



Introdução ao Reconhecimento

Prof. Dr. Geraldo Braz Junior

O que você vê?



Pergunta:

- Essa imagem tem um prédio?
- **Classificação**



Pergunta:

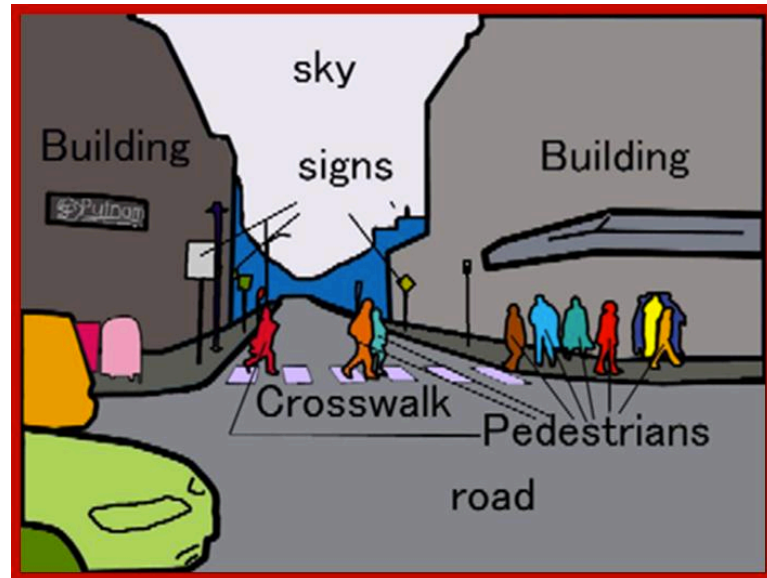
- Essa imagem possui carro(s)?
- **Detecção de Objetos**

Vários



Pergunta:

- O que essa cena apresenta?
- **Segmentação** (em alguns casos, semântica) – localização precisa



Dilema

- Reconhecer por Categorização
 - Reconhecimento de grupos de objetos que compartilham características
- Ou reconhecer por instância
 - Reconhecimento de um objeto específico

Dilema

- Supervisionado
 - Existe uma base de conhecimento anterior
- Não supervisionado
 - Não existe base de conhecimento anterior
- Modelo ou função
 - Classificação x Regressão

Variação intra-grupo



Aplicações?

- Tecnologias assistivas (direção, deficiências...)
- Monitoramento e segurança
- Tratamento de Doenças
- Turismo

- Você tem uma?

Conceitos Básicos

Um Exemplo

- Para ilustrar a complexidade desse tipo de sistema, considere o seguinte exemplo:
 - Uma indústria recebe dois tipos de peixe: Salmão e Robalo. Os peixes são recebidos em uma esteira, e o processo de classificação é manual.
 - A indústria gostaria de automatizar esse processo, usando para isso uma câmera CCD.

Um Exemplo (cont)



Um Exemplo (cont)

- Primeiramente devemos encontrar as **características** que distinguem um salmão de um robalo.
 - Altura, largura, coloração, posição da boca, etc...
- Características (**Features**): Qualquer medida que se possa extrair de um determinado objeto.

Um Exemplo (cont)

- Características podem ser:
 - Simbólicas.
 - Numéricas (Contínuas ou binárias).
- **Ruídos** (Erros):
 - Conceito originário da teoria das comunicações;
 - Podem estar presente no objeto.
 - Uma mancha no peixe, por exemplo.
 - Devidos ao sistema de aquisição.
 - Iluminação por exemplo.

Um Exemplo (cont)

- Dado as diferenças entre as populações de Salmão e Robalo, podemos dizer que cada uma possui um **modelo** específico.
- Modelo: Um descritor, geralmente representado através de uma função matemática.
 - Bom modelo é capaz de absorver ruídos.

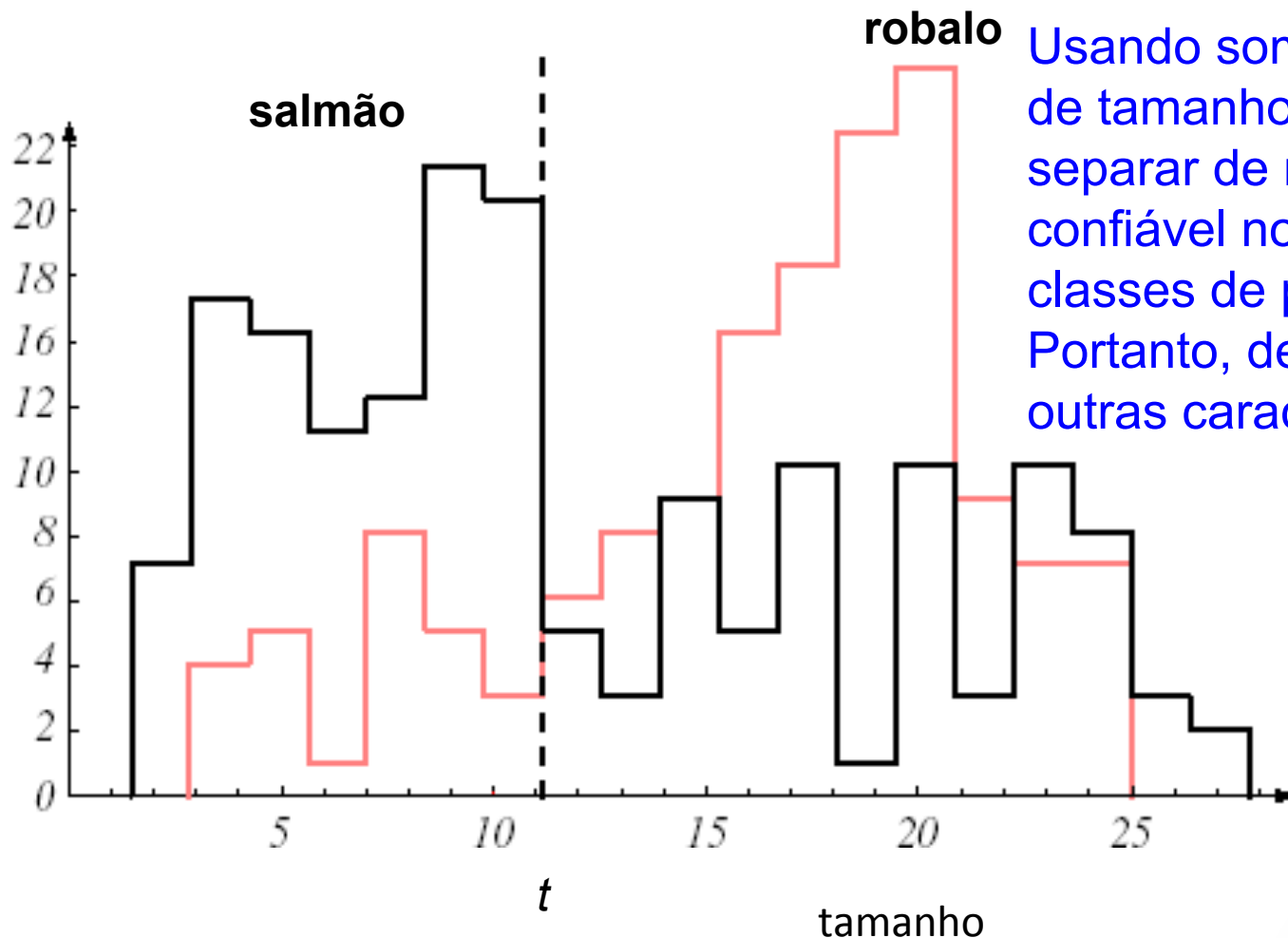
Conceitos básicos de classificação

- Suponha que alguém nos diga que:
 - **Robalos *geralmente* são maiores que salmões.**
 - Isso nos dá uma direção para modelar nosso problema, ou seja,
 - Se o peixe ultrapassa um tamanho t , então ele é um robalo, caso contrário, é um salmão.
 - Mas como determinar t ??

Conceitos básicos de classificação

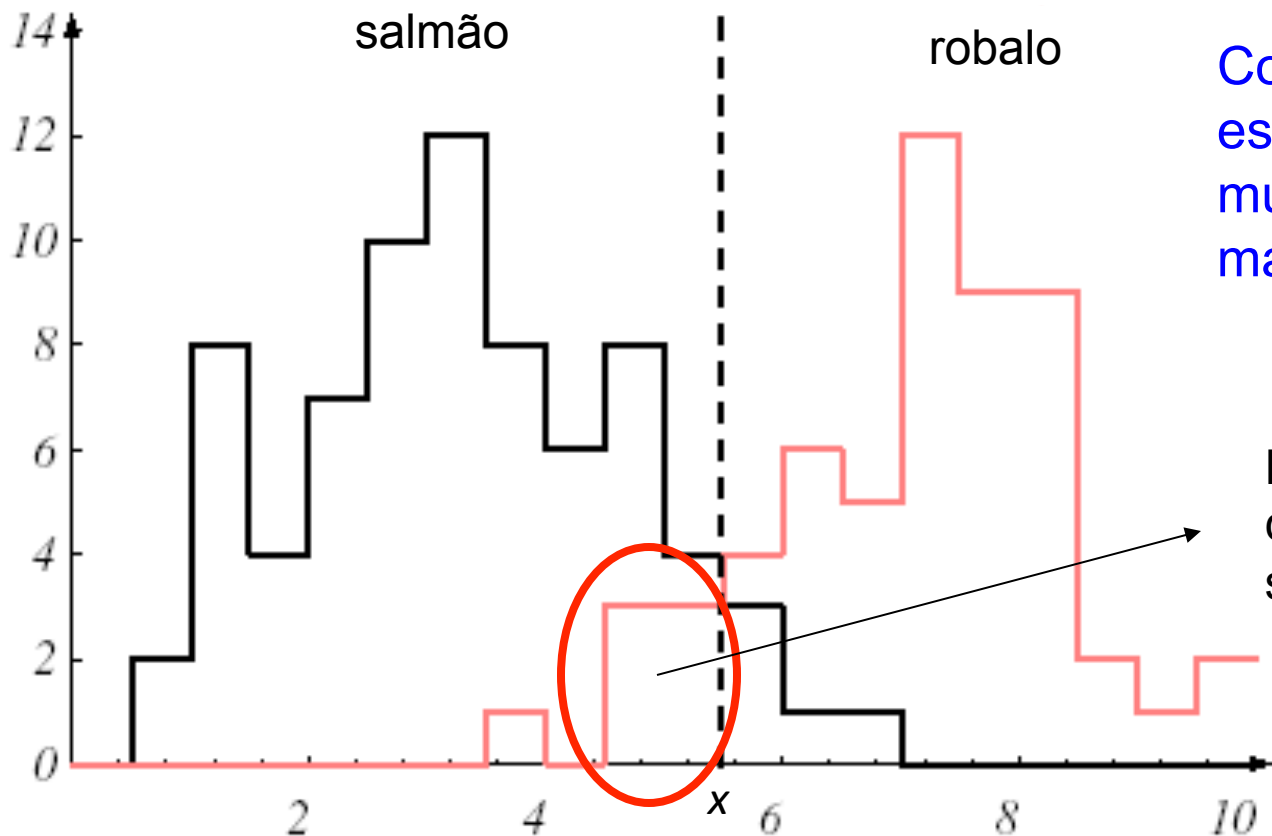
- Podemos selecionar alguns exemplares (**base de treinamento**) de peixe e verificar seus tamanhos.
- Suponha que após analisarmos nossa base de treinamento, tenhamos os seguintes histogramas

Tamanho é suficiente?



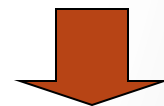
Usando somente a medida de tamanho, não podemos separar de maneira confiável nossas duas classes de peixes. Portanto, devemos tentar outras características.

Existe outra melhor? Quanto?



Como podemos notar, essa característica é muito mais confiável, mas não é perfeita.

Robalos que serão classificados como salmão.



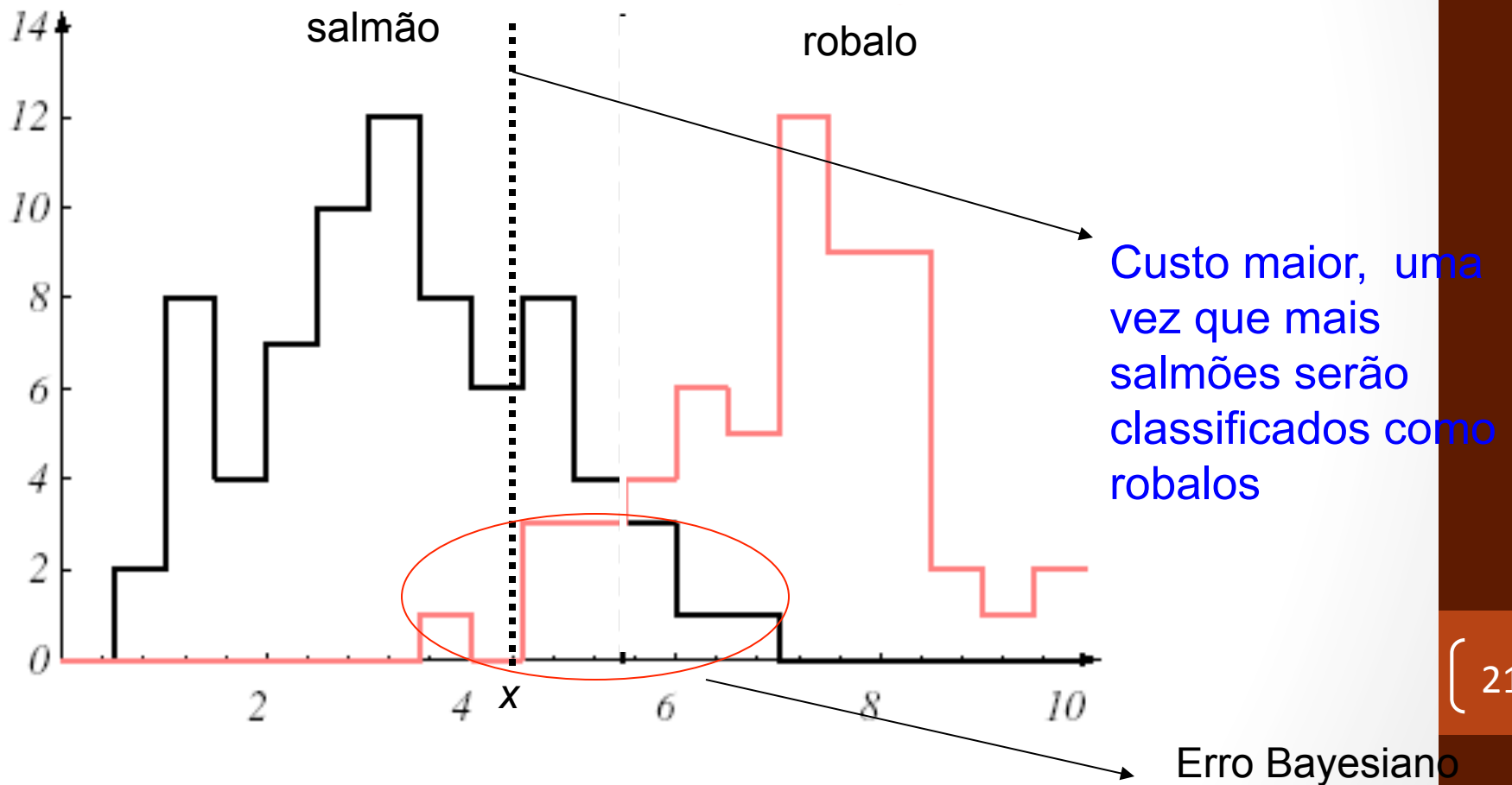
Noção de **CUSTO**

Outra característica: Medida de Claridade (coloração)

Conceitos básicos de classificação

- Suponha que os clientes da nossa indústria aceitem um pedaço de salmão embalado junto com robalo, mas o contrário é inaceitável.
- Devemos então alterar nossa fronteira para que isso não aconteça.

Custo e Erro



Teoria da Decisão

- Isso sugere que existe um **custo** associado com a nossa decisão.
- Nossa tarefa consiste em encontrar uma regra de decisão que minimize o custo.
 - Isso é o papel central da **Teoria da Decisão**.
 - Também pode ser visto como um problema de otimização.

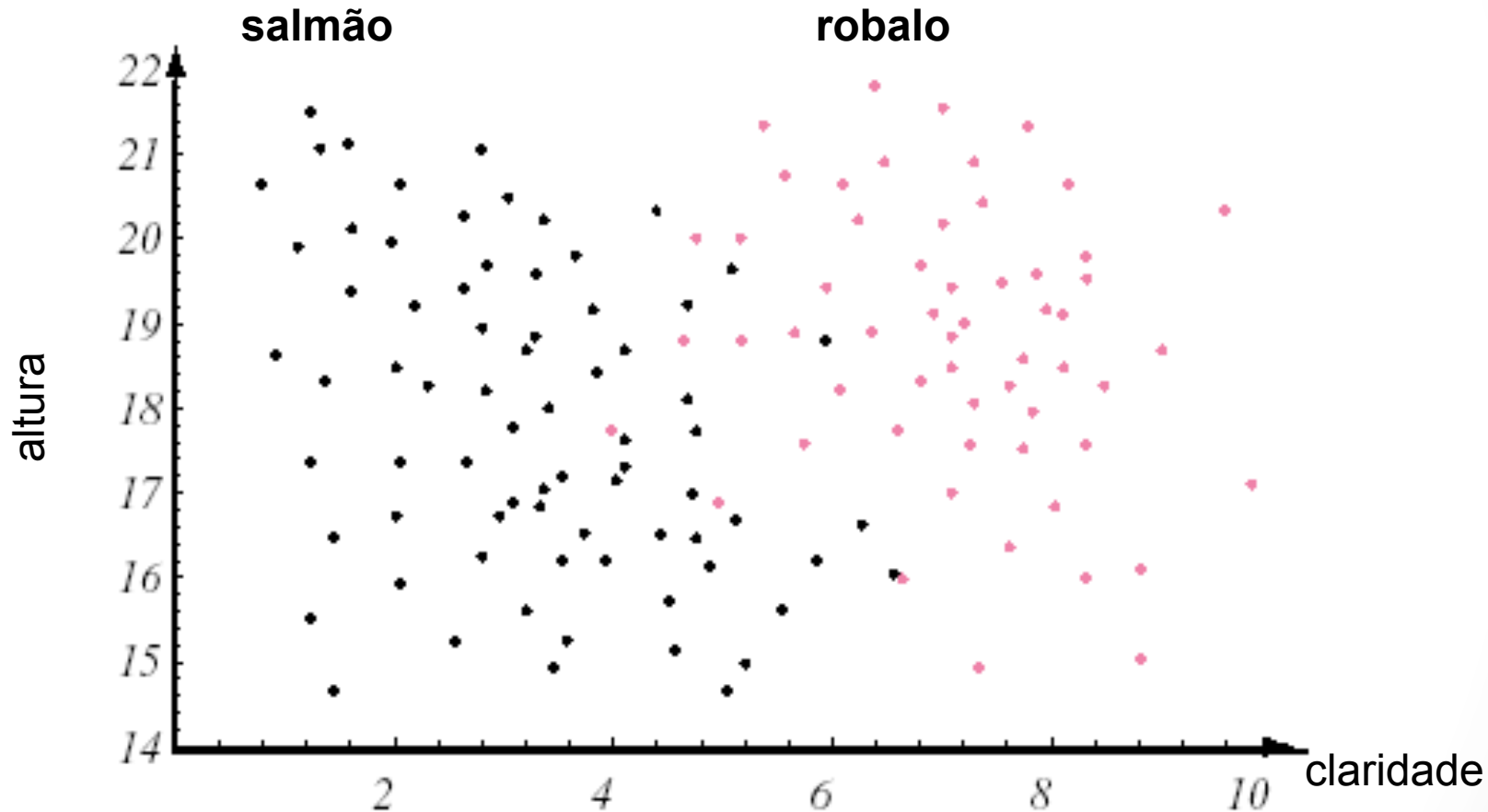
Como minimizar o custo?

- Como minimizar o custo?
 - Buscar outras características.
 - Suponha que a claridade seja a melhor.
 - Podemos utilizar duas características ao mesmo tempo.

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Vetor de características bi-dimensional.

Duas características

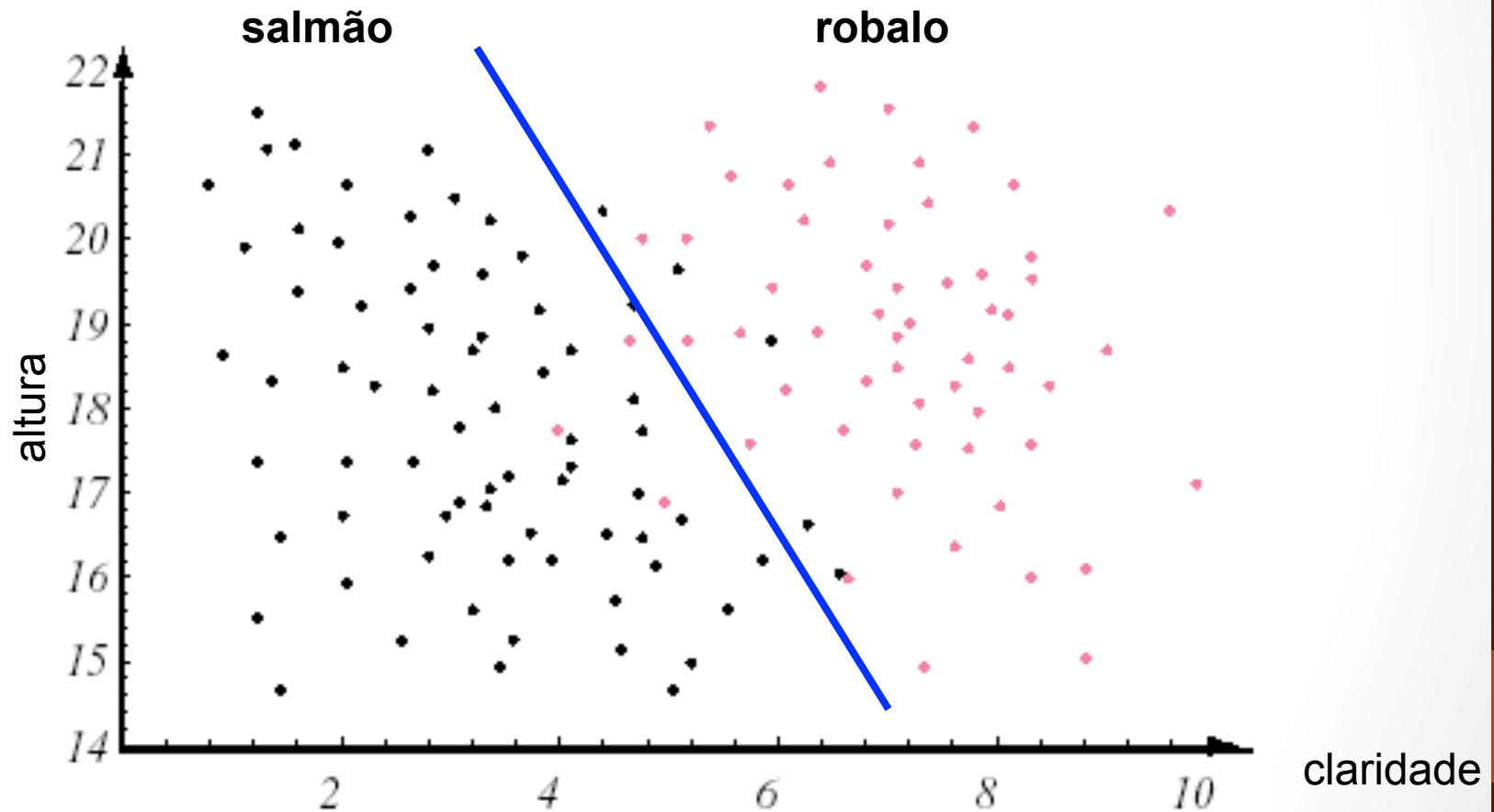


Nossa base de treinamento em um gráfico de dispersão

Fronteira de Decisão

- Nosso problema consiste em encontrar a **fronteira de decisão** que minimize o custo.
- Modelo mais simples:
 - Separação linear, **$y = ax+b$**
 - Algoritmos tradicionais:
 - KNN, Perceptron (Redes Neurais), Funções Discriminantes Lineares (LDA).

Exemplo de fronteira linear



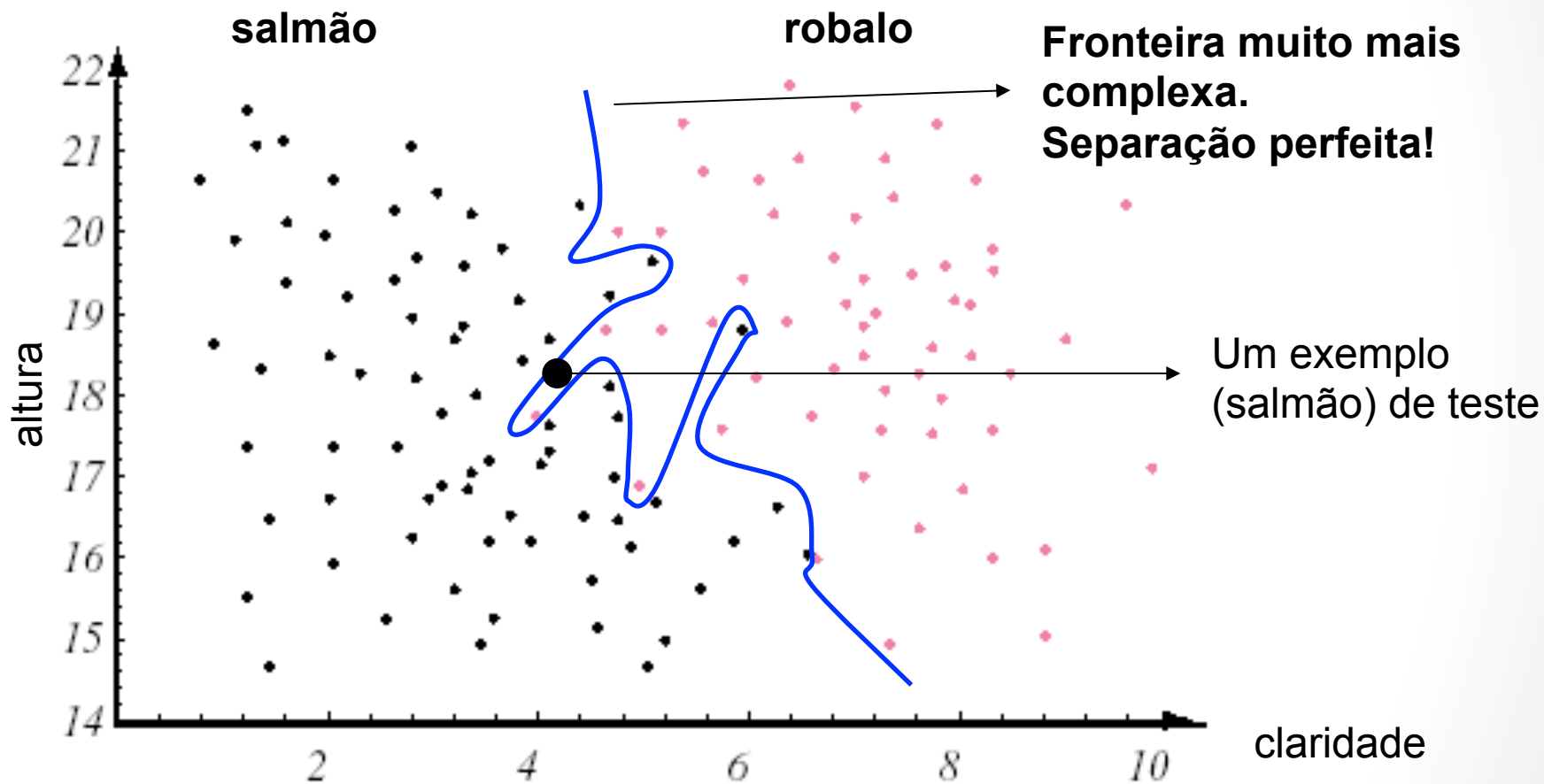
Podemos adicionar mais?

- Essa regra fornece uma boa fronteira de decisão, entretanto pode ser melhorada.
 - Adicionar mais características, como por exemplo, dimensões do peixe, posição relativa dos olhos, etc..., poderiam ajudar.
- Quanto mais características, mais base de treinamento será necessária.
 - **Maldição da dimensionalidade.**

Podemos aumentar a complexidade

- Suponha que não dispomos de mais características.
- Solução:
 - **Construir um modelo mais complexo do que um modelo linear.**

Nova fronteira

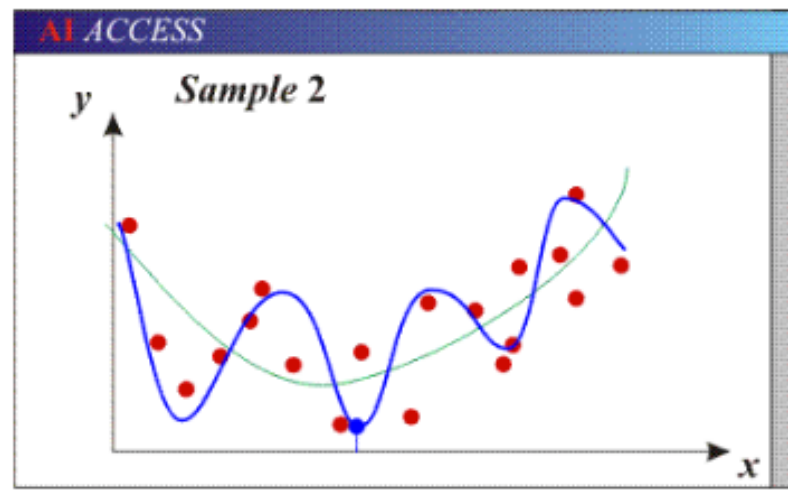
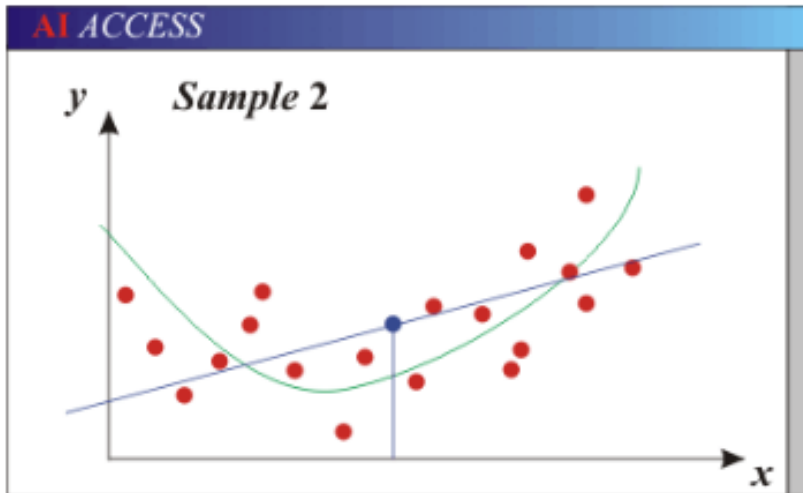


Generalizar

- Apesar de complexo, esse modelo é pior do que o modelo linear.
- A principal característica de um modelo deve ser a sua capacidade de **generalizar**.
 - **BIAS**: quanto o modelo médio de todos os conjuntos de treinamento difere do modelo verdadeiro?
 - **Variância**: o quanto os modelos de treinamento são dispersos entre si?

BIAS x Variância

- Muita BIAS: pouca informação, pouca flexibilidade
- Muita Variância: muita sensibilidade



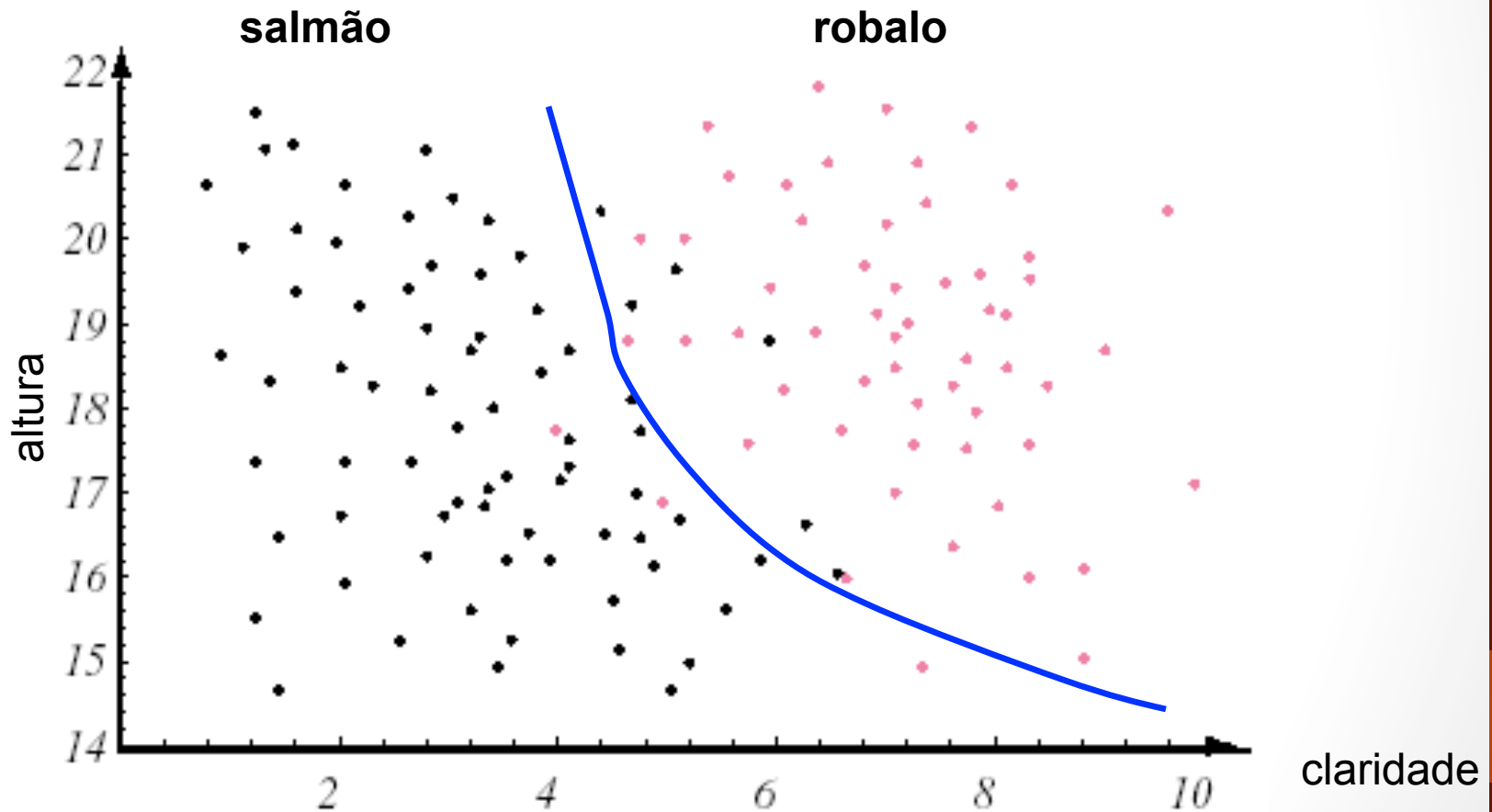
Over-Fitting

- Modelos muito complexos geralmente não generalizam bem, pois “decoram” a base de treinamento (**over-fitting**).
 - Baixa BIAS e alta Variância
 - Baixo erro de treinamento
 - Alto erro de teste

Underfitting

- Por outro lado, modelos muito simples tendem a generalizar demais (**underfitting**).
 - Alto BIAS e baixa variancia
 - Alto erro de treinamento e teste

Conceitos básicos de classificação



Etapas basicas

Etapas Básicas



E depois? Etapas:



Extração de
Características



Modelos
“Aprendidos”



Predição /
Classificação



Pontos a decidir no pipeline

- Quais características usar?
- Qual técnica de aprendizado de máquina usar?

Quais características usar?

- Analise dentro do seu espaço de objetos:
 - Cor? através de histograma
 - Medidas globais de forma? circularidade, especificidade, ...
 - Medidas locais de forma? curvatura, HOG, ...
 - Textura? distribuição estatística e direcional, ...
- Qual usar?

Qual técnica de aprendizado?

- Simples:
 - Dados facilmente separáveis? Alguma técnica simples e linear
 - Dados não são facilmente separáveis ou esse fator é desconhecido: utilize técnicas robustas
- Técnicas robustas = mais tempo de processamento

KNN

Uma técnica simples

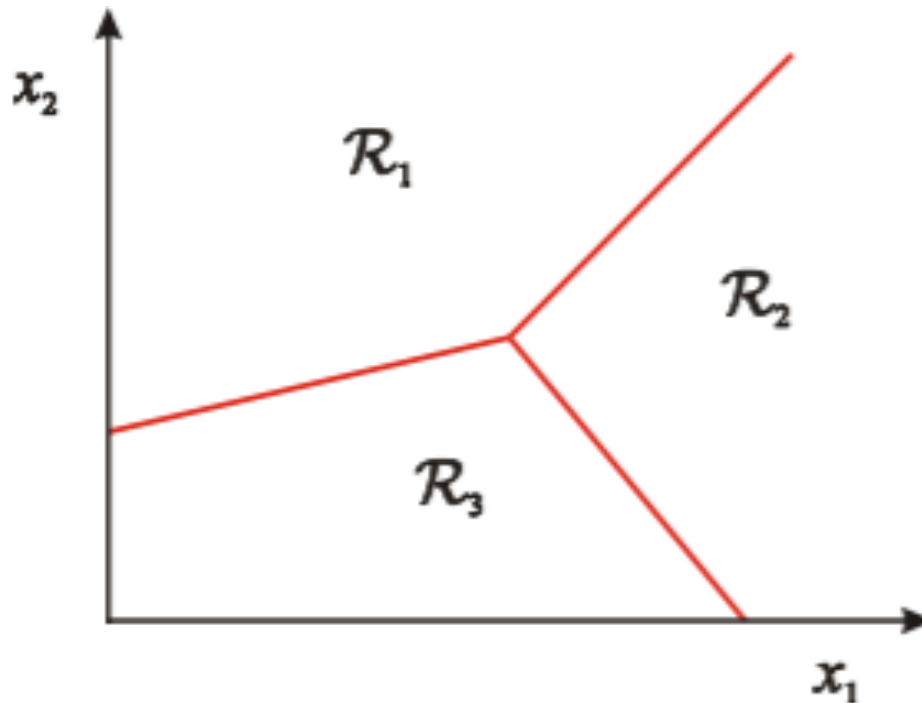
k-nearest neighbor

- Dado uma coleção de objetos conhecidos (com seus labels e características) para formar a função $F \rightarrow$ **base de treinamento**
- Utilize F para classificar um exemplo x de novo objeto \rightarrow **teste**

$$y = f(\mathbf{x})$$

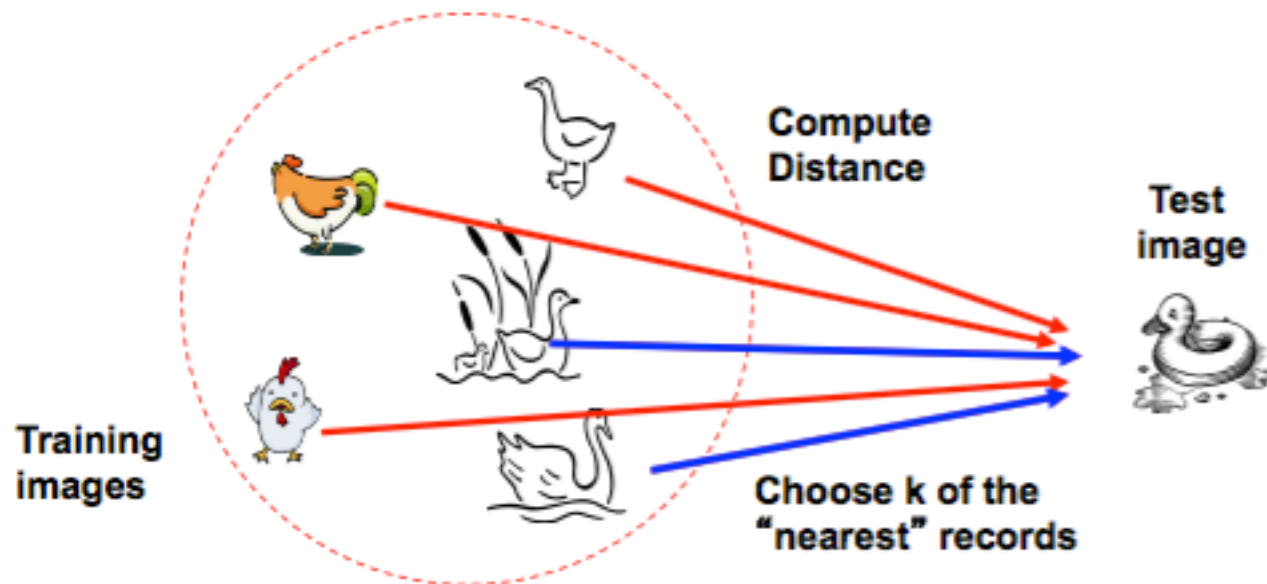
Classificação

- Associe o elemento de entrada a 1 ou mais classes
- Cada decisão divide o espaço de entrada em uma ou mais fronteiras



KNN

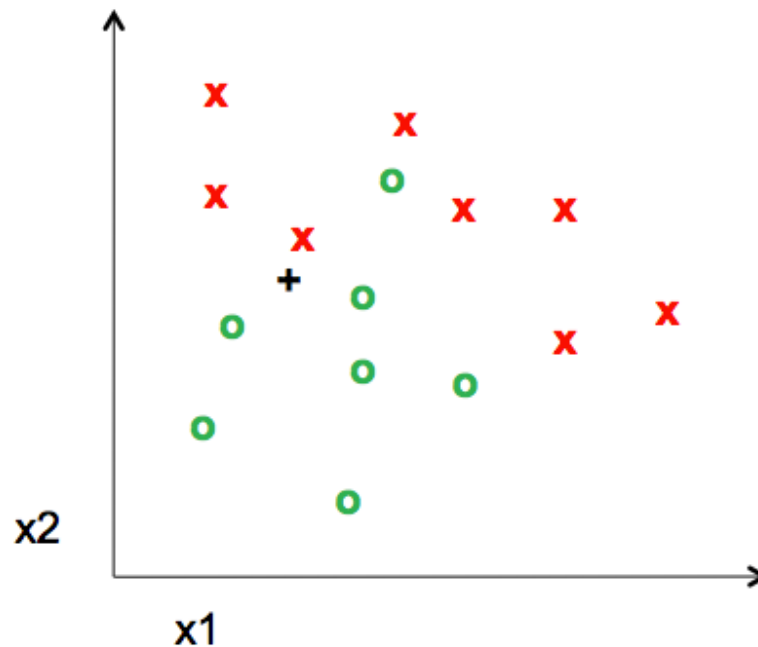
- Associe o elemento a 1 ou mais classes, levando em consideração a similaridade



Distância = Similaridade

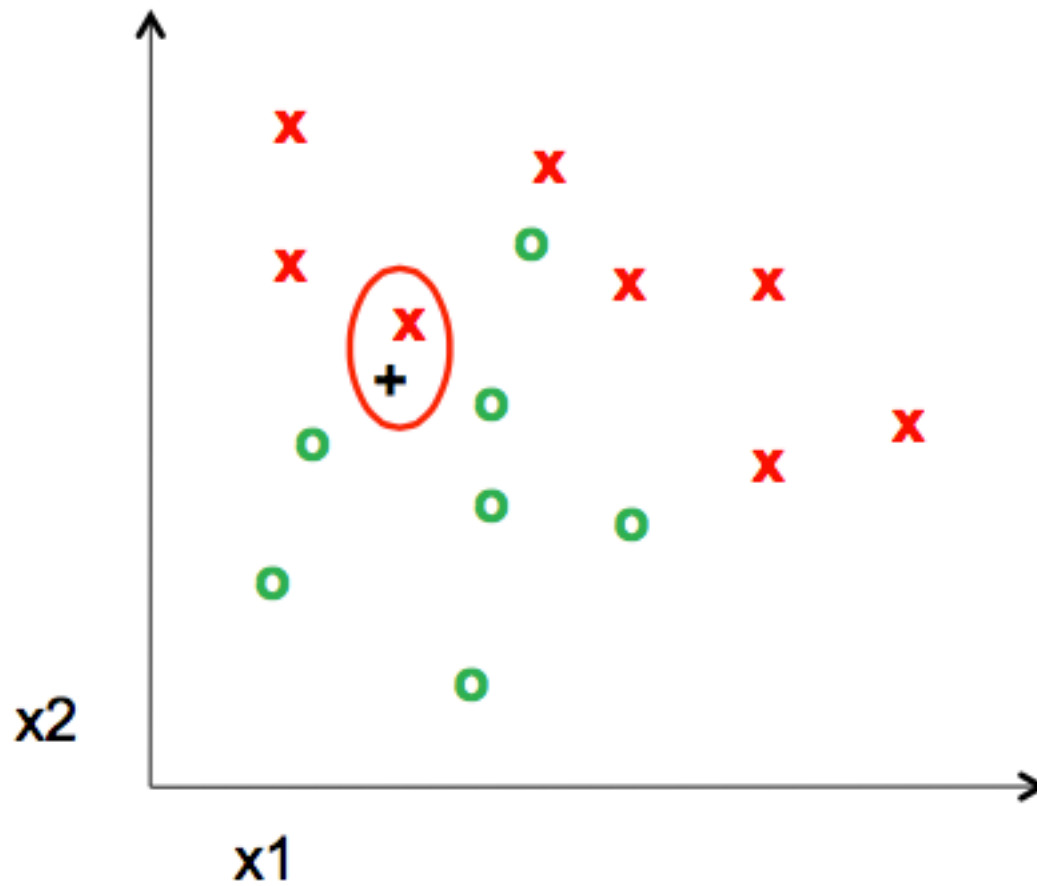
- Norma euclidiana

$$Dist(X^n, X^m) = \sqrt{\sum_{i=1}^D (X_i^n - X_i^m)^2}$$



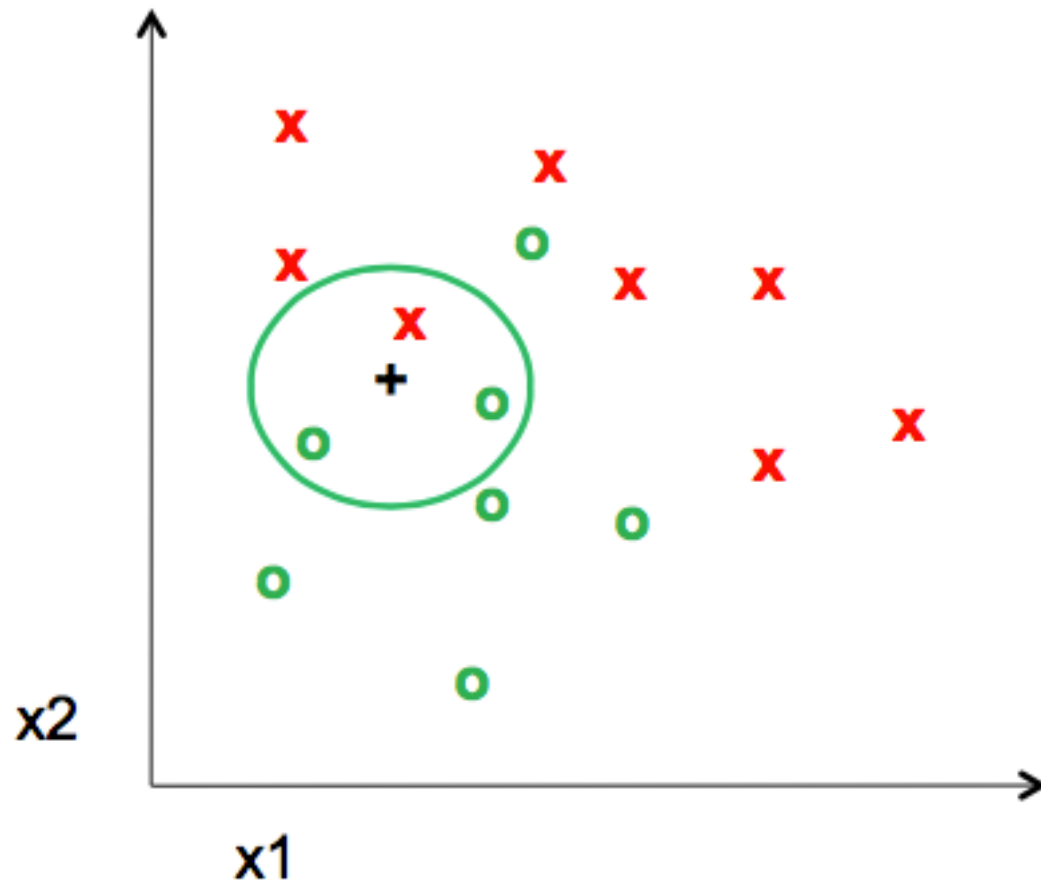
1-NN

- Assume a classe do mais próximo



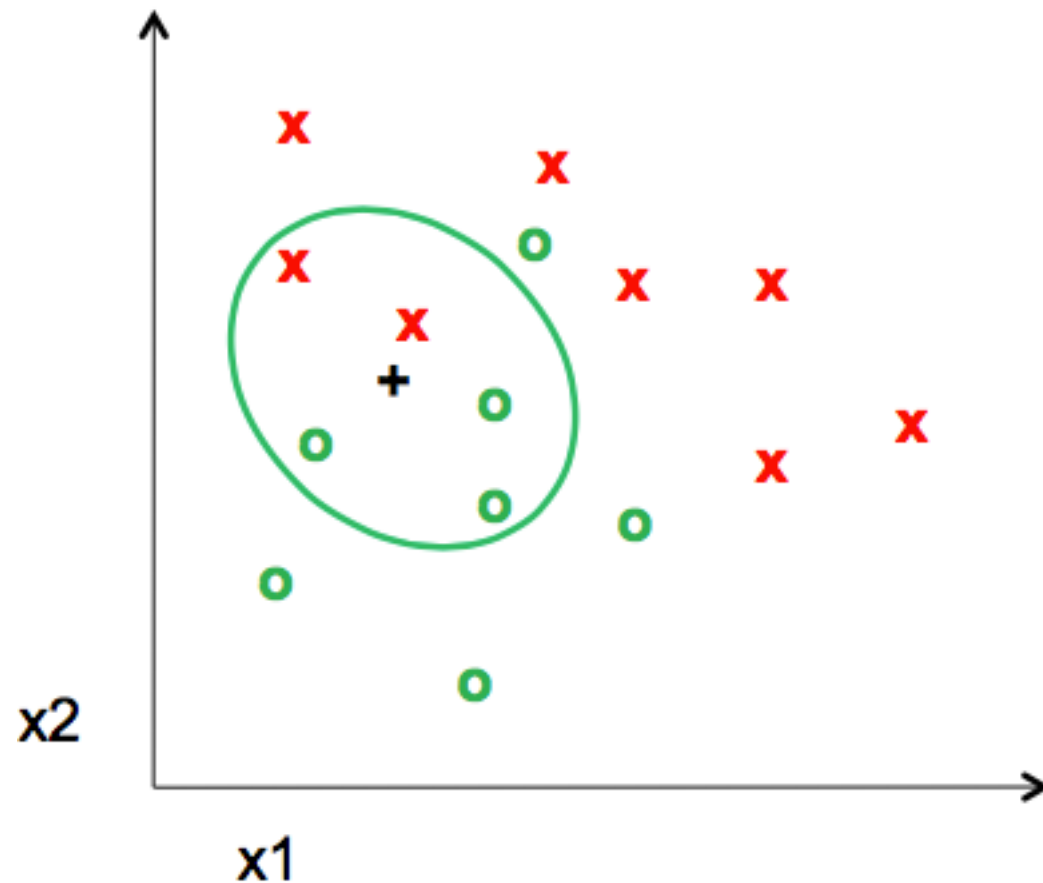
3-NN

- Assume a classe numa votação de 3 mais próximos



5-NN

- Assume a classe numa votação de 3 mais próximos



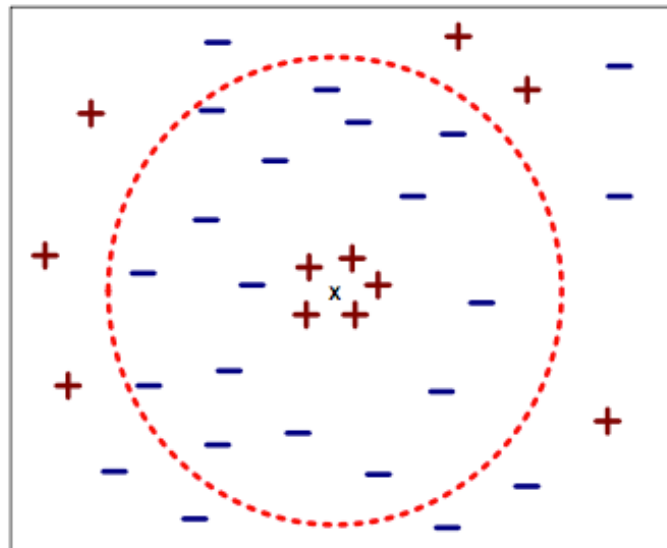
KNN

- Simples
- Flexível nas fronteiras
- Resultados interessantes (mas não os melhores)

- Problemas?
 - Escolher o valor de K
 - Problema de dimensionalidade

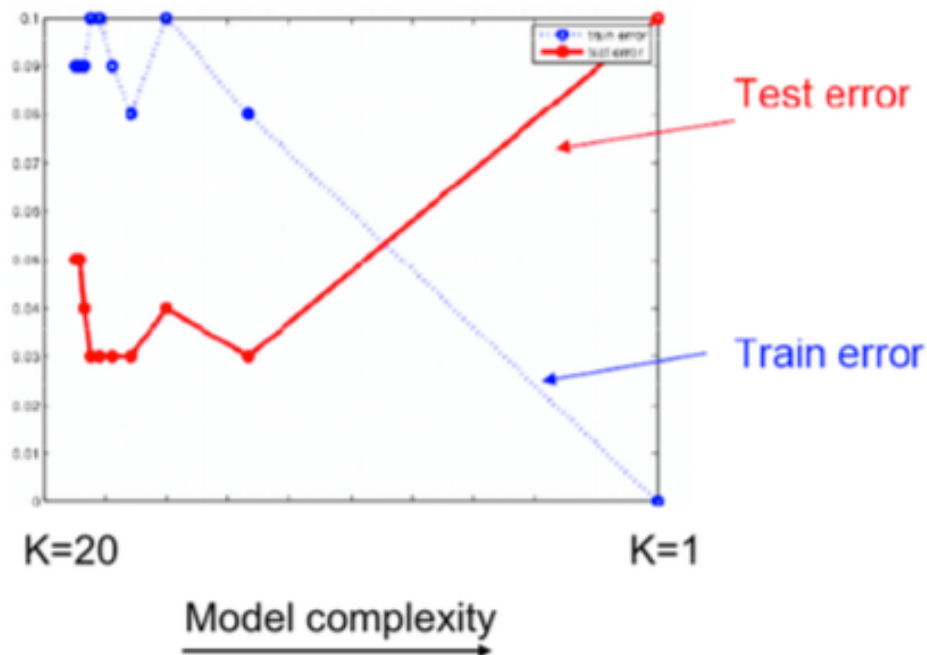
Escolher o valor de K

- O que acontece quando é grande demais?
 - Muitos pontos de outras classes
- E quando é pequeno demais?
 - Muito sensível. Não consegue lidar com ruído



Possíveis soluções

- Estimar o melhor valor de K através de testes que variem o seu valor numa faixa (cross-validate)



Outros classificadores

- **K-nearest neighbor**
- SVM
- Neural networks
- Logistic regression
- Boosted
- Decision Trees

- E muitos outros