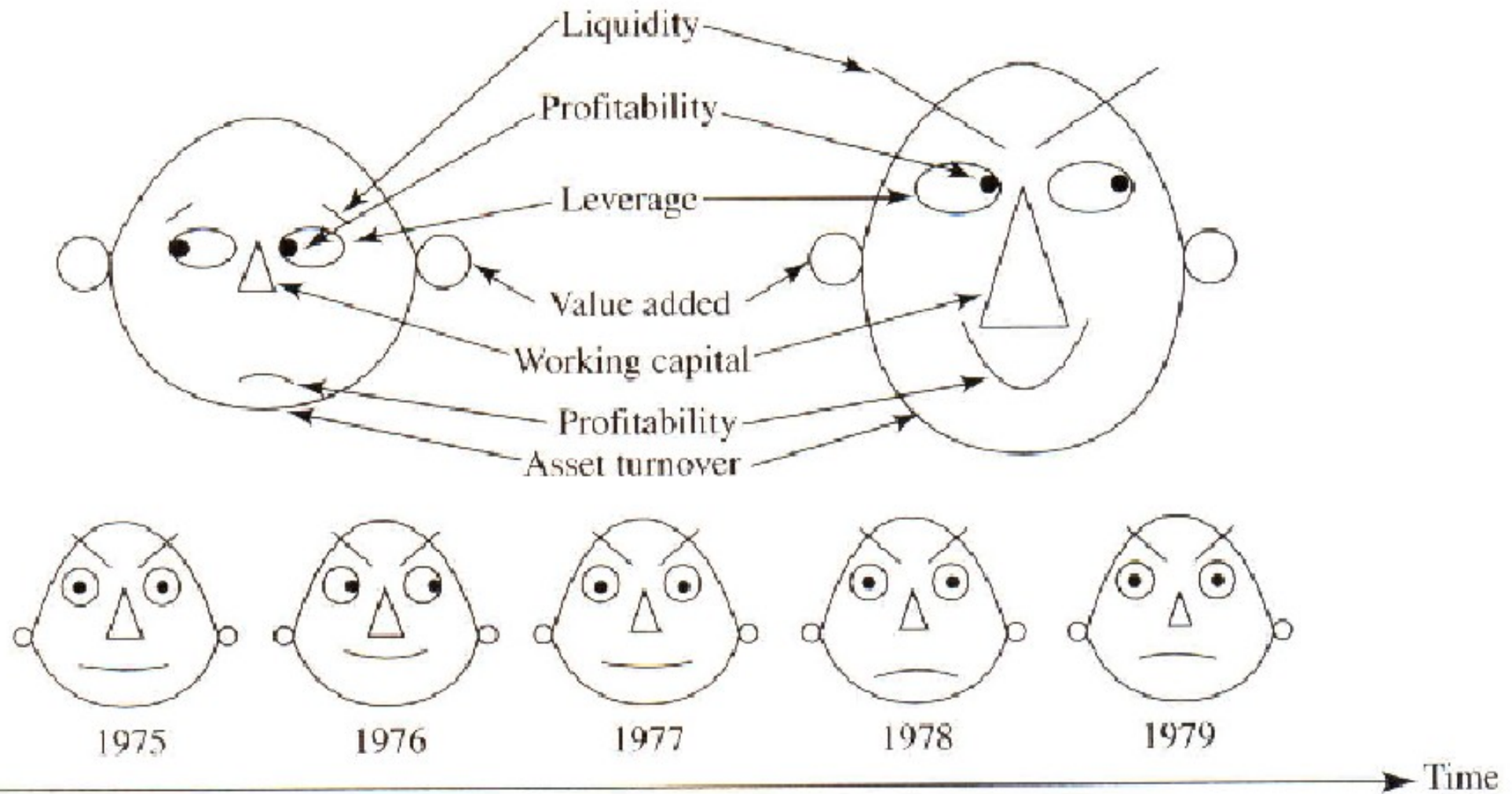


Visualização: Self-Organizing Maps

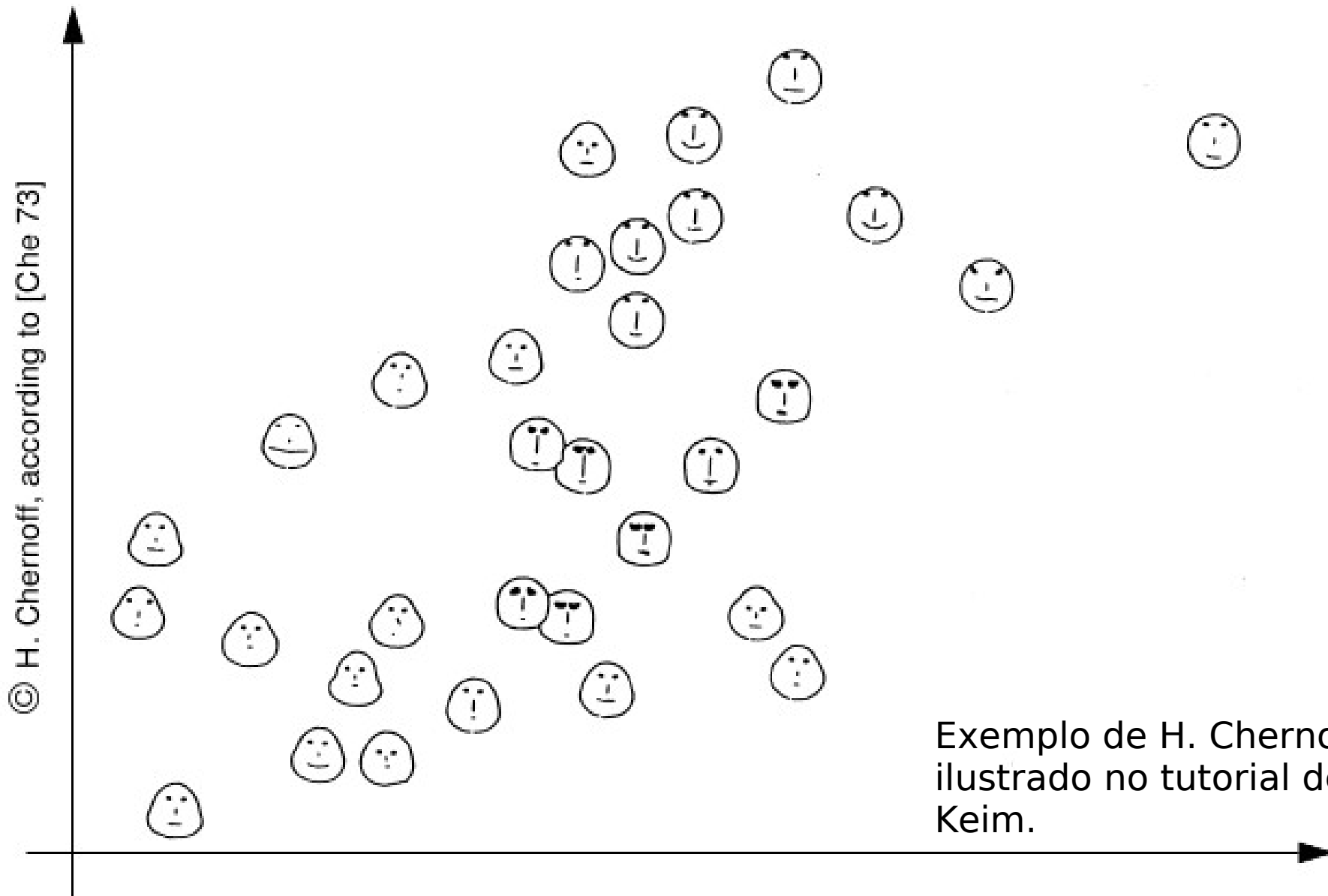


- Idéia básica: usamos duas dimensões para mostrar ícones que representam outras dimensões adicionais.
 - Interpretação deve ser feita com legendas!
 - *Chernoff faces*: atributos das faces (geometria, olhos, excentricidade, curvaturas, etc.) representam outras dimensões.
 - *Stick figures*: dimensões adicionais mapeadas para ângulos e comprimentos de segmentos de retas.

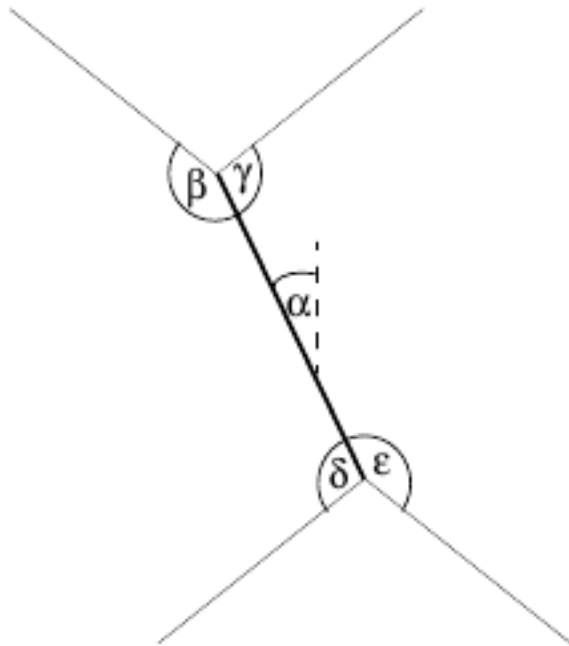
Visualização: Chernoff Faces



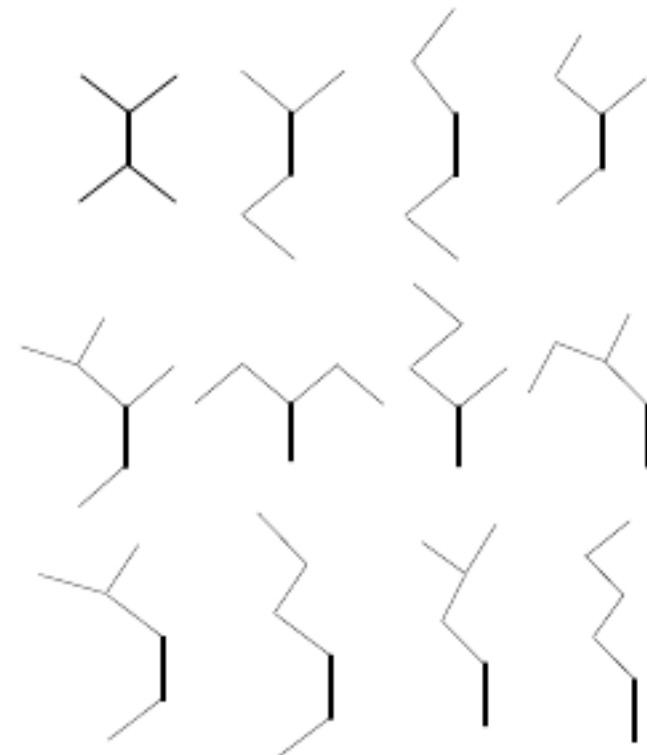
Visualização: Chernoff Faces



Exemplo de H. Chernoff, ilustrado no tutorial de Daniel Keim.



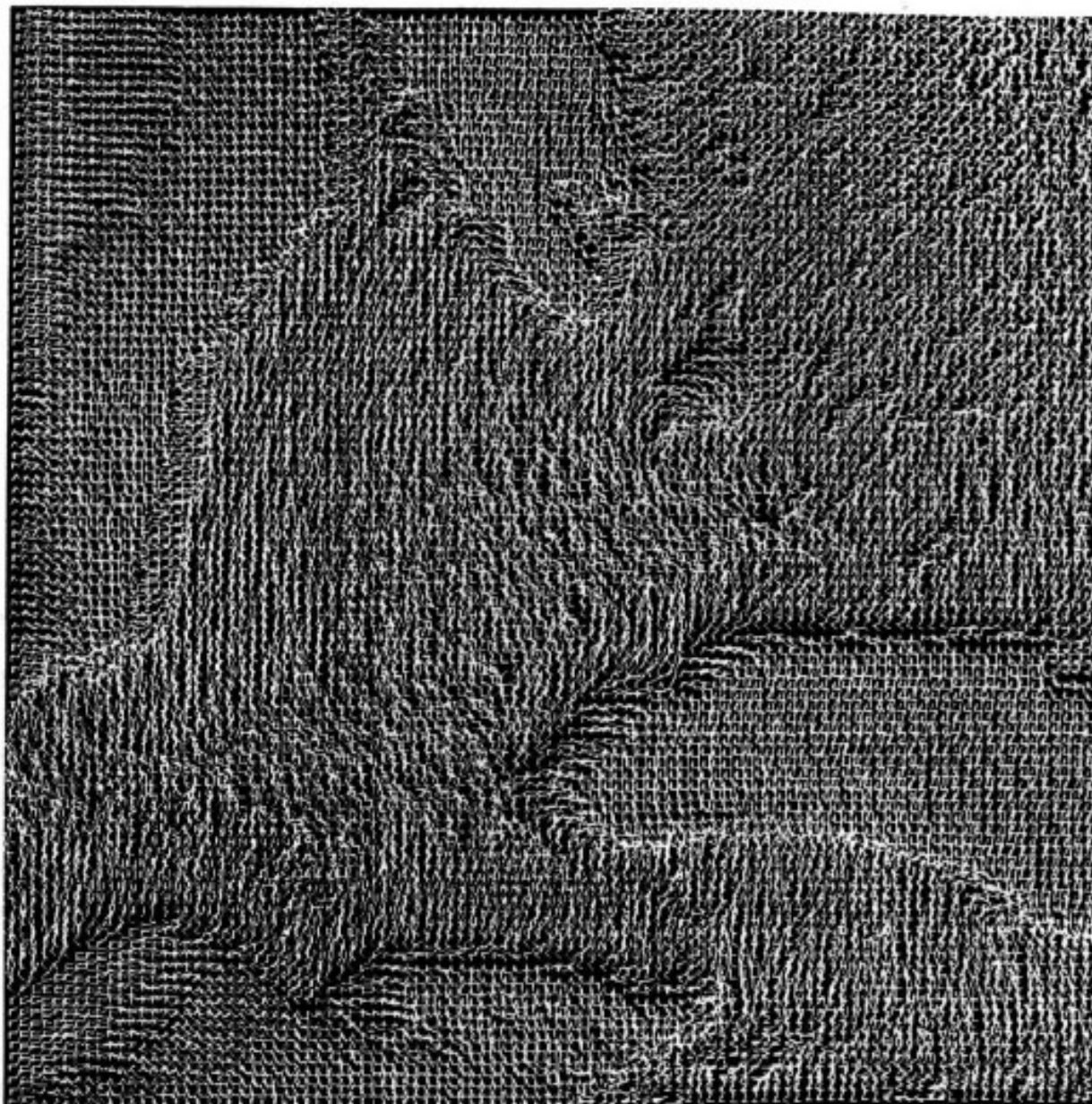
Stick Figure Icon



A Family of Stick Figures

- Uso de duas dimensões mais textura

Fonte: Tutorial de Daniel Keim.

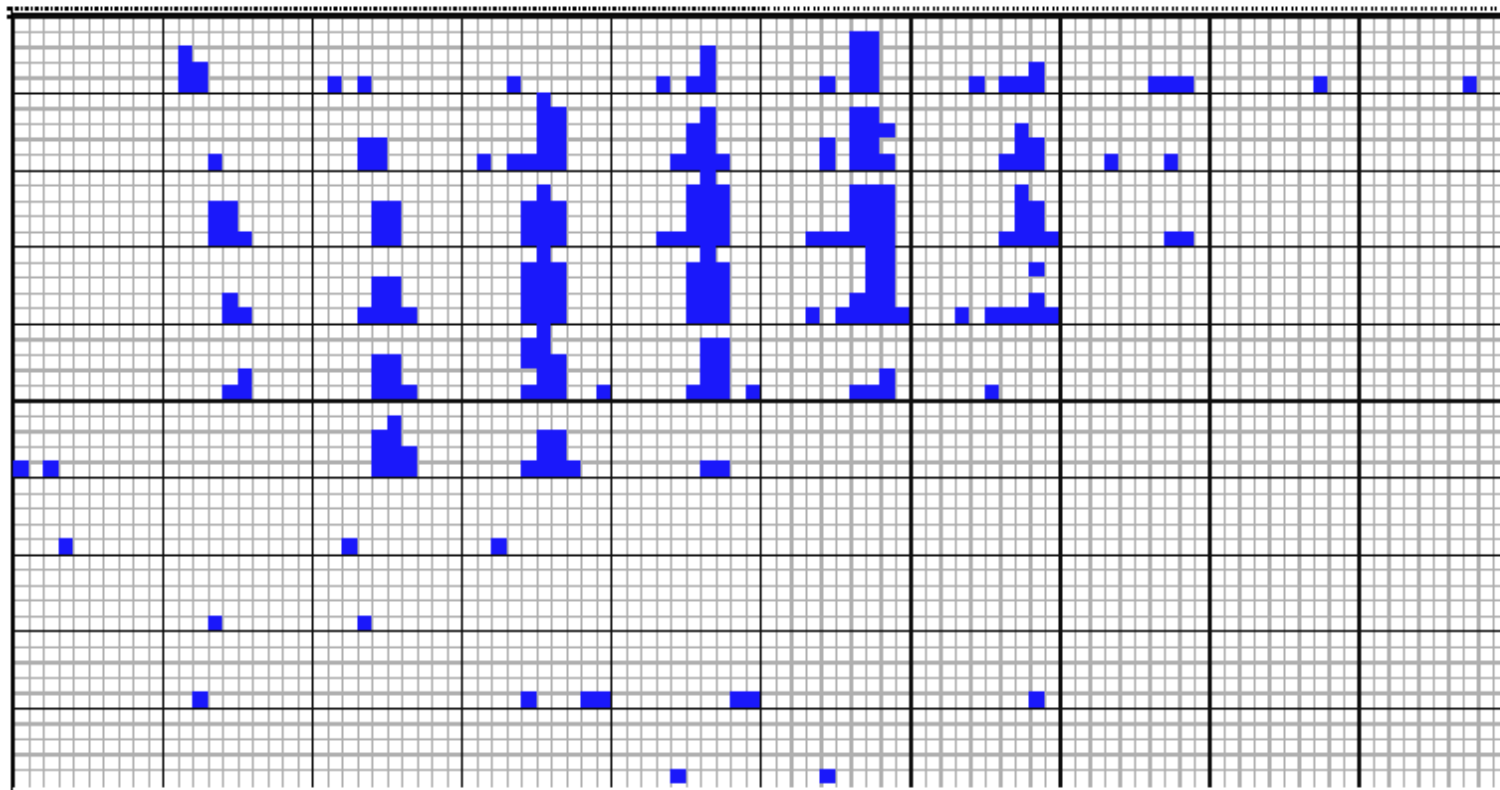
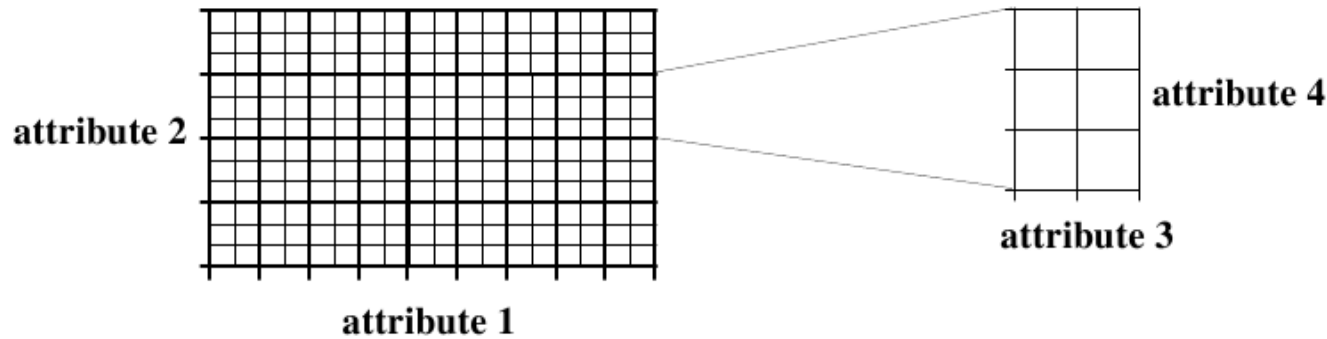


5-dim. image
data from the
great lake region

Fonte: Tutorial de Daniel Keim.

- Idéia básica: particionamento das dimensões em subdimensões.
 - *Dimensional Stacking*: Particionamento de N dimensões em conjuntos de 2 dimensões.
 - *Worlds-within-Worlds*: Particionamento de N dimensões em conjuntos de 3 dimensões.
 - *Treemap*: Preenche área de visualização alternando eixos X e Y.
 - *Cone Trees*: Visualização interativa de dados hierárquicos.
 - *InfoCube*: Visualização hierárquica com 3D e transparência.

Visualização: *Dimensional Stacking*

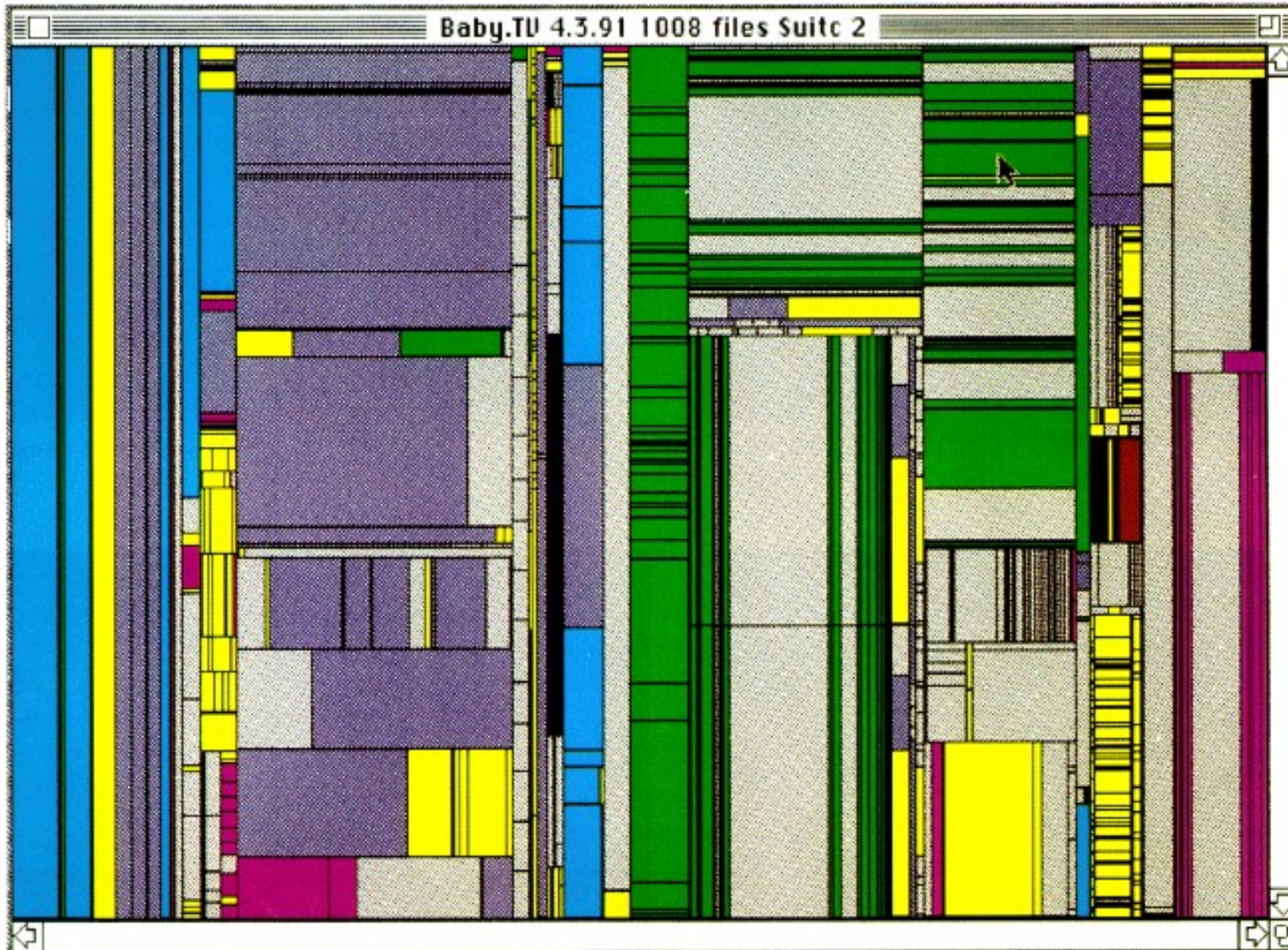


visualization of oil mining data with longitude and latitude mapped to the outer x-, y- axes and ore grade and depth mapped to the inner x-, y- axes

used by permission of M. Ward, Worcester Polytechnic Institute

Fonte: Tutorial de Daniel Keim.

Visualização: *Treemap*



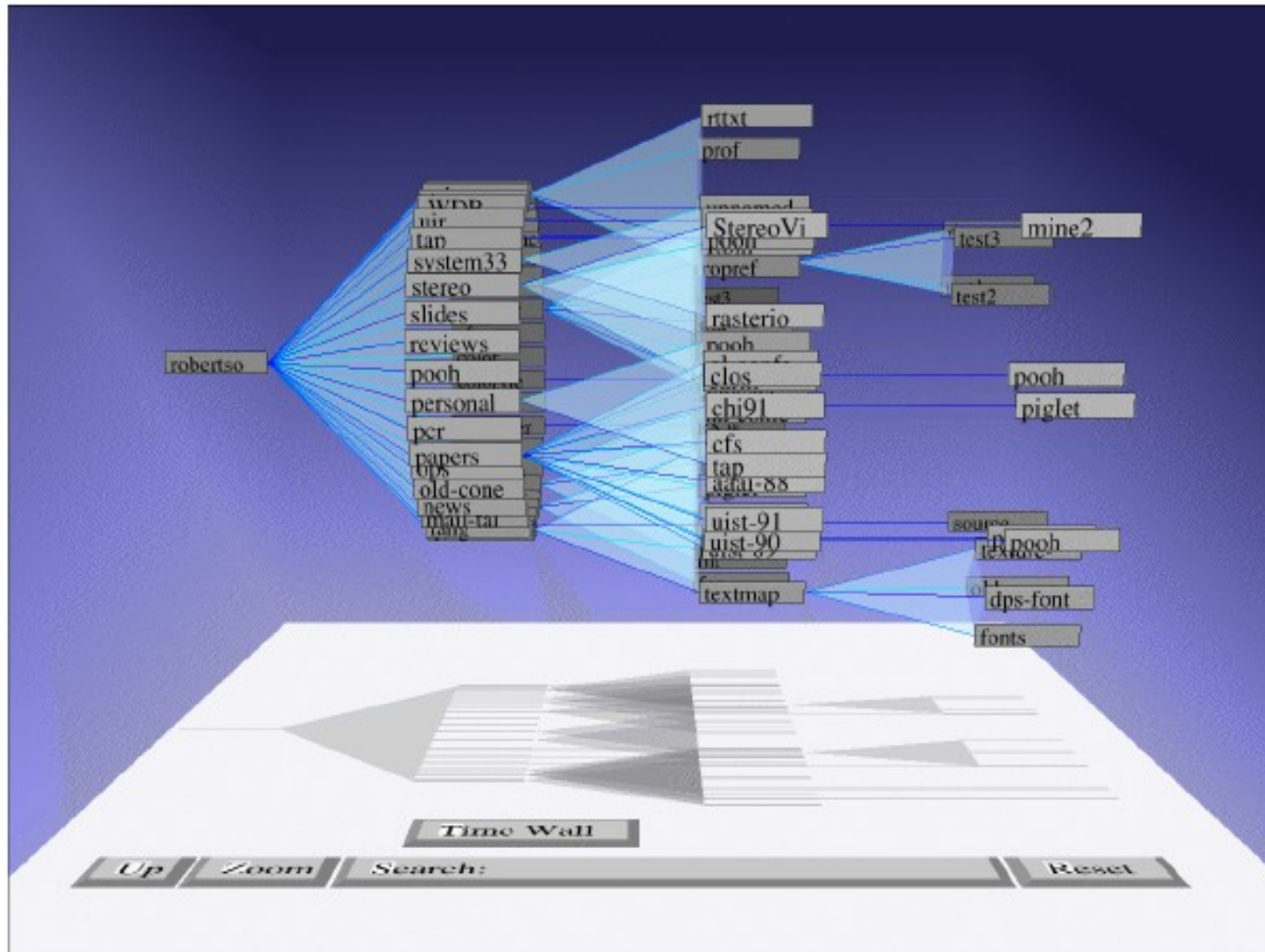
treemap of a
file system
containing about
1000 files

Fonte: Tutorial de Daniel Keim.

Visualização: Cone Trees



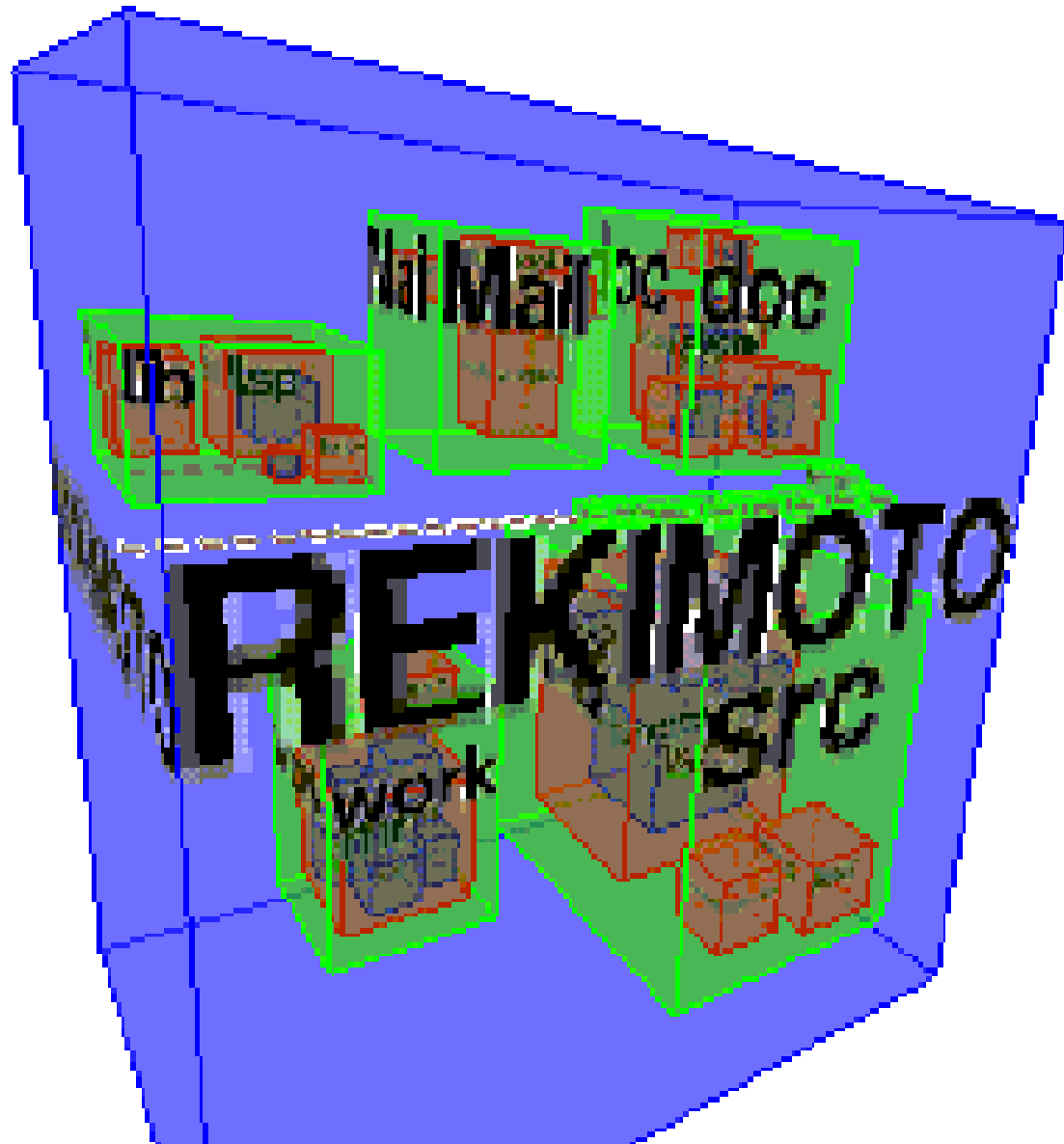
used by permission of S. Card, Xerox PARC



file system structure
visualized as a
cone tree

Fonte: Tutorial de Daniel Keim.

used by permission of J. Rekimoto, Sony CS Lab Inc.



Fonte: Tutorial de Daniel Keim.

Mineração de Dados Multimídia

Introdução

- O que é diferente?
 - Atributos e normalização.
- **Multimídia**: várias fontes de dados correlacionados, de tipos diferentes.
 - Complexidade de extração de atributos para cada mídia mais complexidade de integração.
 - A definição do problema pode corresponder a boa parte da solução.
 - Raros casos de mineração de dados multimídia no sentido estrito...
- Não existe **blind data mining** – muito menos **blind multimedia mining!**



- Alguns tipos de dados multimídia tem metadados associados aos dados de conteúdo.
- **Metadados:** dados organizados, resumidos, de alto nível.
- **Dados de conteúdo:** estruturados de acordo com a mídia, extensos, “baixo nível”.
- Idealmente poderíamos extrair informações dos dados de “baixo nível” para complementar os metadados.

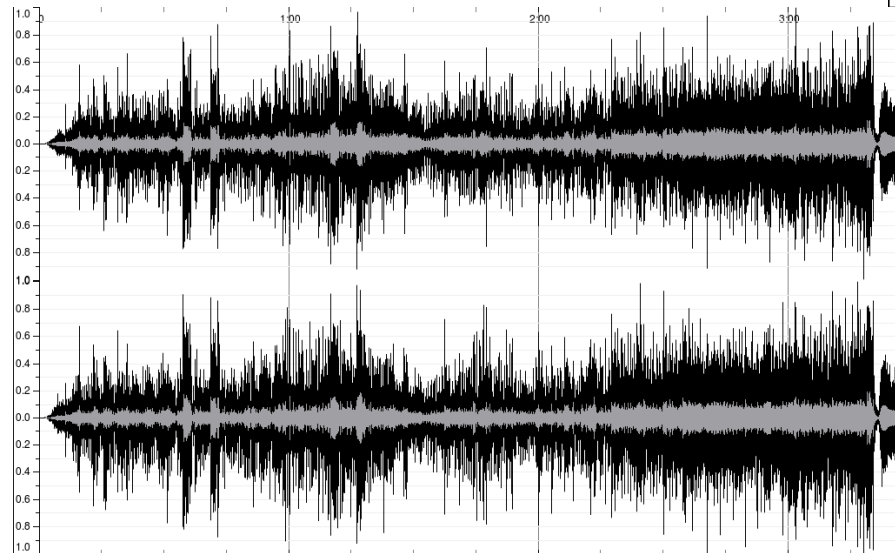
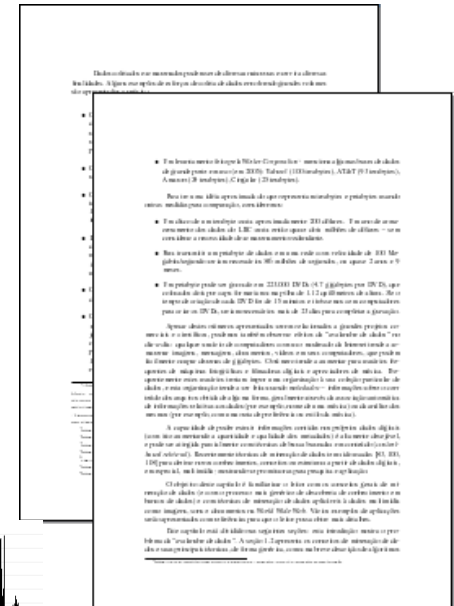
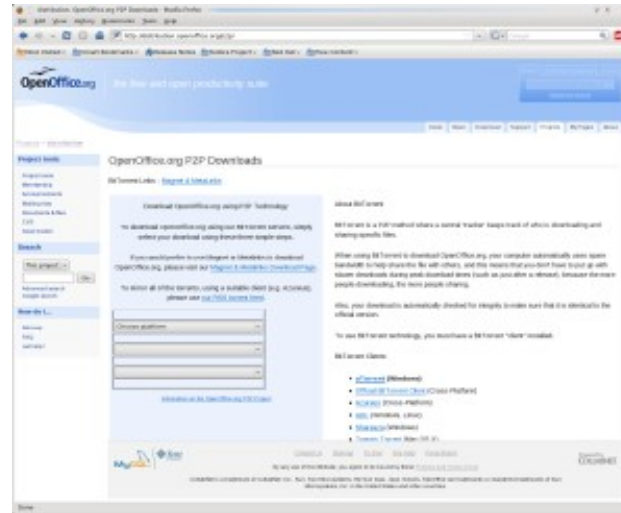
Exemplo	Conteúdo	Metadado
Imagens de câmeras digitais	Cor predominante em regiões, histogramas, formas, cores e texturas de áreas perceptualmente homogêneas, disposição geométrica das áreas e relações espaciais entre elas, etc.	Dimensões da imagem, abertura, exposição, foco, data e hora da imagem, modo de operação, modelo da câmera, etc.
Arquivos de áudio digital (ex. música)	Ritmo ou tempo, timbre, pausas, informações derivadas de análise dos sinais de áudio, texto extraído da música através de reconhecimento de fala, etc.	Nome da música, autor, cantor, categoria ou estilo, duração, ano de produção, letra da música, etc.
Vídeo digital	Atributos das imagens estáticas que compõem o vídeo mais os obtidos da análise da diferença de imagens em sequência, atributos de áudio, correlação do conteúdo de áudio com o de vídeo, estimativa de movimento de objetos e da câmera, etc.	Contexto, duração, praticamente os mesmos atributos de imagens e de áudio, etc.

Exemplo	Conteúdo	Metadado
Texto em geral (não estruturado ou parcialmente estruturado)	Presença e ausência de palavras e associações, histogramas de palavras, comprimento de sentenças, métricas de complexidade do texto, etc.	Autor, língua, contexto, objetivo, tamanho do texto, palavras-chave, etc.
Documentos na WWW	Atributos de texto (similares aos usados para texto não estruturado), estrutura do documento, <i>links</i> , etc.	Dados sobre servidor, endereço, informações de <i>logs</i> de acesso, metadados do texto, etc.

- Organização dos metadados depende do domínio da aplicação.
 - Exemplo: metadados de imagens digitais (fotos, documentos, raios-X, etc.)
- Alguns metadados podem existir logicamente dissociados dos dados.
 - Exemplo: dados sobre músicas (que podem ser transformados em *tags*).

- Mineração de metadados:
 - Organização dos metadados é simples.
 - Simples, praticamente imediata.
 - Não necessita acesso aos dados “crus”.
- Mineração de conteúdo:
 - O que realmente é interessante.
 - Uso direto de conteúdo difícil e custoso.
- Mineração de atributos do conteúdo:
 - Forma ideal!
 - O problema é retirar atributos do conteúdo...

- Como extrair dados de conteúdo multimídia?



- Em geral, o volume do conteúdo é muito maior do que dos dados extraíveis.
 - Extração de dados de conteúdo volumoso pode ter custo computacional elevado.
- Que atributos podem/devem ser extraídos do conteúdo?
 - Nem sempre simples de definir ou extrair: obter uma tabela normal pode ser inviável.
 - Relevância dos atributos pode ser bem diferente para tarefas semelhantes.
- Enorme diferença entre os dados crus e informações sobre o conteúdo: **Gap semântico**.
 - Humanos podem usar conhecimento adicional aos dados!

O Gap Semântico



O Gap Semântico



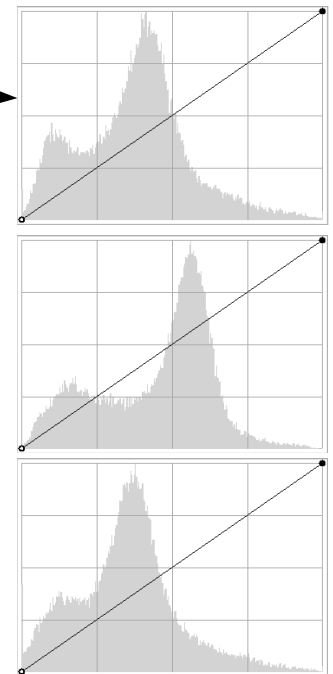
Largura, altura, abertura, tempo de exposição, local, ISO, etc.

X,Y	R,G,B
0,0	9,8,4
1,0	11,10,6
2,0	15,15,15
3,0	15,15,15
4,0	10,11,15
5,0	14,15,19
...	...
278,209	44,65,34
279,209	34,55,24

Esquilo cinza em cima de uma cerca verde, em um gramado no Parque de Kensington, Londres, com moitas com flores vermelhas ao fundo.

Mamífero atrás de uma cerca.

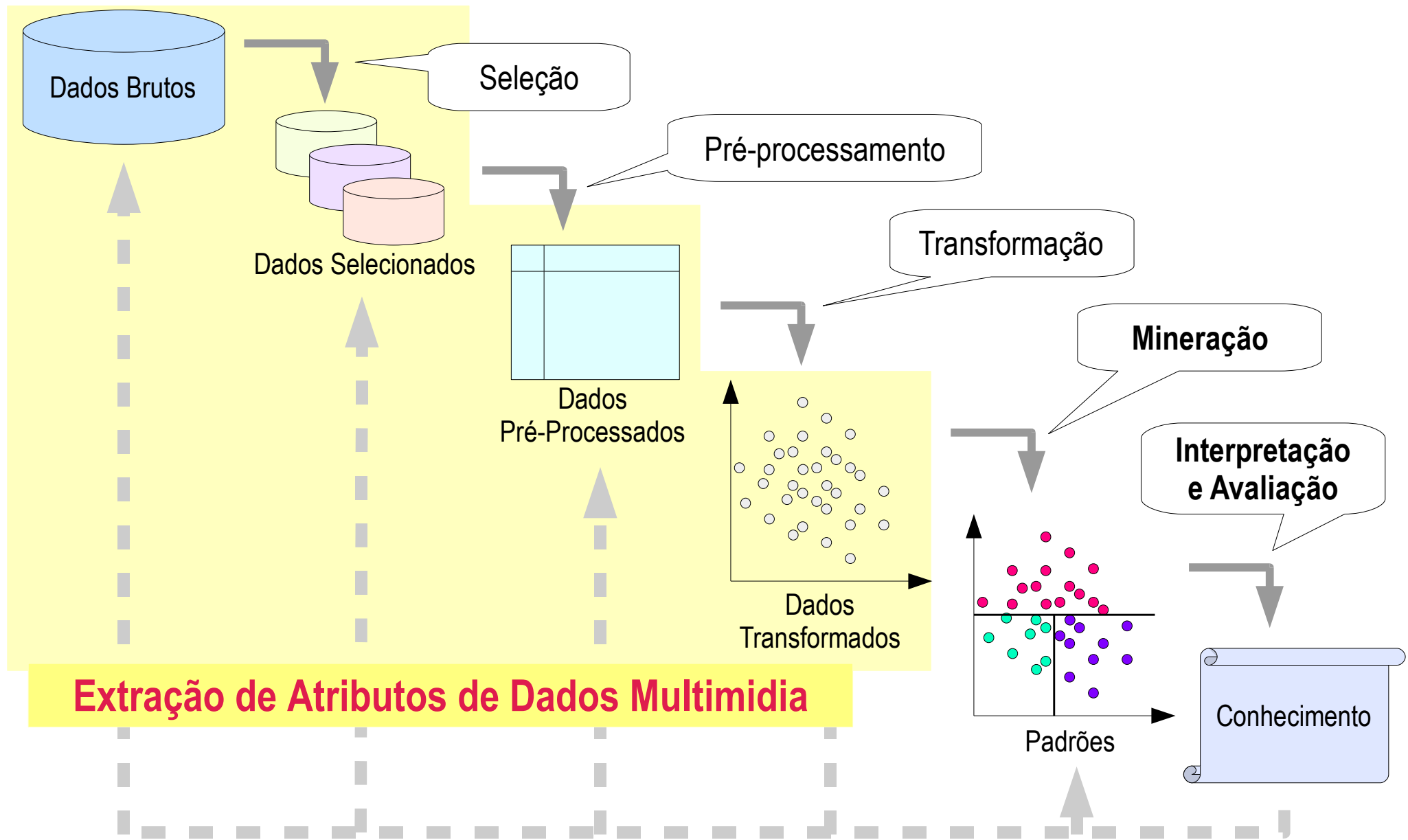
"Bob".



- Procure a palavra X em um conjunto de textos.
- Procure a expressão regular X em um conjunto de textos.
- Procure uma combinação de termos e exclusões...
-
- Procure uma frase estruturada de determinada forma...
- ...
- Procure nomes próprios.....
-
- Procure nomes de políticos.
- ...
- Procure trechos engraçados em um conjunto de textos.



- Nosso grande problema: reduzir o *semantic gap*...
 - De forma o mais automática possível,
 - Com o menor esforço computacional possível,
 - Com a menor dependência de dados ancilares possível.
- É possível fazer isto?
 - É possível fazer isto *sem* restringir aplicações e domínios?
- Se existe solução, passa pela extração de atributos.
- Lembrete: não existe *blind data mining*!



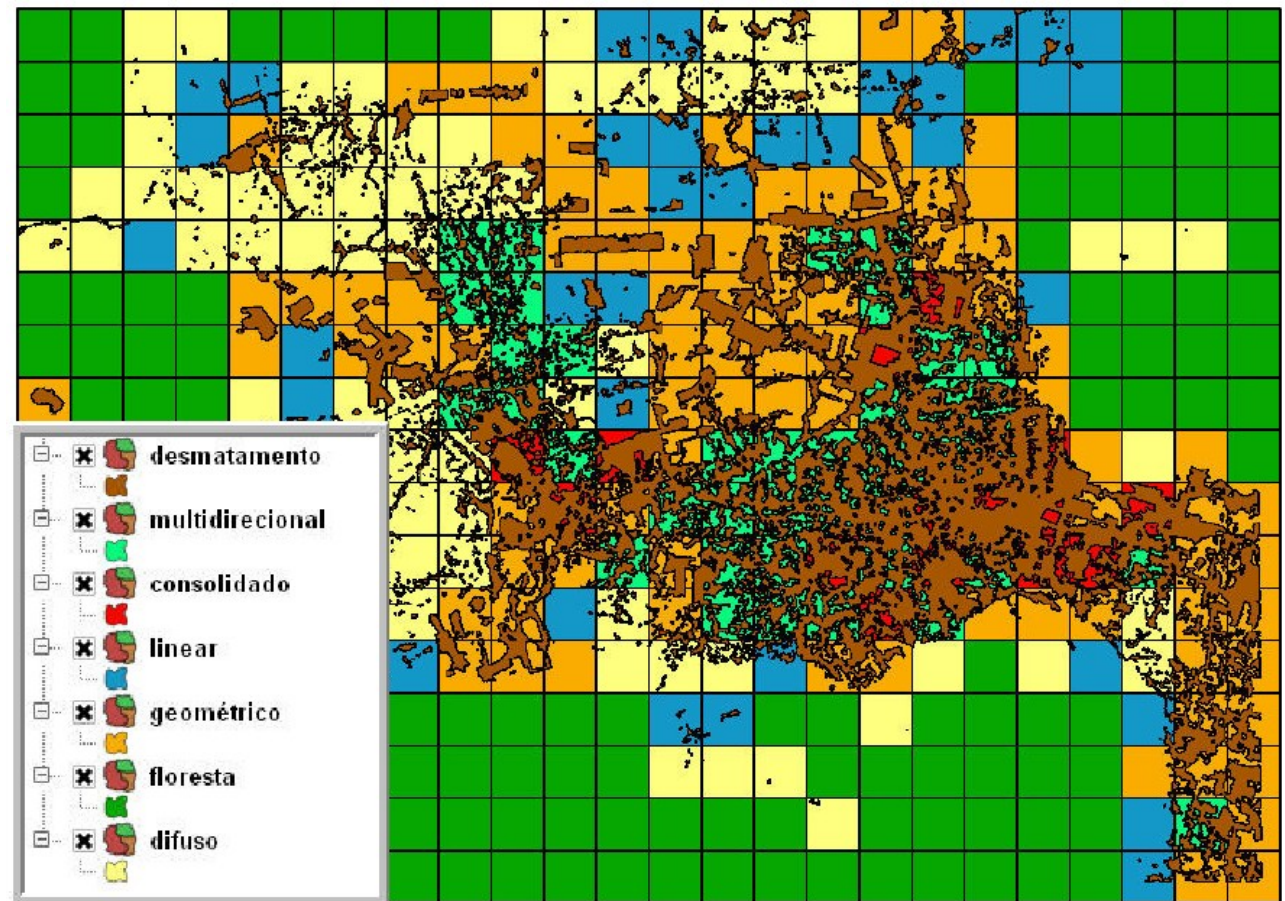
- Próximas seções: Apresentação de técnicas e casos de aplicação.
 - Amostragem bem reduzida!
 - Muitas aplicações e algoritmos descritos foram obtidos do *Lecture Notes in Computer Science* (LNCS, LNAI ou LNBI).
- Veremos poucos casos de mineração multimídia realmente multimídia.
- Alguns exemplos pertencem a mais de uma categoria.
 - Especialmente exemplos que envolvem dados temporais ou imagens.

Mineração de Dados Multimídia

Dados Espaciais

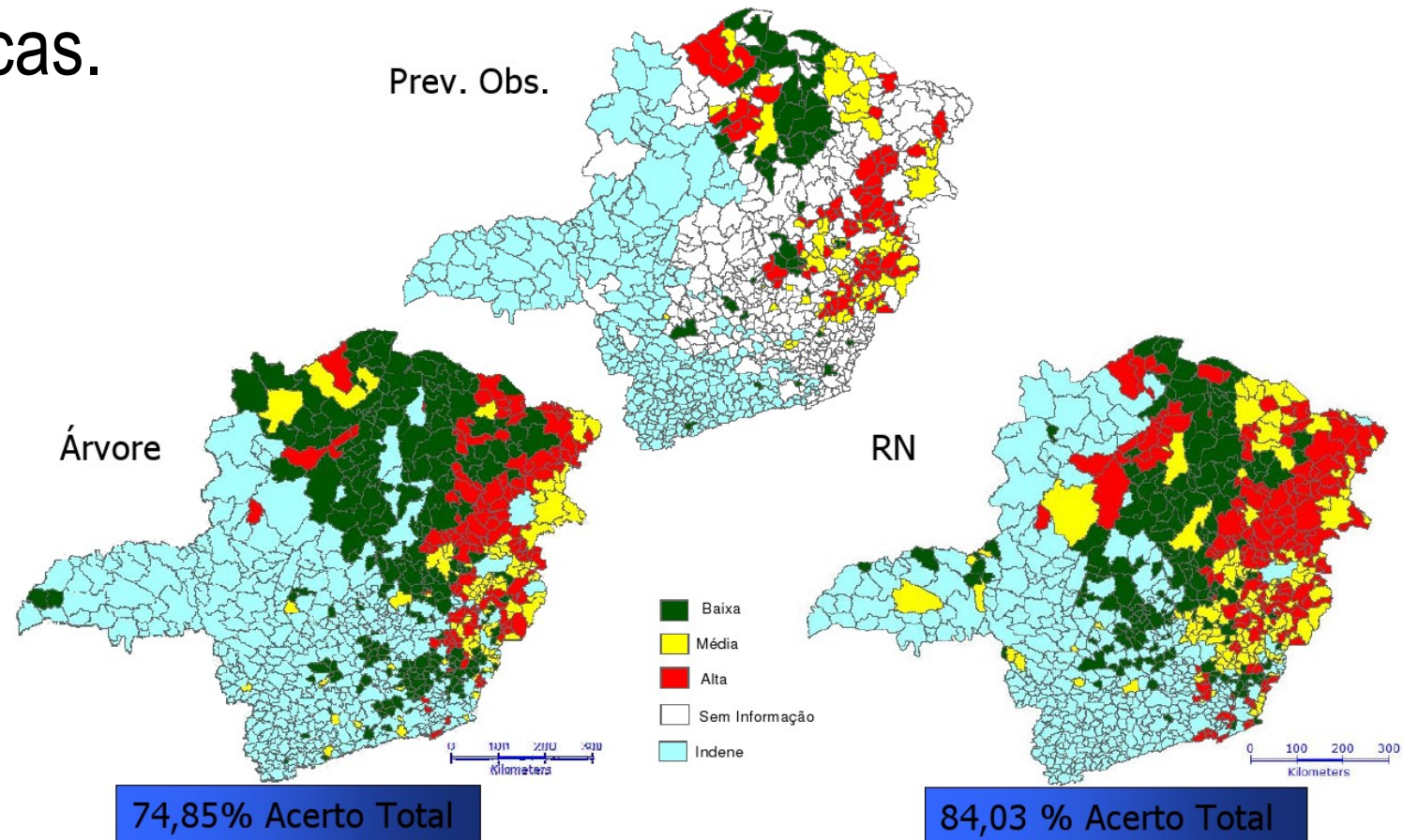
- Motivada pelo uso crescente de bancos de dados espaciais e sistemas de informações geográficas.
- Técnicas de mineração de dados espaciais associam objetos espaciais a técnicas de DM tradicionais.
 - Objetos espaciais podem ser pontos, linhas, polígonos, etc.
 - Relações espaciais podem incluir *perto*, *dentro*, *fora*, etc.
- Foco pode ser na descoberta global ou na descoberta de fenômenos espacialmente locais.

- Classificação do desmatamento usando padrões de ocupação humana.



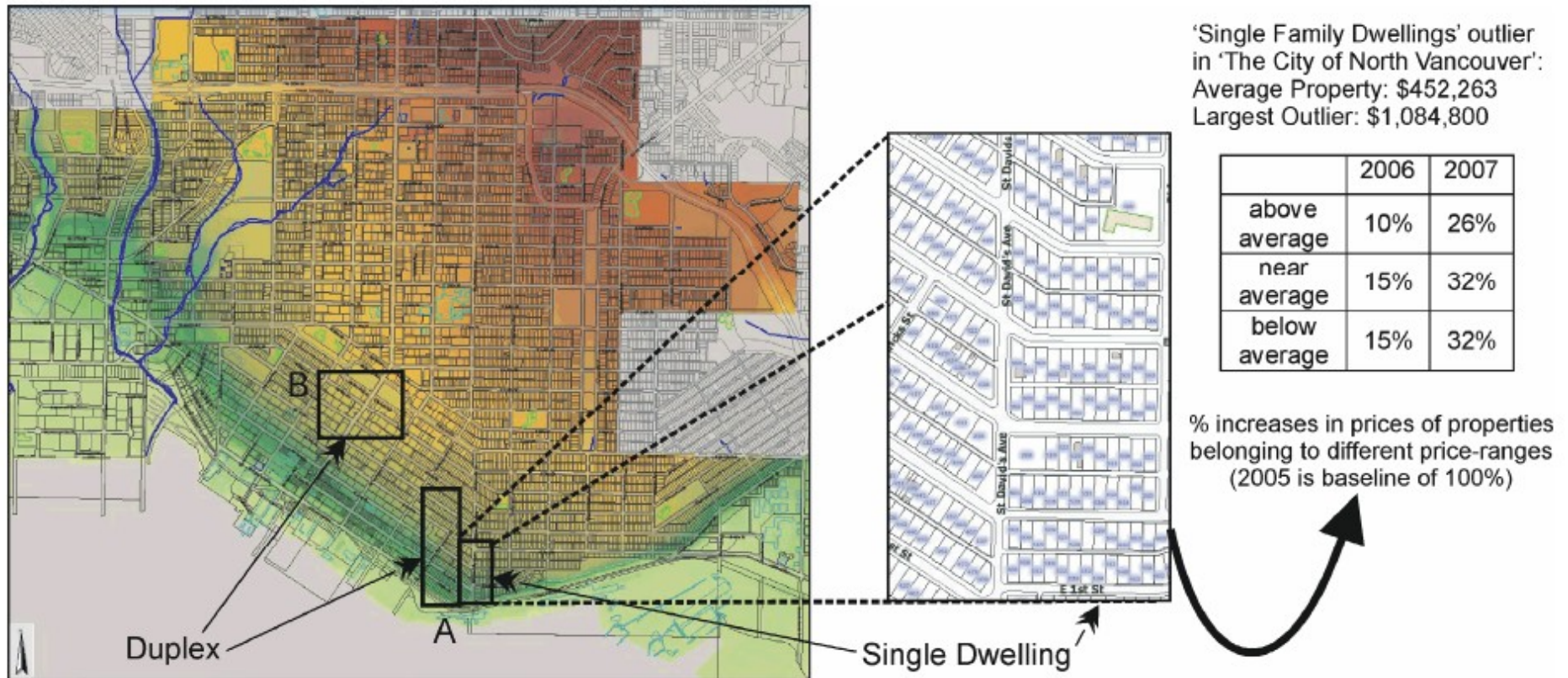
Mineração de Dados Espaciais Utilizando Métricas de Paisagem, Relatório Final de Márcio Azeredo, disciplina CAP-359, INPE.

- Classificação da prevalência de ocorrência de esquistossomose em MG usando variáveis ambientais e socioeconômicas.



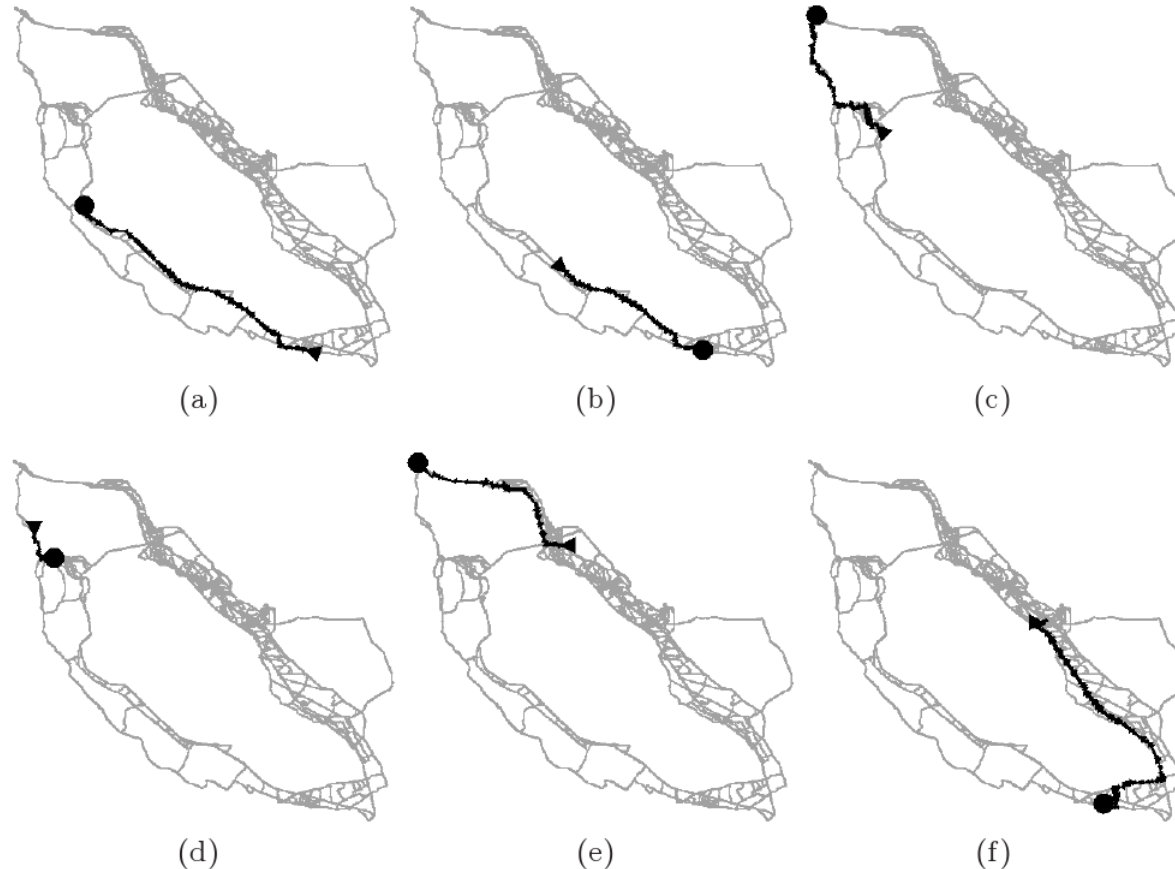
Classificação do risco da esquistossomose no estado de Minas Gerais, Relatório Final de Flávia de Toledo Martins, disciplina CAP-359, INPE.

- Identificação de regiões que se comportam como *outliers* em bases de dados sociogeográficas.



Richard Frank, Wen Jin, and Martin Ester, *Efficiently Mining Regional Outliers in Spatial Data*, LNCS 4605, 2007.

- Identificação de padrões de fluxo de tráfego (“*hot routes*”) por agrupamento do volume de tráfego em segmentos de estradas.



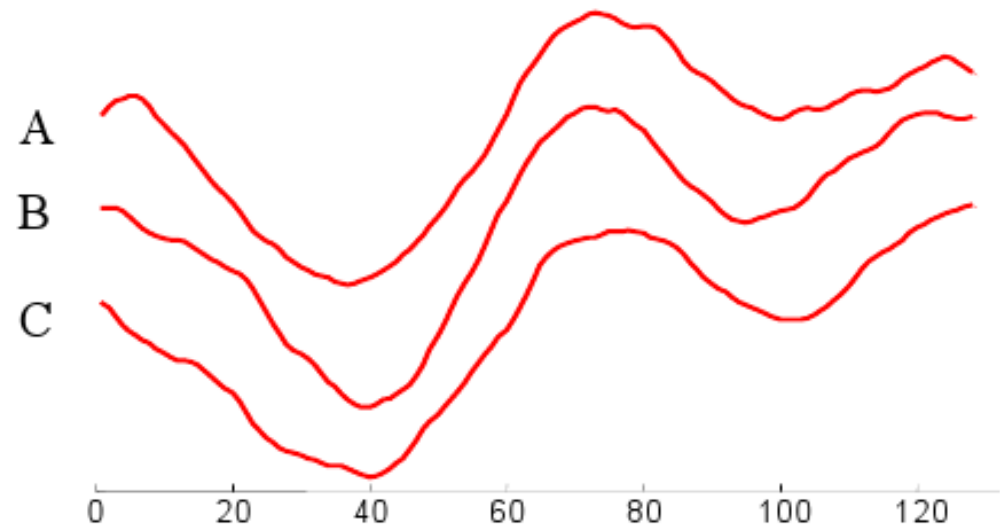
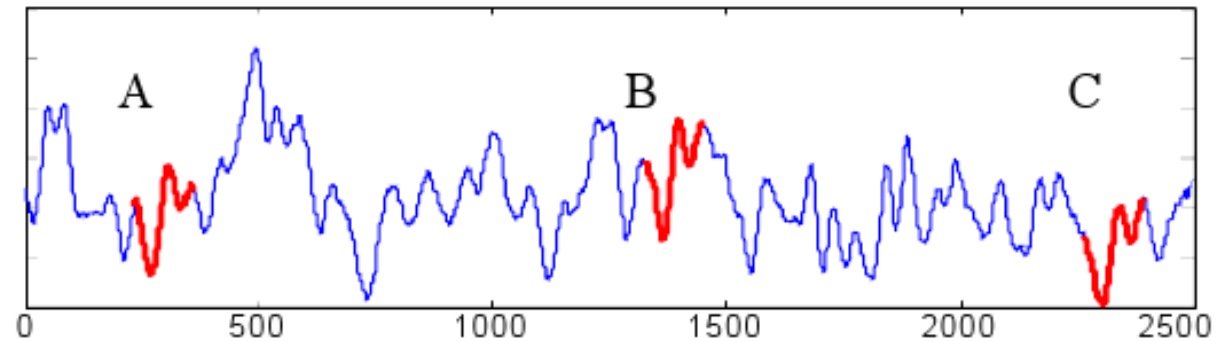
Xiaolei Li, Jiawei Han, Jae-Gil Lee and Hector Gonzalez, *Traffic Density-Based Discovery of Hot Routes in Road Networks*, LNCS 4605, 2007.

Mineração de Dados Multimídia

Dados Temporais

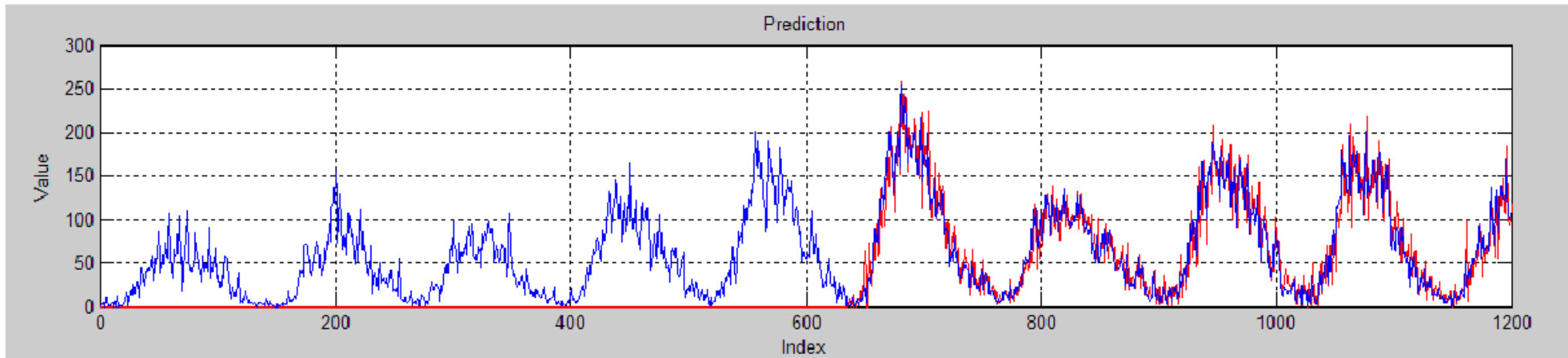
- A maior parte dos dados coletados automaticamente tem um componente temporal.
- Uso em identificação de eventos, previsão de valores futuros, localização de padrões, identificação de associações temporais, etc.
- Dados temporais podem ser contínuos ou esparsos, ter um índice temporal ou ser multivariado no tempo, ter características espaçotemporais, etc.
- Maior problema: modelagem do índice temporal (ex. Sazonalidade, associação com atividade humana).

- Técnicas para identificação de padrões recorrentes (*motifs*) em séries temporais.



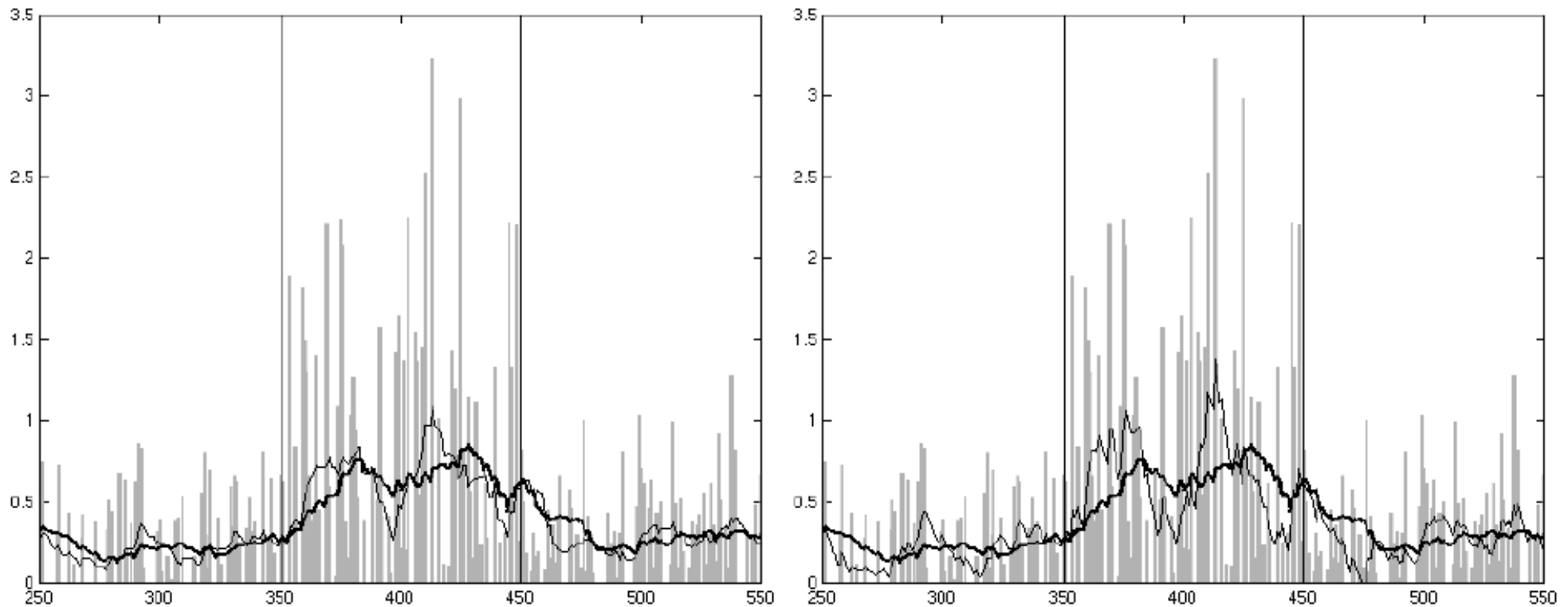
Jessica Lin, Eamonn Keogh, Stefano Lonardi and Pranav Patel, *Finding Motifs in Time Series*, Proceedings of the Second Workshop on Temporal Data Mining, 2002..

- Uso de redes neurais para previsão em séries temporais pseudo-periódicas.
 - Atributos são médias móveis das diferenças de N-ésima ordem entre valores.



Yang Lan and Daniel Neagu, *Applications of the Moving Average of n^{th} -Order Difference Algorithm for Time Series Prediction*, LNCS 4632, 2007.

- Detecção de prováveis epidemias a partir da análise de *logs* de acesso a uma base de dados sobre doenças.



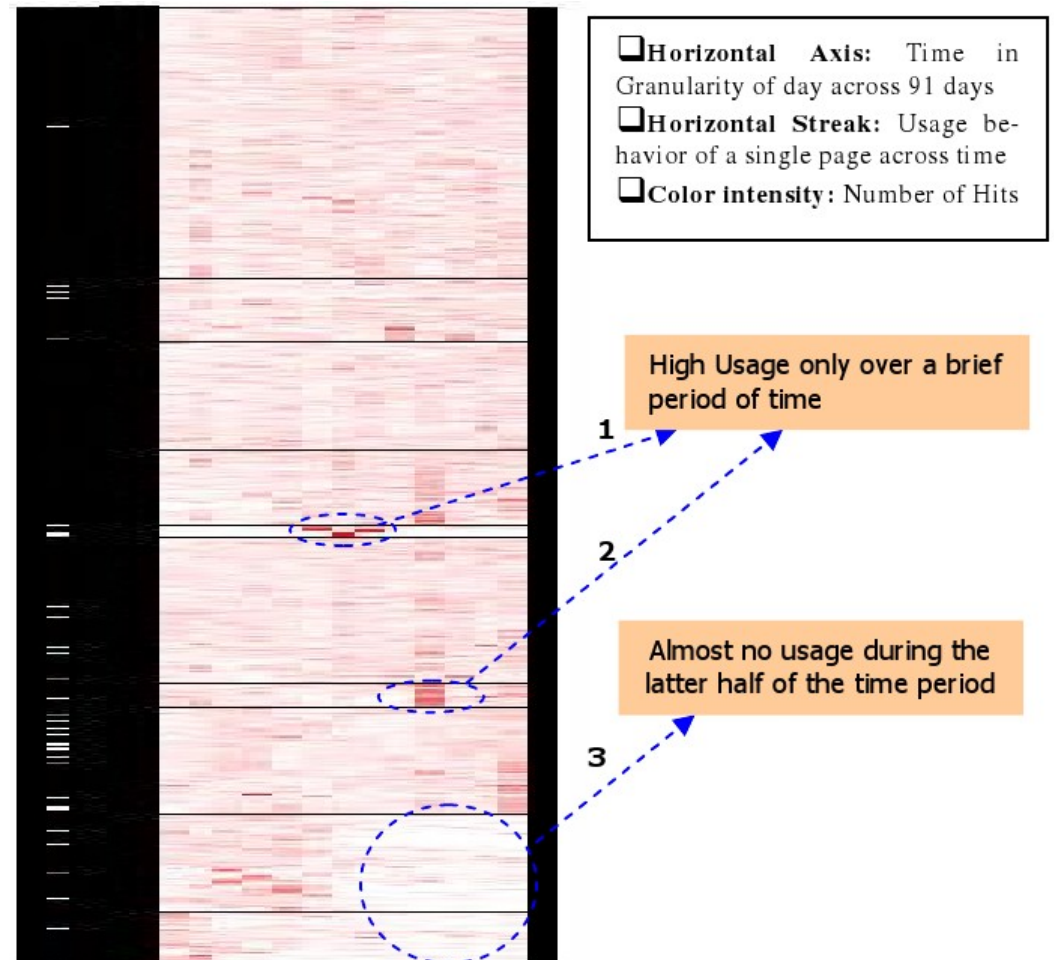
Jaana Heino and Hannu Toivonen, *Automated Detection of Epidemics from the Usage Logs of a Physicians' Reference Database*, LNCS 2838, 2003.

- Árvores de decisão para busca de co-ocorrências em dados do mercado de ações da Malásia.

No.	Classification Rules
1	<i>if AMMB=UP and RHBCAP=UP and TIME=UP then COMPOSITE=UP (191, 8)</i>
2	<i>if AMMB=DOWN and MBB=DOWN and MAS=DOWN then COMPOSITE=DOWN (118, 5)</i>
3	<i>if AMMB=DOWN and MBB=SAME and COMMERZ=DOWN then COMPOSITE=DOWN (80, 12)</i>
4	<i>if AMMB=DOWN and MBB=UP and MMBCORP=DOWN and TA=DOWN then COMPOSITE=DOWN (38,4)</i>
5	<i>if AMMB=UP and RHBCAP=UP and TIME=SAME then COMPOSITE=UP (36, 3)</i>
6	<i>if AMMB=SAME and MAS=UP and YTL=UP then COMPOSITE=UP (34, 1)</i>
7	<i>if AMMB=UP and RHBCAP=DOWN and TENAGA=UP and GENTING=UP then COMPOSITE=UP (28)</i>
8	<i>if AMMB=UP and RHBCAP=UP and TIME=DOWN and BERNAS=UP then COMPOSITE=UP (22)</i>
9	<i>if AMMB=SAME and MAS=DOWN and GUTHRIE=DOWN then COMPOSITE=DOWN (18)</i>
10	<i>if AMMB=UP and RHBCAP=DOWN and TENAGA=DOWN and DRBHCOM=DOWN and COMMERZ=DOWN then COMPOSITE=DOWN (18)</i>

Lay-Ki Soon and Sang Ho Lee, *Explorative Data Mining on Stock Data – Experimental Results and Findings*, LNCS 4632, 2007.

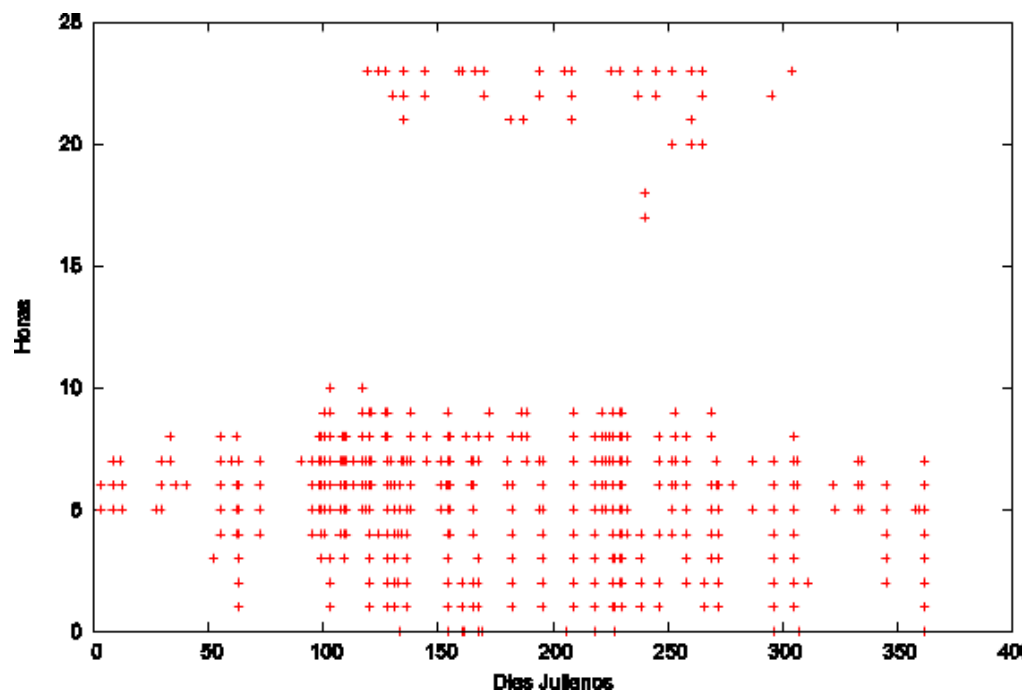
- Análise da mudança de acesso a páginas na Web.
- Agrupamento em *clusters* de padrões de acesso.



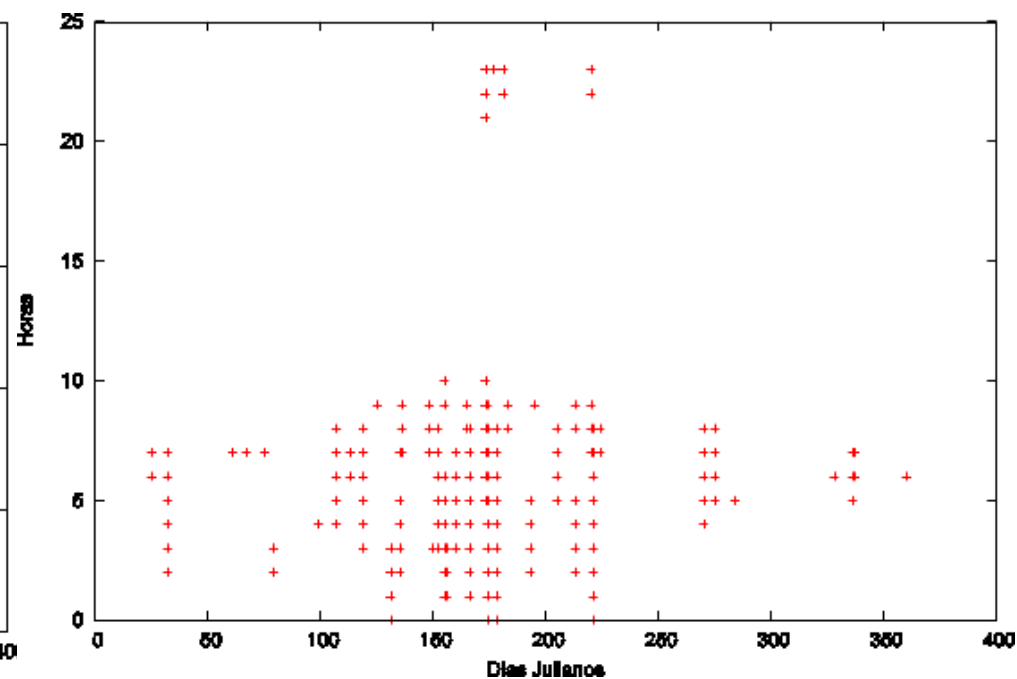
Prasanna Desikan and Jaideep Srivastava, *Mining Temporally Changing Web Usage Graphs*, LNCS 3932, 2006.

- Caracterização do que causa ocorrência de nevoeiros ao longo do tempo.

1951

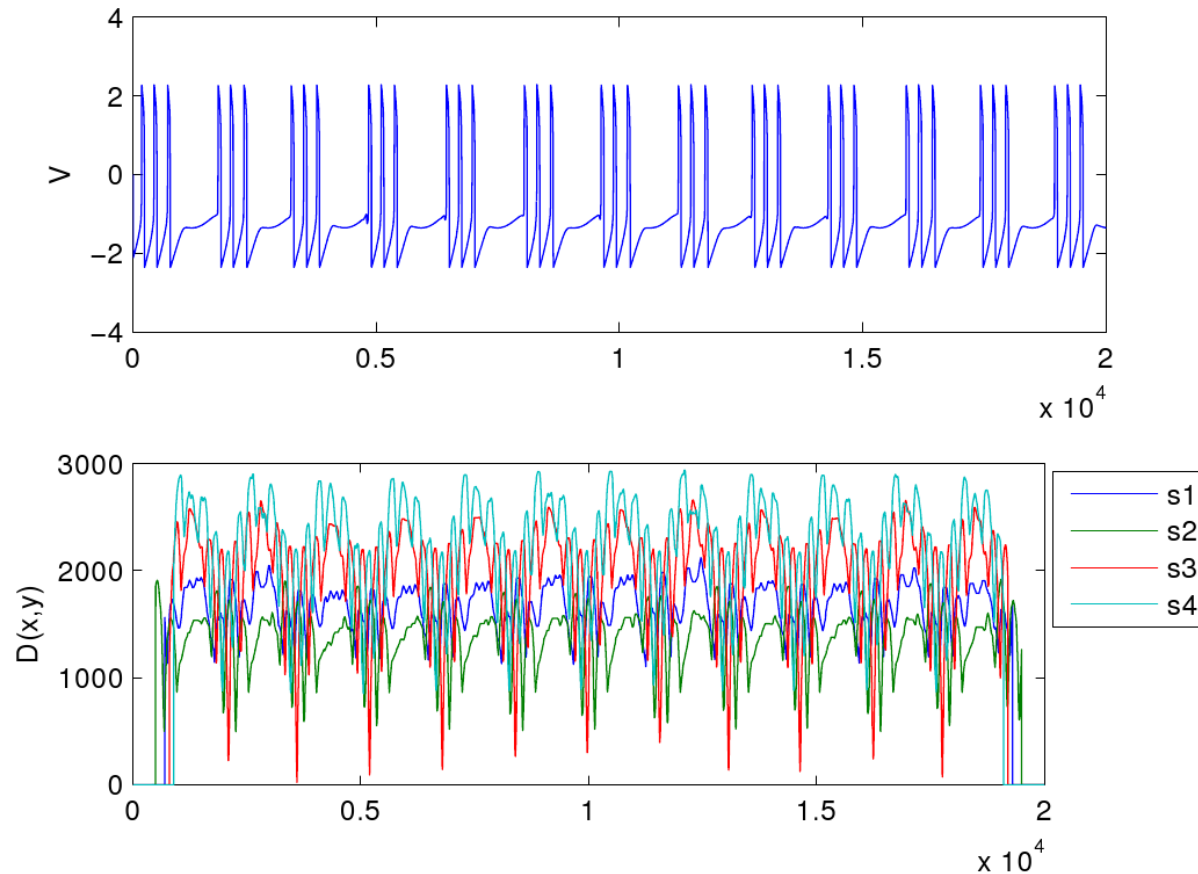


1995



Mineração de Dados para Previsão de Nevoeiros, Relatório Final de André Muniz Marinho da Rocha, disciplina CAP-359, INPE.

- Mineração de sinais caóticos (séries temporais) para descoberta de propriedades gerais.



Mineração de Dados para encontrar Motifs em Séries Temporais, Relatório Final de Rosangela Follmann Bageston, disciplina CAP-359, INPE.

Mineração de Dados Multimídia

Imagens

- Muito esforço em *CBIR* (*Content-Based Image Retrieval*) e *RBIR* (*Region-Based Image Retrieval*).
- Complexidades:
 - Facilidade de aquisição.
 - Alto conteúdo semântico.
 - Conteúdo subjetivo.
- Ainda maior problema: *semantic gap*.
 - De pixels a descritores: como?

- Duas abordagens (extremos) para geração de dados com maior conteúdo semântico:
- Anotação Manual
 - Trabalho excessivamente manual.
 - Alto nível de subjetividade.
- Indexação baseada em conteúdo
 - **Semantic gap**: causa baixa performance.
- Resultados insatisfatórios.

- Abordagem entre os extremos: anotação automática.
- Uso de
 - *Bag of features*: indicação de que objetos podem ser associados à imagem (ex. praia, barcos, coqueiros).
 - *Semantic link*: que conceitos podem estar ligados a outros (ex. praia = areia + mar) → *semantic trees* e *semantic graphs*.
- Ainda existem desafios!
 - Nível de detalhamento dos atributos.
 - Vocabulário e mapeamento entre linguagens.
 - Descrições longas.
 - Relevância de conteúdo.
 - Qualidade de conteúdo.

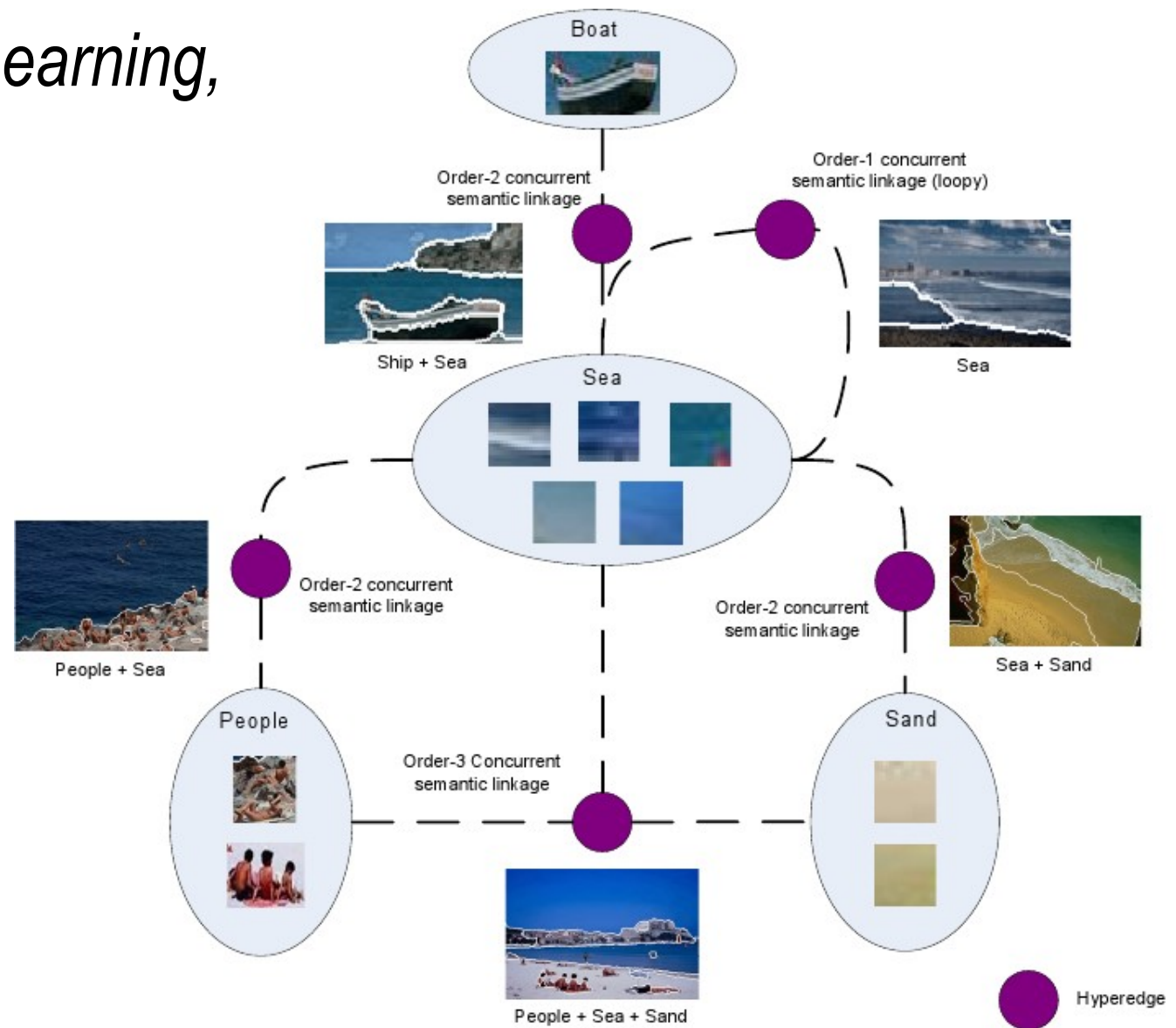
- Melhor apreciação se conhecermos alguns conceitos:
 - Pixels e multiescala.
 - Espaço de cores (ex. RGB x CMYK x HSV x La*b*).
 - Bordas.
 - Regiões e descritores.
 - Textura.
 - *Bag-of-features*.



Mineração de Imagens – Conceitos

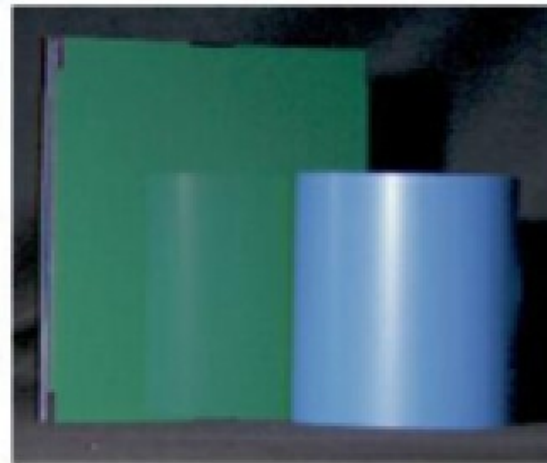


- *Multiple Instance Learning, Bag-of-features, Image Annotation.*

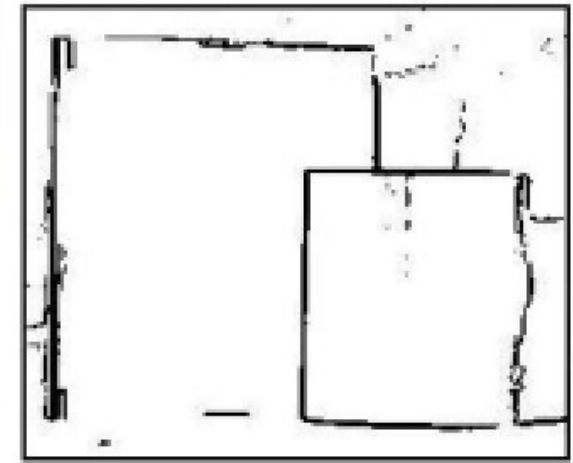


Yong Rui and Guo-Jun Qi, *Learning Concepts by Modeling Relationships*, LNCS 4577, 2007.

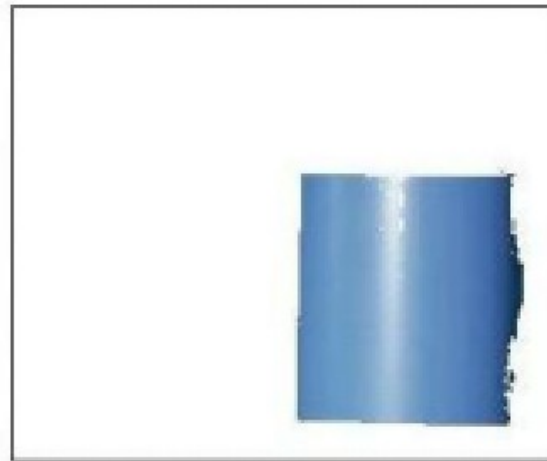
- Segmentação controlada para extração de objetos (ainda sem semântica).



(a)



(b)



(c)



(d)

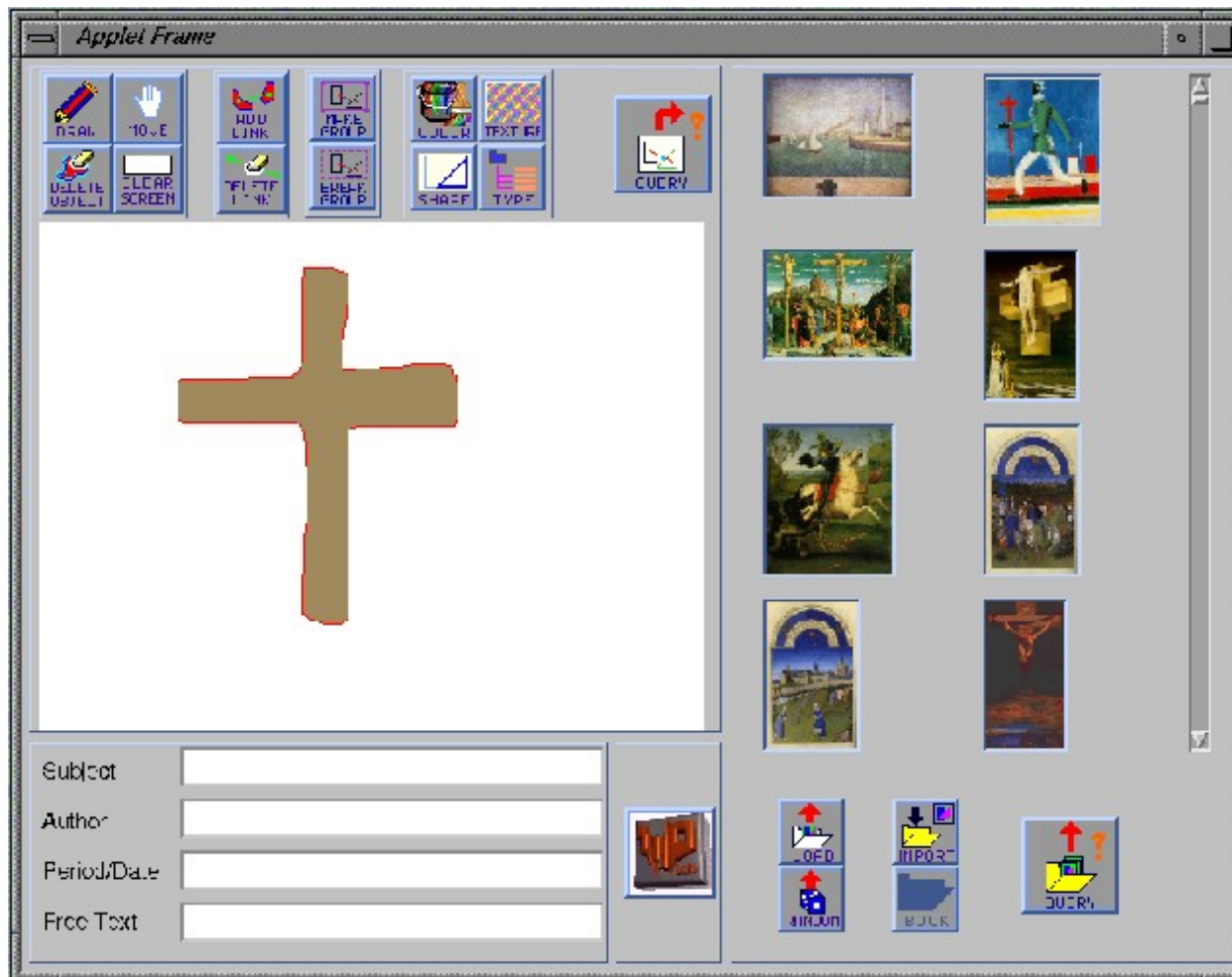
Latifur Khan and Lei Wang, *Object Detection for Hierarchical Image Classification*, LNCS 2797, 2003.

- Segmentação baseada em regiões (cores e texturas) e bordas.



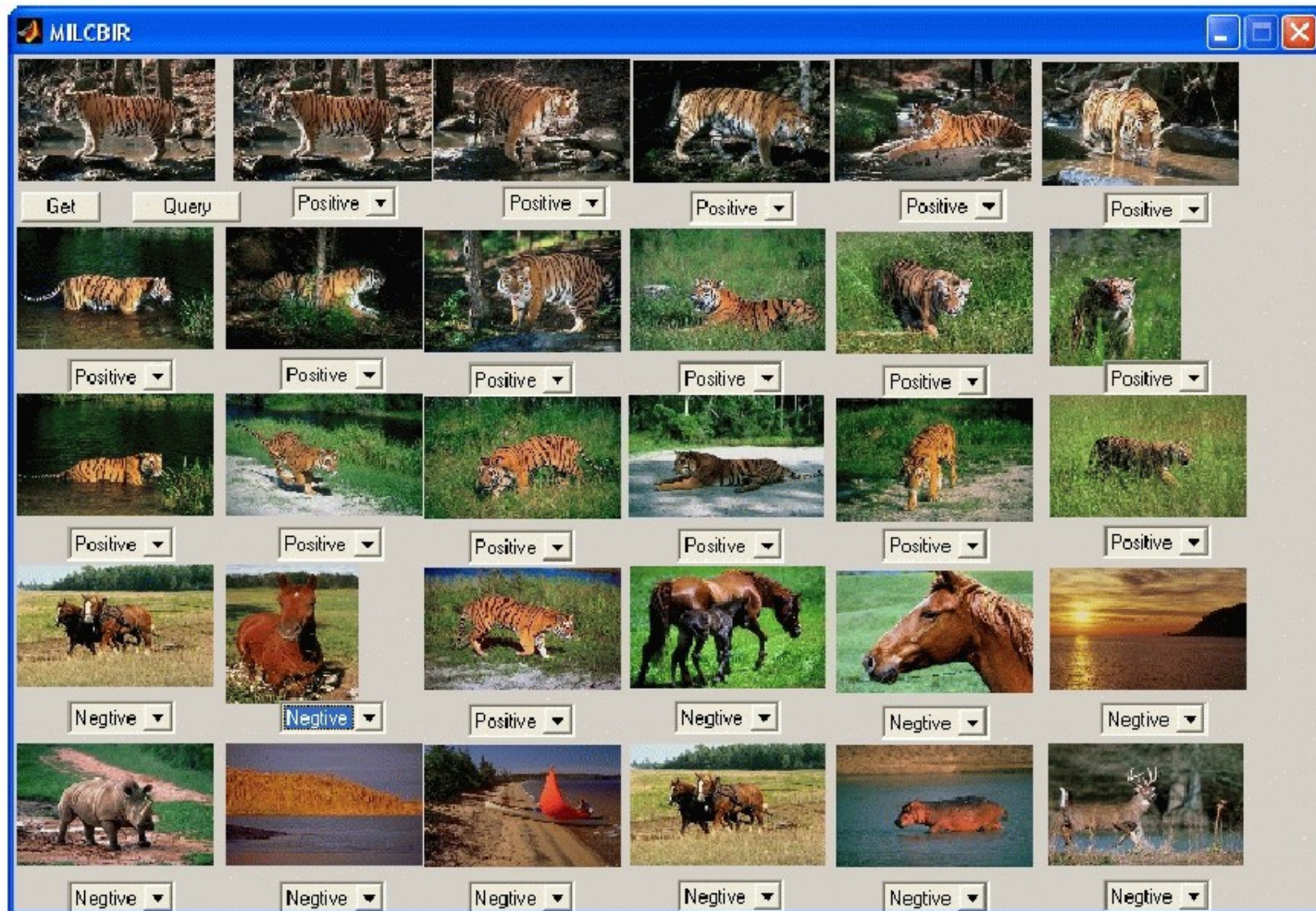
Shengyang Yu, Yan Zhang, Yonggang Wang and Jie Yang, *Color-Texture Image Segmentation by Combining Region and Photometric Invariant Edge Information*, LNCS 4577, 2007.

- Exemplo de *CBIR* (*Visible Image Retrieval*).



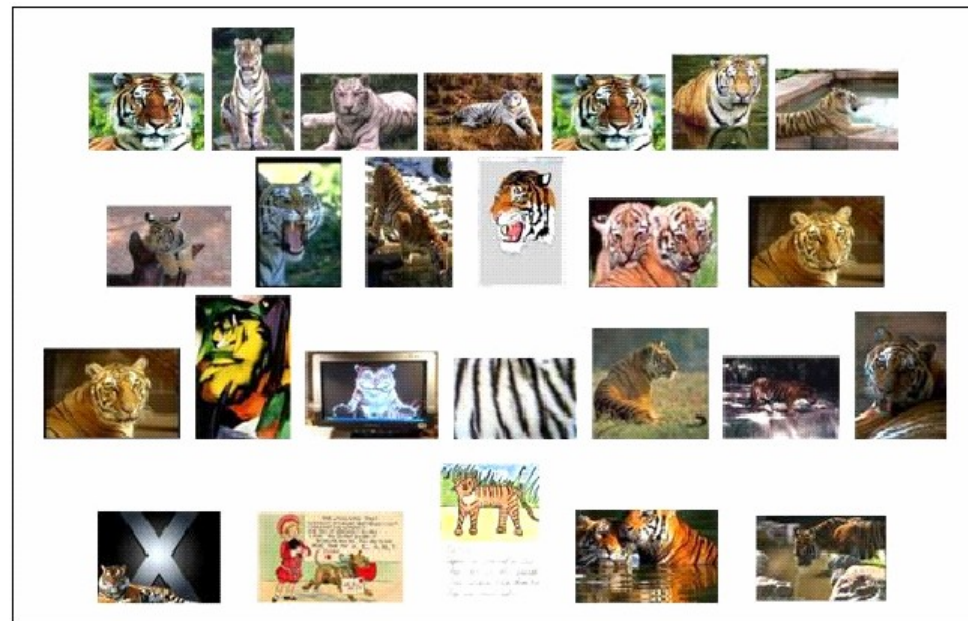
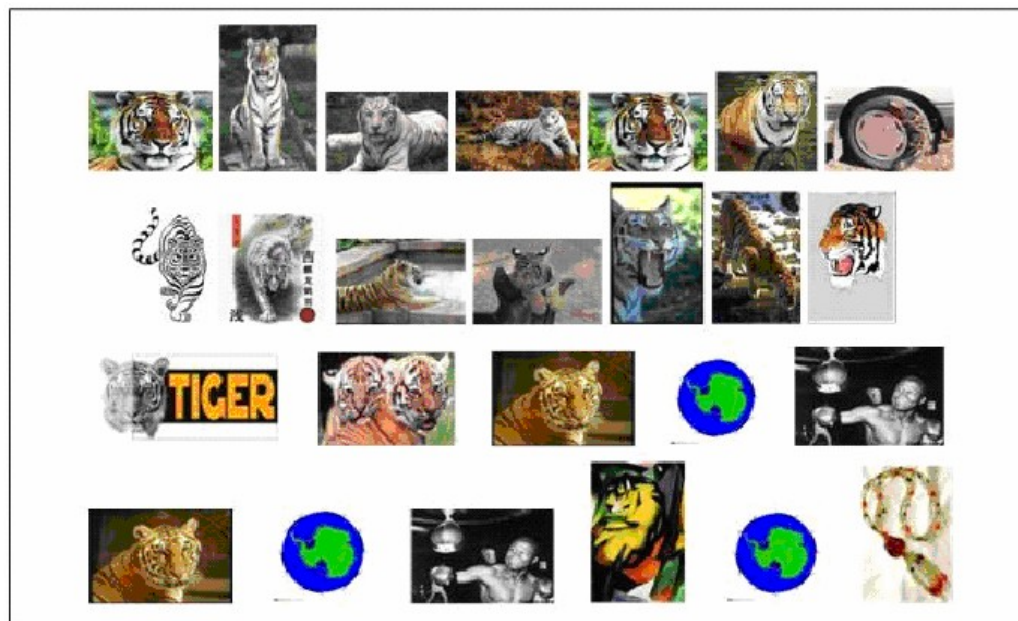
Carlo Colombo and Alberto del Bimbo, *Visible Image Retrieval in Image Databases: Search and Retrieval of Digital Imagery*, 2002.

- *CBIR* com *feedback* de usuários para identificação de objetos relevantes.



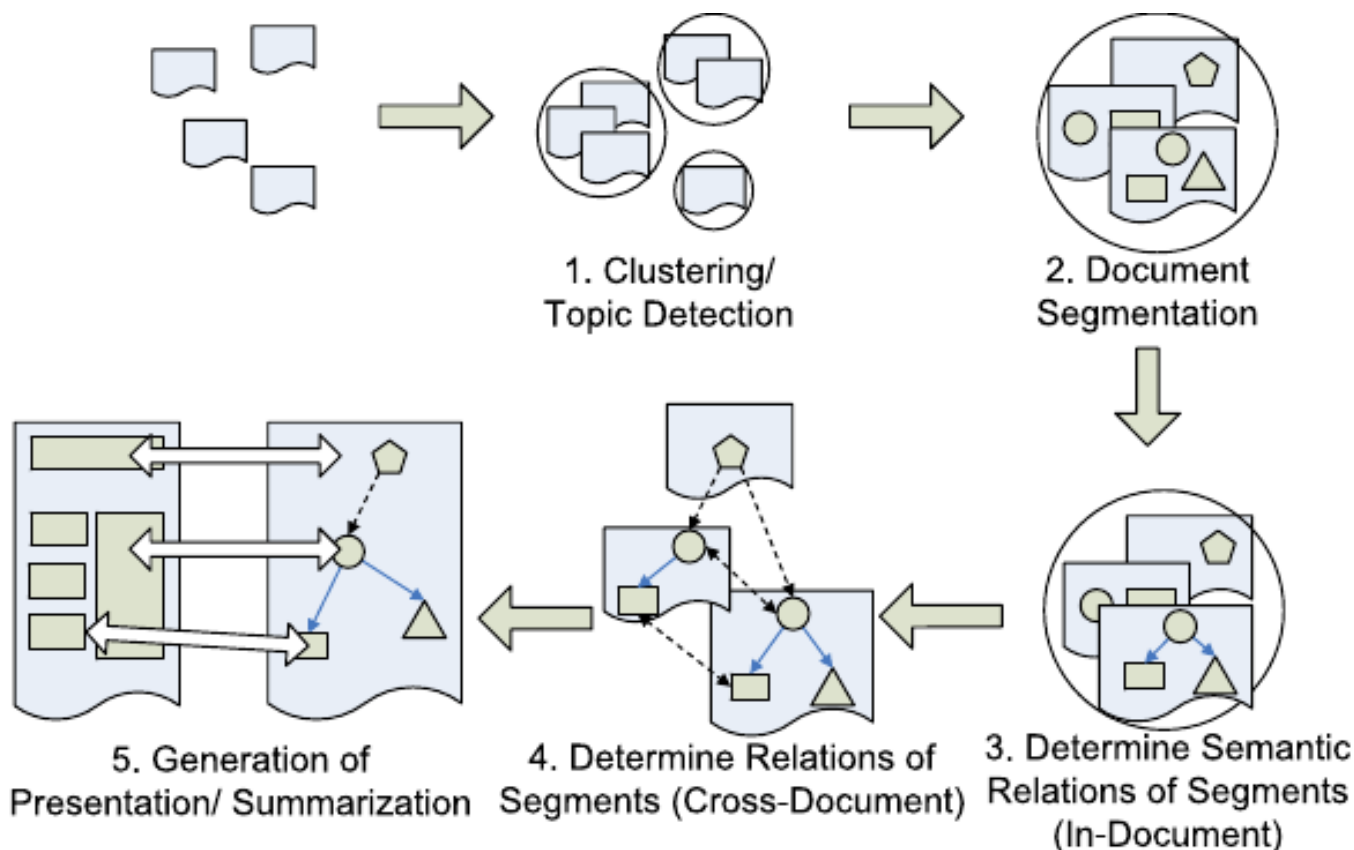
X. Huang , S-C. Chen , M-L. Shyu and C. Zhang, *Mining High-Level User Concepts with Multiple Instance Learning and Relevance Feedback for Content-Based Image Retrieval*, LNCS 2797, 2003.

- Filtragem de resultados de sistemas de busca de imagem por textos adjacentes.
 - Atributos de baixo nível são extraídos e imagens consideradas irrelevantes são removidas do resultado da busca.



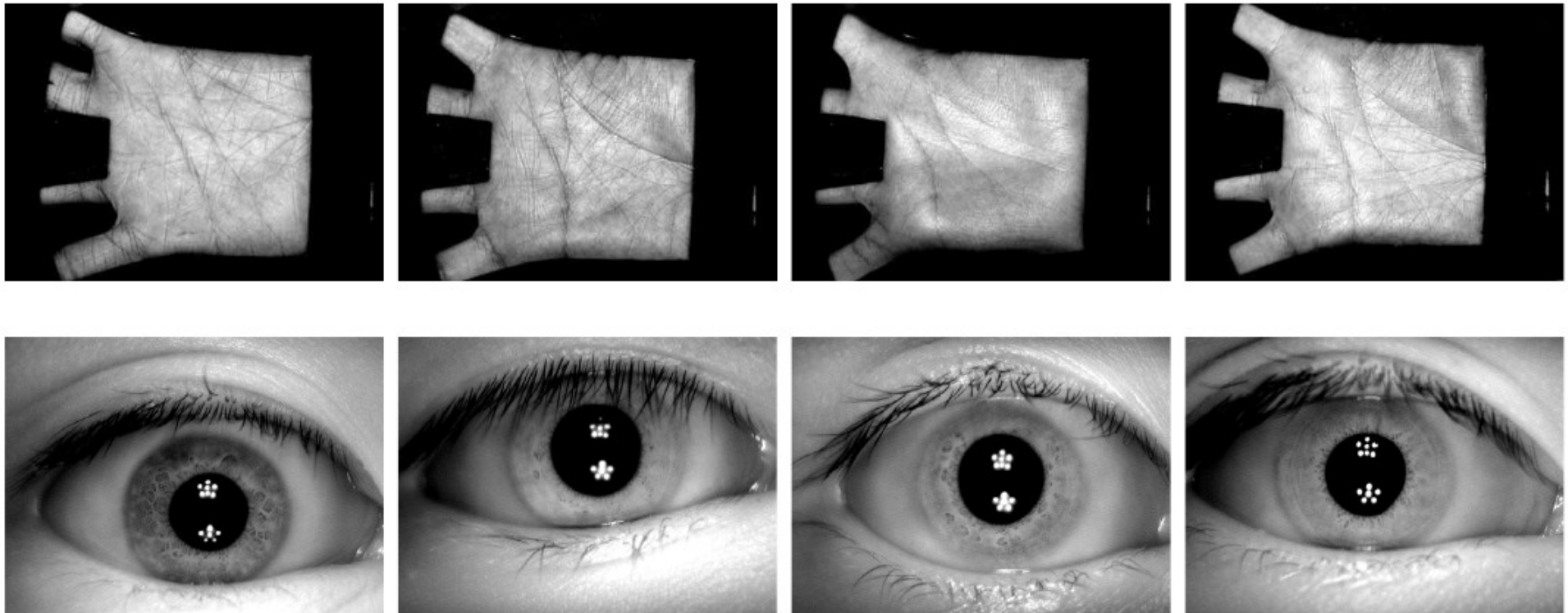
Ying Liu, Dengsheng Zhang and Guojun Lu, *SIEVE – Search Images Effectively Through Visual Elimination*, LNCS 4577, 2007.

- Descoberta de associação entre imagens e trechos de texto.
 - Uso de medida de similaridade entre texto e imagem.



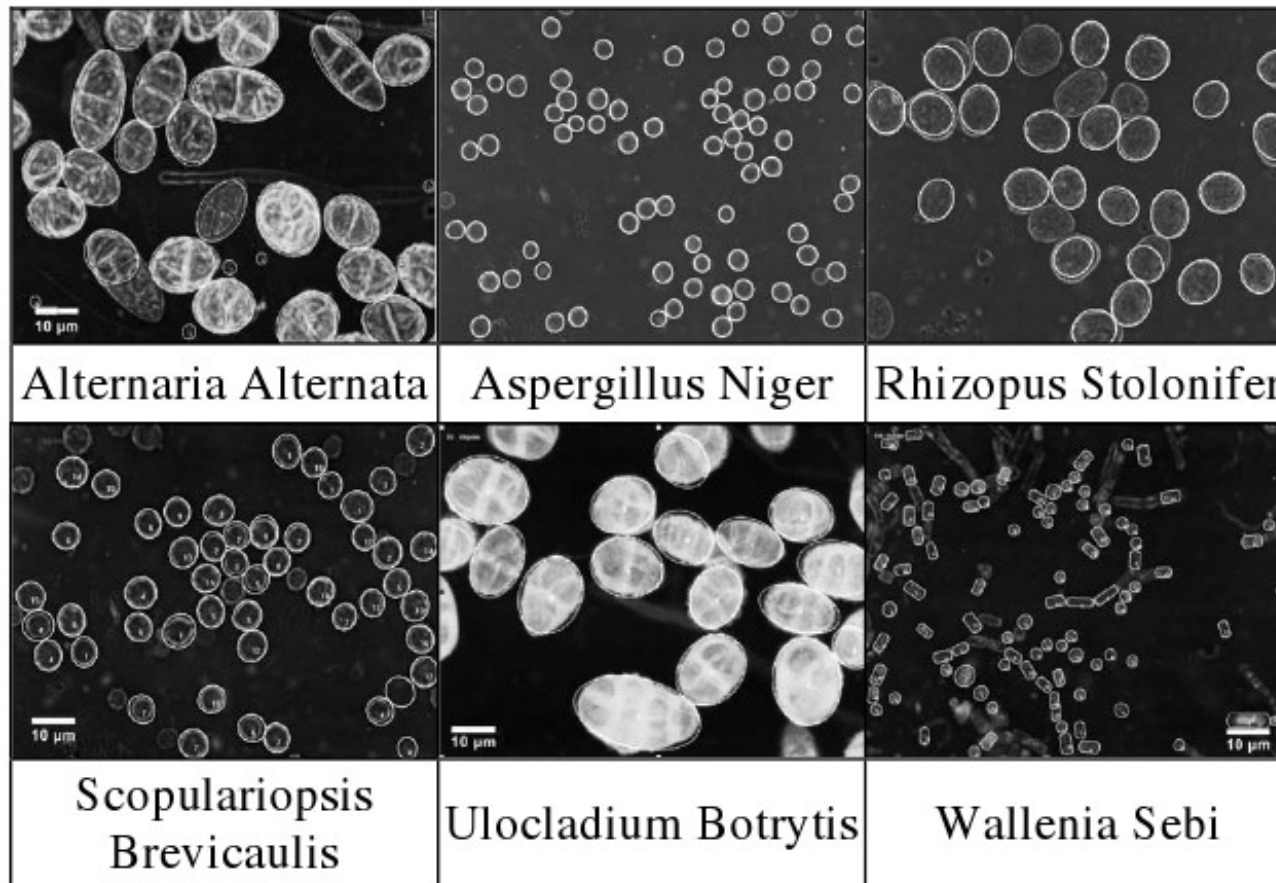
Tao Jiang and Ah-Hwee Tan, *Discovering Image-Text Associations for Cross-Media Web Information Fusion*, LNCS 4213, 2006.

- Fusão de imagens de íris e palmas da mão para autenticação.
 - Resultados bem superiores às técnicas aplicadas isoladamente.



Xiangqian Wu, David Zhang, Kuanquan Wang, and Ning Qi, *Fusion of Palmprint and Iris for Personal Authentication*, LNCS 4632, 2007.

- Reconhecimento de tipos de fungos em imagens microscópicas usando forma.



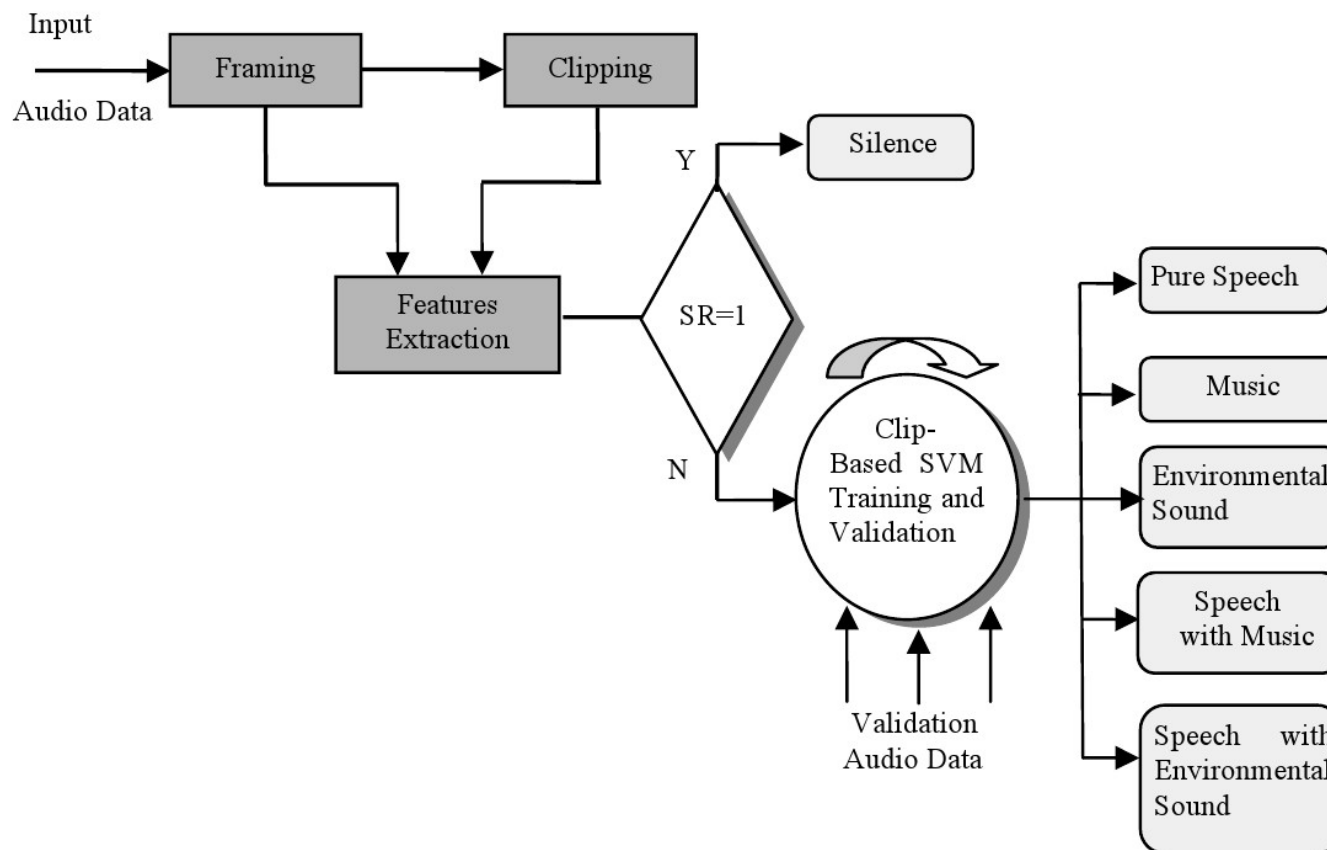
Petra Perner, Horst Perner, Angela Bühring and Silke Jänichen, *Mining Images to Find General Forms of Biological Objects*, LNCS 3275, 2004.

Mineração de Dados Multimídia

Áudio

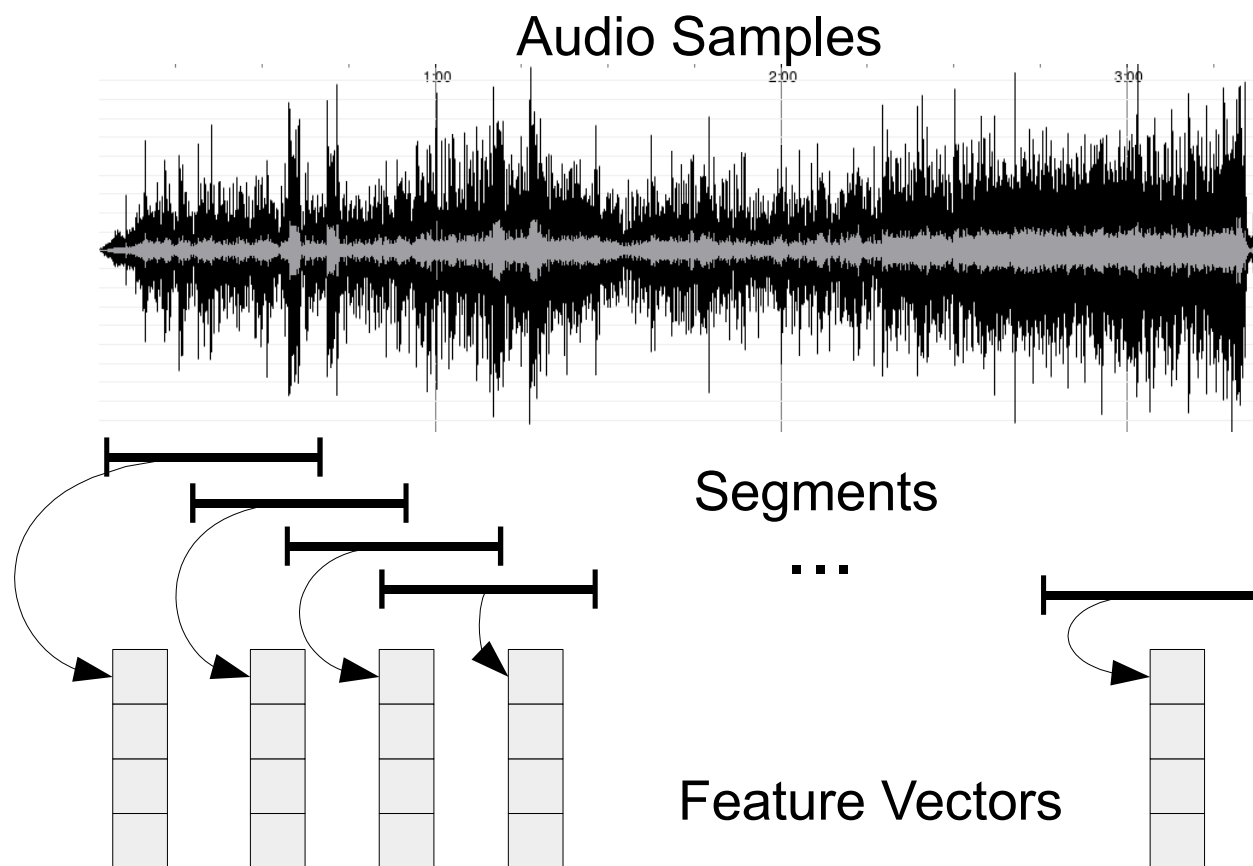
- Como extrair atributos de áudio?
- Música, fala tem finalidades diferentes → devem ter atributos diferentes.
- Quais atributos?
 - Enrique Alexandre, Lucas Cuadra, Lorena Álvarez, Manuel Rosa-Zurera, and Francisco López-Ferrerias: *Automatic Sound Classification for Improving Speech Intelligibility in Hearing Aids Using a Layered Structure*, LNCS 4224, 2006.
 - Carlos N. Silla Jr, Alessandro L. Koerich and Celso A. A. Kaestner, *A Machine Learning Approach to Automatic Music Genre Classification*, JBCS v.3 n. 14, 2008.

- Classificação de trilhas de áudio em seis classes usando vários tipos de atributo.



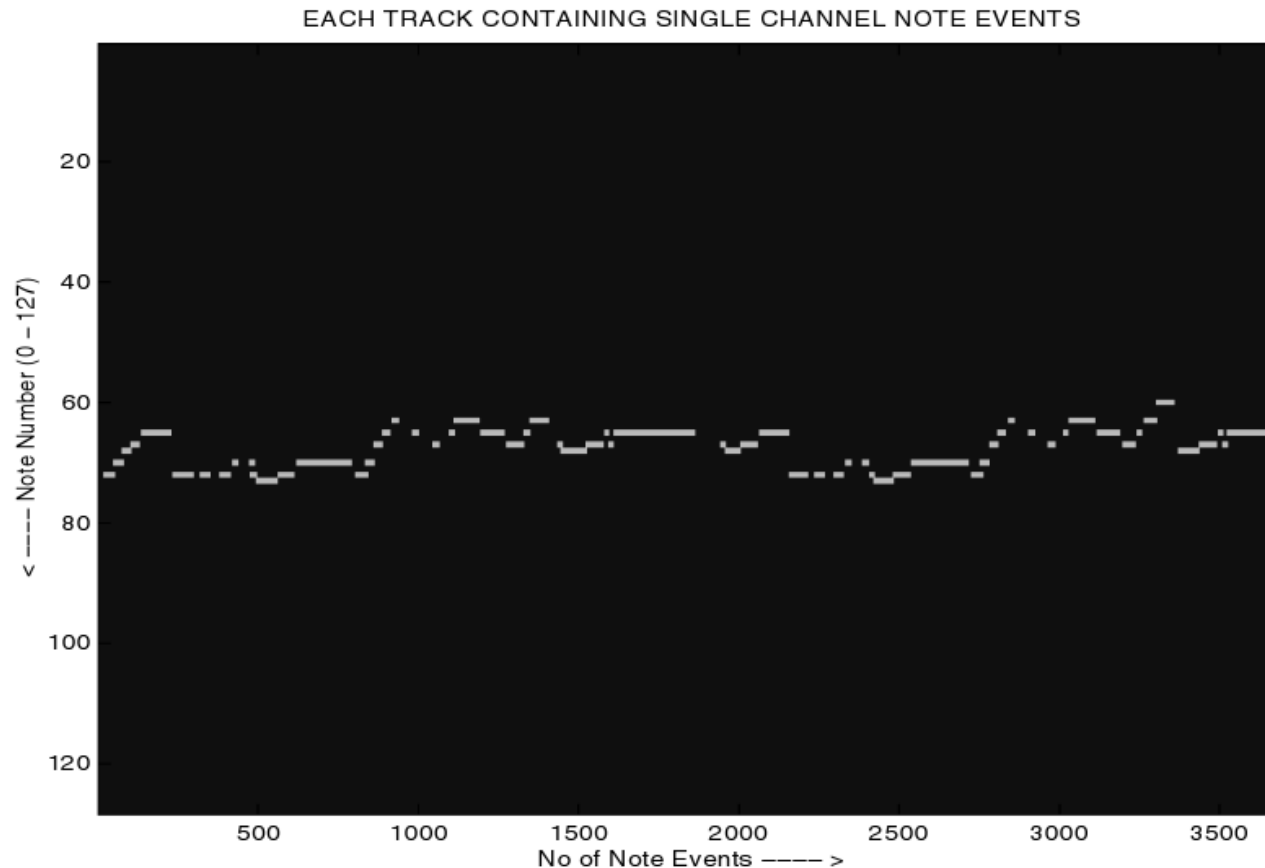
Yingying Zhu, Zhong Ming and Qiang Huang, *SVM-Based Audio Classification for Content-Based Multimedia Retrieval*, LNCS 4577, 2007.

- Classificação de gênero musical usando decomposição temporal, 30 atributos e múltiplos classificadores.



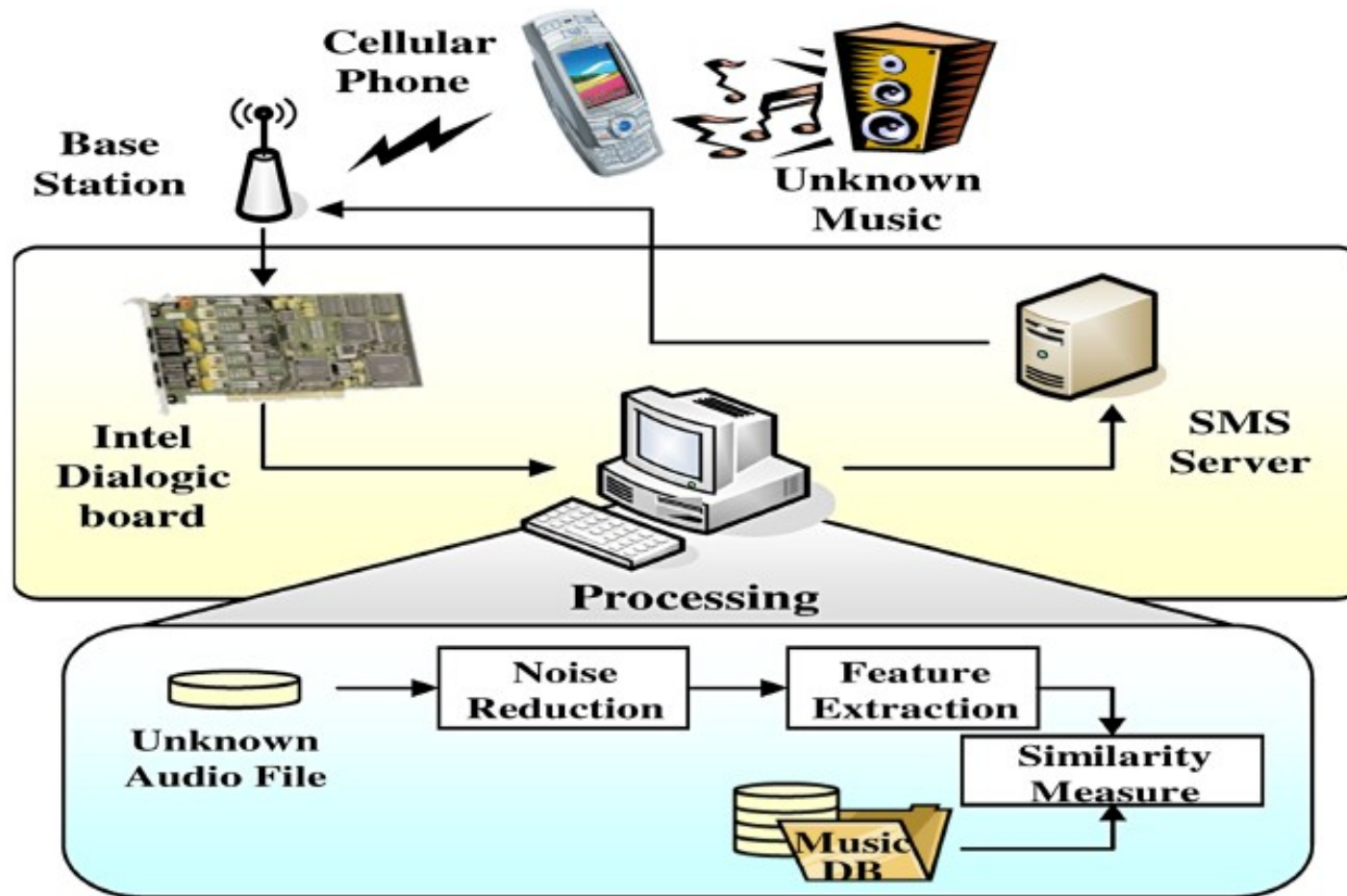
Carlos N. Silla Jr, Alessandro L. Koerich and Celso A. A. Kaestner, *A Machine Learning Approach to Automatic Music Genre Classification*, JBCS v.3 n. 14, 2008.

- Identificação da linha melódica em arquivos MIDI para sistema *Query By Humming*.



Sudha Velusamy, Balaji Thoshkahna and K.R. Ramakrishnan, *A Novel Melody Line Identification Algorithm for Polyphonic MIDI Music*, LNCS 4352, 2007.

- Identificação de música através de dispositivos móveis.



Won-Jung Yoon, Sanghun Oh and Kyu-Sik Park, *Robust Music Information Retrieval on Mobile Network Based on Multi-Feature Clustering*, LNCS 4093, 2006.

Mineração de Dados Multimídia

Vídeo

- Problemas similares ao de análise de imagens, com:
 - Sinais de áudio que podem ou não ser correlacionados;
 - Técnicas que exploram diferença temporal entre *frames*;
 - Informações auxiliares como legendas e transições.
- Duas grandes áreas de interesse:
 - Mineração/Processamento de vídeo de esportes;
 - Mineração/Processamento de vídeos de notícias.


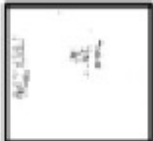












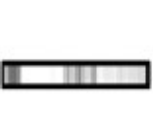




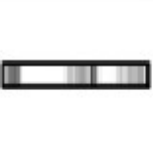
- *LSCOM – Large Scale Concept Ontology for Media: Conjunto de anotações em vídeo sobre 449 conceitos visuais sobre mais de 60.000 cenas.*
- *MediaMill Challenge Problem: Base de eventos em vídeo para estudo e comparação de algoritmos: 85 horas de vídeo com 101 conceitos.*

<http://www.lsc.com.org/>



<http://www.science.uva.nl/research/mediamill/index.php>

- Caracterização de eventos por direção de movimentos (análise de *frames*).

Category	Segments	AM	TMMI	AMMI	SR	SC
1		3.8				
		3.9				
2		4.4				
		4.9				

JungHwan Oh, JeongKyu Lee, Sanjaykumar Kote, and Babitha Bandi, *Multimedia Data Mining Framework for Raw Video Sequences*, LNCS 2797, 2003.

- Detecção de objetos, *clusters* e trilhas de objetos para mineração.



(a)



(b)

Arasanathan Anjulan and Nishan Canagarajah, *Video Object Mining with Local Region Tracking*, LNCS 4577, 2007.

- Detecção de legendas em sequências de vídeo (análise de coeficientes DCT, coerência temporal e crescimento de regiões).



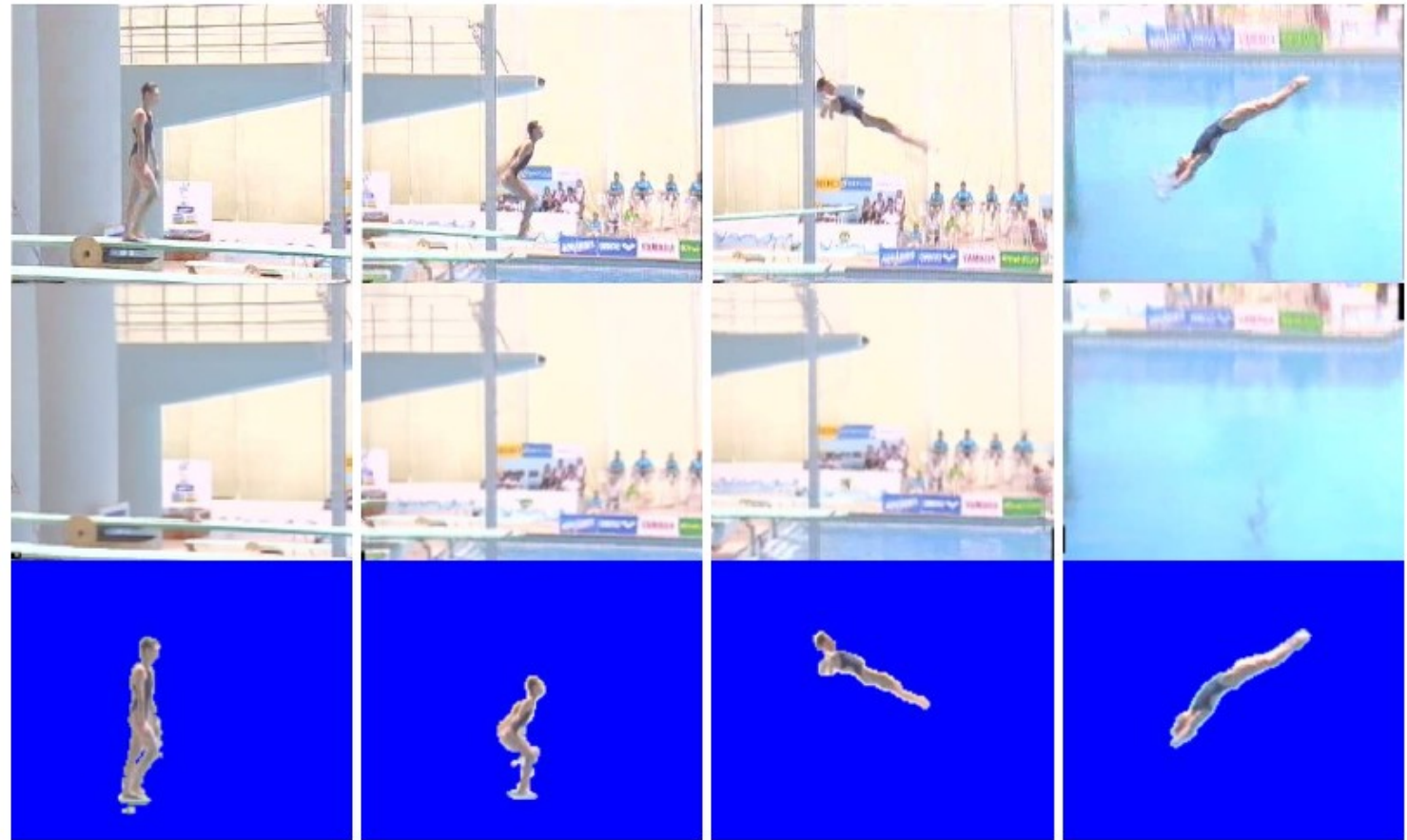
Yaowei Wang, Limin Su and Qixiang Ye, *A Robust Caption Detecting Algorithm on MPEG Compressed Video*, LNCS 4577, 2007.

- Detecção e remoção de legendas em sequências de vídeo (com *tracking*).



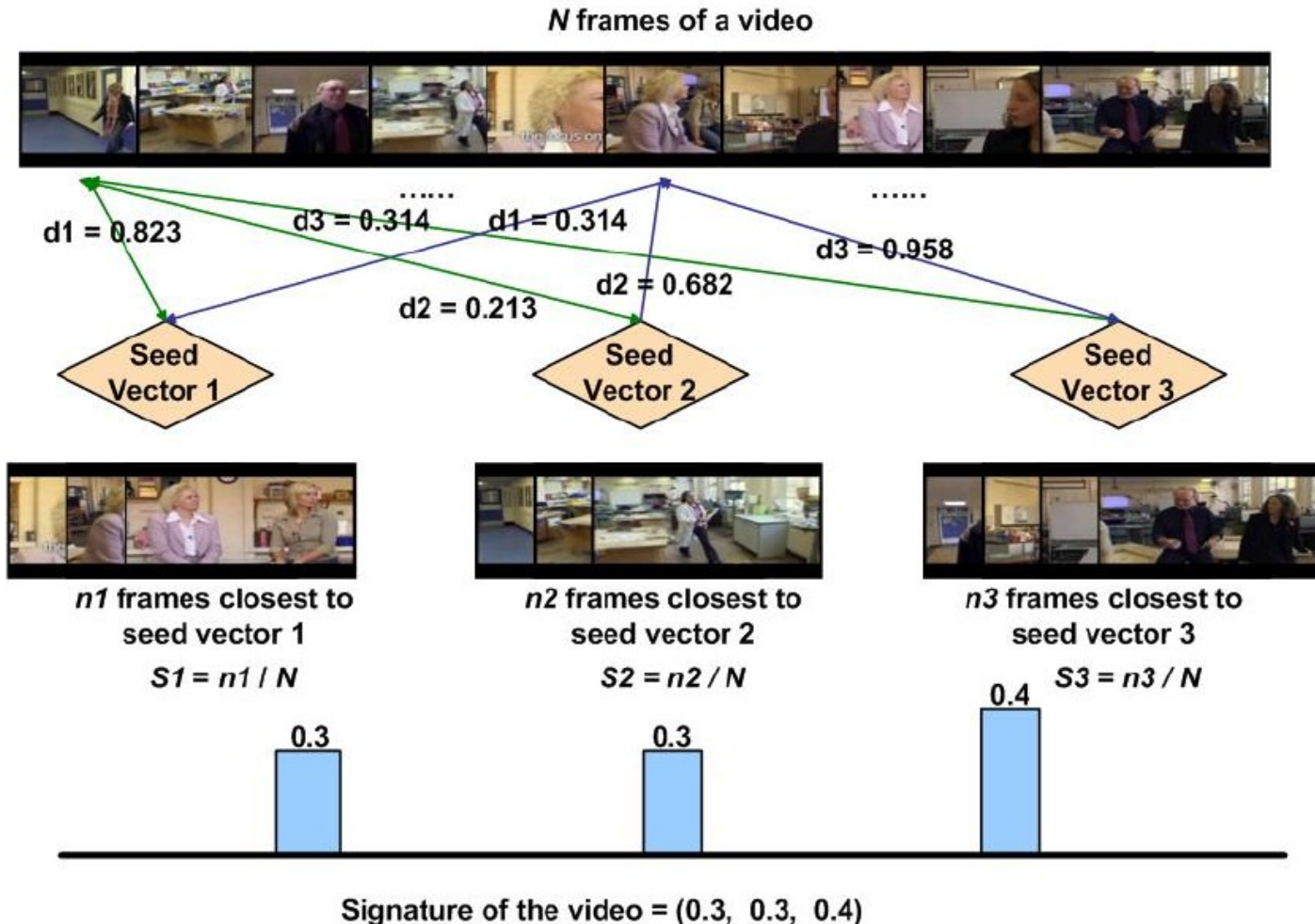
Jinqiao Wang, Qingshan Liu, Lingyu Duan, Hanqing Lu and Changsheng Xu, *Automatic TV Logo Detection, Tracking and Removal in Broadcast Video*, LNCS 4352, 2007.

- Segmentação e caracterização do movimento de atletas em vídeos de mergulho.



Haojie Li, Si Wu, Shan Ba, Shouxun Lin and Yongdong Zhang, *Automatic Detection and Recognition of Athlete Actions in Diving Video*, LNCS 4352, 2007.

- Caracterização robusta de vídeos (para detecção de cópias).



Lu Liu, Wei Lai, Xian-Sheng Hua and Shi-Qiang Yang, *Video Histogram: A Novel Video Signature for Efficient Web Video Duplicate Detection*, LNCS 4352, 2007.

- Segmentação de jogadores e bola por segmentação de imagens e análise de formas.



Yu Huang, Joan Llach, and Sitaram Bhagavathy, *Players and Ball Detection in Soccer Videos Based on Color Segmentation and Shape Analysis*, LNCS 4577, 2007.

Mineração de Dados Multimídia

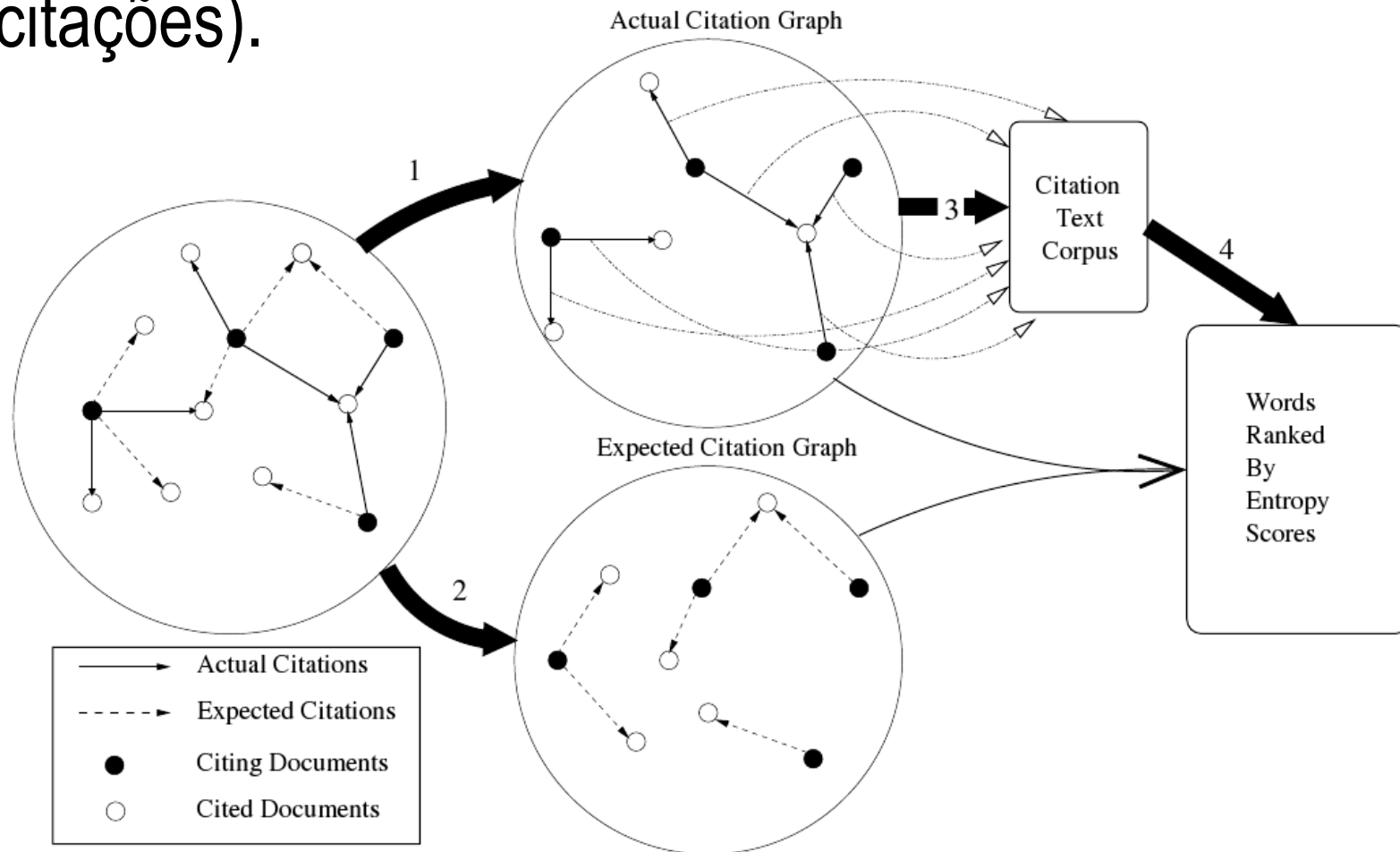
Texto

- Grande parte dos dados digitais é em forma não estruturada: texto sem formatação semântica.
- Ainda o problema de extração de atributos e caracterização.
- Pequeno volume de bytes em um documento.
 - Dentro deste, médio volume de dados estruturais e pequeno volume semântico.
- Palavras podem não ser bons atributos:
 - Ex. “Albert Einstein”: alta co-ocorrência, falta de semântica.

- Pré-processamento:
 - Filtro de formas (ex. mensagens em listas de discussão, *e-mails*, *blogs*, etc.)
 - Tokenização (como tratar símbolos como ponto final e hífen?)
 - Agrupamento em sentenças e frases.
 - Normalização com sinônimos (carro = automóvel?) e extração de radicais (formas de verbos).
 - Análise sintática.
 - Análise semântica (?)
- Grande problema de processamento de linguagem natural!

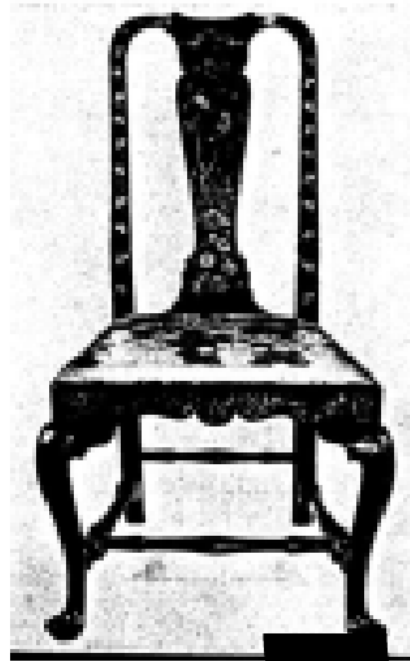
- Técnicas dependem do conteúdo e do tipo de documento.
 - Estrutura básica pode ser simples (*e-mails, blogs*) ou complexa (livros, artigos).
 - Filtragens e simplificações são dependentes da aplicação (porco = carne = comida?)
 - Tudo é facilitado se usamos domínios e dados específicos.
-
- Mineração de XML: caso específico (não coberto).

- Agrupamento de documentos científicos pela análise dos *links* (citações).



Aruna Lorensuhewa, Binh Pham, and Shlomo Geva, *Clustering Scientific Literature Using Sparse Citation Graph Analysis*, LNCS 4213, 2006.

- Análise de palavras-chave sobre conceitos abstratos (estilo) para classificação de documentos descritivos.



174. New England walnut chair from a set which descended in the Winthrop-Blanchard family. The Oriental-lacquer decoration of birds, landscape, and flowers in gold, green, and red on a black ground is combined with Western acanthus scrolls and a coat-of-arms. It is thought that these chairs, made in the mid-1700s, were sent to China to be decorated at the end of the century. The chair offers an interesting contrast to the furniture in Western styles made and decorated in China for the Bowditch and Low families, which was shown in the China Trade Exhibition at the Metropolitan Museum of Art in 1941. (*Bayou Bend Collection, Houston.*)

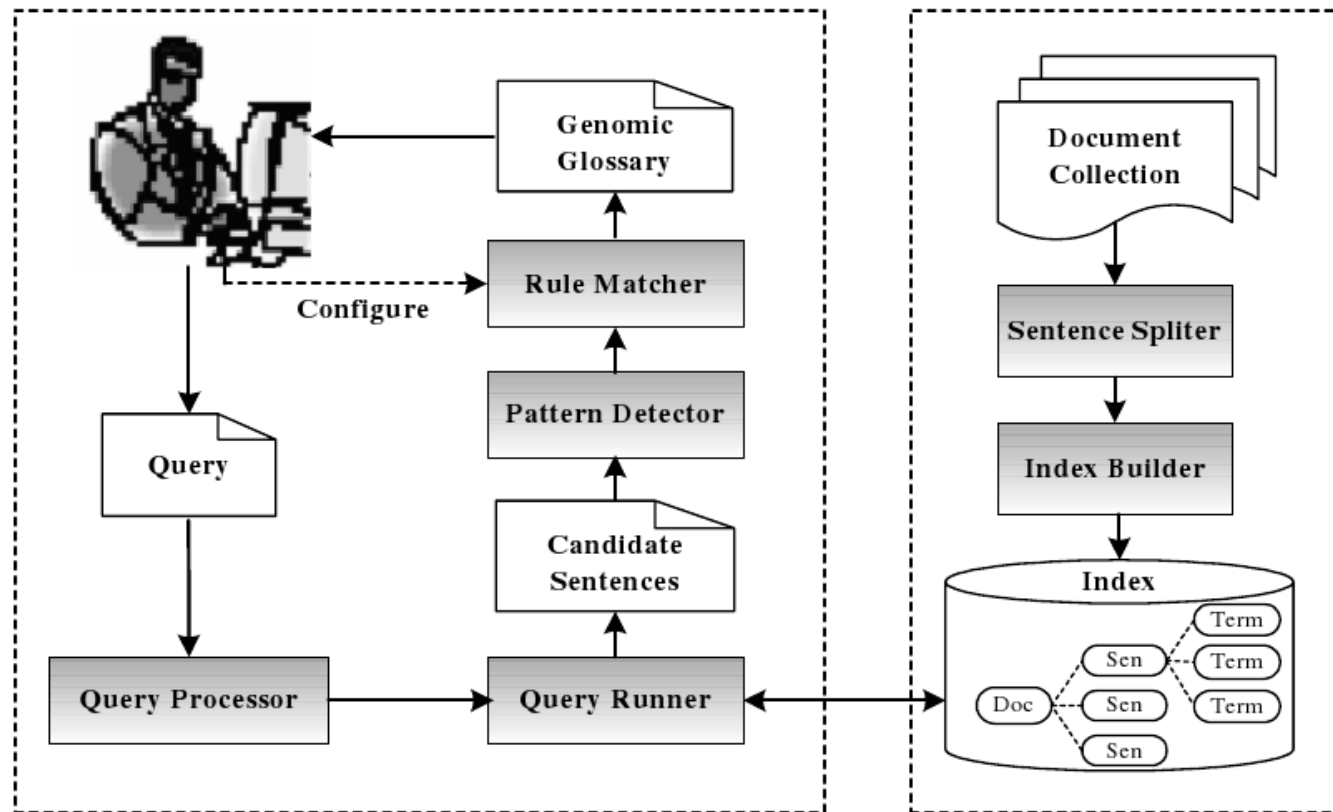
Aruna Lorensuhewa, Binh Pham, and Shlomo Geva, *Style Recognition Using Keyword Analysis*, LNCS 2797, 2003.

- Busca em textos de patentes chinesas usando *sememes*.
 - *Sememes*: unidades semânticas que não podem ser decompostas.

ID	Phrase	Sememe
061554	gentleman 男人	human 人, family 家, male 男
005241	must 必须	{modality 语气}
114646	plagiarism 剽窃	steal 偷, *copy 抄写
011940	outflank 抄袭	attack 攻打, military 军

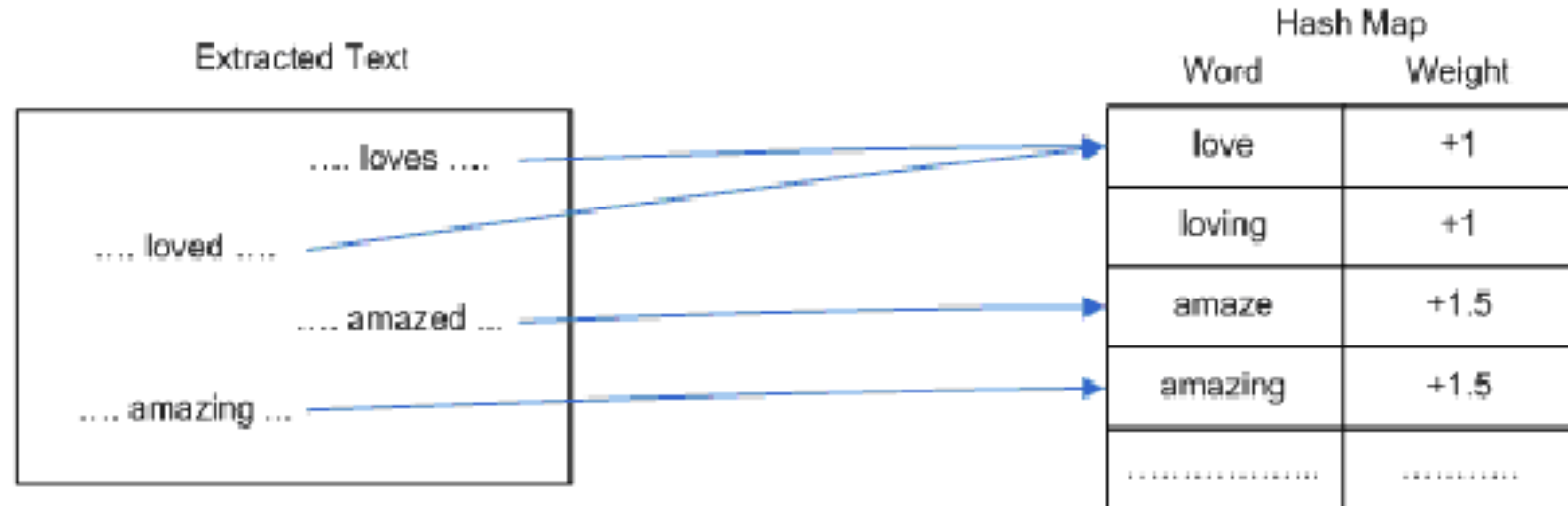
Bo Jin, Hong-Fei Teng, Yan-Jun Shi and Fu-Zheng Qu, *Chinese Patent Mining Based on Sememe Statistics and Key-Phrase Extraction*, LNCS 4632, 2007.

- Recuperação de textos do *MEDLINE* a partir de nomes de genes (extração de glossários).



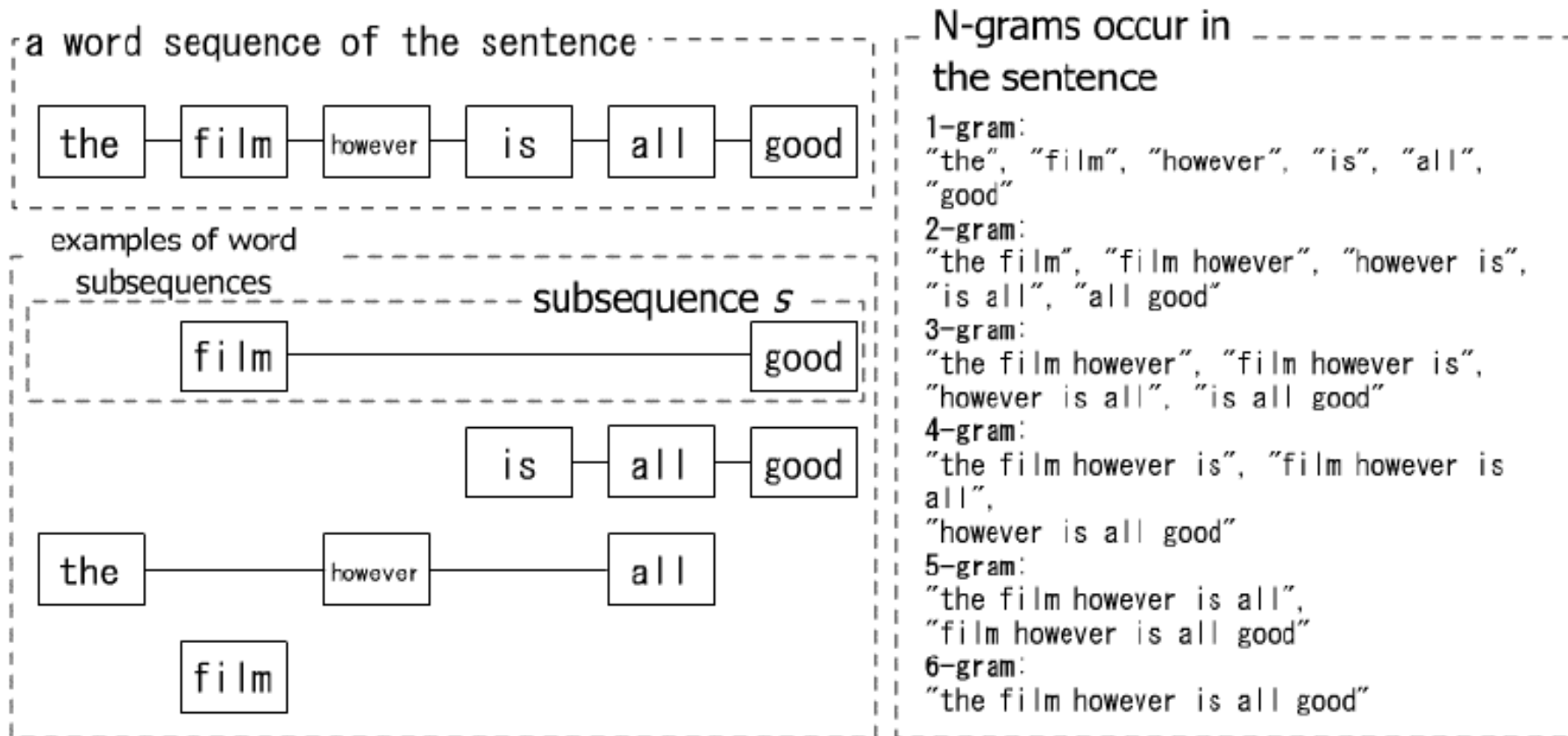
Jiao Li and Xiaoyan Zhu, *Automatic Extraction of Genomic Glossary Triggered by Query*, LNCS 3916, 2006.

- Tentativa de correlação de termos emotivos (opiniões) em *blogs* sobre companhias.



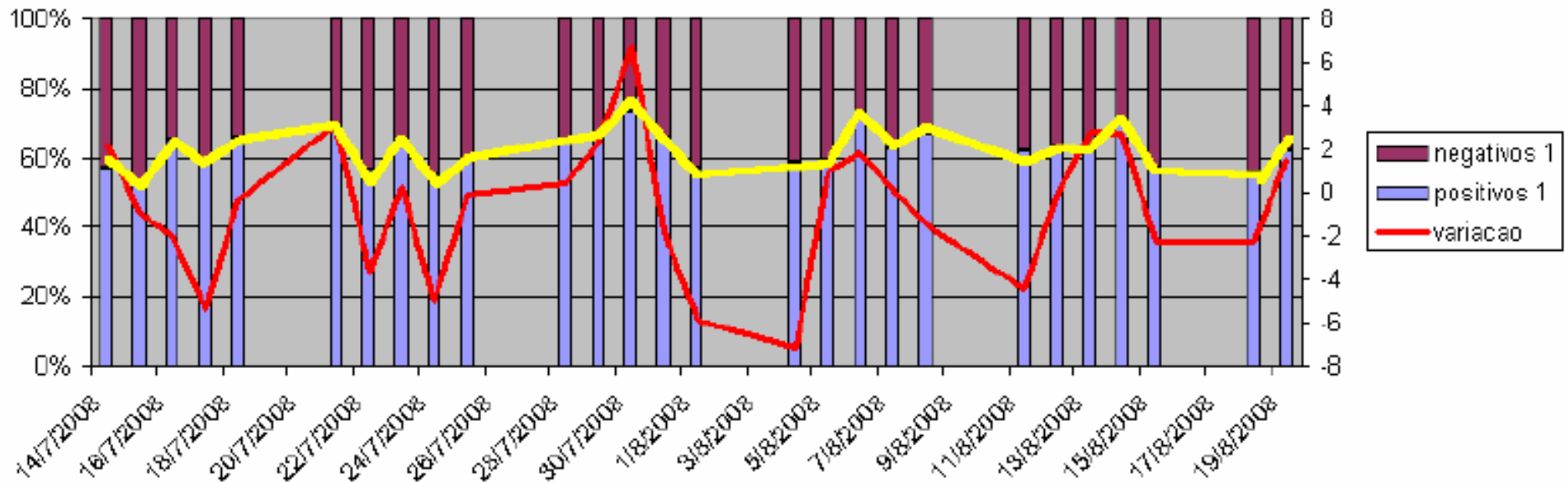
James Geller, Sapankumar Parikh and Sriram Krishnan, *Blog Mining for the Fortune 500*, LNCS 4571, 2007.

- Classificação do sentimento de um documento por subsequências e subárvores.



Shotaro Matsumoto, Hiroya Takamura and Manabu Okumura, *Sentiment Classification Using Word Sub-sequences and Dependency Sub-trees*, LNCS 3518, 2005.

- Correlação de uso de termos positivos e negativos em fórum de discussão sobre ação de uma empresa.



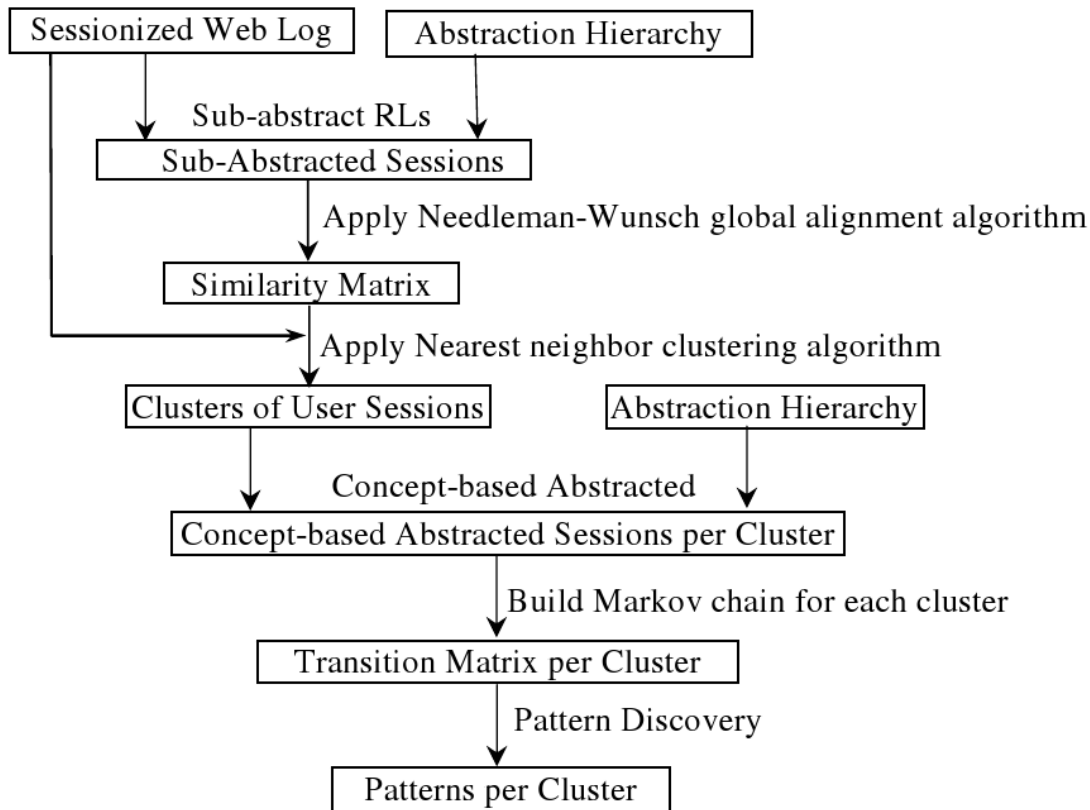
Suporte à Análise de Risco Financeiro Utilizando Mineração de Texto para a Descoberta de Variáveis Subjetivas, Relatório Final de Rogério Ishibashi , disciplina CAP-359, INPE.

Mineração de Dados Multimídia

WWW

- Muitos problemas e aplicações relacionados com mineração de textos.
- Grandes diferenças:
 - Estrutura na WWW *pode* ser mais simples!
 - Informação auxiliar obtida dos servidores.
- Aplicações (algumas comentadas):
 - Aplicações em sistemas de recomendação e filtragem colaborativa.
 - Caracterização, agrupamento e classificação de *Web Usage*.
 - Mineração de *logs* para segurança e detecção de intrusão.
 - Mineração de redes sociais, *blogs*, etc.

- Descoberta e agrupamento de SUPs (*Significant Usage Patterns*) usando sessões e conceitos sobre o site.



Cluster No.	No. of Sessions	Threshold (θ)	Average Session Length	No. of States	SUPs
1	1746	0.3	9.6	98	1. S-C ₁ -C ₁ -C ₂ -C ₃ -C ₄ -C ₅ -C ₆ -C ₇ -E 2. S-C ₁ -C ₁ -C ₂ -C ₃ -C ₄ -C ₅ -E 3. S-C ₁ -C ₁ -C ₂ -C ₃ -E 4. S-C ₁ -C ₂ -C ₃ -C ₃ -C ₄ -C ₅ -C ₆ -C ₇ -E 5. S-C ₁ -C ₂ -C ₃ -C ₄ -C ₄ -C ₅ -C ₆ -C ₇ -E 6. S-C ₁ -C ₂ -C ₃ -C ₄ -C ₅ -C ₅ -C ₆ -C ₇ -E 7. S-C ₁ -C ₂ -C ₃ -C ₄ -C ₅ -C ₆ -C ₆ -C ₇ -E 8. S-C ₁ -C ₂ -C ₃ -C ₄ -C ₅ -C ₆ -C ₇ -C ₇ -E 9. S-C ₁ -C ₂ -C ₃ -C ₄ -C ₅ -C ₆ -C ₇ -C ₈ -E 10. S-C ₁ -C ₂ -C ₃ -C ₄ -C ₅ -C ₆ -C ₇ -E 11. S-C ₁ -C ₂ -C ₃ -C ₄ -C ₅ -C ₆ -E 12. S-C ₁ -C ₂ -C ₃ -C ₄ -C ₅ -E 13. S-C ₁ -C ₂ -C ₃ -C ₄ -E 14. S-C ₁ -C ₂ -C ₃ -E
2	241	0.37	6.6	38	1. S-P ₁ -P ₂ -P ₃ -P ₃ -E 2. S-P ₁ -P ₂ -P ₃ -P ₄ -P ₄ -P ₅ -E 3. S-P ₁ -P ₂ -P ₃ -P ₄ -P ₄ -E 4. S-P ₁ -P ₂ -P ₃ -P ₄ -P ₅ -P ₄ -E 5. S-P ₁ -P ₂ -P ₃ -P ₄ -P ₅ -P ₅ -E 6. S-P ₁ -P ₂ -P ₃ -P ₄ -P ₅ -P ₆ -C ₁ -E 7. S-P ₁ -P ₂ -P ₃ -P ₄ -P ₅ -P ₆ -P ₇ -E 8. S-P ₁ -P ₂ -P ₃ -P ₄ -P ₅ -P ₆ -E 9. S-P ₁ -P ₂ -P ₃ -P ₄ -P ₅ -E 10. S-P ₁ -P ₂ -P ₃ -P ₄ -C ₁ -E 11. S-P ₁ -P ₂ -P ₃ -P ₄ -E 12. S-P ₁ -P ₂ -P ₃ -C ₁ -E 13. S-P ₁ -P ₂ -P ₃ -E 14. S-P ₁ -P ₂ -E
3	13	0.3	3.0	6	1. S-C ₁ -P ₁ -P ₁ -P ₂ -E 2. S-C ₁ -P ₁ -P ₁ -E 3. S-C ₁ -P ₁ -P ₂ -E 4. S-C ₁ -P ₁ -E 5. S-I ₁ -P ₁ -P ₁ -P ₂ -E 6. S-I ₁ -P ₁ -P ₁ -E 7. S-I ₁ -P ₁ -P ₂ -E 8. S-I ₁ -P ₁ -E

Lin Lu, Margaret Dunham, and Yu Meng, *Mining Significant Usage Patterns from Clickstream Data*, LNCS 4198, 2006.

- Descoberta de padrões sequenciais em ataques multiestágio em *IDSs*.

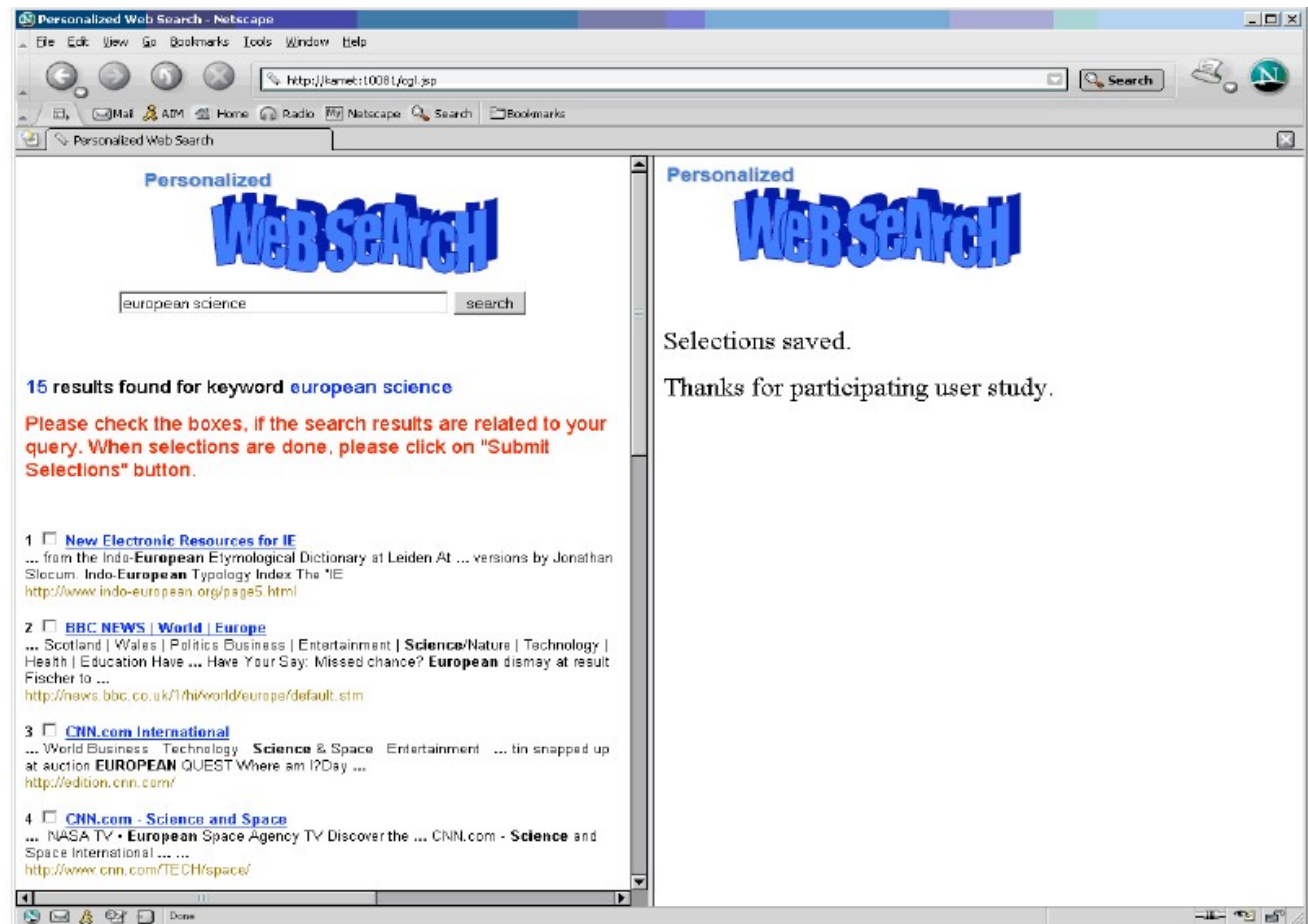
Attack Class	DetectTime	Other attributes
...
7	06-08-09-11:12:14	...
4	06-08-09-11:13:18	...
5	06-08-09-11:34:46	...
2	06-08-09-11:34:46	...
3	06-08-10-00:06:53	...
5	06-08-10-00:12:47	...
...

Subsequence id	Attack class sequence
1	<(7,5),2,6,4>
2	<7,6,4,(6,5)>
3	<7,2,6,4>
4	<7,6,5>

Subsequence id	1-sequence	2-sequence	3-sequence	4-sequence
<(7,5),2,6,4>	<7> <2>	<7,2>,<7,6>,<7,4>,<7,5>,<2,6>,<2,4>,<6,4>,<6,5>,<4,5>	<7,2,6 >,<7,2,4>,<7,6,4>,<7,6,5>,<2,6,4>	<7,2,6,4>
<7,6,4,(6,5)>	<6> <4>			
<7,2,6,4>	<5>			
<7,6,5>				

Zhitang Li, Aifang Zhang, Dong Li, and Li Wang, *Discovering Novel Multistage Attack Strategies*, LNCS 4632, 2007.

- Melhoria do resultado do PageRank usando preferências e perfis de usuários.



Mehmet S. Aktas , Mehmet A. Nacar and Filippo Menczer, *Using Hyperlink Features to Personalize Web Search*, LNCS 3932, 2006.

- Sistema de busca (SPIRIT) que usa termos geográficos/referenciais para busca em documentos indexados geograficamente.

Query Type	Example
1. Distance	1. schools <i>within 10 km</i> of Zurich city centre 2. hotels <i>near</i> Cardiff University
2. Topological	1. hospitals <i>in</i> London
3. Directional	1. holiday resorts <i>north of</i> Milan

Subodh Vaid, Christopher B. Jones, Hideo Joho and Mark Sanderson, *Spatio-textual Indexing for Geographical Search on the Web*, LNCS 3633, 2005.

Comentários Finais

- Vimos alguma coisa...
- .. faltou MUITA coisa!
- Meu *semantic gap*: distância entre algoritmos e aplicações.
- Pontos para melhoria: exemplos concretos de extração de atributos de imagens, texto e áudio....
 - ... precisaremos de mais do que oito horas!
- Somente poucos exemplos foram mostrados.
 - Muitos do *Lecture Notes in Computer Science*.
 - Existem vários congressos, alguns no Brasil.

- <http://www.lac.inpe.br/~rafael.santos>
 - Material de outros cursos, aulas, etc.
- KD Nuggets.com
- Propaganda descarada: *“Então você quer ser um minerador de dados?”*
- **Perguntas?**