Introdução a Algoritmos de Computação Natural para Mineração de Dados

Gisele L. Pappa
Universidade Federal de Minas Gerais
glpappa@dcc.ufmg.br

Organização do Curso

- Parte 1: Introdução a Mineração de Dados
- Parte 2: Introdução a Algoritmos de Computação Natural
- Parte 3: Mineração de Dados + Algoritmos de Computação Natural
- Parte 4: Prática – WEKA

Parte 1:
Introdução a
Mineração de Dados

Mineração de Dados

- Por quê estudar?
  - Crescimento explosivo na quantidade de dados coletados e disponíveis
- De onde vêm esses dados?
  - Negócios: Web, comércio eletrônico, mercado de ações
  - Ciência: Bioinformática, sensoramento remoto, simulações
  - Sociedade: redes sociais, câmeras digitais, notícias

Google trends

Search Trends

Tip: Use commas to compare multiple search terms.

Examples

newspapers, blogs, magazines — daytona 500, indy 500

abc.com, cbs.com, nbc.com, fox.com — apple.com, microso

Hot Topics[New!] (USA)

1. ipad 3g
2. jesus
3. good friday
4. sania mirza
5. shoaib malik
6. hyderabad
7. miami medical
8. john forsythe
9. erykah badu
10. erin andrews

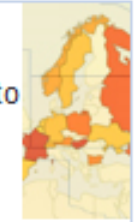
More Hot Topics:

Search latest

Google Flu Trends

Google Flu Trends uses aggregated Google search data to estimate flu activity in near real-time in 20 countries.

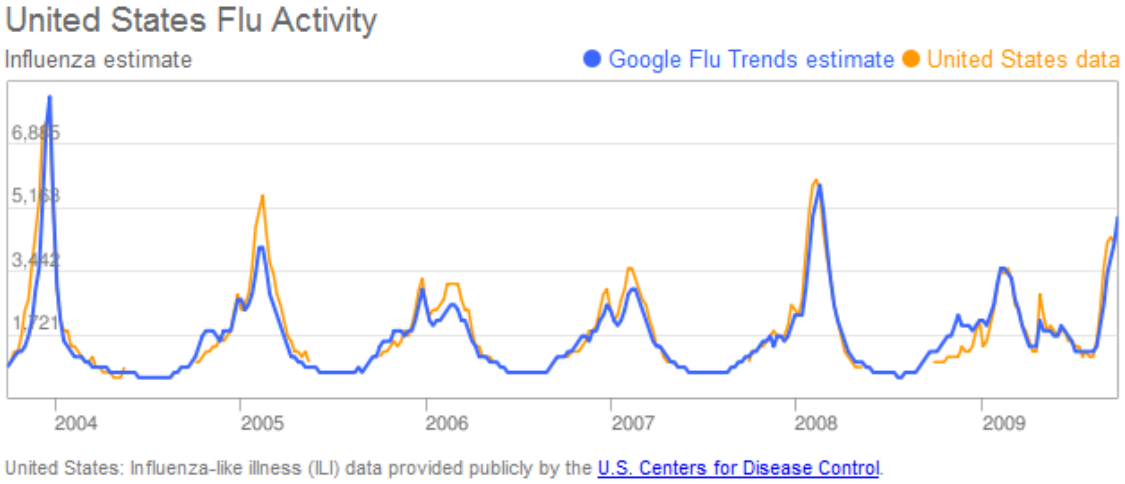
Learn more



Detecting influenza epidemics using search engine query data

Jeremy Ginsberg[1], Matthew H. Mohebbi[1], Rajan S. Patel[1], Lynnette Brammer[2], Mark S. Smolinski[1] & Larry Brilliant[1]

[1]Google Inc. [2]Centers for Disease Control and Prevention



http://www.google.org/flutrends/about/how.html

Eleições 2010

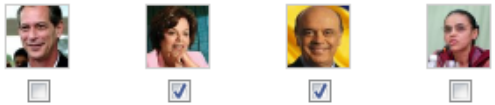
observatóriodaweb

Winweb

Instituto Nacional de Ciência e Tecnologia para a Web

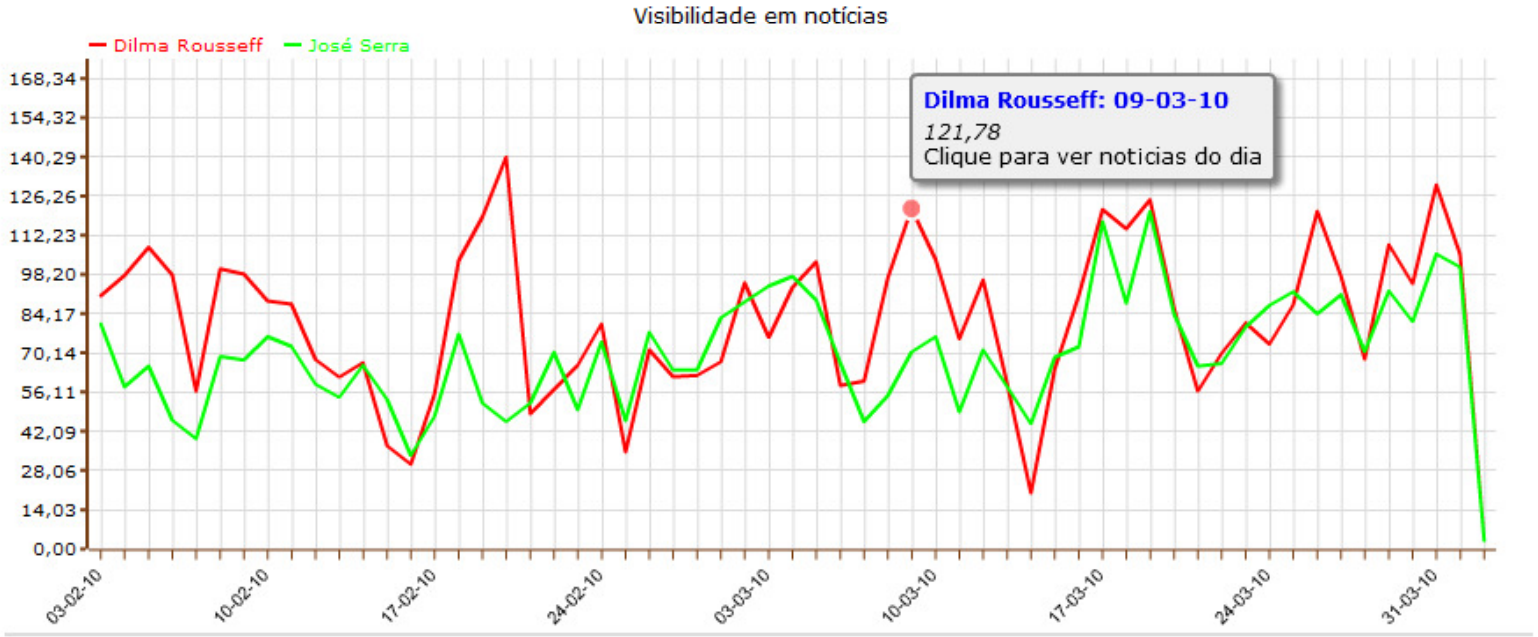
Visibilidade

1- Escolha duas personalidades para a análise e informe o período:



Selecione o período: Últimos dois meses

Ok, mostre-me os resultados



http://observatorio.inweb.org.br

Era da Informação

ou

Era dos Dados???

Sumário

- O que é mineração de dados?
- Visão Geral de Mineração de Dados
  - Tipo de conhecimento a ser extraído
  - Tipo dos dados a serem minerados
  - Tipos das técnicas a serem utilizadas
  - Aplicações consideradas
- Conhecendo seus dados
- Pré-processamento
- Classificação
- Avaliação dos Algoritmos de Classificação

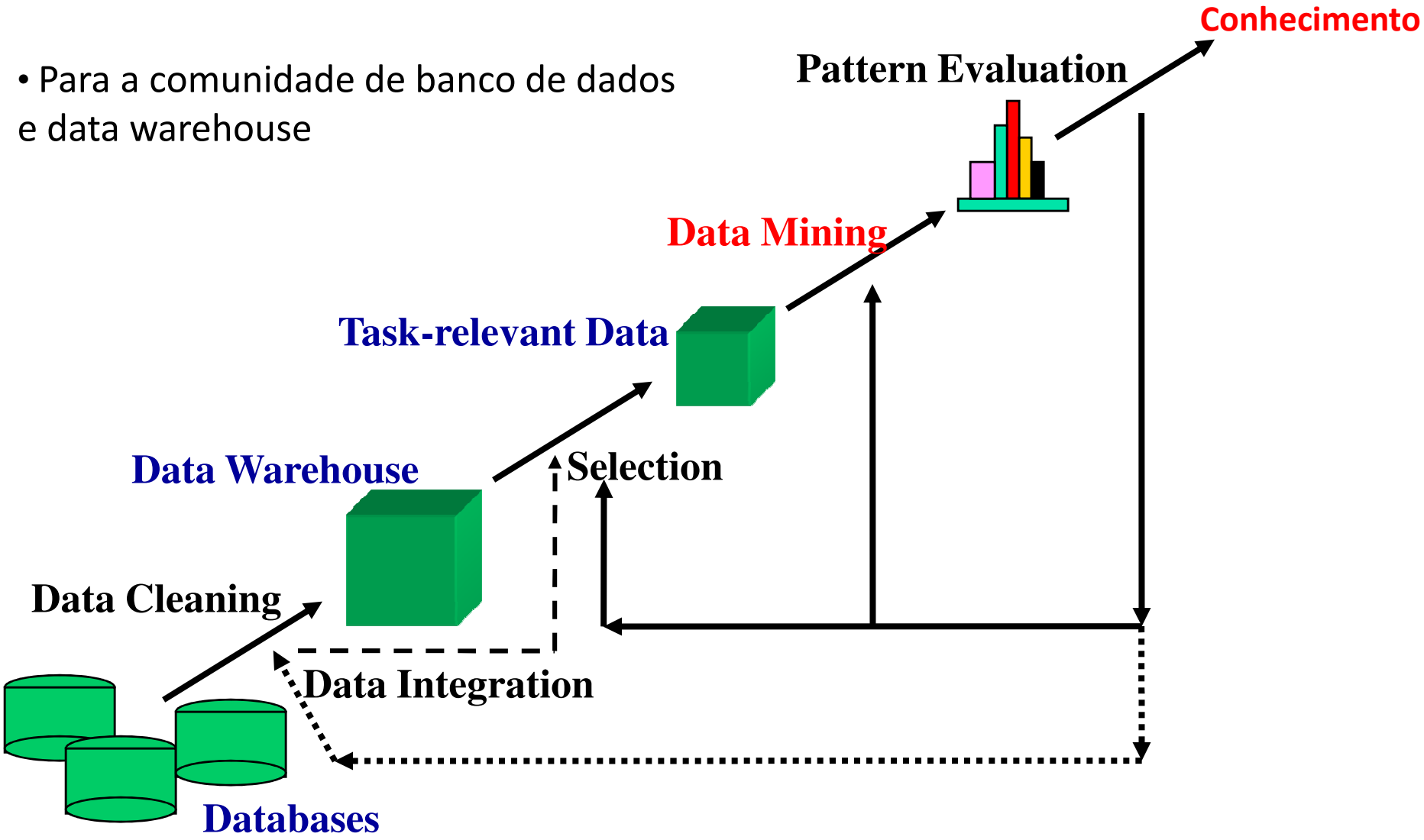
O que é Mineração de Dados?

- Extração de padrões interessantes ou conhecimento de um grande volume de dados
- Também conhecido como KDD (*Knowledge Discovery in Databases)*
- O que é um padrão interessante?
  - Não-trivial
  - Implícito
  - Anteriormente desconhecido
  - Útil

Se (sexo paciente == feminino)
então grávida

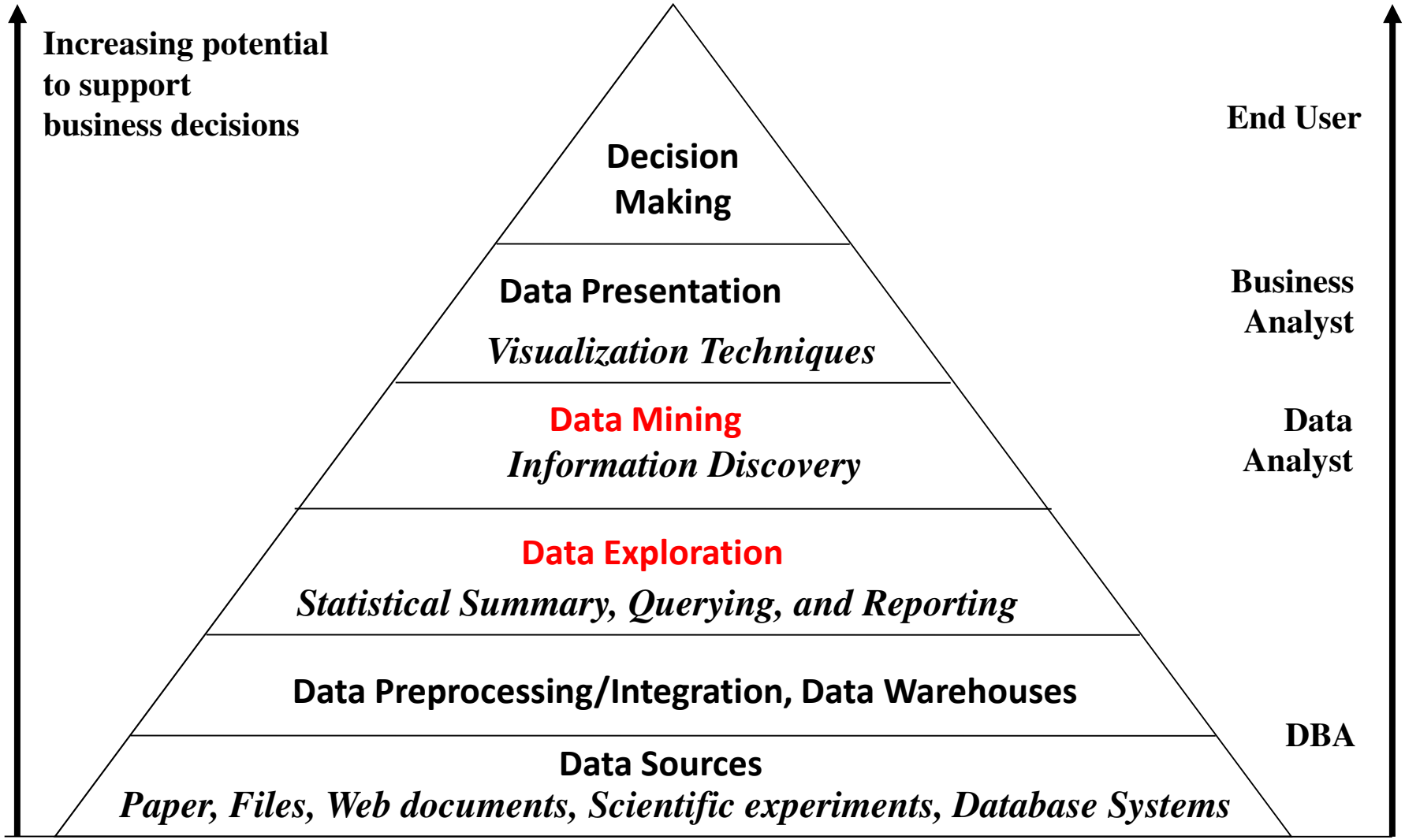
O que é Mineração de Dados?

• Para a comunidade de banco de dados e data warehouse



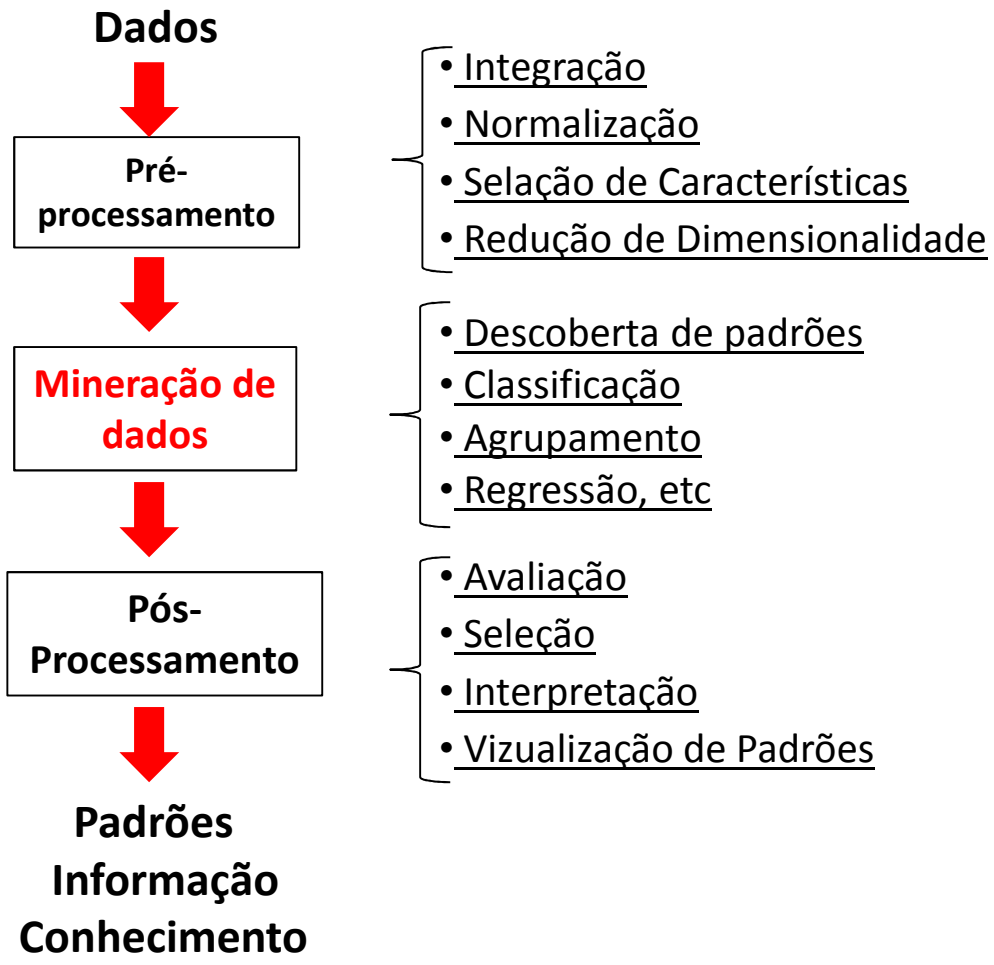
O que é Mineração de Dados?

- Para a comunidade *business intelligence*



O que é Mineração de Dados?

- Para as comunidades de aprendizado de máquina e estatística



Visão Geral de Mineração de Dados

1. Tipo de conhecimento a ser minerado
   - Mineração de padrões frequentes, Classificação, Agrupamento, etc
   - Preditivo ou descritivo
2. Tipo dos dados a serem minerados
   - Relacionais, sequências, temporais, espaciais, *streams*, textos, grafos, etc
3. Técnicas a serem utilizadas
   - Estatística, aprendizado de máquina, visualização, reconhecimento de padrões, etc
4. Aplicações consideradas
   - *Web mining*, bio-data mining, análise de fraudes, análise de tendências no mercado financeiro, etc

Tipo de conhecimento a ser minerado

- Mineração de padrões ou itens frequentes
  - Que itens são frequentemente comprados juntos no Wallmart?
- Tarefa de Associação
  - Fraldas -> cerveja [0.5%, 75%] (suporte,confiança)
- Problemas:
  - Como encontrar padrões de forma eficiente em grandes bases?
  - Como usar esses padrões nas tarefas de classificação e agrupamento?

Tipo de conhecimento a ser minerado

- Aprendizado supervisionado
  - Rótulo das classes conhecido no conjunto de treinamento
- Classificação e Previsão
  - Construir modelos baseados em um conjunto de treinamento
  - Descrever ou distinguir classes para previsões futuras
- Métodos comuns
  - Árvores de decisão, regras de associação, redes Bayesianas, SVM, regressão logística, redes neurais, algoritmos evolucionários, etc

Tipo de conhecimento a ser minerado

- Aprendizado não-supervisionado
  - Agrupamento
- Agrupar dados similares criando categorias (ou grupos)
- Princípio: maximizar a similaridade inter-categoria e minimizar a similaridade intra-categoria

Tipo dos dados a serem minerados

- Bases de dados relacionais ou transacionais
- Bases de dados avançadas:
  - Multimídia - Imagens ou vídeos
  - Séries temporais
  - Dados espaciais ou espaço-temporais
  - Texto
  - Web
- Bases de dados representadas por grafos
  - Compostos químicos e redes sociais

Maiores desafios da área

- Mineração dados heterogêneos
- Lidar com dados de alta dimensão
- Lidar com dados incompletos, incertos e com ruído
- Incorporar exceções e conhecimento a priori sobre os problemas sendo resolvidos
- Eficiência e escalabilidade dos algoritmos

Conhecendo seus Dados:
Foco em Dados Estruturados

Instâncias

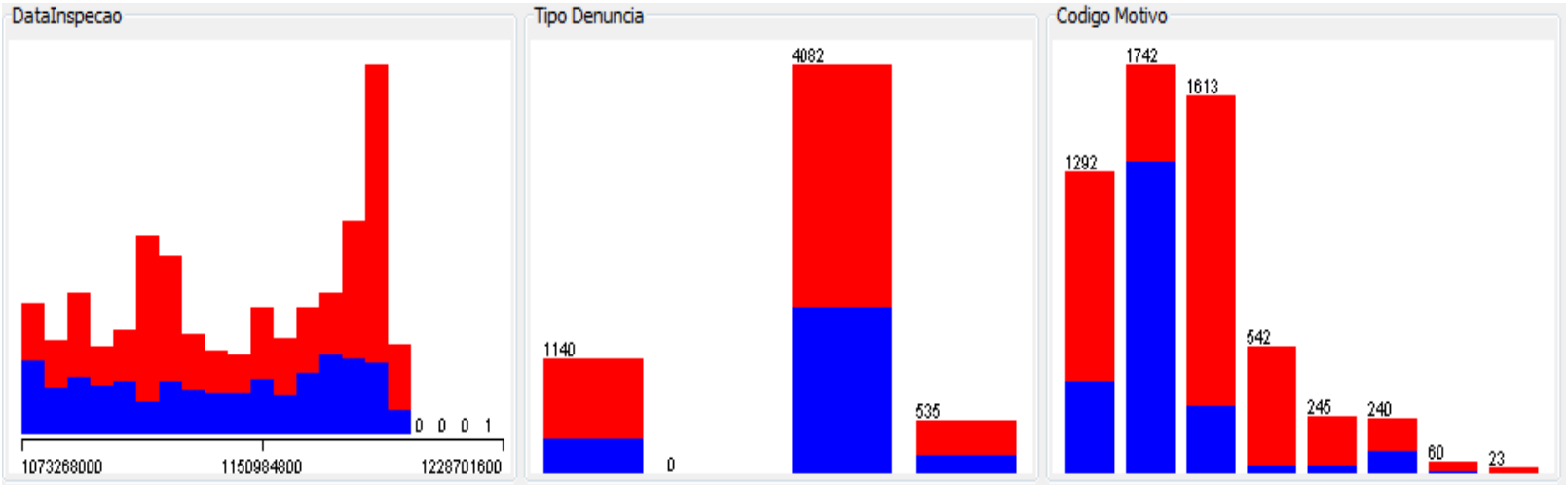
- Bases de dados são compostas por instâncias
- Uma instância representa uma entidade no mundo real
  - Ex: consumidores, pacientes, proteínas
- Instâncias são descritas por um conjunto de atributos
- Numa base de dados, linhas representam instâncias e colunas atributos

Atributos

- Também conhecidos como dimensões, características, variáveis
  - Ex: nome, endereço, telefone
- Tipos de atributos:
  - Nominais: categorias, estados
    - Ex: cor do cabelo: {loiro, ruivo, preto, branco, castanho}
  - Ordinais : a ordem importa, mas o valor que representada cada categoria não é conhecido
    - Ex: altura: {baixo, médio, alto, muito alto}
  - Numéricos
    - Ex: salário, temperatura
- Atributo discreto (número finito de estados) versus contínuo (normalmente representado por um número real)

Estatísticas básicas sobre os dados

- Tendência central, variação e espalhamento (*spread*)
- Dispersão dos dados
  - Média, mínimo e máximo
  - Exemplo: problema de 2 classes



Pré-Processamento:
Visa aumentar a qualidade dos dados

Principais tarefas de pré-processamento

- Limpeza de dados
  - Tratar dados faltantes (*missing values)*, resolver inconsistências, identificar e remover *outliers*
- Integração de dados
- Redução de dados
  - Redução de dimensionalidade, numerosidade e compressão de dados
- Transformação e discretização de dados
  - Normalização de dados

Limpeza de dados

- Dados reais são
  - Incompletos: faltam valores para atributos, faltam atributos, ou contém dados agregados
    - Ex: Profissão = “ ” (dado faltante)
  - Contêm ruídos, erros e *outliers*
    - Ex: Salário = “-10” (ERRO!)
  - Inconsistência: discrepâncias em códigos ou nomes
    - Ex: idade = “42” e Data de nascimento = 03/07/1997”
    - Atributo assume valores “1,2,3”, e depois passa a assumir “A, B, C”

Dados Incompletos

- Dados nem sempre estão disponíveis
  - Podemos ter instâncias com valores faltantes
- Causas mais comuns para dados faltantes:
  - Mal funcionamento de um equipamento
  - Dado não considerado importante
  - Dado inconsistente com outro registro, e por isso removido da base

Dados Incompletos

- Como tratar dados incompletos?
  - Ignorar a instância
  - Preencher os valores manualmente
  - Preencher os valores automaticamente:
    - Usando uma constante global
    - Usando a média
    - Usando a média de todas as instâncias pertencentes a mesma classe
    - Inferência baseada em um método Bayesiano ou uma árvore de decisão

Dados com ruído

- Ruído: erro aleatório ou variância no valor de uma variável
- Principais causas:
  - Problemas na entrada dos dados
  - Problemas na transmissão dos dados
  - Inconsistência de nomenclatura

Como tratar dados com ruído?

- Regressão
  - Fazer um *fitting* dos dados usando uma função
- Agrupamento
  - Detectar e remover *outliers*
- Combinar inspeção automática com inspeção humana
  - Detectar valores possivelmente ruidosos, e deixar com que eles sejam verificados por humanos

Principais tarefas de pré-processamento

- Limpeza de dados - OK
- Integração de dados
- Redução de dados
- Transformação e discretização de dados

Integração de Dados

- Combinar dados de diferentes fontes, normalmente em diferentes formatos
- Problemas de identificação e deduplicação de entidades
- Mesmo atributo em fontes diferentes possui valores diferentes:
  - Sistema métrico diferente
  - Escala diferente
- Problema: pode gerar redundância
  - Tratada com teste de correlação e co-variância

Principais tarefas de pré-processamento

- Limpeza de dados - OK
- Integração de dados - OK
- Redução de dados
- Transformação e discretização de dados

Redução de Dados

- Obter uma representação reduzida dos dados que, quando analisada, leve aos mesmos resultados obtidos com os dados completos
- Por quê?
  - Bases de dados são normalmente imensas, acarretando alto custo computacional
- Estratégias:
  - Redução de dimensionalidade
  - Redução de dados (numerosidade)
  - Compressão de dados

Redução de Dimensionalidade

- Evita a maldição da dimensionalidade
  - Quanto mais dados, mais esparsa a base de dados e mais difícil de aprender
- Ajuda a reduzir o número de atributos irrelevantes e remover ruído
- Reduz o tempo necessário para a mineração
- Facilita a visualização dos dados

Redução de Dimensionalidade

- Técnicas:
  - Transformadas *Wavelet*
  - PCA (*Principal Component Analysis*) (WEKA)
  - Métodos supervisionados e não lineares, como seleção de atributos
- Seleção de atributos (WEKA)
  - Dados normalmente não são criados para serem minerados
  - Remove atributos irrelevantes para o processo, tais como identificadores

Seleção de Atributos

- Dados *K* atributos, existem 2^K combinações possíveis
- Utilização de heurísticas para seleção:
  - O melhor atributo é selecionado através de testes de significância assumindo independência entre eles
  - Seleção *Greedy*
    - Seleciona o melhor atributo
    - Seleciona o segundo melhor condicionado ao primeiro, e assim por diante
  - Eliminação *Greedy*
    - Elimina o pior atributo a cada iteração
  - Seleção baseada em busca em largura
    - Seleciona sempre os *n* melhores, ao invés do melhor
  - Algoritmos evolucionários

Redução de Dados

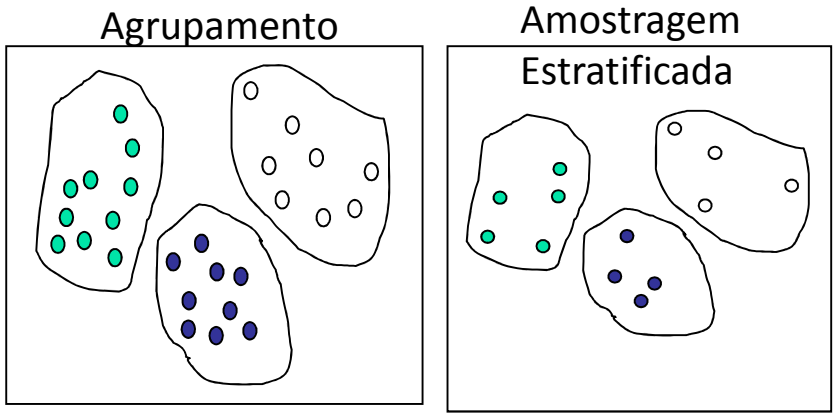
- Por quê?
  - Permite que os algoritmos de mineração de dados sejam mais eficientes
- Pode ser feita de pelo menos duas maneiras:
  1. Escolhe formas alternativas de representar os dados
     - Ex: rodar uma regressão linear e guardar apenas os coeficientes da função encontrada
     - Agrupar os dados e escolher uma representação para cada grupo

Redução de Dados

2. Amostragem
   - Obter uma amostra *m* capaz de representar o conjunto completo de dados *N*
   - Como amostrar os dados???
     - Aleatoriamente (não funciona bem para dados com classes desbalanceadas)
     - Estratificada
       - » Mantém a distribuição das classes nos dados originais

- 2 técnicas podem

Ser utilizadas em conjunto



Principais tarefas de pré-processamento

- Limpeza de dados - OK
- Integração de dados - OK
- Redução de dados - OK
- Transformação e discretização de dados

Transformação dos Dados

- Encontrar uma função que mapeie todos os valores de um atributo para um novo conjunto de valores
- Principais técnicas:
  - Construção de atributos
  - Agregação - sumarização
  - Normalização
    - Min-max
    - Z-score
    - Escala decimal
  - Discretização

Normalização

- Min-max: novo intervalo[$nmin_A$, $nmax_A$]

$$v' = \frac{v - min_A}{max_A - min_A}(nmax_A - nmin_A) + nmin_A$$

  - Ex: Se os salários variam de 12.000 a 98.000 e queremos normalizá-los entre [0,1], entaõ 73.000 será mapeado para

$$\frac{73.600 - 12.000}{98.000 - 12.000}(1 - 0) + 0 = 0.716$$

- Z-score (μ: mean, σ: standard deviation): $v' = \frac{v - \mu_A}{\sigma_A}$
  - seja μ = 54,000, σ = 16,000, então $\frac{73{,}600 - 54{,}000}{16{,}000} = 1.225$

- Normalização por escala decimal:

$$v' = \frac{v}{10^j}$$, onde j é o menor inteiro de forma que Max(|v'|) < 1

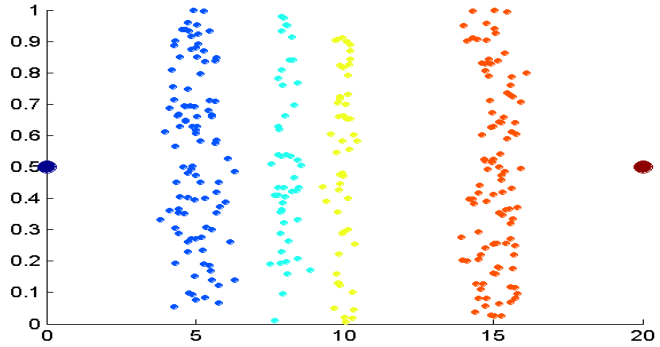
Discretização

- 3 tipos de atributos:
  - Nominais, Ordinais e Contínuos
- Discretização: divide o intervalo de um atributo contínuo em intervalos
  - Os “nomes” de cada intervalo podem então substituir os valores contínuos
  - Pode levar em conta a classe dos exemplos ou não
- Métodos comuns:
  - Divisão em intervalos
  - Análise de histogramas
  - Agrupamento

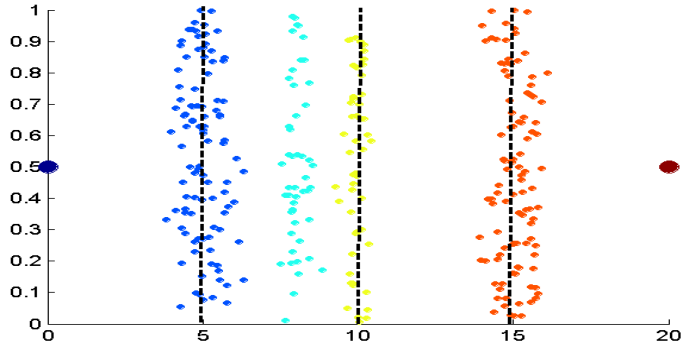
Discretização

- Divisão em intervalos:
  - Partição em intervalos de mesmo tamanho
    - Divide os valores em *n* intervalos de mesmo tamanho
    - Se A é o menor e B o maior valor do atributo, o intervalo é representado por (A-B)/*n*
  - Partição em intervalos de mesma frequência
    - Divide os valores em *n* intervalos com o mesmo número de amostras

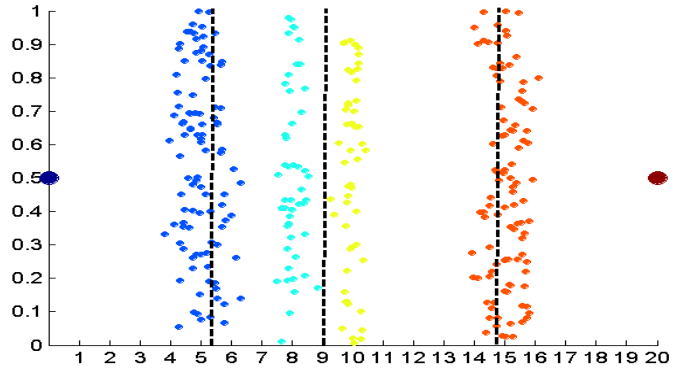
Discretização desconsiderando a classe



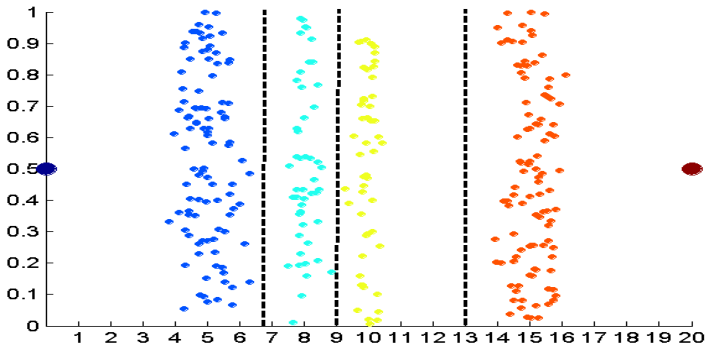
Data



Intervalos iguais

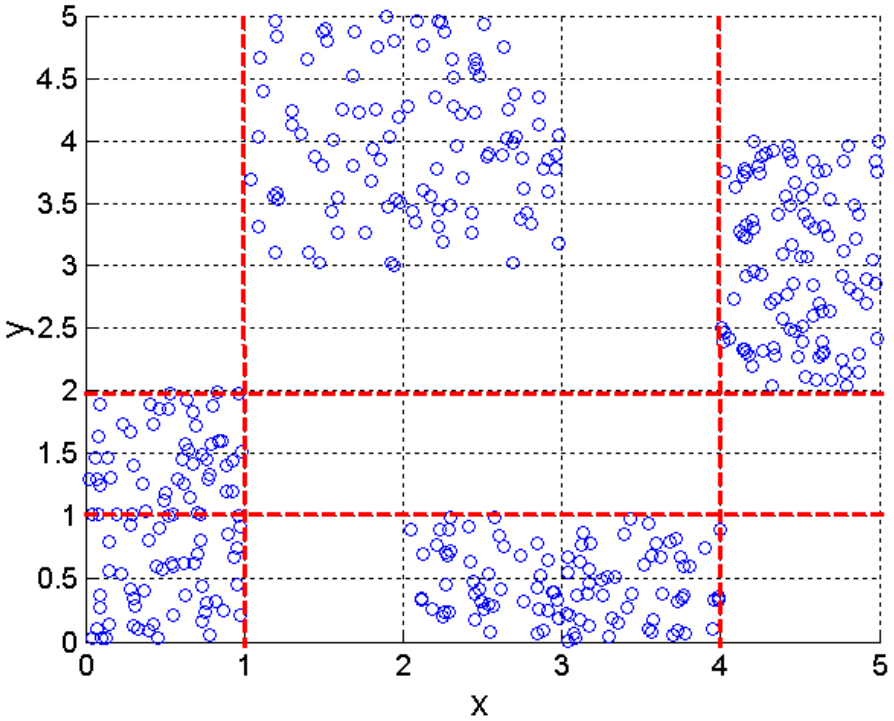


Intervalos com mesma frequência



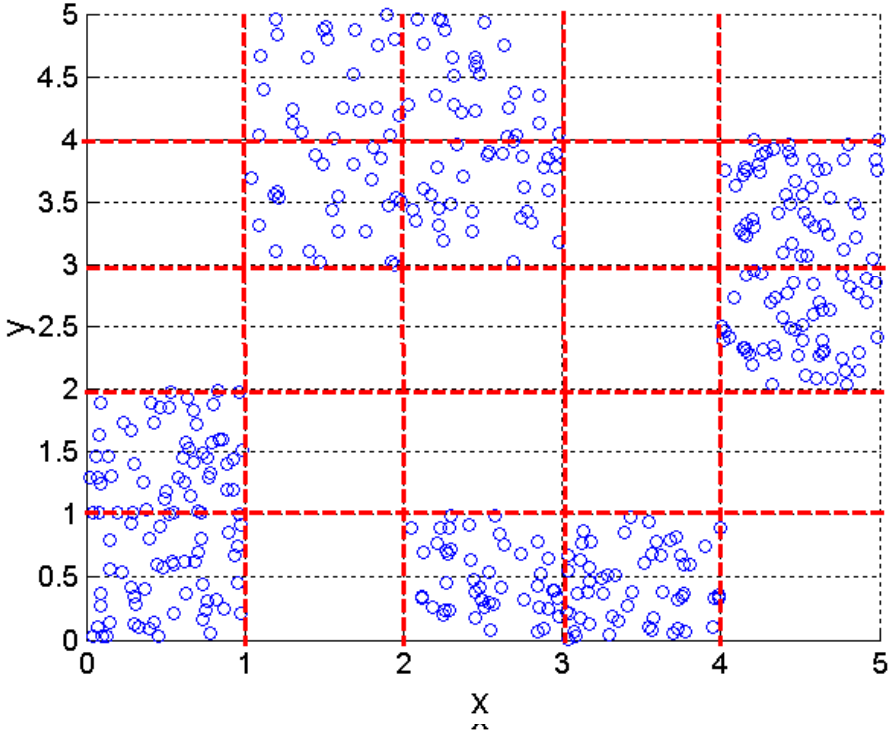
Agrupamento com K-means

Discretização baseada em entropia



3 categories for both x and y

3 intervalos para x e y



5 categories for both x and y

5 intervalos para x e y

Principais tarefas de pré-processamento

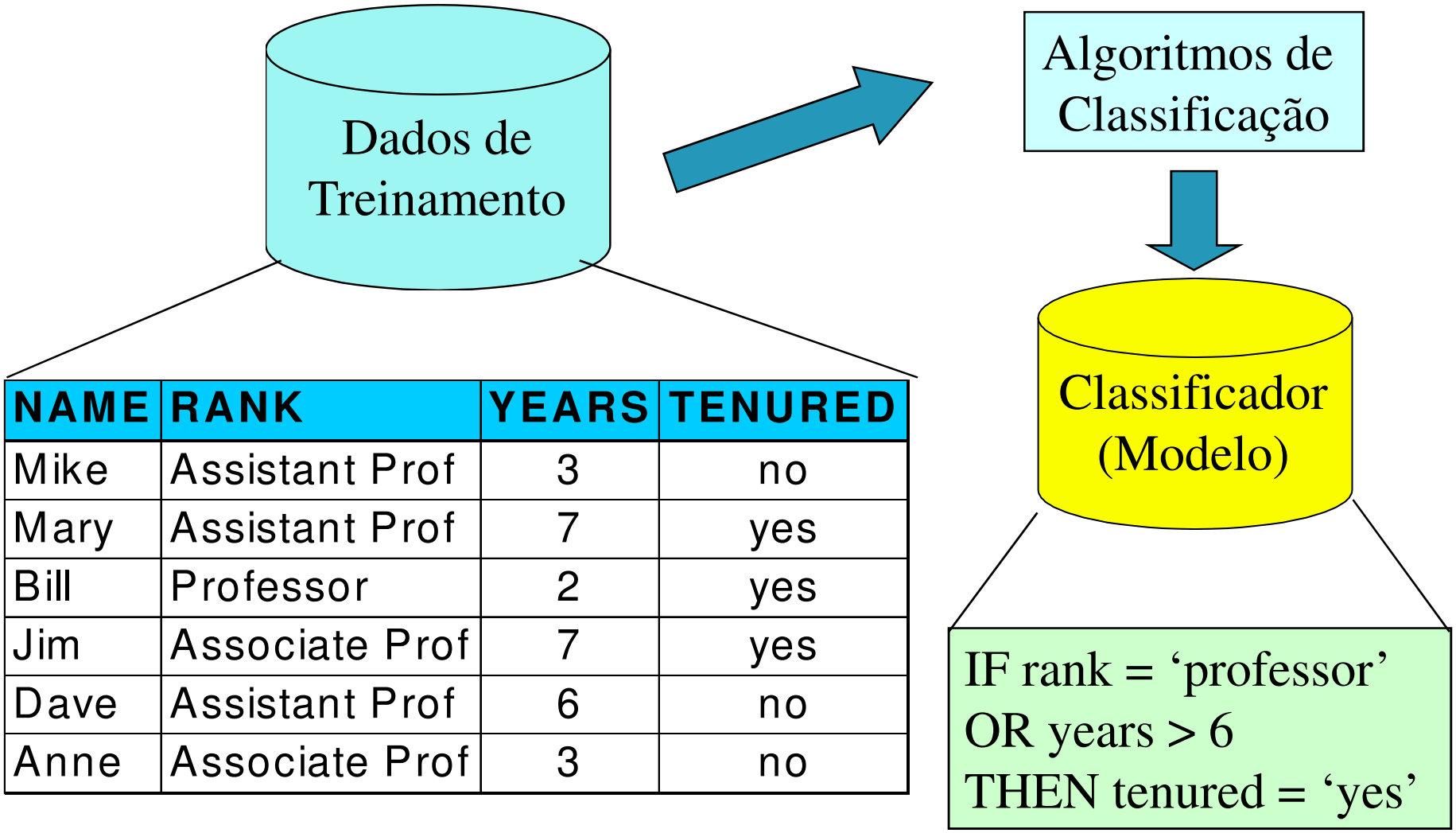
- Limpeza de dados - OK
- Integração de dados - OK
- Redução de dados - OK
- Transformação e discretização de dados - OK

Mineração de Dados:
Foco em Classificação

Classificação

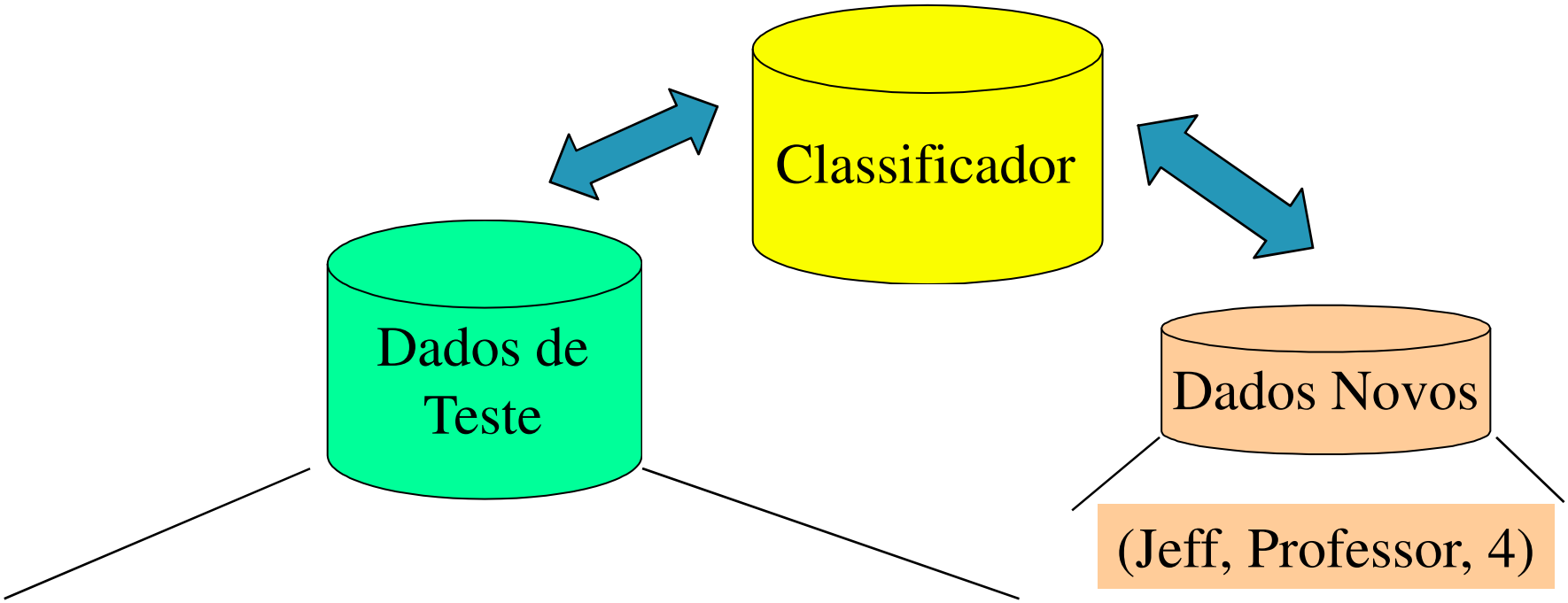
- Conjunto de exemplos cuja classe é conhecida
- Induz um modelo a partir dos exemplos de treinamento
  - Modelo define como o conhecimento será representado
- Testa o modelo em um conjunto de teste, diferente do conjunto de treinamento
- 2 fases: treinamento e teste

Fase 1:Treinamento



NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no

Fase 2: Teste



NAME	RANK	YEARS	TENURED
Tom	Assistant Prof	2	no
Merlisa	Associate Prof	7	no
George	Professor	5	yes
Joseph	Assistant Prof	7	yes

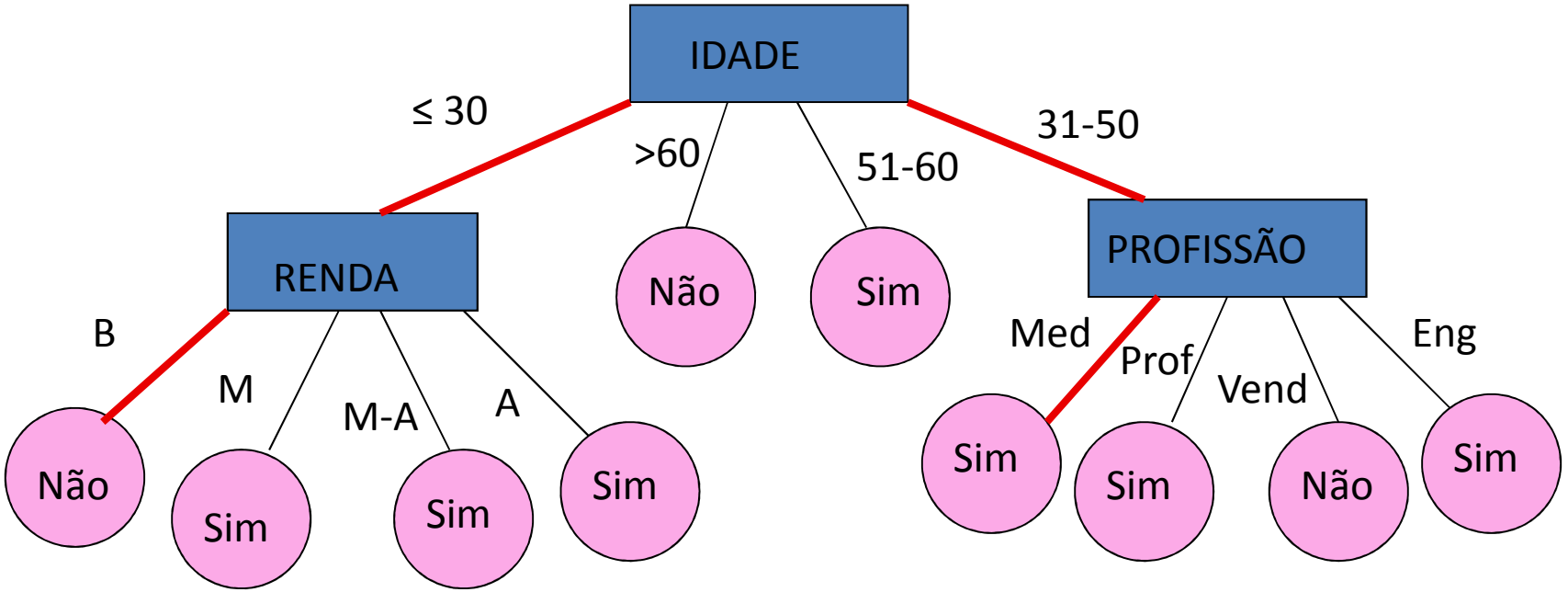
Tipos de Modelos de Classificação

- Modelos de conhecimento compreensível
  - Regras de decisão
  - Árvore de decisão
  - Redes Bayesianas
- Modelos "caixa preta"
  - SVMs *(Support Vector Machines)*
  - Redes neurais
  - KNN
- Algoritmos evolucionários podem ser usados para gerar modelos dos 2 tipos

Árvore de Decisão

- Baseado no conjunto de treinamento, encontra os atributos que trazem maior ganho de informação (redução de entropia)
- Cria um nó de decisão utilizando esse atributo
- Repete esse processo recursivamente, até que divisões não sejam mais possíveis, e os nós folhas da árvore representem classes
- Ex: cliente compra ou não um eletrônico?

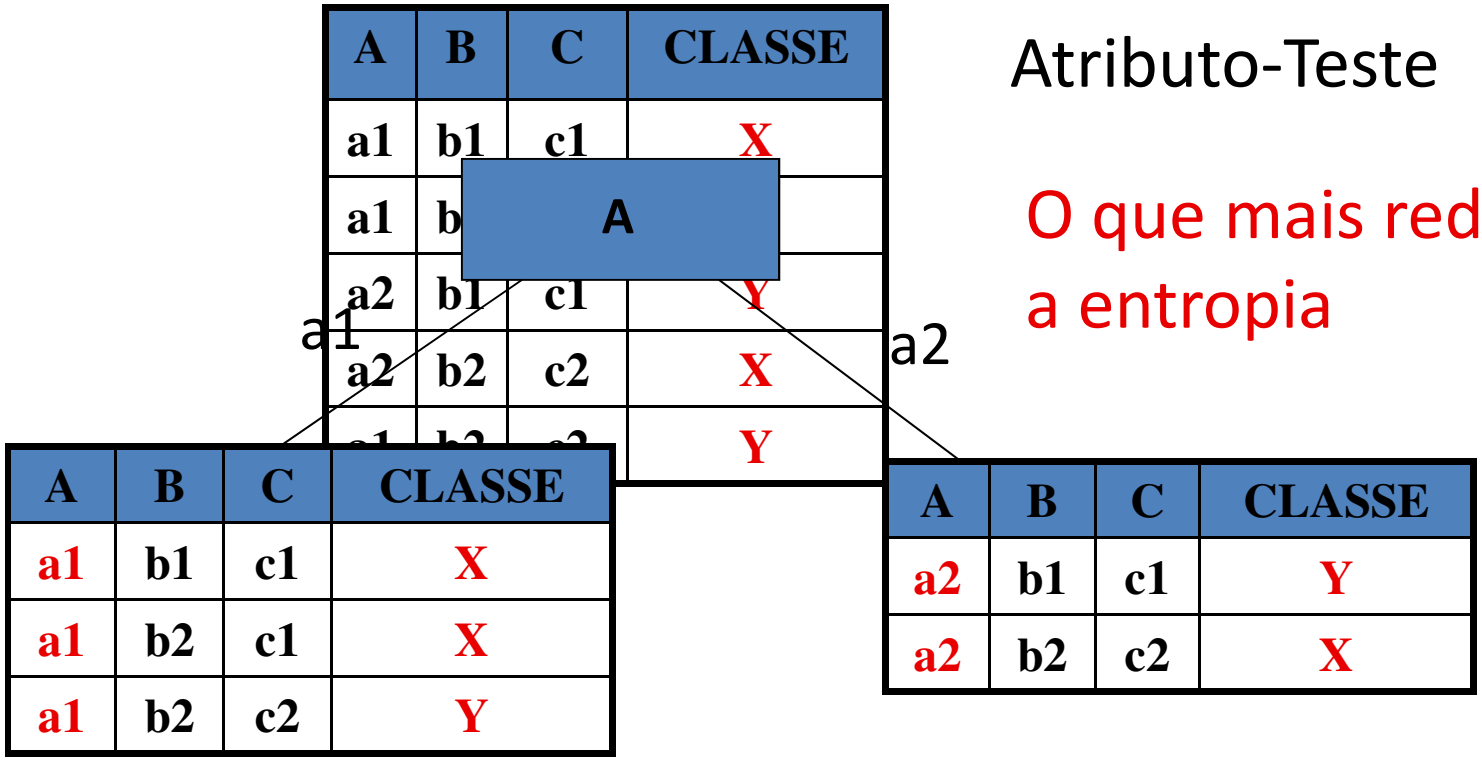
Árvore de Decisão



Se Idade ≤ 30 e Renda é Baixa então **Não compra Eletrônico**

Se Idade = 31-50 e Prof é Médico então **compra Eletrônico**

Como criar uma Árvore de Decisão



A	B	C	CLASSE
a1	b1	c1	X
a1	b		
a2	b1	c1	Y
a2	b2	c2	X
			Y

A	B	C	CLASSE
a1	b1	c1	X
a1	b2	c1	X
a1	b2	c2	Y

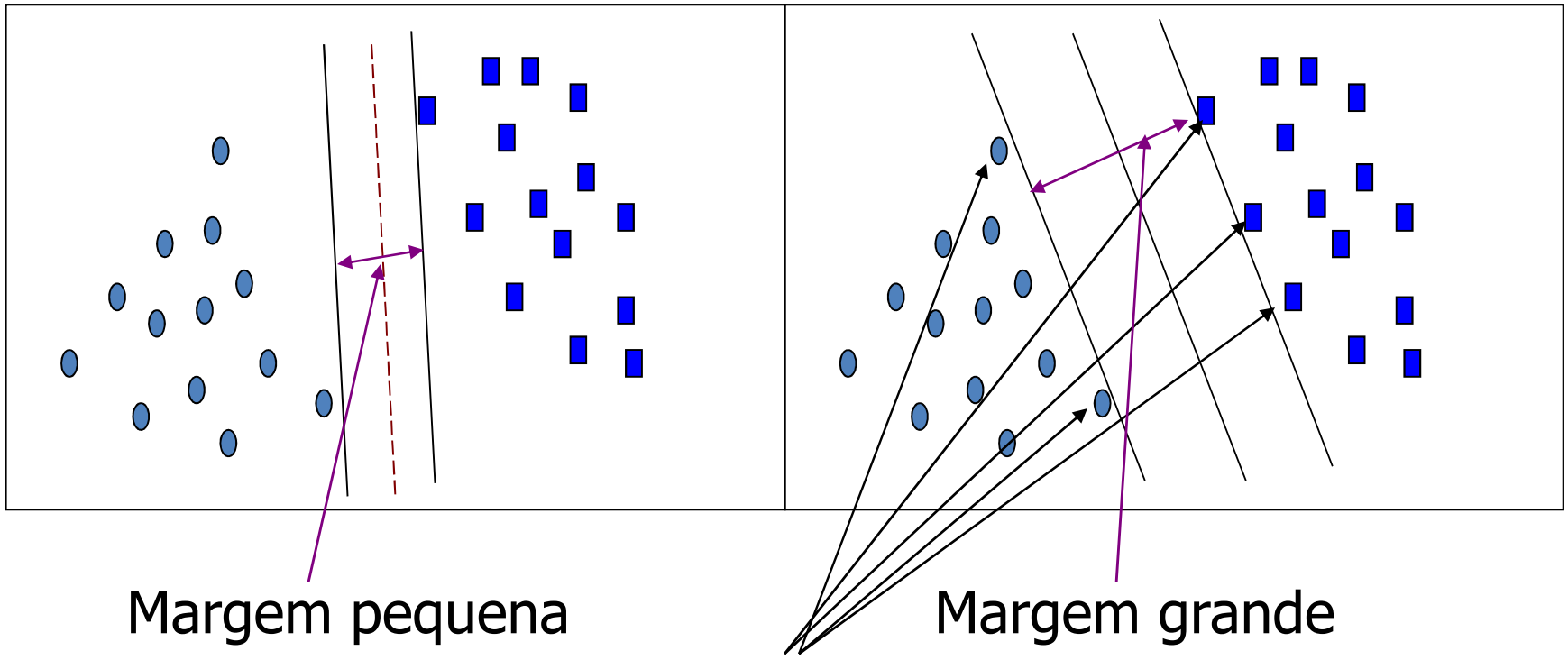
A	B	C	CLASSE
a2	b1	c1	Y
a2	b2	c2	X

LISTA-ATRIBUTOS = { **A, B, C** }

SVM (*Support Vector Machine*)

- A tarefa de classificação consiste em encontrar a melhor forma de separar classes distintas.
- SVM encontra os melhores pontos de separação (vetores de suporte) em um hiper-plano e constrói classificadores sobre eles
- SVMs podem ser lineares ou não-lineares

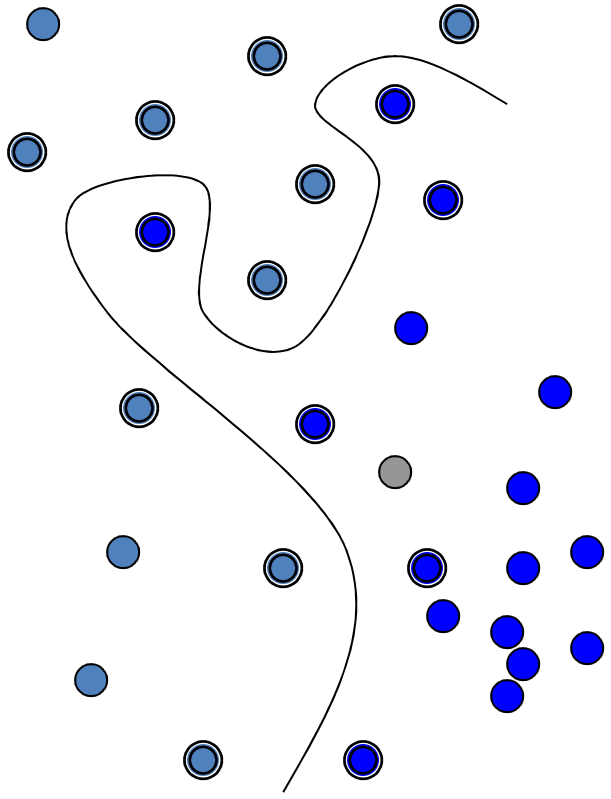
SVM – Support Vector Machines



Exemplo de SVM

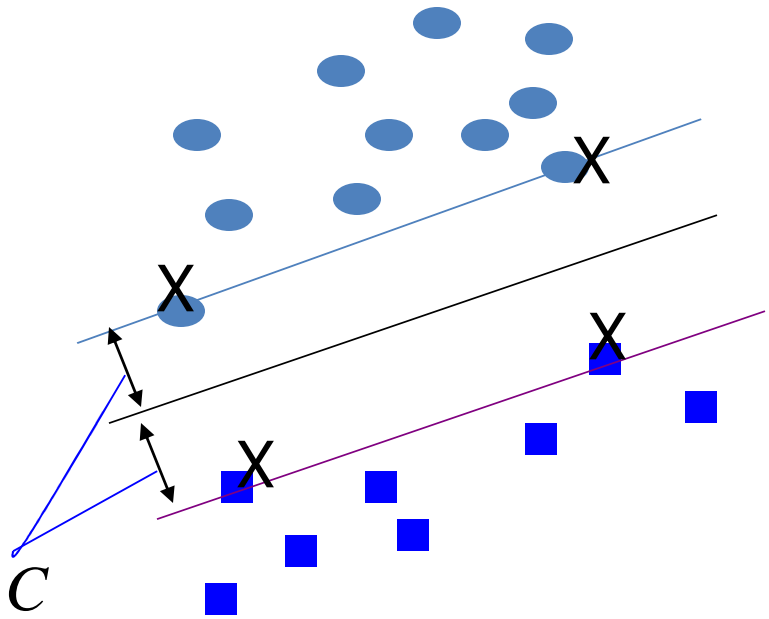
- Os pontos circulados são os vetores de suporte, ou seja, os melhores pontos para representar uma “borda de separação” entre as classes .

  A curva representa essa borda de separação.



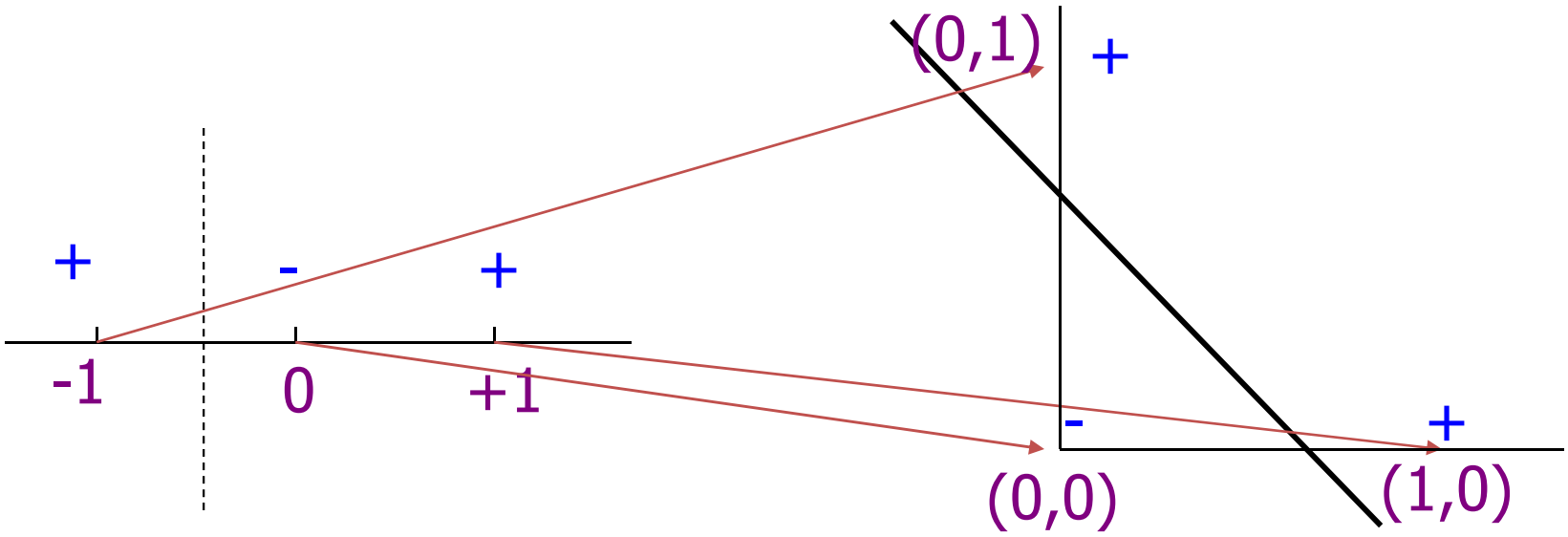
Classes linearmente separáveis

- Classes 1 e 2 são separáveis.
- Os pontos são escolhidos de forma a maximizar a margem.
- Os pontos marcados com um X são os vetores de suporte.



Classes não-linearmente separáveis

- Projeto os dados em um novo espaço onde eles sejam linearmente separáveis – faço isso usando o *kernel*



Avaliação dos algoritmos de classificação

Avaliação

- Normalmente é feita utilizando a taxa de acerto ou acurácia
  - Problemas com classes não-balanceadas
- Outras métricas mais pertinentes
  - Sensitividade
  - Especificidade
  - Precisão
  - F1

Acurácia – Taxa de erros

- Acc(M) = porcentagem das tuplas dos dados de teste que sao corretamente classificadas.
- Err(M) = 1 – Acc(M)
- Matriz de Confusão

Classes Preditas

Classes Reais	C1	C2
C1	**Positivos verdadeiros**	**Falsos Negativos**
C2	**Falsos Positivos**	**Negativos verdadeiros**

Classes “não-balanceadas”

Exemplo : acc(M) = 90%

C1 = tem-câncer (4 pacientes)

C2 = não-tem-câncer (500 pacientes)

- Classificou corretamente 454 pacientes que não tem câncer
- Não acertou nenhum dos que tem câncer
- Pode ser considerado como “bom classificador”

mesmo com acurácia alta ?

Medidas para classificadores (classes não-balanceadas)

Sensitividade (*recall*) $= \frac{\text{true-pos}}{\text{pos}}$

% pacientes classificados **corretamente como positivos** dentre todos os que **realmente são positivos**

Especificidade $= \frac{\text{true-neg}}{\text{neg}}$

Precisão $= \frac{\text{true-pos}}{\text{true-pos + falso-pos}}$

% pacientes classificados **corretamente como positivos** dentre todos os que **foram classificados como positivos**

Precisão e Recall : medidas originadas em *Recuperação de Informação* utilizadas em Classificação, quando se lida com "classes não-balanceadas"

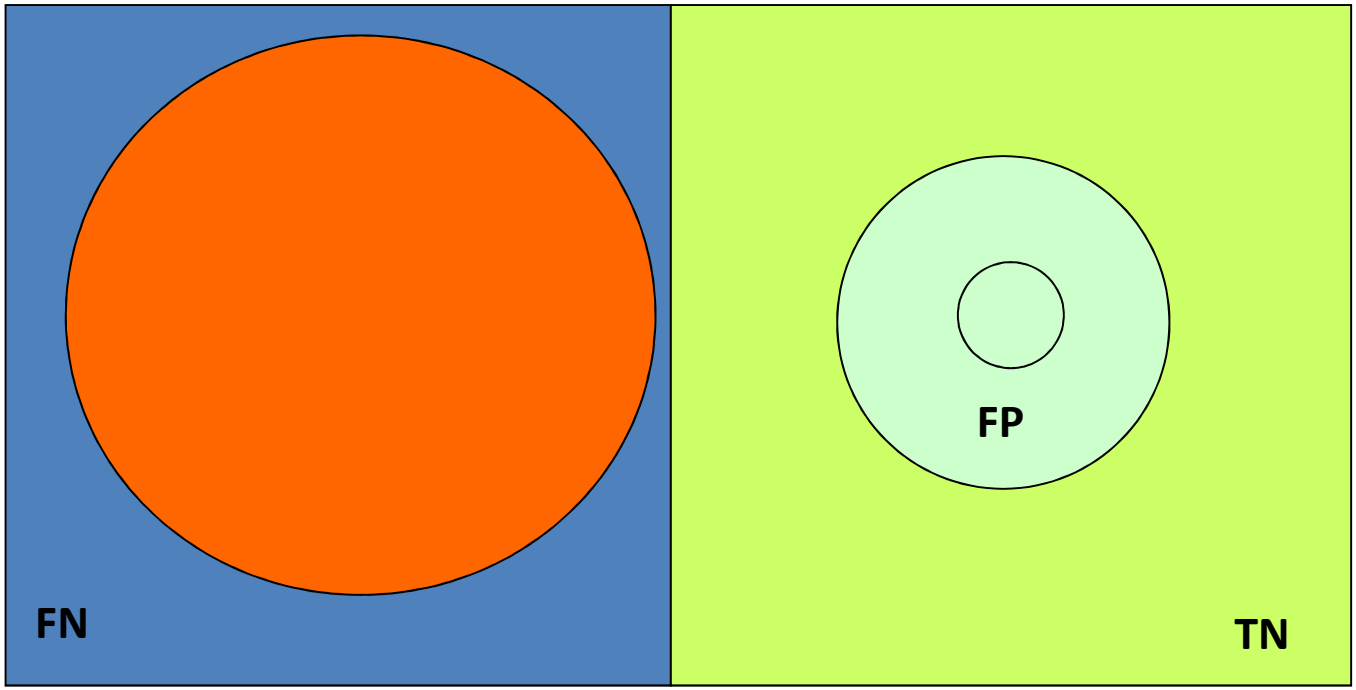
Medida F1 : Média harmônica de Precisão e Recall

- Média harmônica entre dois números x e y tende a ser próxima de min(x,y)

- F1 alto implica que precisão e *recall* são razoavelmente altos.

$$F1 = \frac{2\,rp}{r + p}$$

Trade-off entre TP e FP



POSITIVOS REAIS

NEGATIVOS REAIS

Curva ROC

- ROC = **R**eceiver **O**perating **C**haracteristic Curve
- Ideal para comparar classificadores
- Enfoque gráfico que mostra um *trade-off* entre as taxas de TP (TPR) e FP (FPR) de um classificador.

- TPR = TP/(TP + FN) (= recall)
- FPR = FP/(TN + FP)
- **Ideal : TPR = 1 e FPR = 0**

Como gerar a curva ROC de um classificador ?

- O classificador precisa produzir, para cada instância X, a probabilidade de X ser classificada na classe **Positiva**.
- Classificadores como redes neurais e redes bayesianas produzem tais probabilidades.
- Para outros tipos de classificadores, é preciso calcular esta probabilidade.

Como gerar a curva ROC de um classificador ?

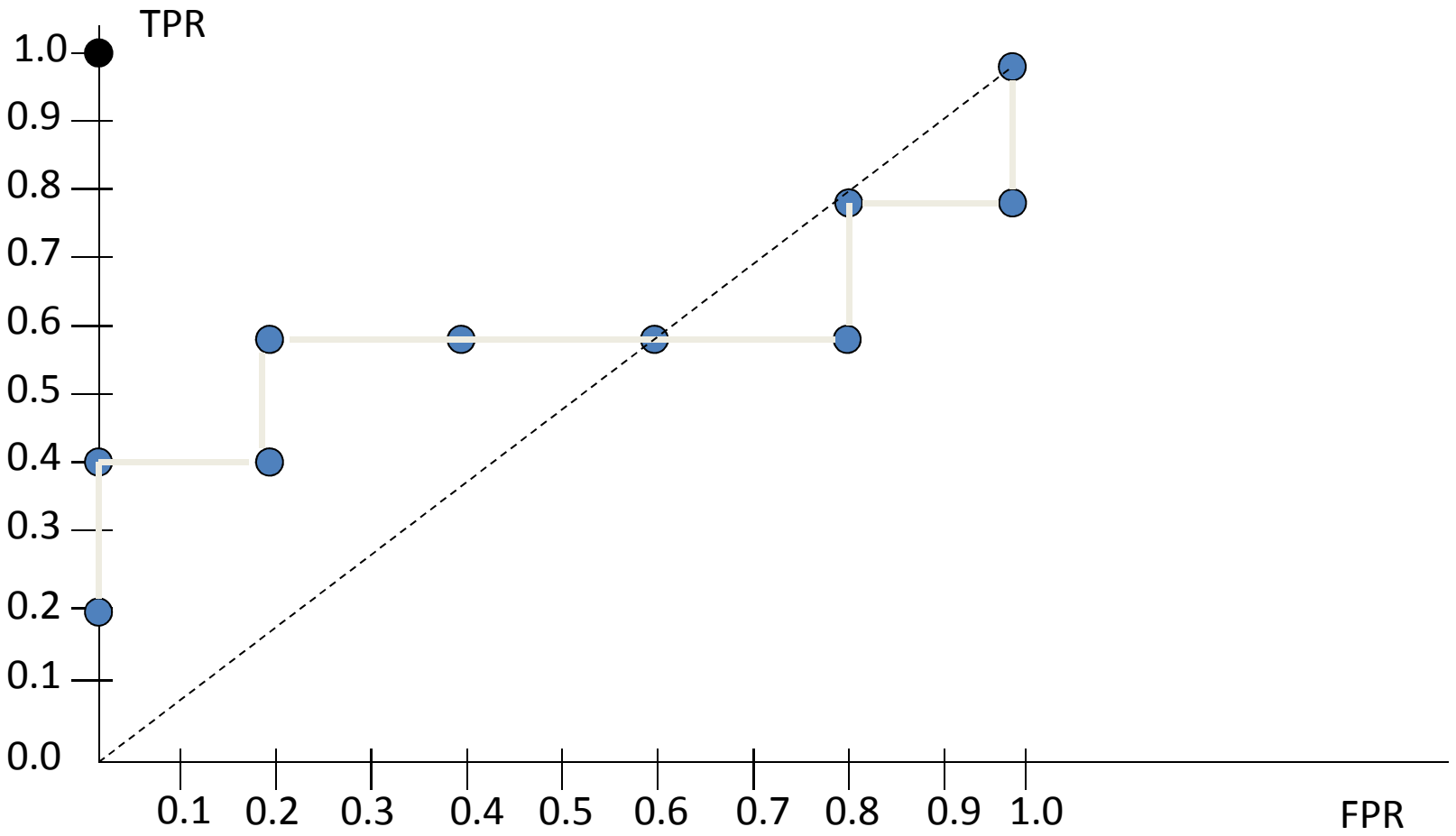
1. Ordene as instâncias por ordem crescente de seus valores de *output* (prob. de estar na classe positiva)
2. Selecione a primeira instância X1 e
   - Classifique X1 como **POSITIVA**
   - Classifique todas as instâncias com *outputs* maiores do X1 como **POSITIVAS**

3. Selecione a segunda instância X2 e
   - Classifique X2 como **POSITIVA**
   - Classifique todas as instâncias com outputs maiores do X2 como **POSITIVAS** e as com outputs menores como **NEGATIVAS**
   - Calcule os novos valores de TP e FP
     - **Se a classe de X1 é positiva então TP é decrementado de 1 e FP continua o mesmo**
     - **Se a classe de X1 é negativa então TP continua o mesmo e FP é decrementado.**
4. Repita o processo para a terceira instância a até varrer todo o conjunto de treinamento
5. Faça o gráfico dos valores de TPR (eixo y) por FPR (eixo x)

Exemplo

Classe	+	-	+	-	-	-	+	-	+	+	
	0.25	**0.43**	**0.53**	**0.76**	**0.85**	**0.85**	**0.85**	**0.87**	**0.93**	**0.95**	**1.00**
TP	5	4	4	3	3	3	3	2	2	1	0
FP	5	5	4	4	3	2	1	1	0	0	0
TN	0	0	1	1	2	3	4	4	5	5	5
FN	0	1	1	2	2	2	2	3	3	4	5
TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0

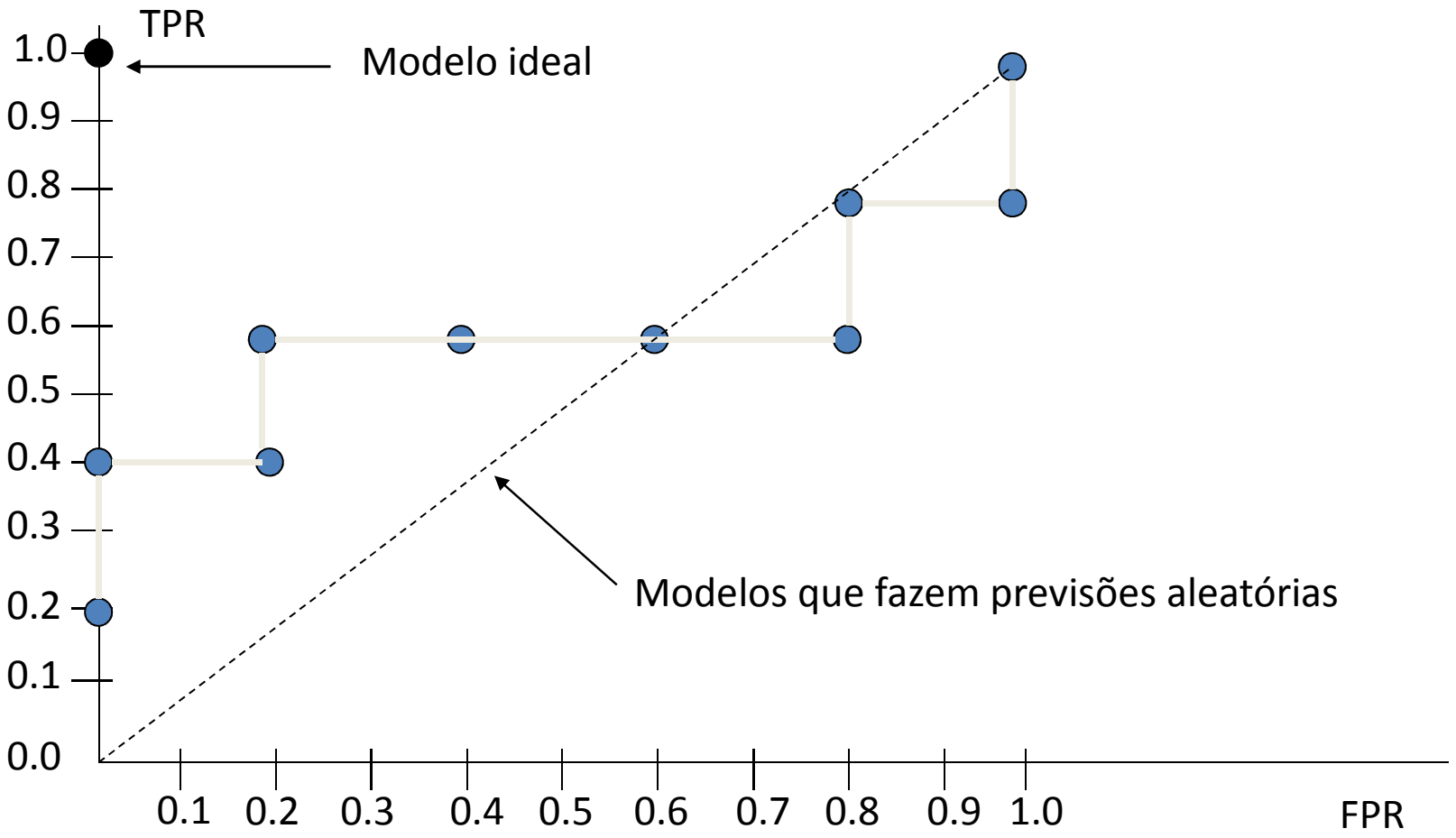
Exemplo



Curva Roc

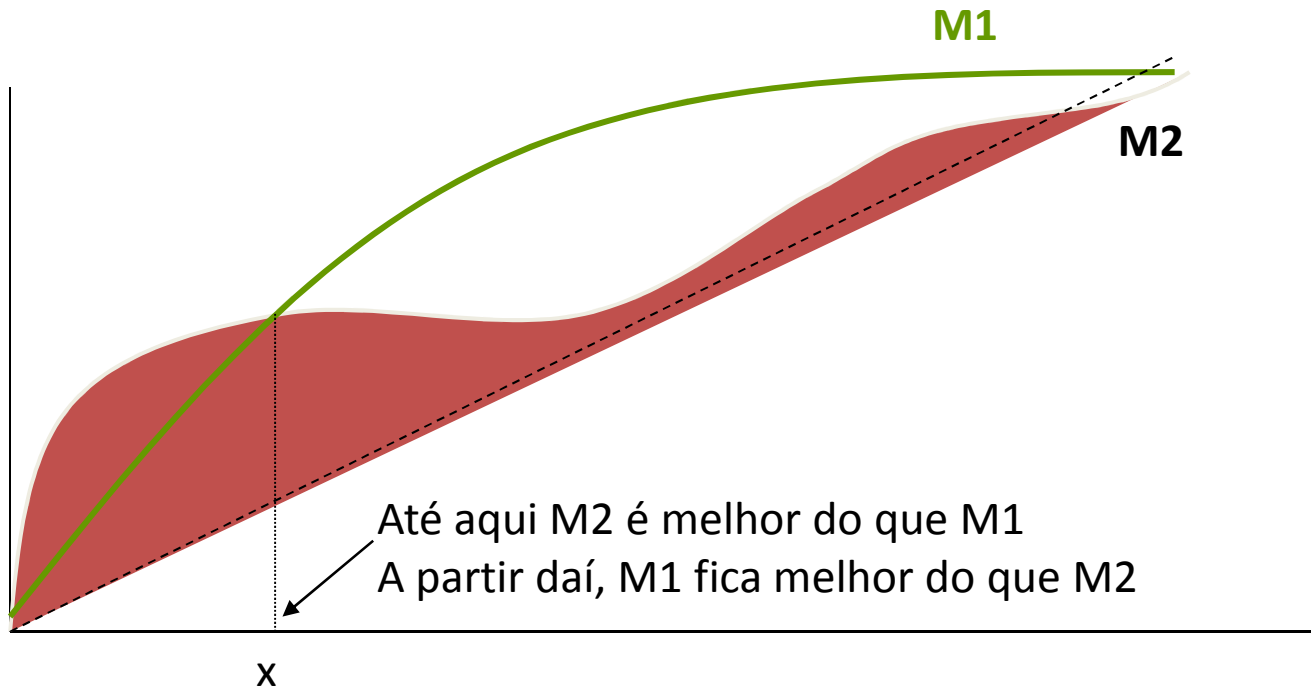
- Cada ponto na curva corresponde a um dos modelos induzidos pelo classificador
- Um bom modelo deve estar localizado próximo do ponto (0,1)
- Modelos localizados na diagonal são modelos aleatórios – TPR = FPR
- Modelos localizados acima da diagonal são melhores do que modelos abaixo da diagonal.

Exemplo


TPR
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
Modelo ideal
Modelos que fazem previsões aleatórias
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
FPR

Comparando performance relativas de diferentes classificadores

- Curvas Roc são utilizadas para se medir a performance relativa de diferentes classificadores.



Area abaixo da curva ROC (AUC)

- A área abaixo da curva ROC fornece medida para comparar performances de classificadores.
- Quanto maior a área AUC melhor a performance global do classificador.
- Classificador optimal: área =1
- Classificador randômico : área = 0.5

Referências

- Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, 2nd ed., Morgan Kaufmann Publishers, 2006.
- Ian H. Witten, Eibe Frank , Data Mining: Practical Machine Learning Tools and Techniques, 2nd ed., Morgan Kaufmann, 2005.
- Tom Fawcett, ROC Graphs: Notes and Practical Considerations for Researchers , Relatório Técnico, 2004.

Agradecimentos

- Alguns desses slides foram baseados nas notas de aula de J. Han, Chris Clifton, e S. de Amo