FIAP GRADUAÇÃO

INTELIGÊNCIA ARTIFICIAL E COMPUTACIONAL

Profª . Miguel Bozer da Silva e
Prof. Henrique Ferreira dos Santos

profmiguel.silva@fiap.com.br

2020

Prof. Miguel Bozer da Silva

APRENDIZADO DE MÁQUINA

Aprendizado de Máquina

O Machine Learning (ML), ou em português Aprendizado de Máquina, envolve técnicas de **inteligência artificial baseada em dados**;

Os algoritmos de ML podem ser de diferentes tipos e estratégias de aprendizado;

Se um algoritmo prevê uma determinada estrutura matemática rígida do modelo ele é denominado **paramétrico** como os algoritmos de Regressão Linear e Regressão Logística;

Existem algoritmos **não paramétricos** como o KNN e a Árvore de Decisão;

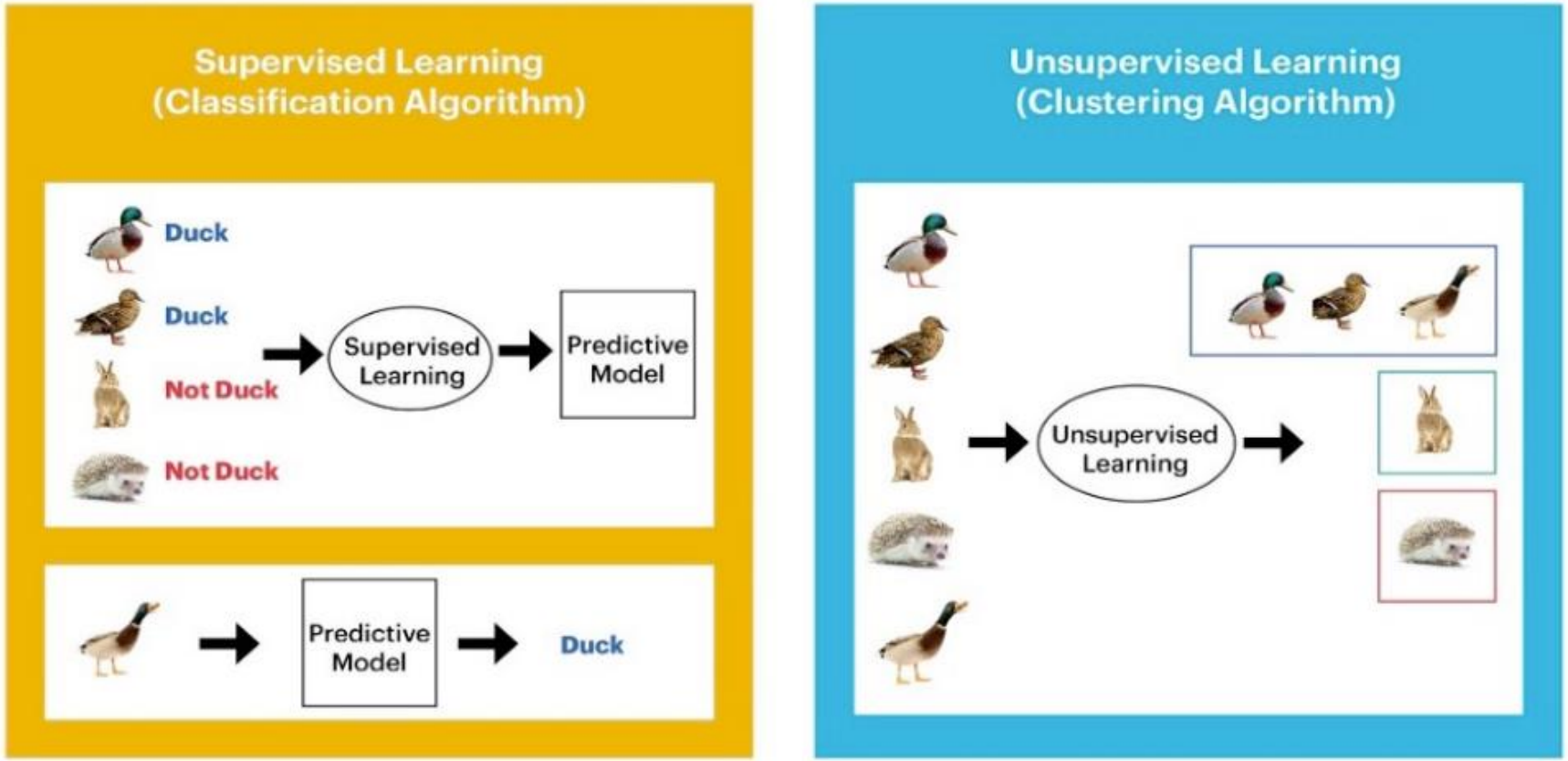
Além da qualidade paramétrica, os algoritmos de ML podem ser divididos em dois grupos:

Algoritmo de **Aprendizado Supervisionado**;

Algoritmos de **Aprendizado Não Supervisionado**;

Aprendizado de Máquina

No aprendizado **supervisionado** temos rótulos para cada entrada de dado. No **não supervisionado**, não fornecemos nenhuma informação (rótulo) para o agrupa



https://miro.medium.com/max/1000/0*4q_X_xQxevOQY_0u

Western Digital.

Prof. Miguel Bozer da Silva

APRENDIZADO SUPERVISIONADO

Aprendizado Supervisionado

- **O que é o aprendizado supervisionado?**
- No aprendizado supervisionado temos os dados de entrada do nosso modelo e também conhecemos os *labels* deles, isto é o valor esperado da saída do modelo para cada entrada:

$\mathbf{x}^{(i)}$ — i-ésima entrada do nosso modelo. Aqui temos todas as características diferentes que iremos utilizar para fazermos uma predição da saída

$y^{(i)}$ — Label com a saída esperada pelo nosso modelo

Aprendizado Supervisionado

- **O que é o aprendizado supervisionado?**
- No aprendizado supervisionado temos os dados de entrada do nosso modelo e também conhecemos os *labels* deles, isto é o valor esperado da saída do modelo para cada entrada:

Salario mensal	Nível de educação	Moradia	Aprovação do cartão de crédito
8500,00	Mestrado	Casa/Apartamento Próprio quitado	Aprovado
1950,00	Ensino médio / técnico	Aluguel	Reprovado
⋮	⋮	⋮	⋮
3500,00	Graduação	Casa/Apartamento Próprio financiado	Reprovado

$\mathbf{x}^{(i)}$ Entrada de dados

$y^{(i)}$ Saída de dados

Aprendizado Supervisionado

- Para o caso dos **classificadores**, conhecemos os nossos dados de entrada (x) e conhecemos os labels dele (y) que são **categóricos**

altura	peso	Classe
1,83	95	adulto
1,65	77	adulto
1,25	50	criança
⋮	⋮	⋮
1,77	69	adulto

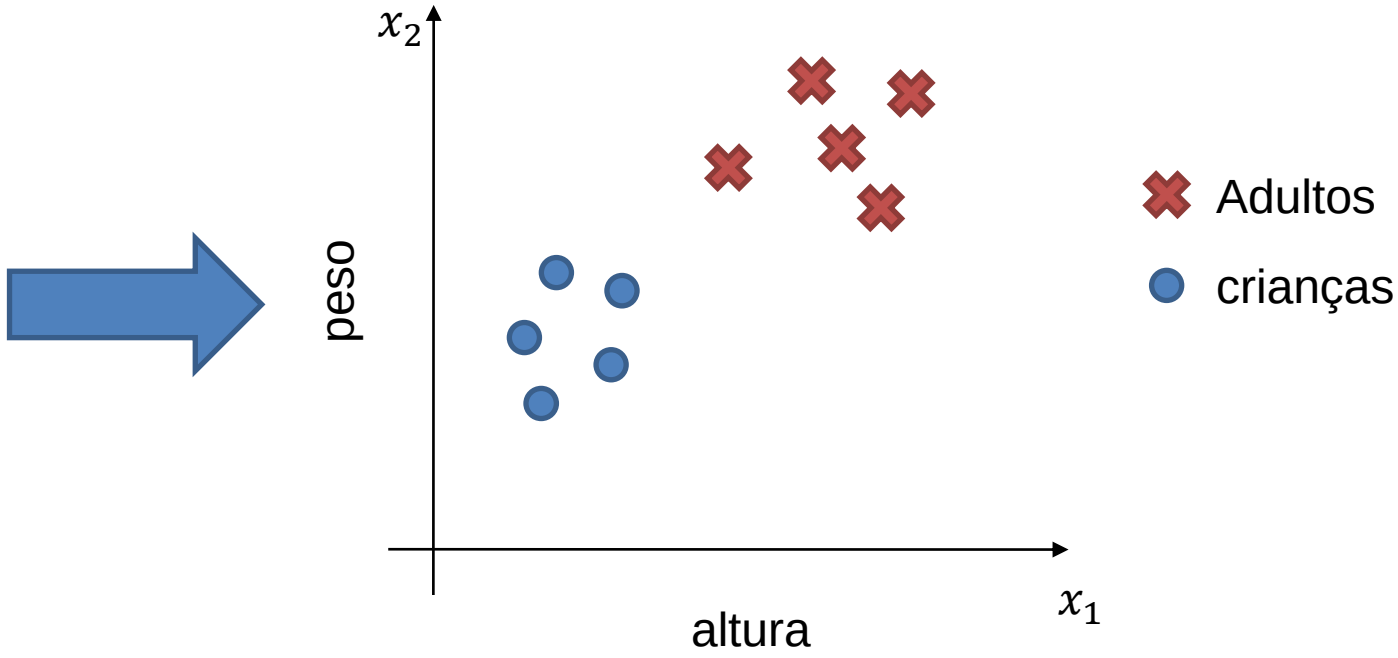
$\mathbf{x}^{(i)}$ Entrada de dados

$y^{(i)}$ Saída de dados

Aprendizado Supervisionado

- Para o caso dos **classificadores**, conhecemos os nossos dados de entrada (x) e conhecemos os labels dele (y)

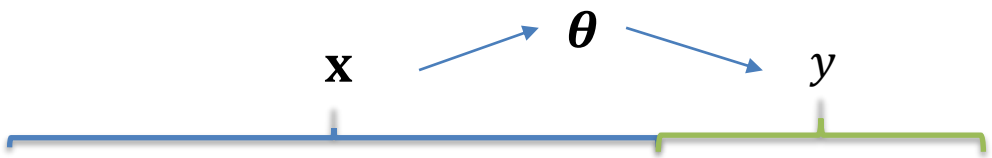
altura	peso	Classe
1,83	95	adulto
1,65	77	adulto
1,25	50	criança
⋮	⋮	⋮
1,77	69	adulto



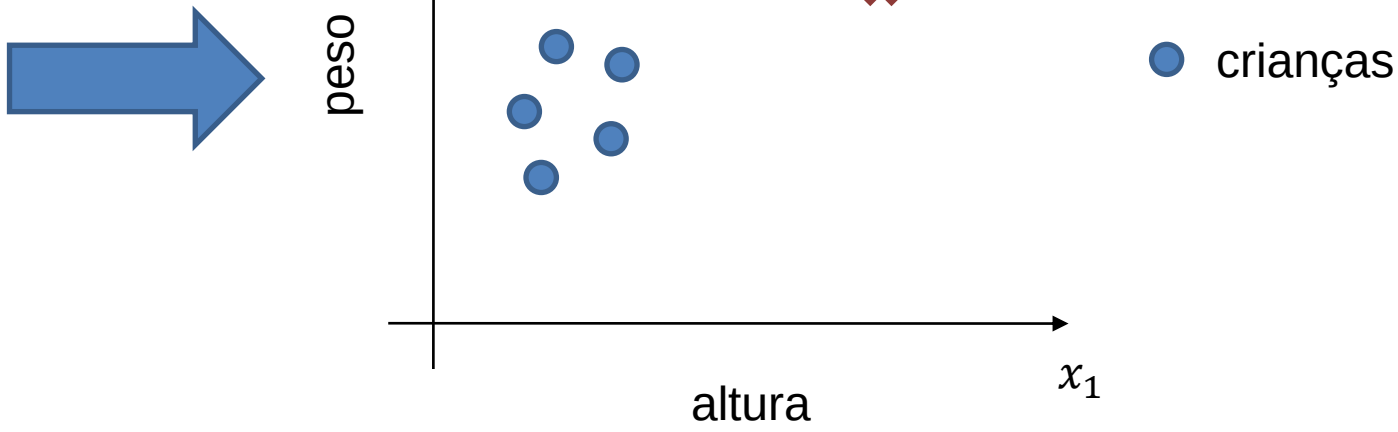
Observação: Podemos ter mais de duas classes nos nossos problemas!

Aprendizado Supervisionado

- Os modelos **classificadores** irão estimar parâmetros ($\boldsymbol{\theta}$) que nos indicam a relação entre as nossas entradas (**x**) e a nossa saída – *label* (*y*)



altura	peso	Classe
1,83	95	adulto
1,65	77	adulto
1,25	50	criança
⋮	⋮	⋮
1,77	69	adulto

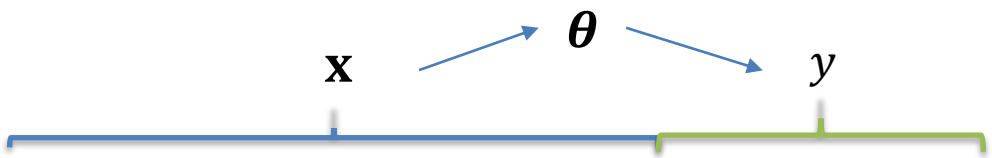


$$\hat{y} = f(\boldsymbol{\theta}) = \begin{cases} 0 \; se \; criança \\ 1 \; se \; adulto \end{cases}$$

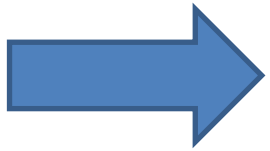
Observação: Podemos ter mais de duas classes nos nossos problemas!

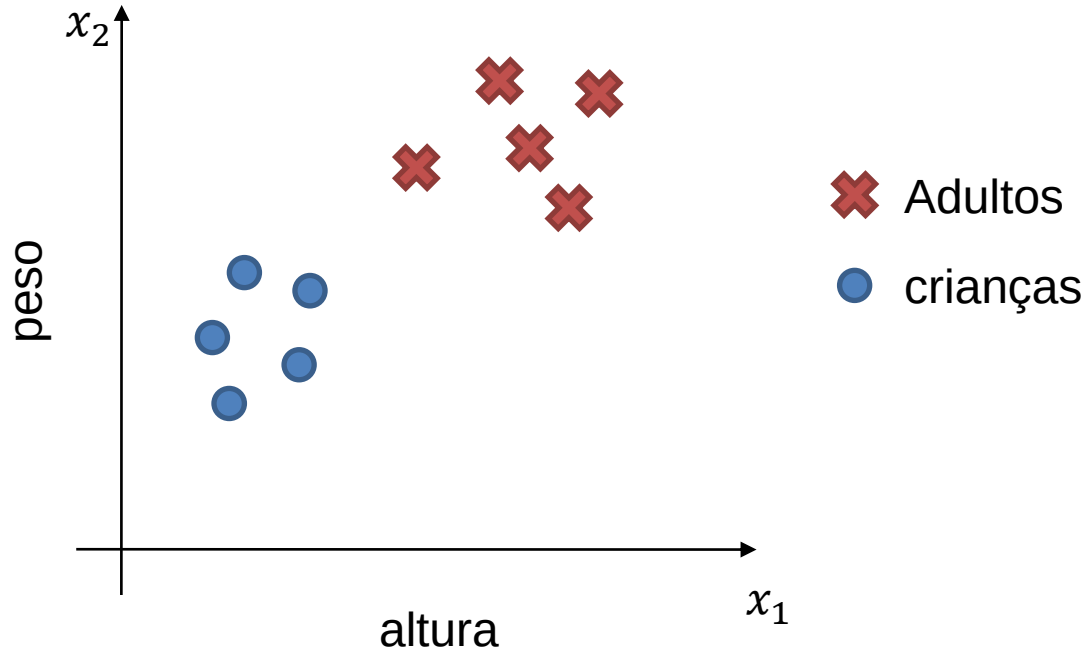
Aprendizado Supervisionado

- A etapa de aprendizado no nosso modelo f($\boldsymbol{\theta}$) é chamada de **treinamento**. Nela o modelo aprenderá a relação das entradas com as saídas.



altura	peso	Classe
1,83	95	adulto
1,65	77	adulto
1,25	50	criança
⋮	⋮	⋮
1,77	69	adulto



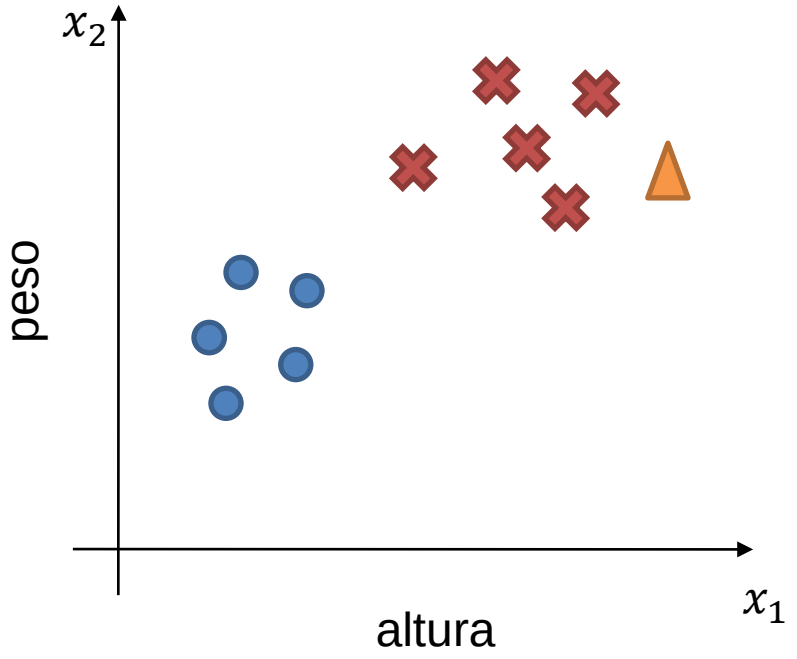


$$\hat{y} = f(\boldsymbol{\theta}) = \begin{cases} 0 \; se \; criança \\ 1 \; se \; adulto \end{cases}$$

Observação: Podemos ter mais de duas classes nos nossos problemas!

Aprendizado Supervisionado

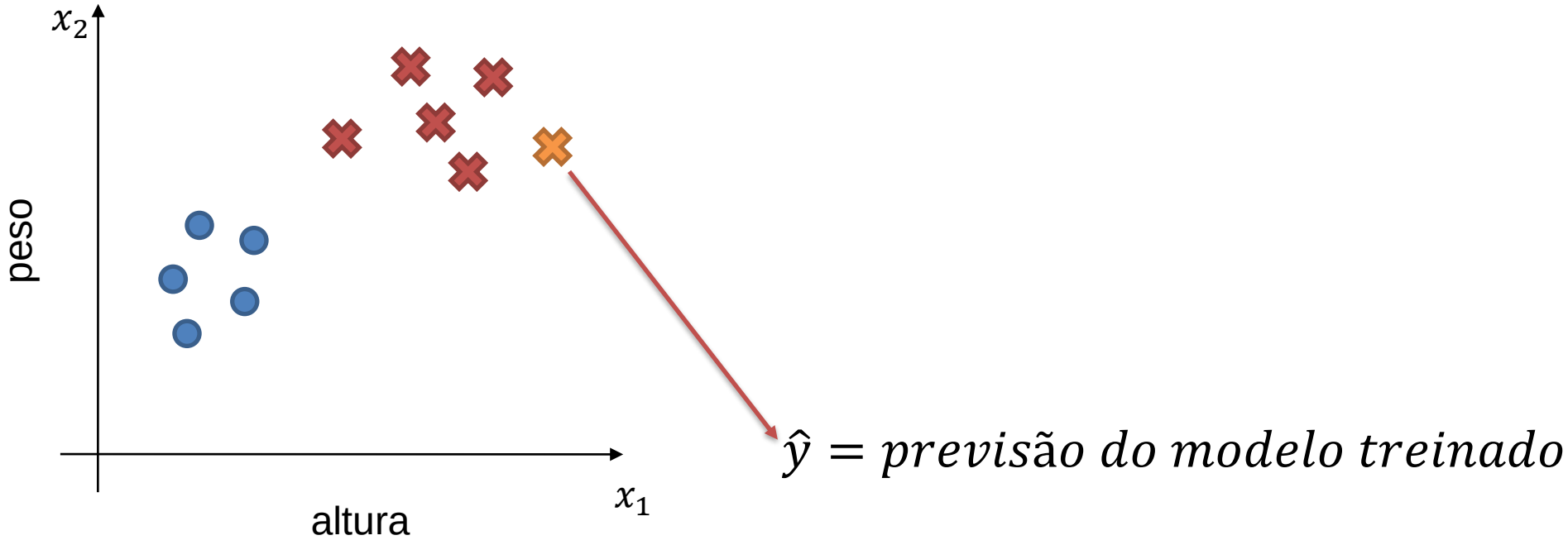
- Após o treinamento, podemos usar o nosso modelo para estimar dados desconhecidos: **Caso um novo dado** cuja classe é desconhecida for apresentado ao modelo, podemos classifica-lo!



Observação: Podemos ter mais de duas classes nos nossos problemas!

Aprendizado Supervisionado

- Após o treinamento, podemos usar o nosso modelo para estimar dados desconhecidos: **Caso um novo dado** cuja classe é desconhecida for apresentado ao modelo, podemos classifica-lo!

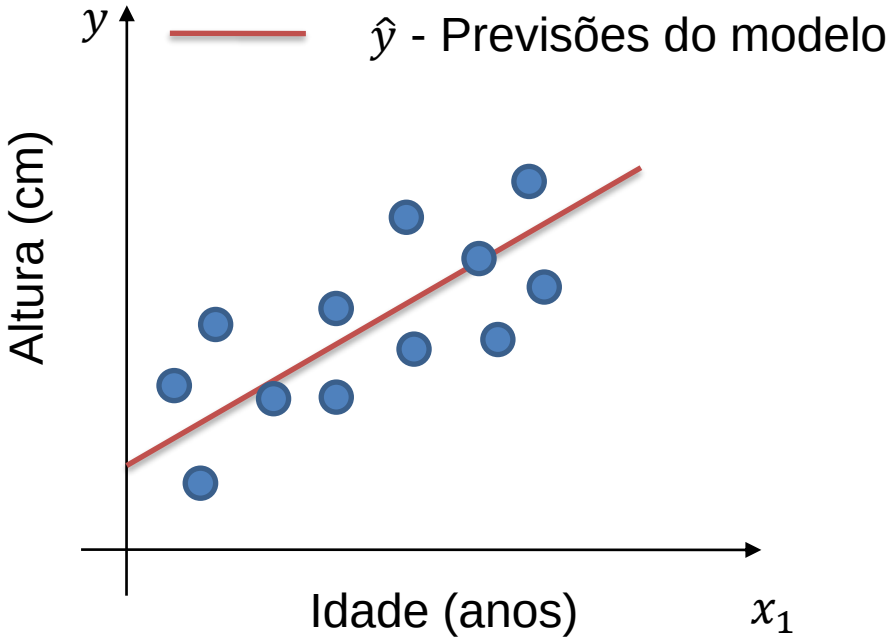


Observação: Podemos ter mais de duas classes nos nossos problemas!

Aprendizado Supervisionado

- Podemos também estimar uma saída de valores **numéricos contínuos** (y) a partir de um conjunto de dados de entrada (**x**)

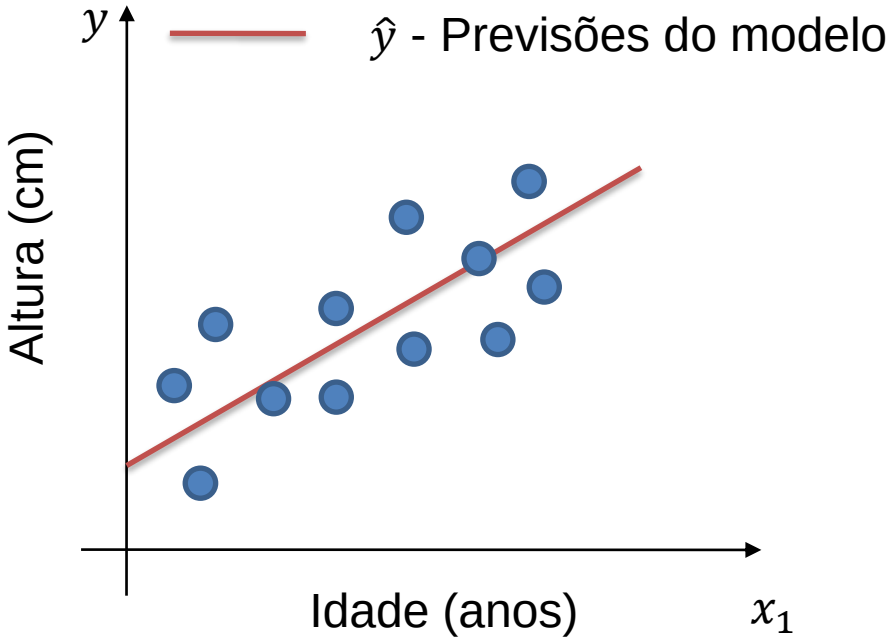
idade	altura
5	1,00
11	1,43
7	1,19
⋮	⋮
16	1,73
$\mathbf{x}^{(i)}$	$y^{(i)}$



Aprendizado Supervisionado

- Nesses casos temos a necessidade de utilizar modelos **regressores** para resolver o problema que estamos trabalhando

idade	altura
5	1,00
11	1,43
7	1,19
⋮	⋮
16	1,73
$\mathbf{x}^{(i)}$	$y^{(i)}$



Prof. Miguel Bozer da Silva

PROJETO DE APRENDIZADO SUPERVISIONADO

Projeto de Aprendizado Supervisionado

- Podemos resumir o passo a passo de um projeto de aprendizado supervisionado como:
  - Receber os dados
  - Analise Exploratória dos dados
    - Tratamento dos dados
  - Divisão do conjunto de dados
  - Realização do treinamento
  - Avaliação do modelo

Projeto de Aprendizado Supervisionado

ETAPA 1:

- Receber os dados
- Analise Exploratória dos dados
  - Tratamento dos dados

Vimos isso nas últimas aulas com o auxílio do pandas!

Projeto de Aprendizado Supervisionado

ETAPA 1 (Recordando):

- Tratamento de valores nulos;
- Tratamento de outliers;
- Em alguns projetos podemos ter conjuntos de dados categóricos. Nesses casos, temos que ajustá-los para valores numéricos.
- Temos duas principais abordagens para quando conseguimos estabelecer uma ordenação dos dados e quando não conseguimos fazer isso.

One Hot Enconding

- Quando não conseguimos ordenar os dados usamos o One Hot Enconding

- O One Hot Enconder transforma colunas categóricas em colunas binárias:

Color
Red
Red
Yellow
Green
Yellow

Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1
0	1	0

Label Enconding

- Quando conseguimos ordenar os dados usamos o Label Enconding

- O Label Encoding é aplicado quando temos números ordenados dos nossos dados:

SAFETY-LEVEL (TEXT)	SAFETY-LEVEL (NUMERICAL)
None	0
Low	1
Medium	2
High	3
Very-High	4

Projeto de Aprendizado Supervisionado

ETAPA 1 (Recordando):

- Somente faltou comentar um tópico de tratamento dos dados:
  - A normalização e padronização dos dados

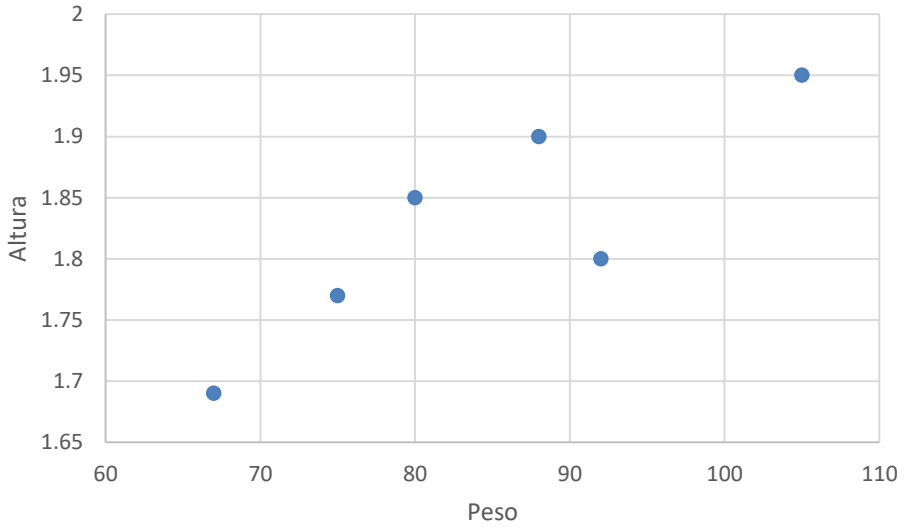
Normalização e Padronização dos Dados

- Alguns modelos de Machine Learning exigem que os valores estejam em escalas similares para que eles não se tornem tendenciosos. Por exemplo:

  - Se temos o peso, altura e o tamanho da camisa que uma pessoa usa. Podemos tentar usar esses dados para estimar qual o tipo de camisa uma pessoa pode comprar
  - Para isso, o nosso modelo recebe os valores do peso e da altura e estima a saída de tamanho da camisa.

Normalização e Padronização dos Dados

id	Peso	Altura(m)	Camisa
1	75	1,77	G
2	80	1,85	G
3	92	1,8	G
4	67	1,69	M
5	88	1,9	GG
6	105	1,95	GG



A escala do peso é muito maior que a altura.

Normalização e Padronização dos Dados

- Caso as grandezas dos dados envolvidos forem muito diferentes, podemos padronizar ou normalizar os nossos dados:

- Padronização:

$$z_i = \frac{x_i - \mu}{\sigma}$$

Onde:

z_i é o i-ésimo valor padronizado;

x_i é o i-ésimo valor original dos nossos dados

σ é o desvio padrão dos dados.

μ é a média dos dados

`Sklearn: StandardScaler()`

- Normalização: $X_{changed} = \frac{X - X_{min}}{X_{max} - X_{min}}$

Onde:

$X_{changed}$ é o valor normalizado

X é o valor antes da normalização

X_{min} é o menor valor do conjunto de dados

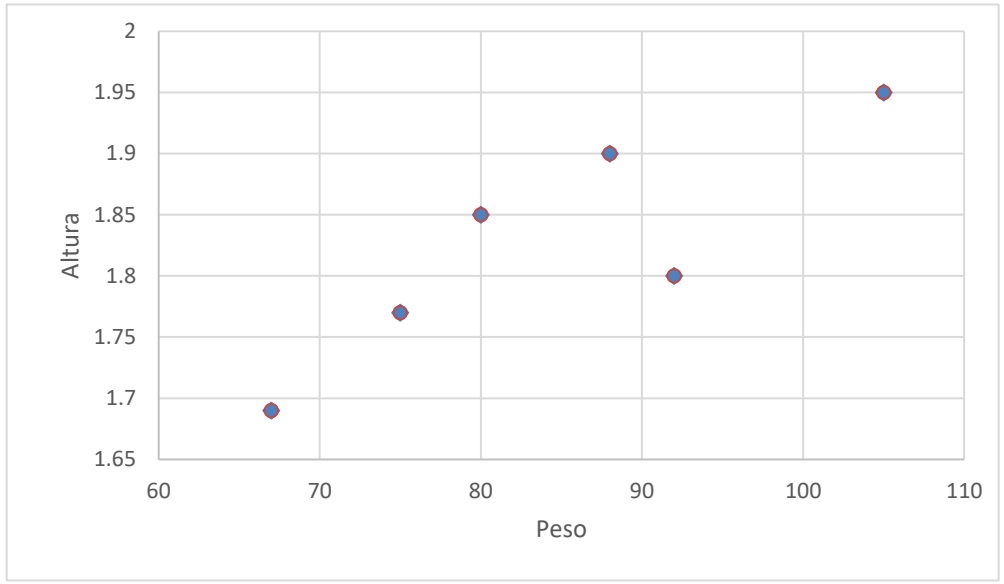
X_{max} é o maior valor do de dados

`Sklearn: MinMaxScaler()`

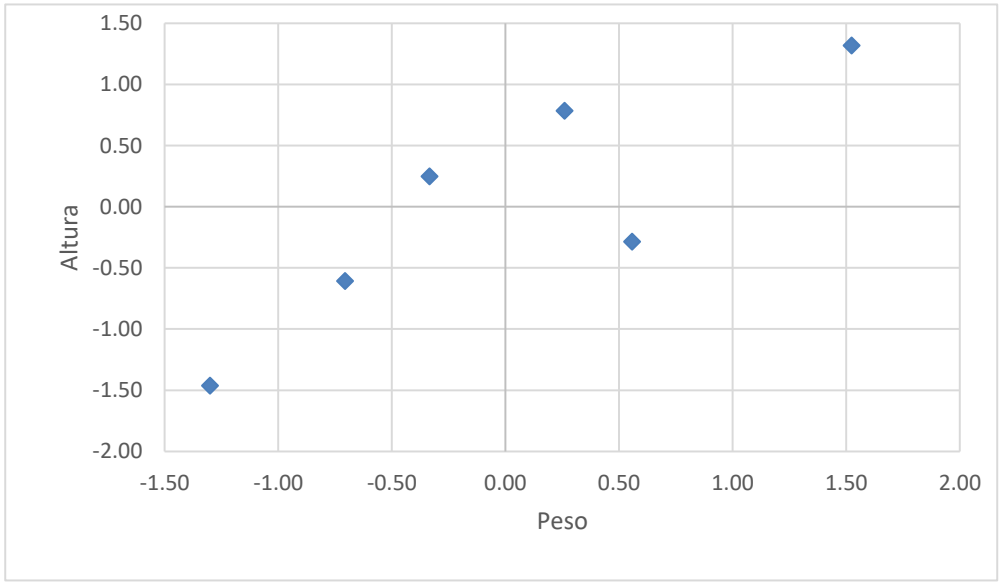
Normalização e Padronização dos Dados

- Após a padronização dos dados:

id	Peso	Altura(m)	Camisa
1	75	1,77	G
2	80	1,85	G
3	92	1,8	G
4	67	1,69	M
5	88	1,9	GG
6	105	1,95	GG

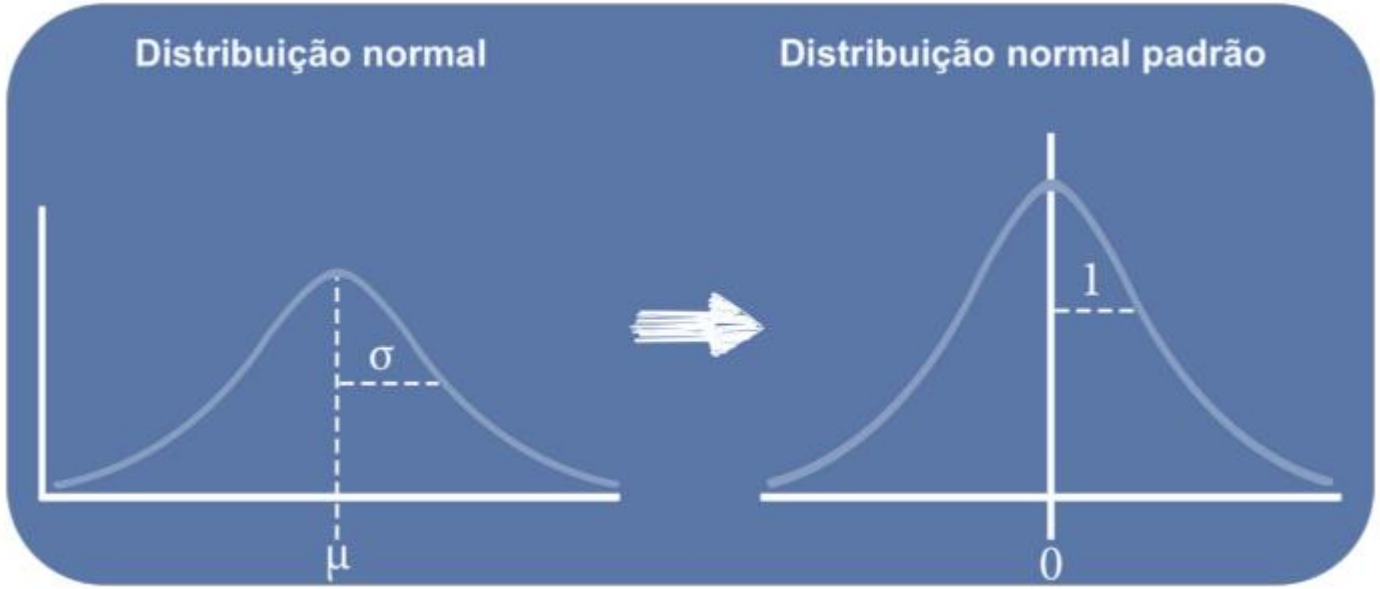


id	Peso	Altura(m)	Camisa
1	-0,71	-0,61	G
2	-0,33	0,25	G
3	0,56	-0,29	G
4	-1,30	-1,46	M
5	0,26	0,78	GG
6	1,52	1,32	GG



Normalização e Padronização dos Dados

- A Padronização dos dados transforma os mesmos em uma distribuição normal padrão

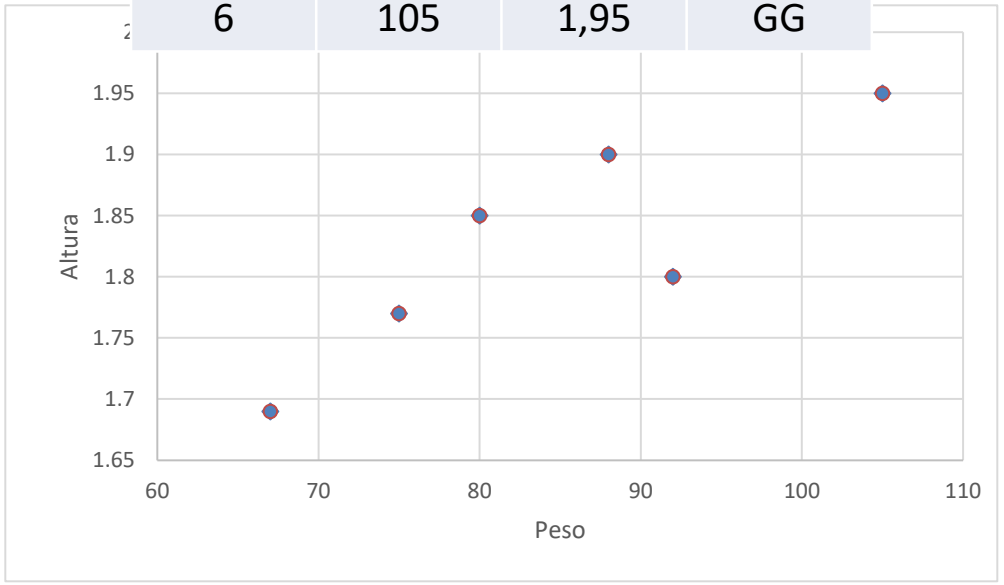


- Recomendado quando os dados estão em uma distribuição normal

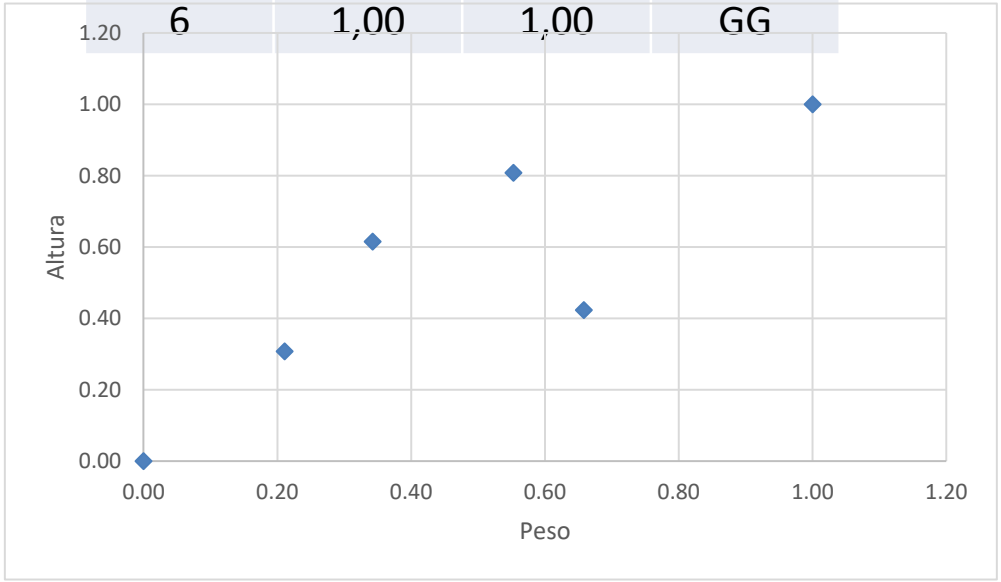
Normalização e Padronização dos Dados

- Após a normalização dos dados:

id	Peso	Altura(m)	Camisa
1	75	1,77	G
2	80	1,85	G
3	92	1,8	G
4	67	1,69	M
5	88	1,9	GG
6	105	1,95	GG



id	Peso	Altura(m)	Camisa
1	0,21	0,31	G
2	0,34	0,62	G
3	0,66	0,42	G
4	0,00	0,00	M
5	0,55	0,81	GG
6	1,00	1,00	GG



Normalização e Padronização dos Dados

- A normalização pode ser aplicada quando a distribuição dos dados não é normal ou se o desvio padrão dos mesmos for muito pequeno.

Projeto de Aprendizado Supervisionado

- ***ETAPA 2:***
- Divisão do conjunto de dados
- Realização do treinamento
- Avaliação do modelo no conjunto de teste

Projeto de Aprendizado Supervisionado

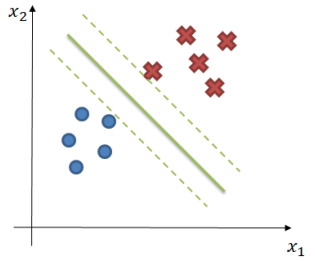
- Num projeto de aprendizado supervisionado teremos um conjunto de dados total que será utilizado

Projeto de Aprendizado Supervisionado

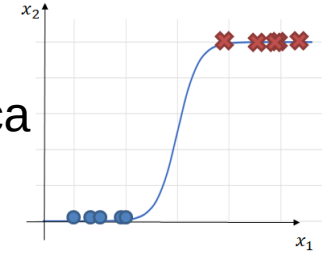
- Podemos escolher alguns modelos para aprender a relação das entradas e saídas de dados na **etapa de treinamento dos modelos**

Dados disponíveis (X; y)

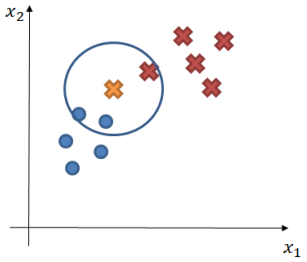
SVM



Reg. Logística



K-Vizinhos mais próximos



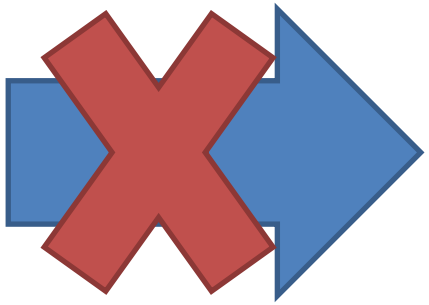
Entre outros ⋮

$$\hat{y} = f(\boldsymbol{\theta}) = \begin{cases} 0\ se\ classe\ 0 \\ 1\ se\ classe\ 1 \end{cases}$$

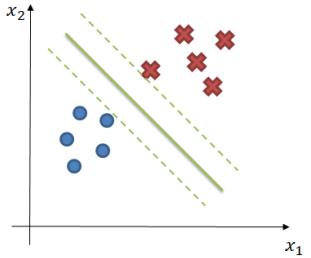
Projeto de Aprendizado Supervisionado

- **Um erro comum** é pensar que podemos usar todos os nossos dados para estimarmos os parâmetros dos modelos (θ) (*treinamento do modelo*)

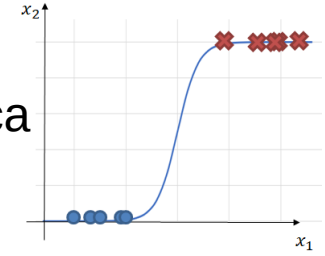
Dados disponíveis (X; y)



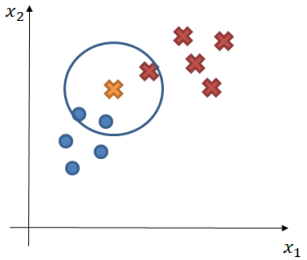
SVM



Reg. Logística



K-Vizinhos mais próximos



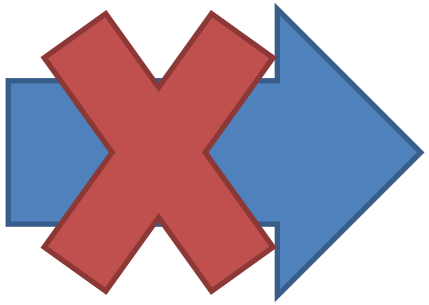
Entre outros ⋮

$$f(\boldsymbol{\theta}) = \begin{cases} 0\ se\ classe\ 0 \\ 1\ se\ classe\ 1 \end{cases}$$

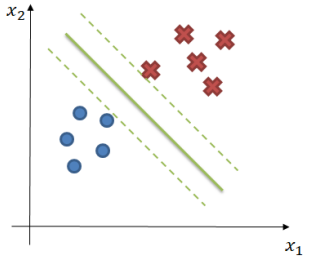
Projeto de Aprendizado Supervisionado

- Dessa forma, não separamos dados para verificar o desempenho de cada modelo com **novos dados dos quais ele nunca foi exposto.**

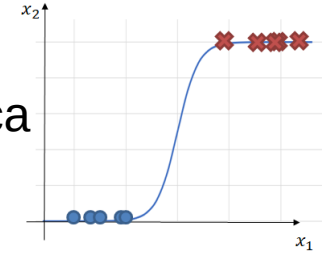
Dados disponíveis (X; y)



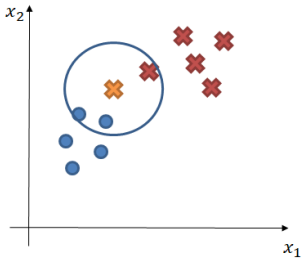
SVM



Reg. Logística



K-Vizinhos mais próximos



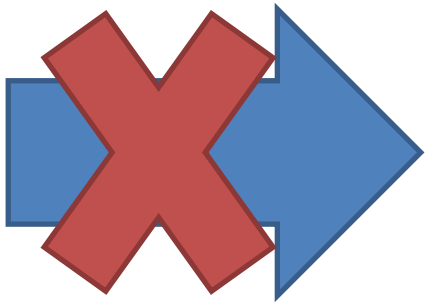
Entre outros ⋮

$$f(\boldsymbol{\theta}) = \begin{cases} 0\ se\ classe\ 0 \\ 1\ se\ classe\ 1 \end{cases}$$

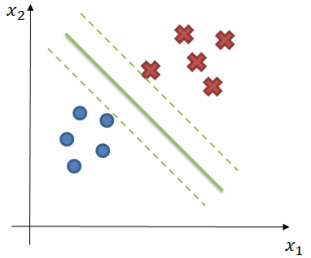
Projeto de Aprendizado Supervisionado

- Reutilizar os dados de treinamento para testarmos o nosso modelo é uma **péssima ideia**. NECESSITAMOS testá-lo em novos dados

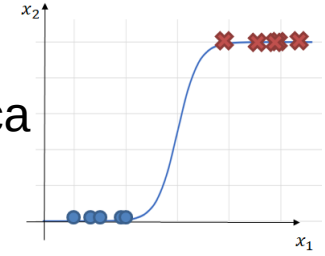
Dados disponíveis (X; y)



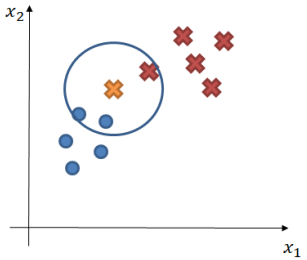
SVM



Reg. Logística



K-Vizinhos mais próximos



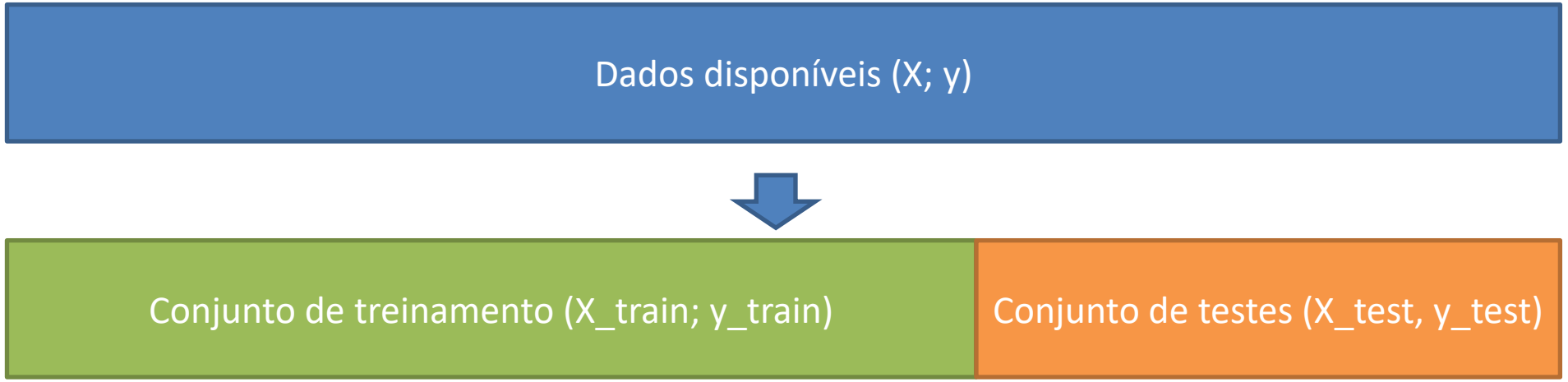
Entre outros ⋮

$$f(\boldsymbol{\theta}) = \begin{cases} 0\ se\ classe\ 0 \\ 1\ se\ classe\ 1 \end{cases}$$

Divisão dos Conjuntos de Dados

- **Conjunto de treinamento:** Utilizado para o modelo aprender as relações entre as entradas e saídas dos meus dados

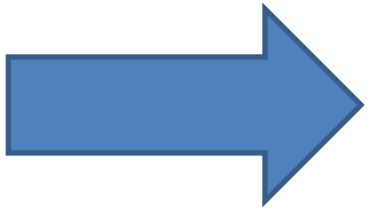
- **Conjunto de teste:** Utilizado para verificar se o nosso modelo foi devidamente treinado e checarmos com métricas de desempenho se o nosso

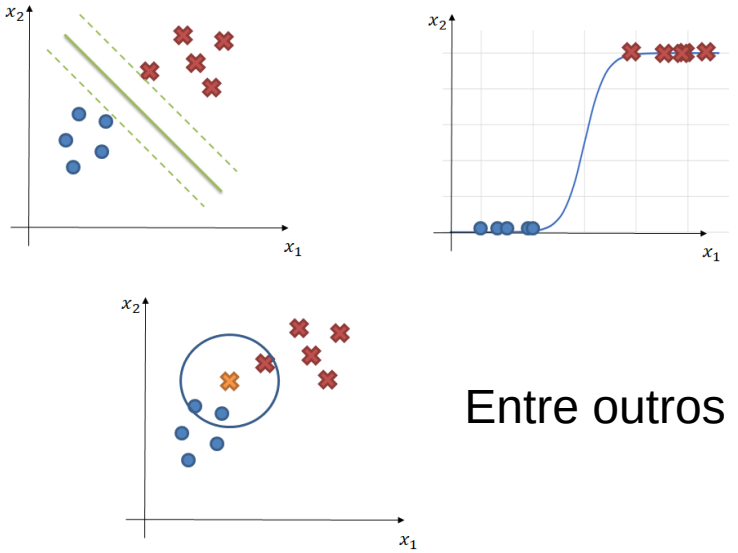


Projeto de Aprendizado Supervisionado

- Com a divisão dos dados podemos atuar da seguinte forma:

Conjunto de treinamento (X_train; y_train)





Entre outros

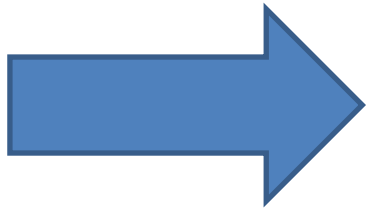
$$f(\boldsymbol{\theta}) = \begin{cases} 0\ se\ classe\ 0 \\ 1\ se\ classe\ 1 \end{cases}$$

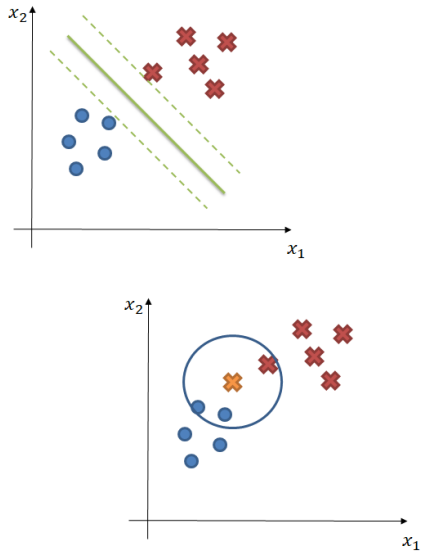
Modelos treinados : Parâmetros θ estimados para cada modelo!

Projeto de Aprendizado Supervisionado

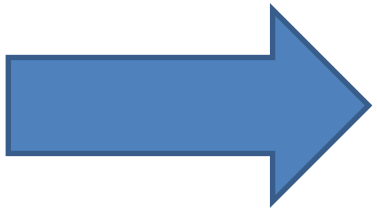
- Com a divisão dos dados podemos atuar da seguinte forma:

Conjunto de testes
(X_test, y_test)





Entre outros



$\hat{y}_{SVM}$

$\hat{y}_{reg}$

$\hat{y}_{KNN}$

Modelos previamente treinados

Podemos estimar as saídas para cada entrada

Projeto de Aprendizado Supervisionado

- Agora podemos comparar o que os modelos estavam ($\hat{y}$) prevendo com o que eles deveriam estar prevendo (y_test)

y_test	$\hat{y}_{SVM}$	$\hat{y}_{reg}$	$\hat{y}_{KNN}$
Adulto	Criança	Adulto	Criança
Adulto	Adulto	Criança	Adulto
Criança	Criança	Criança	Adulto
Adulto	Adulto	Adulto	Criança
Criança	Criança	Adulto	Adulto
Adulto	Criança	Adulto	Adulto
Adulto	Criança	Adulto	Criança
⋮	⋮	⋮	⋮
Criança	Adulto	Criança	Adulto

Valores verdadeiros *Valores Estimados por diferentes modelos*

Projeto de Aprendizado Supervisionado

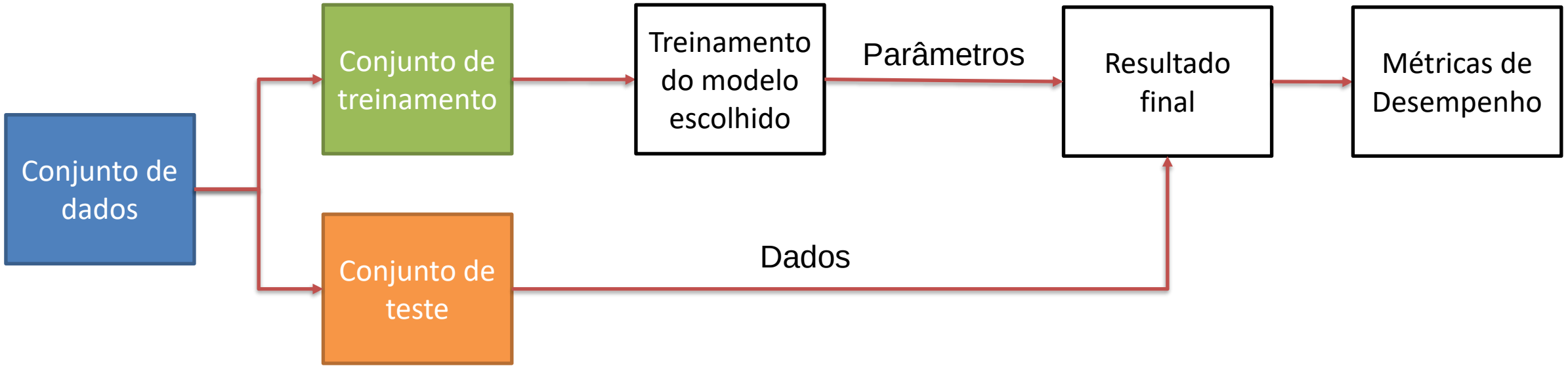
- Podemos comparar os modelos e tentar ver qual deles consegue chegar o mais próximo possível de *y_test.* Para isso, usamos as métricas de desempenho

y_test	$\hat{y}_{SVM}$	$\hat{y}_{reg}$	$\hat{y}_{KNN}$
Adulto	Criança	Adulto	Criança
Adulto	Adulto	Criança	Adulto
Criança	Criança	Criança	Adulto
Adulto	Adulto	Adulto	Criança
Criança	Criança	Adulto	Adulto
Adulto	Criança	Adulto	Adulto
Adulto	Criança	Adulto	Criança
⋮	⋮	⋮	⋮
Criança	Adulto	Criança	Adulto

Valores verdadeiros *Valores Estimados por diferentes modelos*

Treinamento e Teste - Resumo

- Para usarmos modelos de Machine Learning temos que criar os conjuntos de dados para treinamento e teste



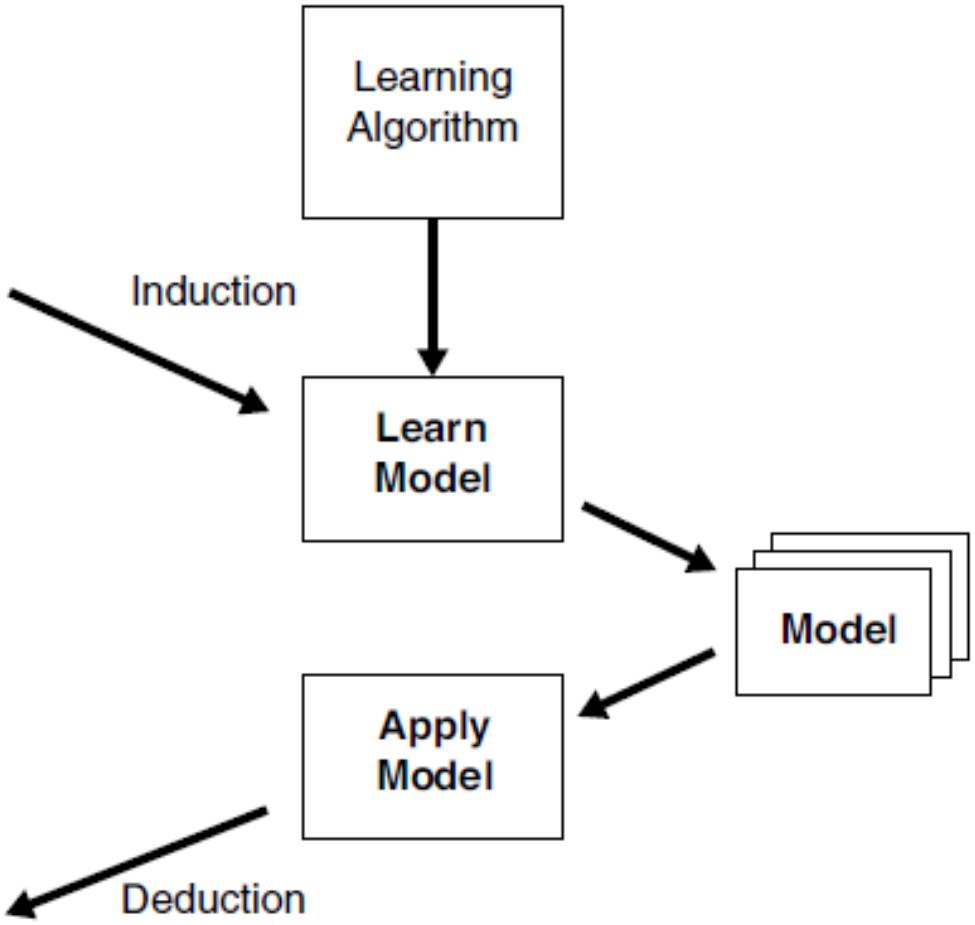
Treinamento e Teste - Resumo

Training Set

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Test Set

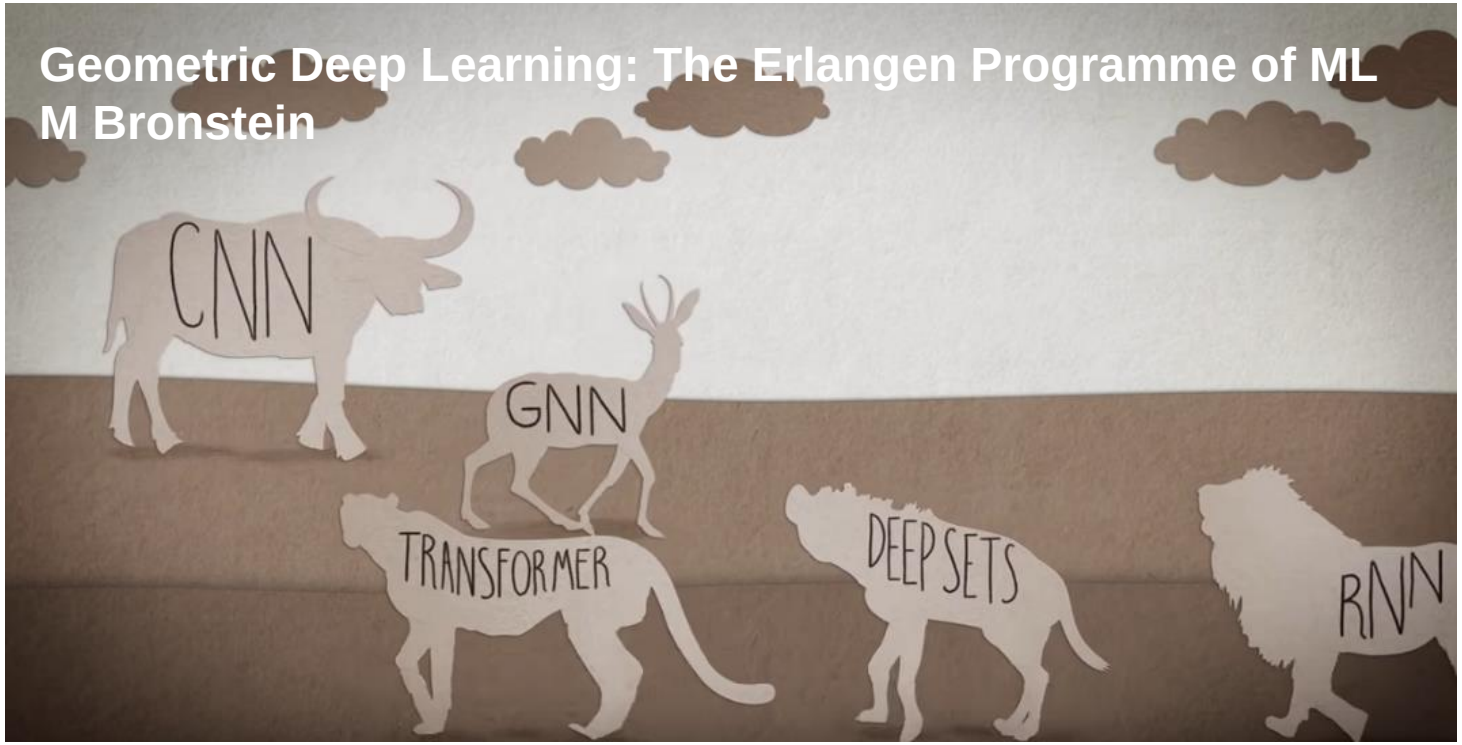
Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?



Treinamento e Teste - Resumo

Como pudemos ver ao longo da aula, existem muitas formas de agrupar os diferentes algoritmos de Machine Learning. Muito desse processo de classificação ainda está sendo feito agora!

É realmente um Zoológico de Algoritmos!



https://www.youtube.com/watch?v=w6Pw4MOzMuo&list=LL&index=129&ab_channel=MichaelBronstein

Treinamento e Teste - Resumo

- Infelizmente, o Zoológico é muito grande para nosso tour: não vamos conseguir conhecer todos os algoritmos que existem esse ano;
- Além disso, muitos outros algoritmos estão sendo propostos todos os meses!
- Vamos estudar alguns dos mais importantes para realizar tarefas básicas de Inteligência Artificial e Ciência de Dados, entre eles:

❖ **Aprendizado supervisionado**:
- ➢ Regressão: regressão linear, SVR (SVM), Árvore de Decisão e KNR;
- ➢ Classificação: KNN, Árvore de Decisão, RandomForest, SVM, Naive Bayes, Regressão Lógistica;

❖ **Aprendizado não supervisionado:**
- ➢ Agrupamento: k-means, hierárquico, DBSCAN, mistura gaussiana;
- ➢ Redução de dimensionalidade: t-SNE, PCA, kPCA, Isomap;

- No segundo semestre iremos ver outra parte do zoológico que realiza essas mesmas tarefas de maneira diferente: as Redes Neurais Artificiais (Deep Learning).

Avaliação do modelo

- Porque devemos avaliar o modelo?

- IA PODE ERRAR!

Erros acontecem... overfitting e underfitting

FIAP

Overfitting

- Modelo que se ajusta aos dados de treinamento muito bem, incluindo outliers
- Impacto negativo na capacidade do modelo em generalizar

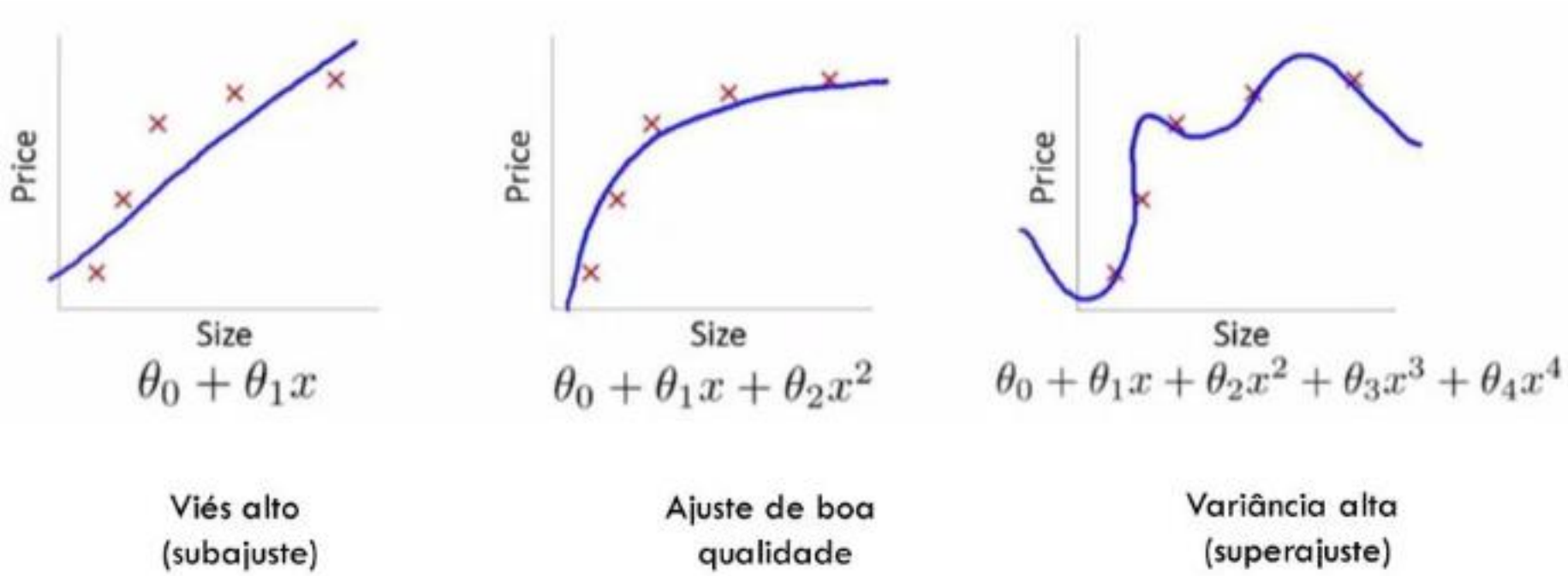
Underfitting

- Um modelos que nem se ajusta bem aos dados de treino, nem generaliza para novos dados

Erros acontecem... overfitting e underfitting

Um modelo com **overfitting** tem mais coeficientes do que o necessário. É um modelo com pouca capacidade de generalização: ele terá alta acurácia para os dados de treinamento e acurácia extremamente baixa para os dados de teste.

•



Erros acontecem...

Bias (enviesamento): Precisamos ser éticos na escolha das colunas que iremos usar e no dados que iremos fornecer aos algoritmos para que injustiças e preconceitos prévios não sejam ensinados aos algoritmos!

Overffiting (sobreajuste): Precisamos fazer separação treino/teste para atestar a generalidade de nosso modelo, levando em consideração o tipo de dados e o propósito para escolher tipo de metodologia de separação (80/20, cross validation, data leakage);

Acurácia e Precisão: Precisamos comparar com resultados de sistemas tradicionais (normalmente denominados de Modelo Base);

FIAP GRADUAÇÃO