Fundamentos de Data Science, Data Mining e Análise Preditiva

Aula 1

Francisco Nauber Bernardo Gois

Uni7

15 de maio de 2

Instrutor

F. Nauber Bernardo Gois

Dsc. Informática Aplicada
Lider de Aprendizado de Máquina na Secretaria de Saúde do Ceará
Engenheiro de Aprendizado de Máquina
Professor da UFC (2018-2019)
Analista de Desenvolvimento Serpro (2004-2018)

Instrutor

https://www.linkedin.com/in/n

https://www.linkedin.com
/in/naubergois/
Envie recomendações, depoimentos e competÊncias

Instrutor

https://www.youtube.com/can

https://www.youtube.com/
canaldaciencia
Instagram: @canaldaciencia

Índice

Índice

Material do Curso

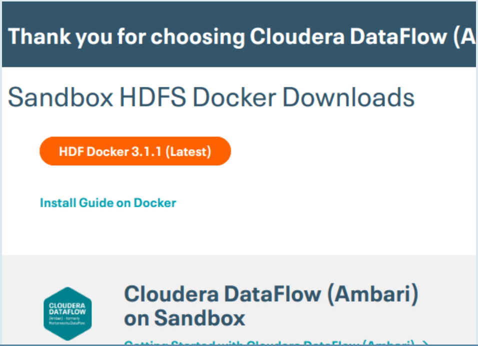
Chris Albon

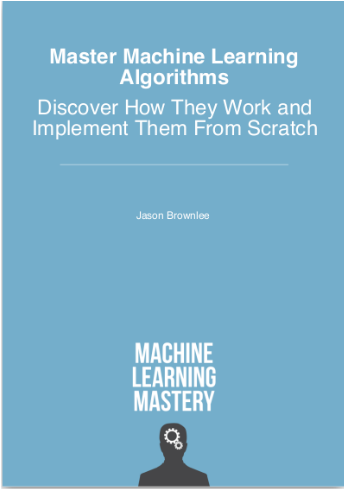
Curso de Aprendizado de Máquina
https://chrisalbon.com/

Material do Curso

HDP 2.5 sandbox image

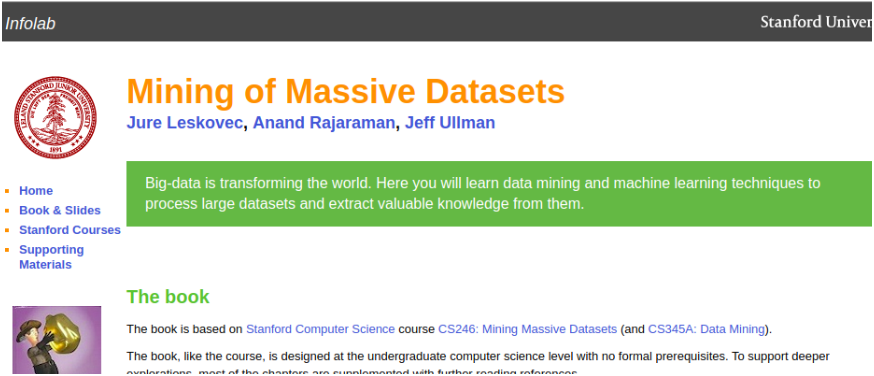
Baixe a imagem no endereço `https://www.cloudera.com/downloads/hortonworks-sandbox/hdf.html?utm_source=`





Livro Recomendado

Livro



`http://mmds.org/`

Data is Power

Data contains value and knowledge

`http://web.stanford.edu/class/cs246/slides/`

DATA IS THE NEW OIL!

"Dados são o novo Petróleo"

Perry Rotella
Contributor

FOLLOW

full bio →

Opinions expressed by Forbes Contributors are their own.

TECH 4/02/2012 @ 11:09AM | 10.791 views

Is Data The New Oil?

+ Comment Now + Follow Comments

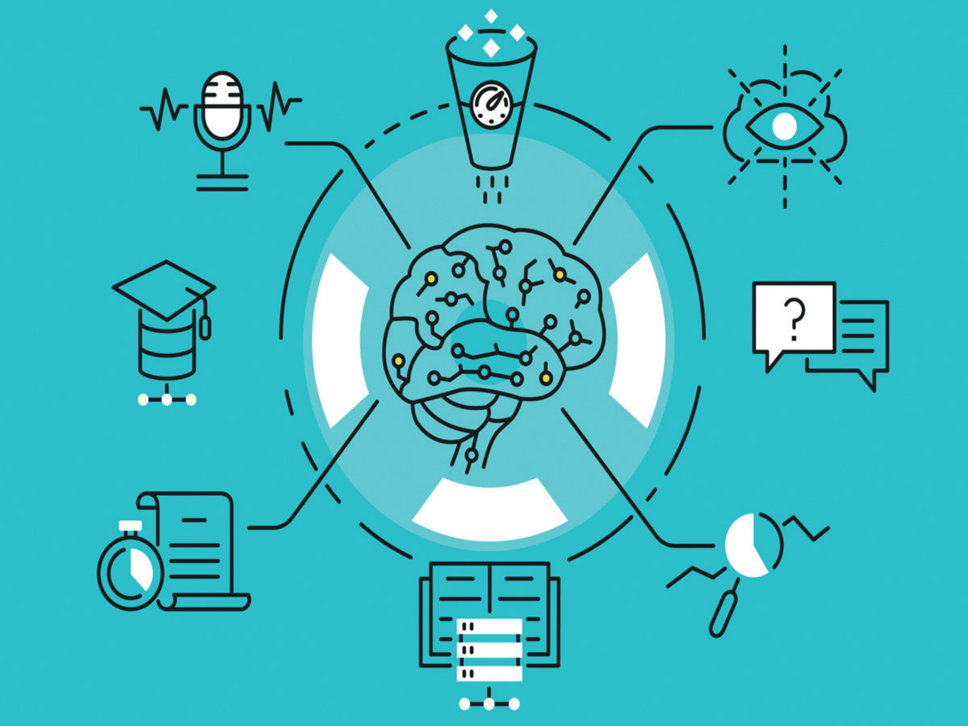
Recently, on a CNBC Squawk Box segment, "The Pulse of Silicon Valley," host Joe Kernan posed the question, "What is the next really big thing?" to Ann Winblad, the legendary investor and senior partner at Hummer-Winblad. Her response: "Data is the new oil."

Como petróleo, precisam ser refinados !

https://www.slideshare.net/MarcoGarcia/construindo-data-lakes-viso-prtica-com-hadoop-e-bigdata

Índice

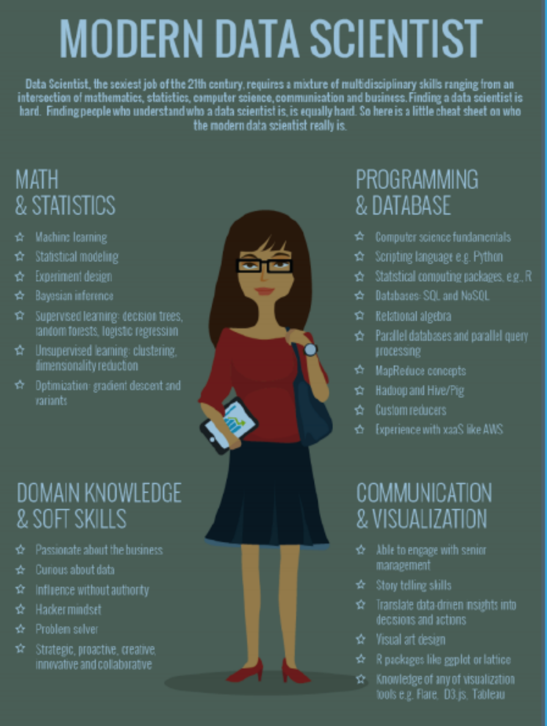
Índice



O QUE É CIÊNCIA DE DADOS?

DATA SCIENCE É MAIS UM TERMO USADO PARA DESCREVER O PROCESSO DE TRANSFORMAÇÃO DE DADOS EM CONHECIMENTO.

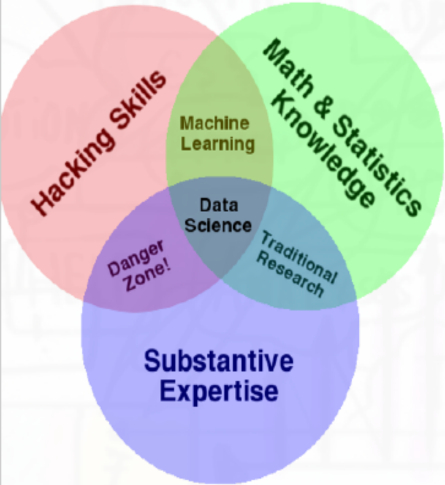
(LOUKIDES, 2016)

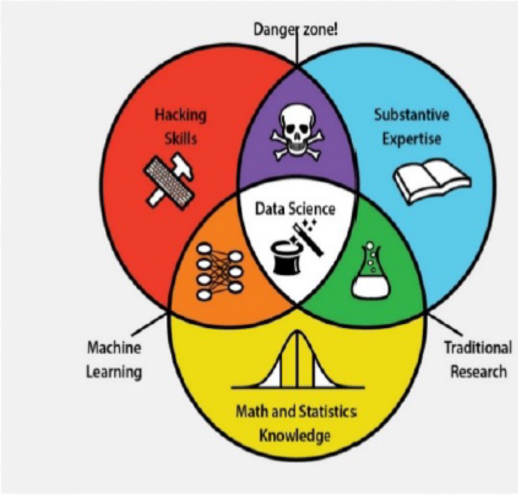


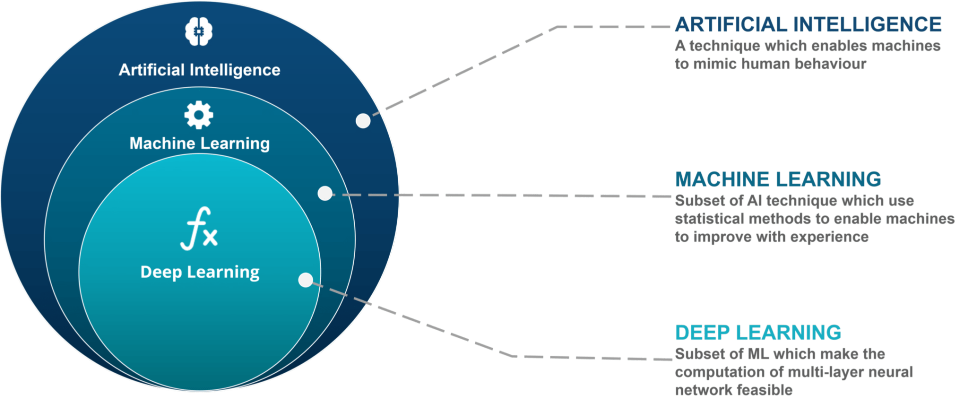
CIENTISTA DE DADOS

- Matemática e estatística
- Banco de Dados e Programação
- Conhecimento de Negócio
- Comunicação

DATA SCIENCE VENN DIAGRAM






Artificial Intelligence
Machine Learning
Deep Learning
ARTIFICIAL INTELLIGENCE
A technique which enables machines to mimic human behaviour
MACHINE LEARNING
Subset of AI technique which use statistical methods to enable machines to improve with experience
DEEP LEARNING
Subset of ML which make the computation of multi-layer neural network feasible

Aprendizado Estatístico

Algoritmo é o processo de aprendizado dos dados

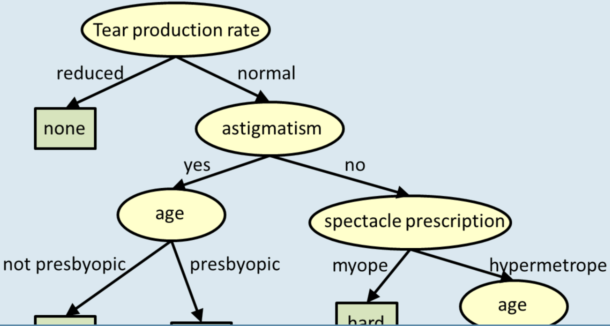
VariavelSaida = função(VetorEntrada)
A variável de saída é a variável dependente. Isso ocorre porque no enquadramento do problema de previsão, a saída é dependente (uma função) da entrada (também chamada de variáveis independentes).

	A	B
1	X (Variável Independente)	Y (Variável Dependente)
2	2	5
3	3	7
4	4	9
5	5	11
6	6	13
7	7	15
8	8	17
9	9	19
10	10	21
11	11	23

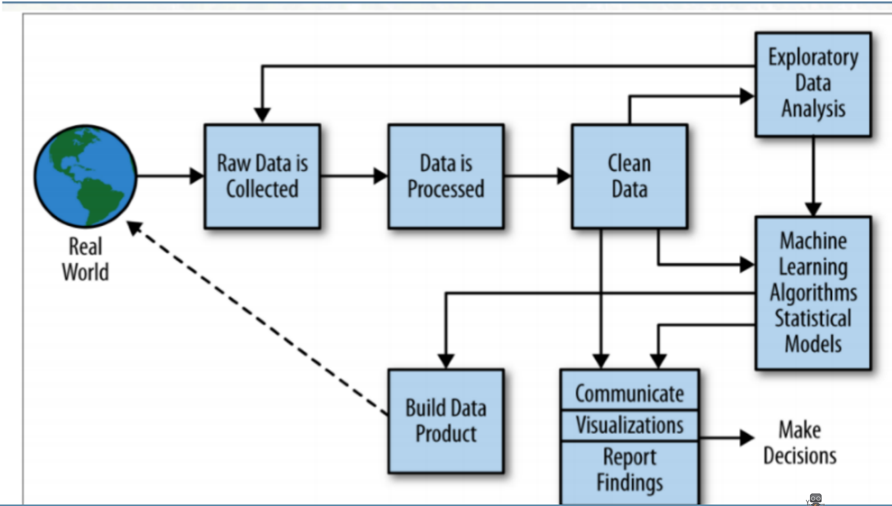
Modelos e Algoritmos

Variáveis dependentes e independentes

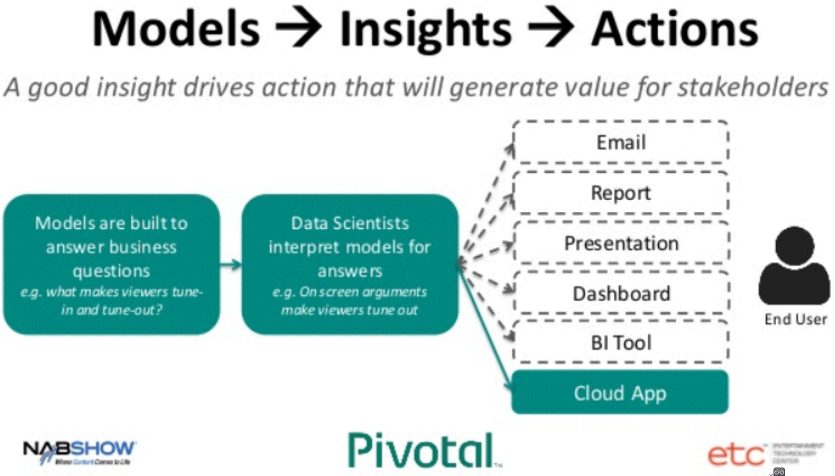
Model = Algorithm(Data)
Modelo é a representação do que foi aprendido com os dados.



Processo Análise de Dados



Insight

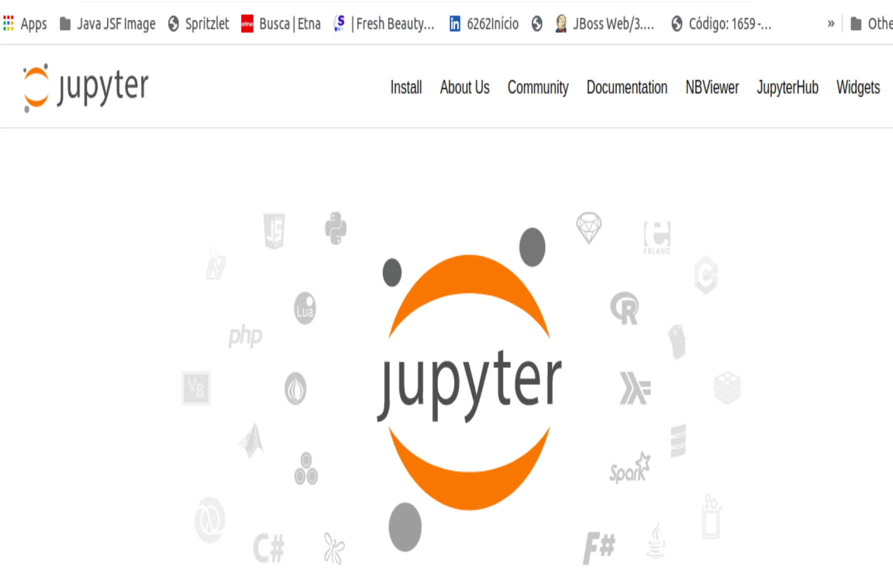


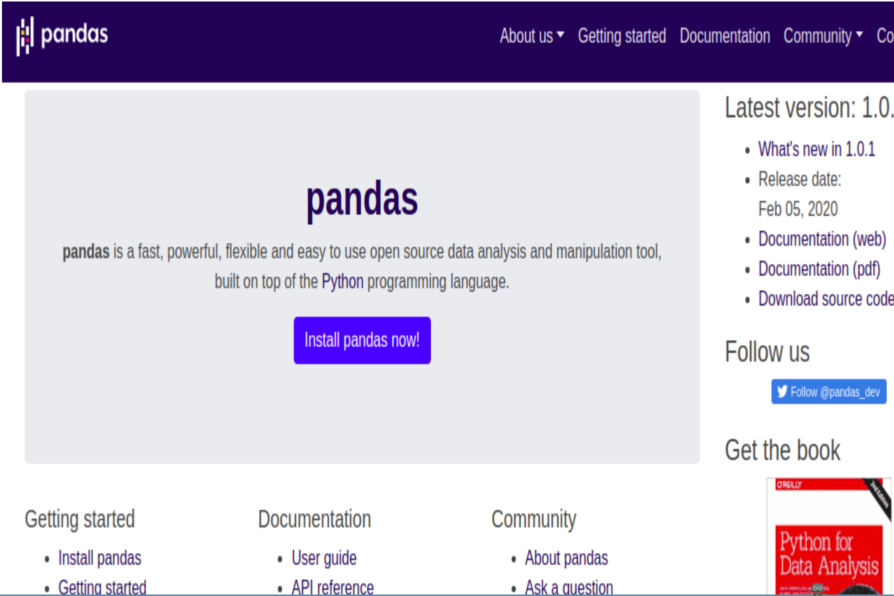
Models → Insights → Actions
A good insight drives action that will generate value for stakeholders
Models are built to answer business questions
e.g. what makes viewers tune-in and tune-out?
Data Scientists interpret models for answers
e.g. On screen arguments make viewers tune out
Email
Report
Presentation
Dashboard
BI Tool
Cloud App
End User
NABSHOW
Pivotal
etc

Tipos de Análise

RELATÓRIOS

- Notebooks online: iPython, Jupyter
- permitem a criação de documentos
- interativos em várias linguagens de análise.
- ¨ Reproducible Research!


Apps
Java JSF Image
Spritzlet
Busca | Etna
| Fresh Beauty...
6262Início
JBoss Web/3....
Código: 1659 -...
Othe
jupyter
Install
About Us
Community
Documentation
NBViewer
JupyterHub
Widgets
jupyter
php
Lua
Spark
C#
F#


pandas
About us
Getting started
Documentation
Community
Co
pandas
pandas is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language.
Install pandas now!
Latest version: 1.0.
What's new in 1.0.1
Release date:
Feb 05, 2020
Documentation (web)
Documentation (pdf)
Download source code
Follow us
Follow @pandas_dev
Get the book
O'REILLY
Python for Data Analysis
Getting started
Install pandas
Getting started
Documentation
User guide
API reference
Community
About pandas
Ask a question


BLOCK 01
NODE 01
NODE 04
NODE 05
NODE 03
BLOCK 01
NODE 01

About Citation Policy Donate a Da

Repository Web

View AL

UCI Machine Learning Repository

Center for Machine Learning and Intelligent Systems

Browse Through: **488** Data Sets

Table V

Default Task
Classification (360)
Regression (107)
Clustering (90)
Other (55)

Attribute Type
Categorical (38)
Numerical (325)
Mixed (55)

Data Type
Multivariate (374)
Univariate (24)
Sequential (51)
Time-Series (99)
Text (55)
Domain-Theory (23)

Name	Data Types	Default Task	Attribute Types	# Instances	# Attrib
Abalone	Multivariate	Classification	Categorical, Integer, Real	4177	8
Adult	Multivariate	Classification	Categorical, Integer	48842	14
Annealing	Multivariate	Classification	Categorical, Integer, Real	798	38
Anonymous Microsoft Web Data		Recommender-Systems	Categorical	37711	294

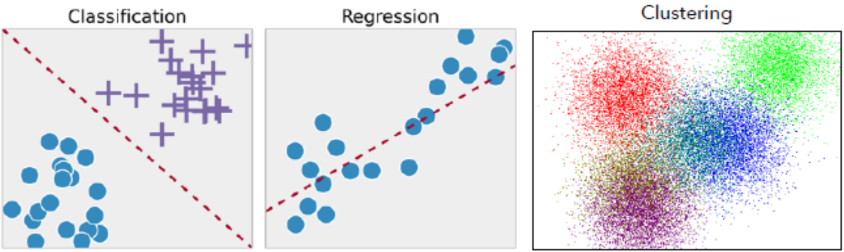
Data Mining Methods

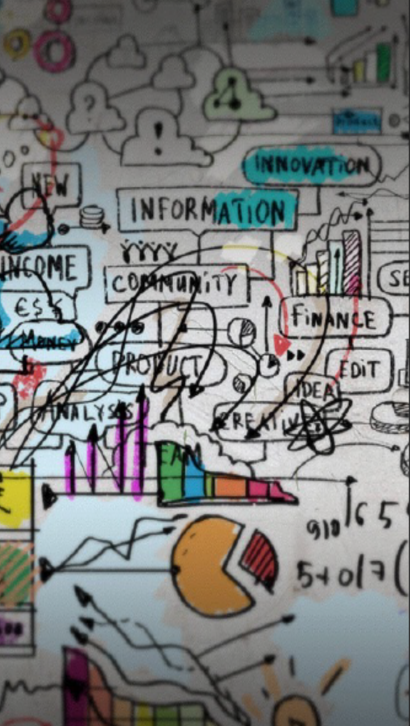
- **Descriptive methods**
  - Find human-interpretable patterns that describe the data
    - **Example:** Clustering
- **Predictive methods**
  - Use some variables to predict unknown or future values of other variables
    - **Example:** Recommender systems

http://web.stanford.edu/class/cs246/slides/01-intro.pdf

FORMULAÇÃO DE UM PROBLEMA

Identificação de uma área de interesse e o tipo de modelo



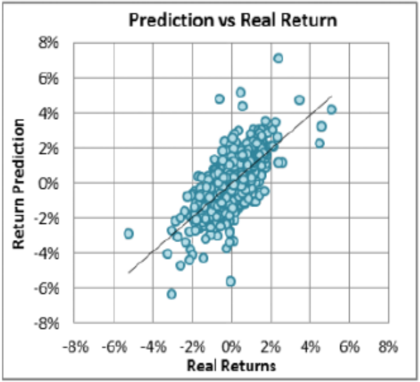


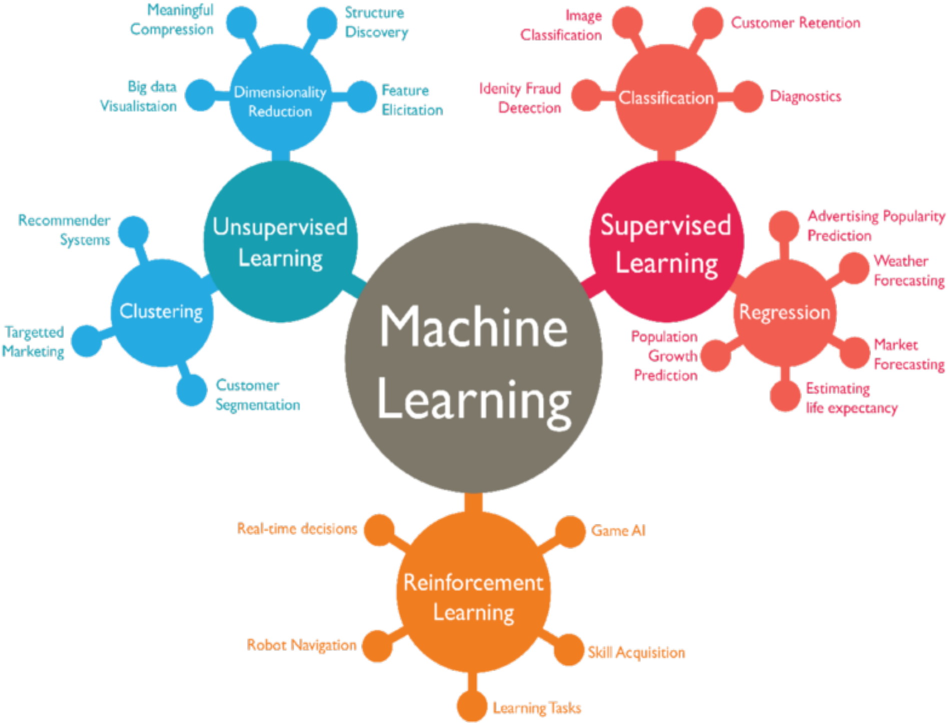
TIPOS DE MODELOS

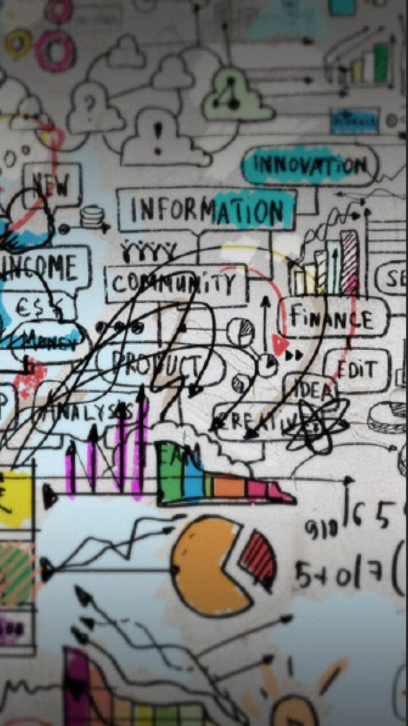
- REGRESSÃO
- CLASSIFICAÇÃO
- AGRUPAMENTO

CLASSIFICAÇÃO VS REGRESSÃO

Regression vs Classification



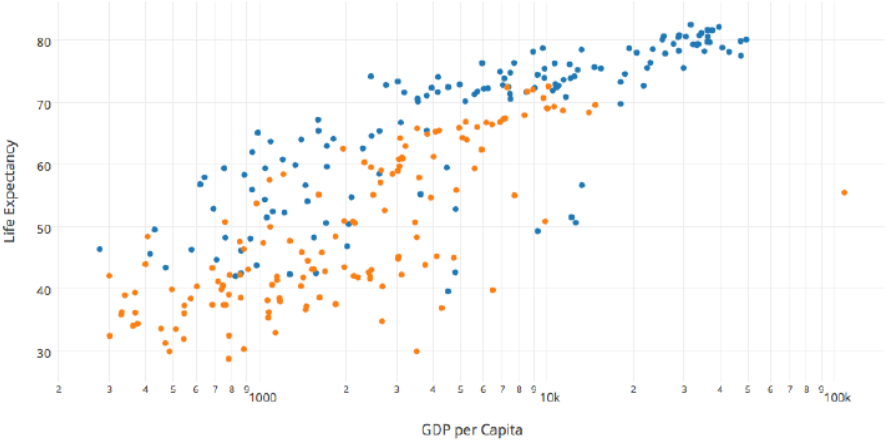

Machine Learning
Unsupervised Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Supervised Learning
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Reinforcement Learning
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks



O QUE É REGRESSÃO ?

REGRESSÃO

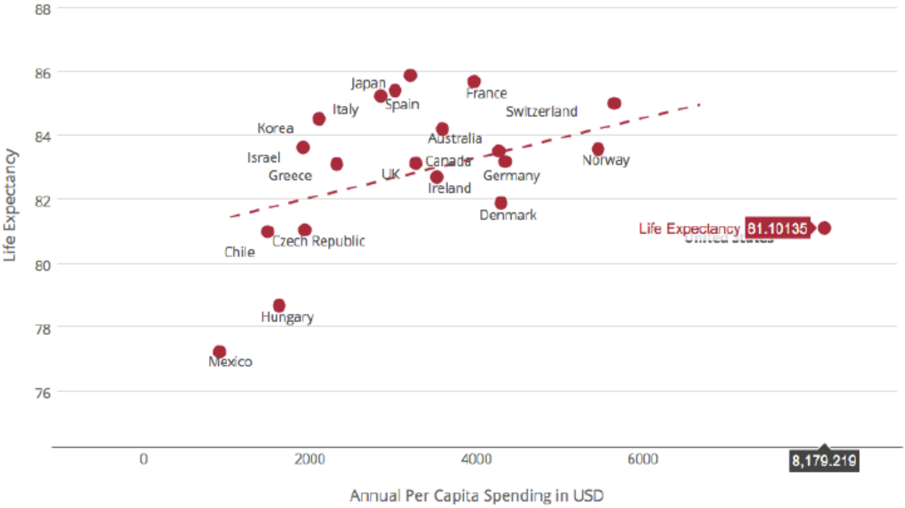
https://plot.ly/pandas/line-and-scatter/



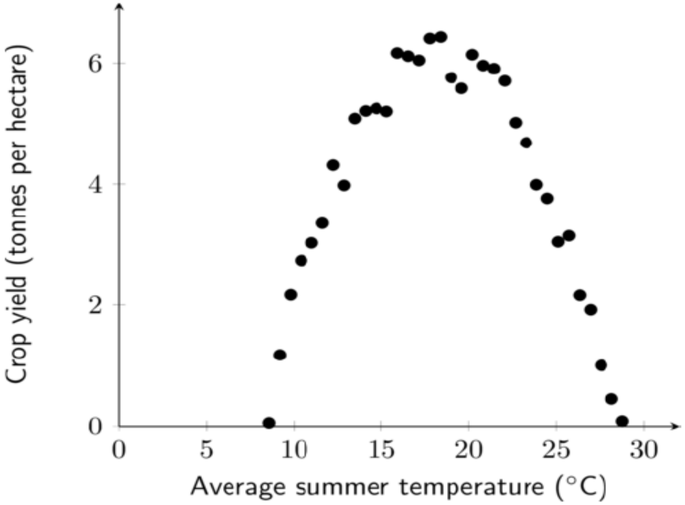
REGRESSÃO

https://plot.ly

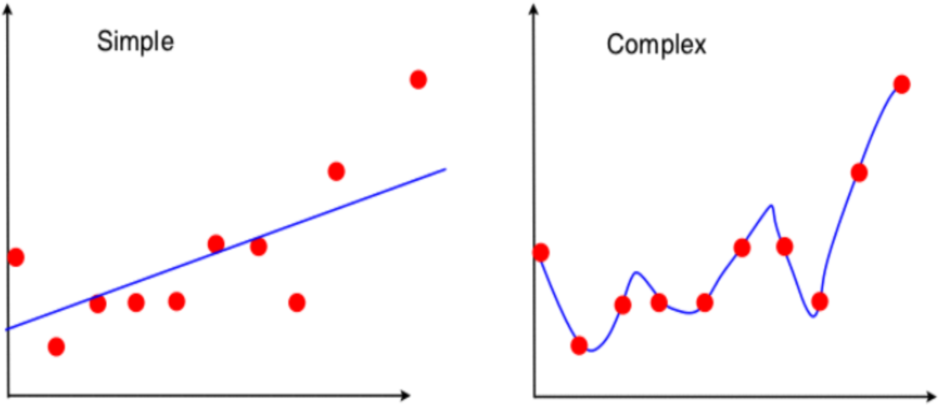
```
('Coefficients: \n', array([ 0.00118801]))
Mean squared error: 9.71
Variance score: -2.38
```



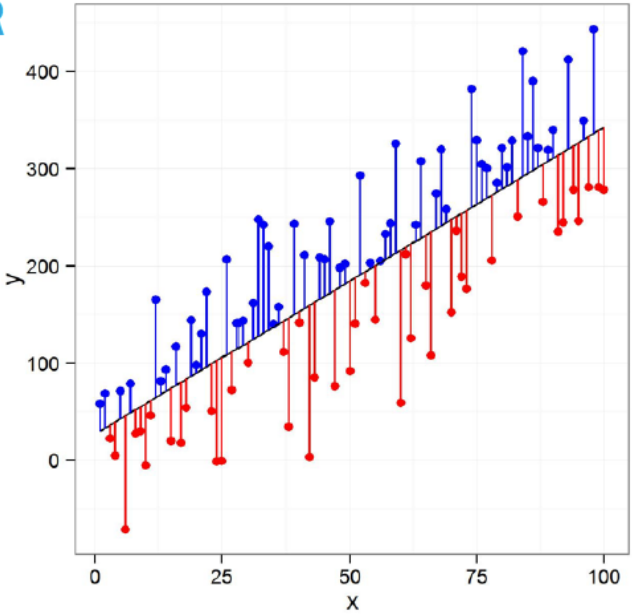
REGRESSÃO


Crop yield (tonnes per hectare)
Average summer temperature (°C)
0
2
4
6
0
5
10
15
20
25
30

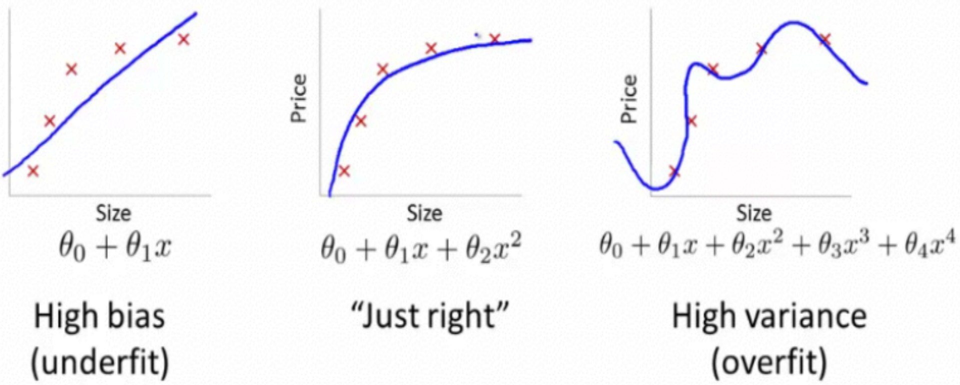
QUAL A MELHOR PREDIÇÃO



REGRESSION ERROR



BIAS E OVERFITTING



OCCAM'S RAZOR

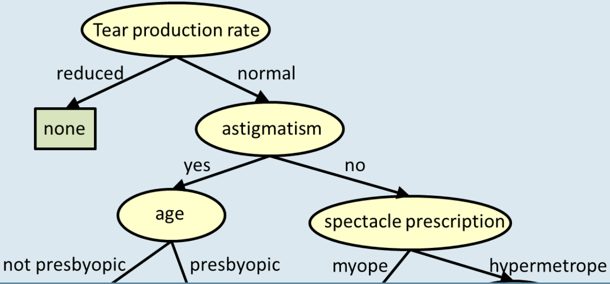
- ***Se os resultados forem semelhantes escolha a solução mais simples.***
- ***Em Data Science prefira sempre o modelo mais simples***

14th-century English logician William of Ockham

Modelos Paramétricos e Não Paramétricos

Modelos Paramétricos e Não Paramétricos

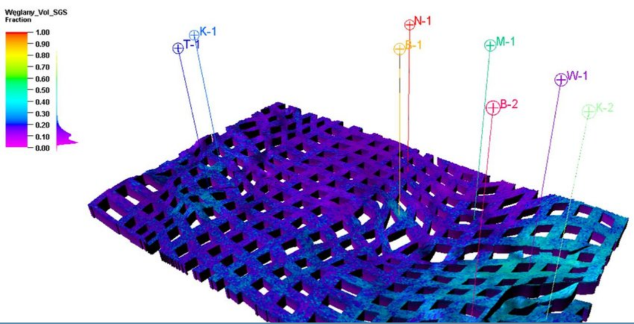
Algoritmos paramétricos de aprendizado de máquina simplificam o mapeamento para uma função funcional
Algoritmos não paramétricos podem aprender ”sozinhos”mapeamento de entradas para saídas. .



Modelos Paramétricos

Passos realizados por modelos paramétricos

Selecione a forma de uma função.
Aprenda os coeficientes para a função a partir dos dados de treinamento.



PROPHET

Docs

Documentation

- Installation
  - Using R
  - Using Python
- Quick Start
  - Python API
  - R API
- Saturating Forecasts
  - Forecasting Growth
  - Saturating Minimum
- Trend Changepoints
  - Automatic changepoint detection in Prophet
  - Adjusting trend flexibility
  - Specifying the locations of the changepoints

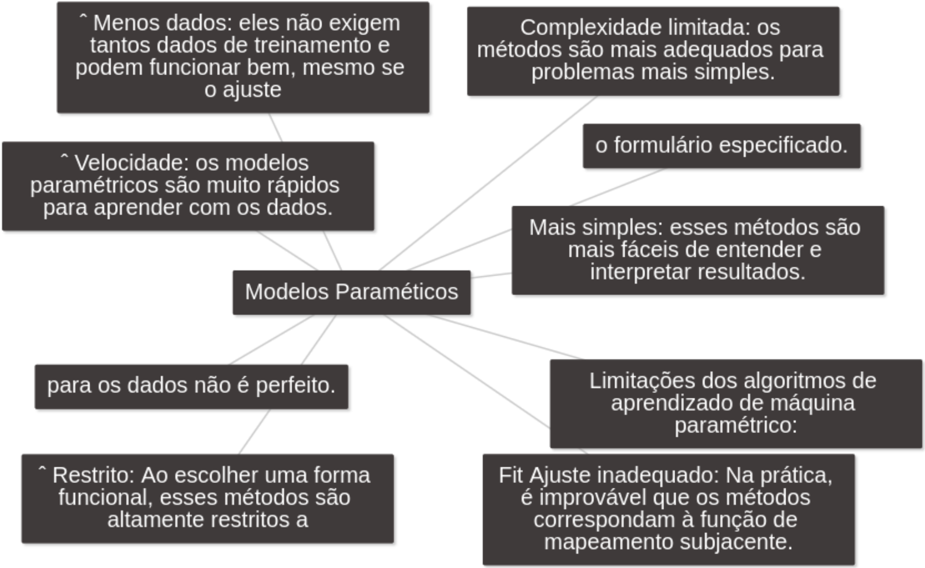
Quick Start

Python API

Prophet follows the `sklearn` model API. We create an instance of the `Prophet` class and then call it
and `predict` methods.

The input to Prophet is always a dataframe with two columns: `ds` and `y`. The `ds` (datestamp) column
should be of a format expected by Pandas, ideally YYYY-MM-DD for a date or YYYY-MM-DD HH:M
for a timestamp. The `y` column must be numeric, and represents the measurement we wish to foreca

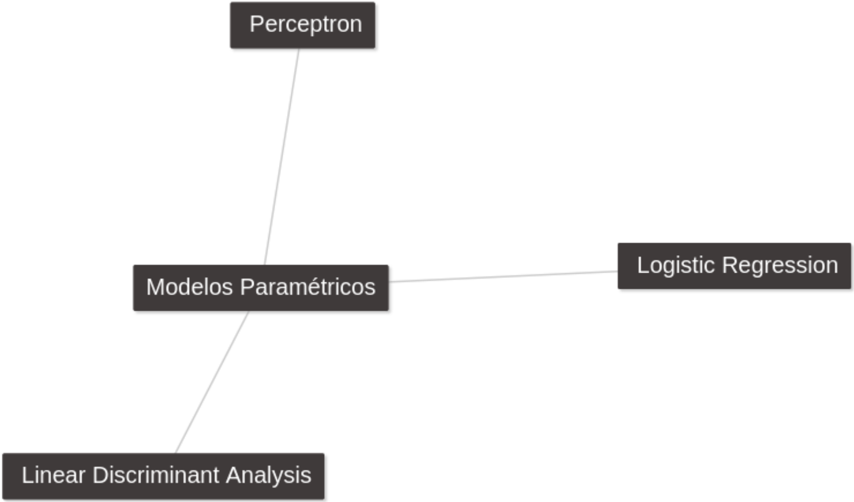
As an example, let's look at a time series of the log daily page views for the Wikipedia page for Peyt


Modelos Paraméticos
ˆ Menos dados: eles não exigem tantos dados de treinamento e podem funcionar bem, mesmo se o ajuste
ˆ Velocidade: os modelos paramétricos são muito rápidos para aprender com os dados.
para os dados não é perfeito.
ˆ Restrito: Ao escolher uma forma funcional, esses métodos são altamente restritos a
Complexidade limitada: os métodos são mais adequados para problemas mais simples.
o formulário especificado.
Mais simples: esses métodos são mais fáceis de entender e interpretar resultados.
Limitações dos algoritmos de aprendizado de máquina paramétrico:
Fit Ajuste inadequado: Na prática, é improvável que os métodos correspondam à função de mapeamento subjacente.

> *A learning model that summarizes data with a set of parameters of fixed size (independent of the number of training examples) is called a parametric model. No matter how much data you throw at a parametric model, it won't change its mind about how many parameters it needs.*

— Artificial Intelligence: A Modern Approach, page 737

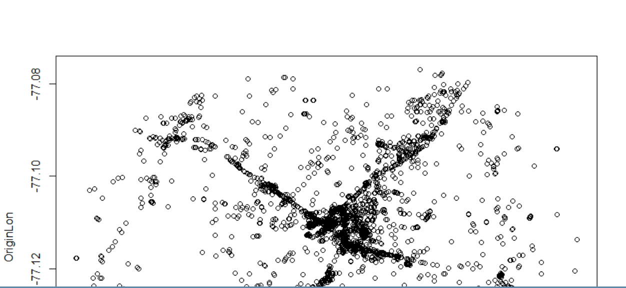
Parametric model	Non-parametric model
It uses a fixed number of parameters to build the model.	It uses flexible number of parameters to build the model.
Considers strong assumptions about the data.	Considers fewer assumptions about the data.
Computationally faster	Computationally slower
Require lesser data	Require more data
Example – Logistic Regression & Naïve Bayes models	Example – KNN & Decision Tree models

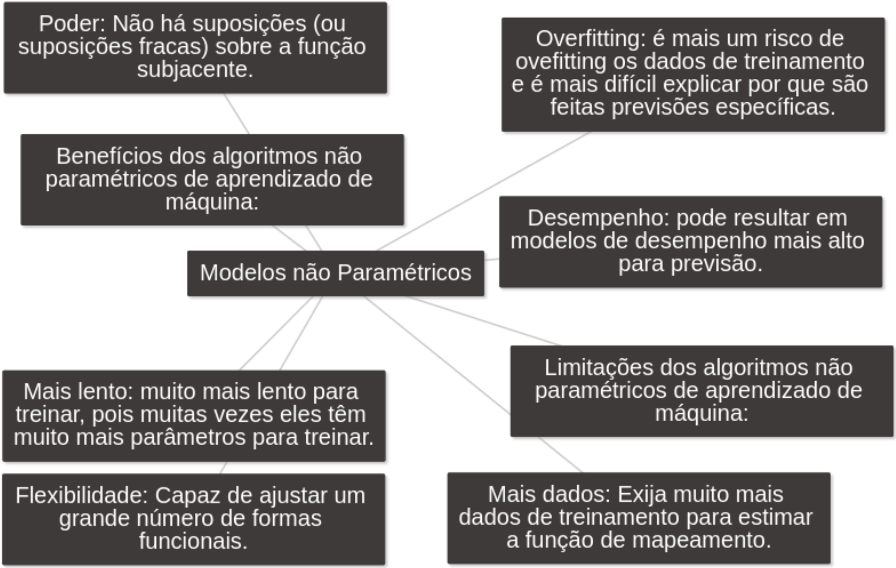

Perceptron
Modelos Paramétricos
Logistic Regression
Linear Discriminant Analysis

Modelos Não Paramétricos

Modelos não Paramétricos

Métodos não paramétricos buscam melhor ajustar os dados de treinamento na construção do mapeamento , mantendo alguma capacidade de generalização para dados não vistos.




Poder: Não há suposições (ou suposições fracas) sobre a função subjacente.
Overfitting: é mais um risco de ovefitting os dados de treinamento e é mais difícil explicar por que são feitas previsões específicas.
Benefícios dos algoritmos não paramétricos de aprendizado de máquina:
Desempenho: pode resultar em modelos de desempenho mais alto para previsão.
Modelos não Paramétricos
Limitações dos algoritmos não paramétricos de aprendizado de máquina:
Mais lento: muito mais lento para treinar, pois muitas vezes eles têm muito mais parâmetros para treinar.
Flexibilidade: Capaz de ajustar um grande número de formas funcionais.
Mais dados: Exija muito mais dados de treinamento para estimar a função de mapeamento.

EXEMPLO DE PROBLEMA - ÁREA (BIOLOGIA) - TIPO: REGRESSÃO

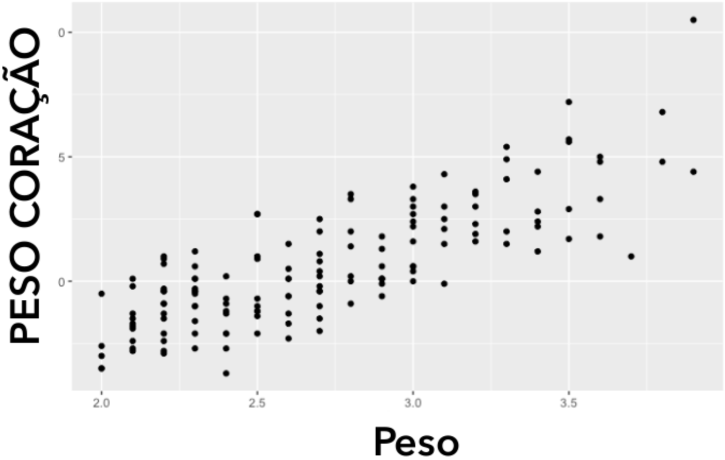
ID	SEXO	CORAÇÃO	PESO
1	F	2.0	7.0
2	F	2.0	7.4
3	F	2.0	9.5
4	F	2.1	7.2
5	F	2.1	7.3
6	F	2.1	7.6
7	F	2.1	8.1
8	F	2.1	8.2

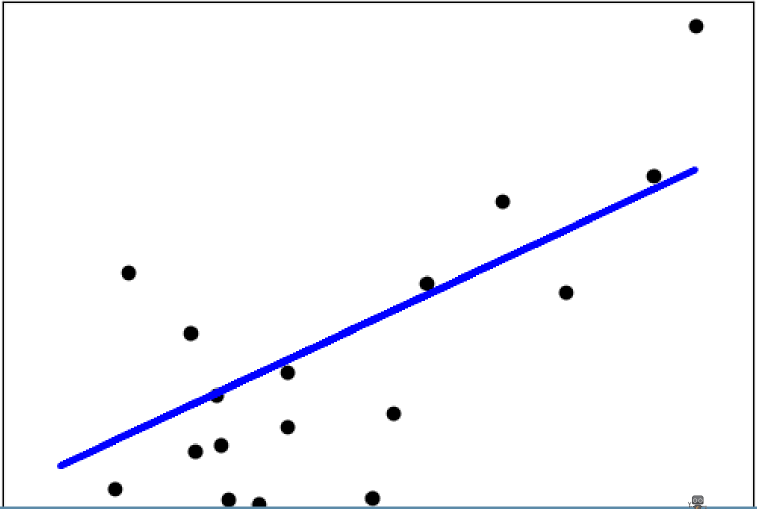
library("MASS")

data(cats)

R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS. To download R, please choose your preferred CRAN mirror.

EXEMPLO DE PROBLEMA – ÁREA (BIOLOGIA) – TIPO: REGRES





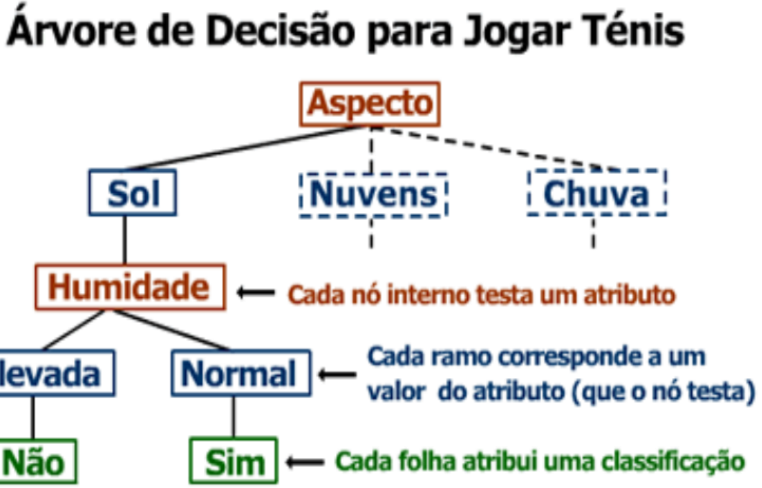

EXEMPLO
PREDIZER
PREÇOS DE
CASAS EM
BOSTON

EXEMPLO DE PROBLEMA – PREÇO DE IMÓVEIS EM BOSTON

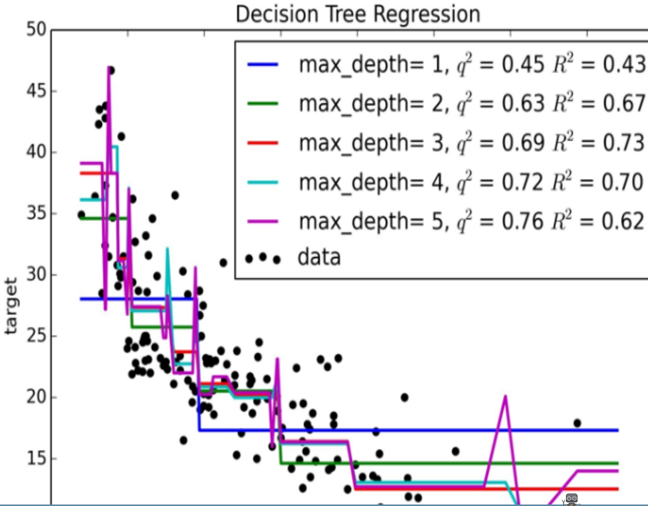
- `RM'` -Média do número de quartos
- `'LSTAT'` percentual de proprietários considerados "lower class" (working poor).
- `'PTRATIO'` razão do número de estudantes por professor no bairro

RM	LMRATIO	PTRATIO	PREÇO
6.575	4.98	15.3	504000
6.421	9.14	17.8	453600
7.185	4.03	17.8	728700
6.998	2.94	18.7	701400
7.147	5.33	18.7	760200
6.43	5.21	18.7	602700
6.012	12.43	15.2	480900
6.172	19.15	15.2	569100
5.631	29.93	15.2	346500
6.004	17.1	15.2	396900
6.377	20.45	15.2	315000
6.009	13.27	15.2	396900
5.889	15.71	15.2	455700
5.949	8.26	21	428400
6.096	10.26	21	382200

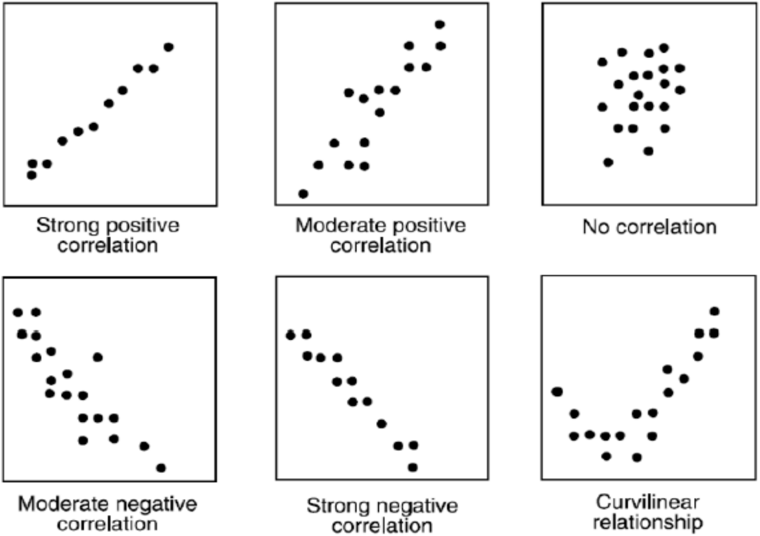
ARVORES DE DECISÃO

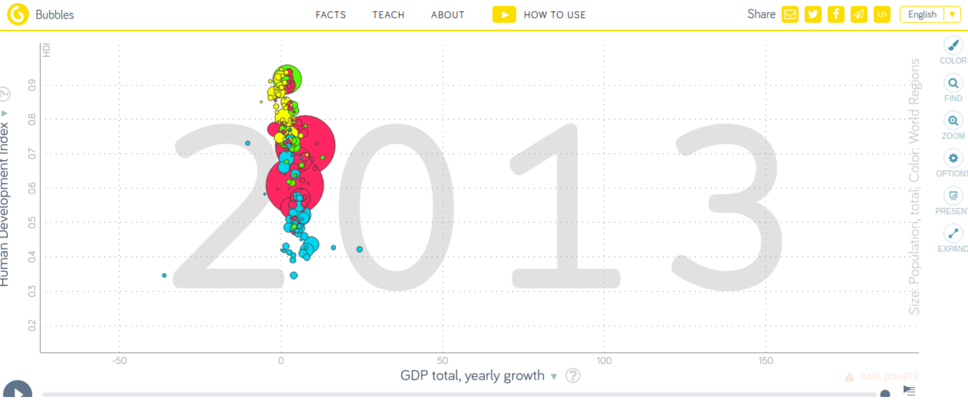


PREÇOS DAS CASAS EM BOSTON

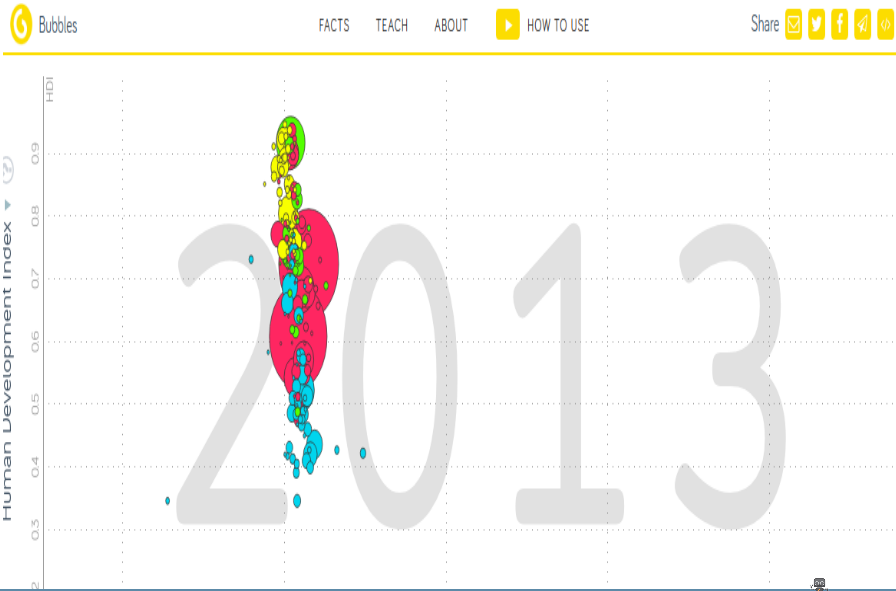


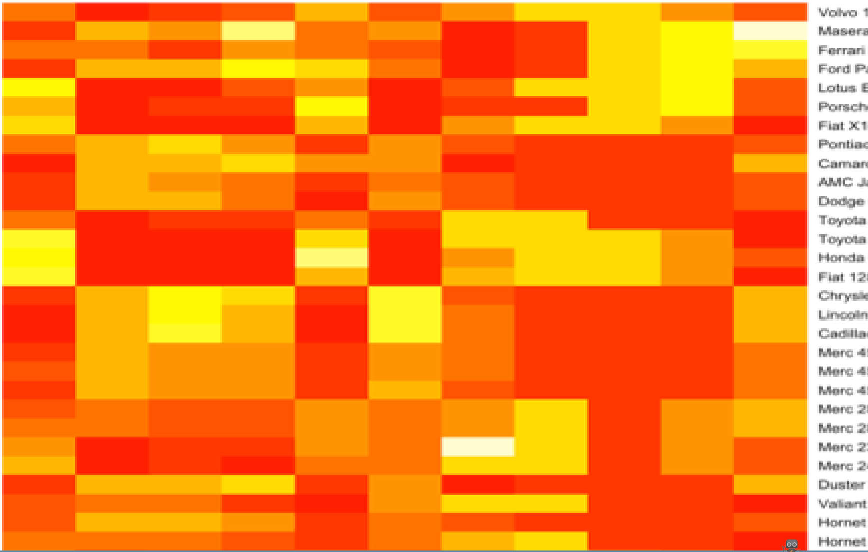
CORRELAÇÃO DOS DADOS





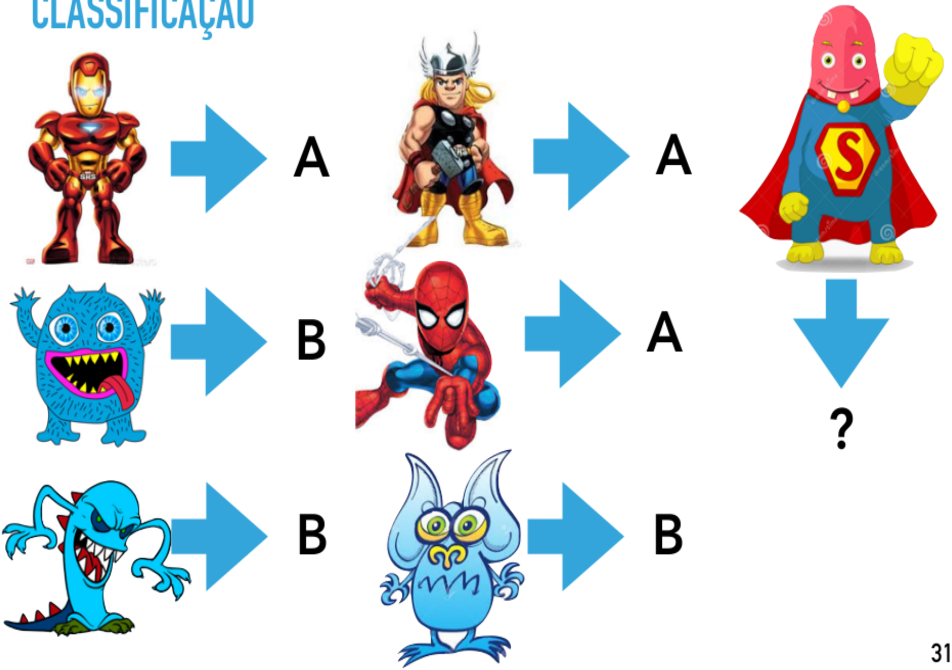
Bubbles
FACTS
TEACH
ABOUT
HOW TO USE
Share
English
HDI
Human Development Index
2013
GDP total, yearly growth
-50
0
50
100
150
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
Size: Population, total, Color: World Regions
COLOR
FIND
ZOOM
OPTIONS
PRESENT
EXPAND
DATA DOUBTS


Bubbles
FACTS
TEACH
ABOUT
HOW TO USE
Share
HDI
Human Development Index
0.9
0.8
0.7
0.6
0.5
0.4
0.3
2013

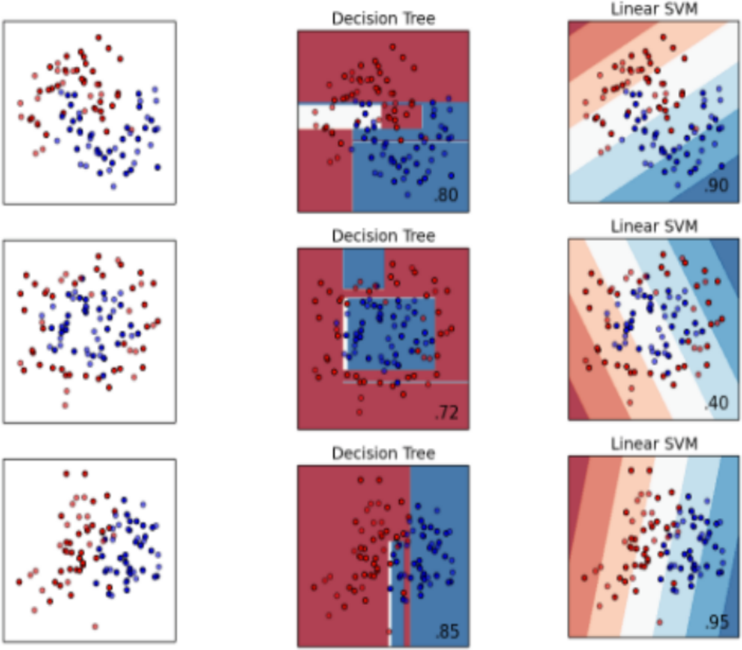

Volvo 1
Masera
Ferrari
Ford P
Lotus E
Porsch
Fiat X1
Pontiac
Camar
AMC J
Dodge
Toyota
Toyota
Honda
Fiat 12
Chrysle
Lincoln
Cadillac
Merc 4
Merc 4
Merc 4
Merc 2
Merc 2
Merc 2
Merc 2
Duster
Valiant
Hornet
Hornet

O QUE É CLASSIFICAÇÃO ?

CLASSIFICAÇÃO



CLASSIFICADORES

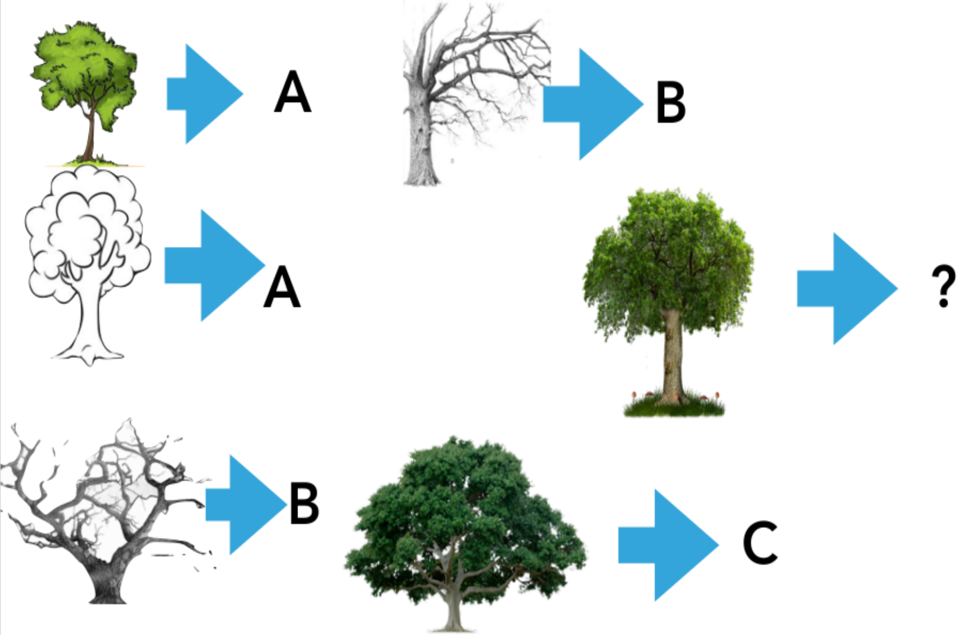


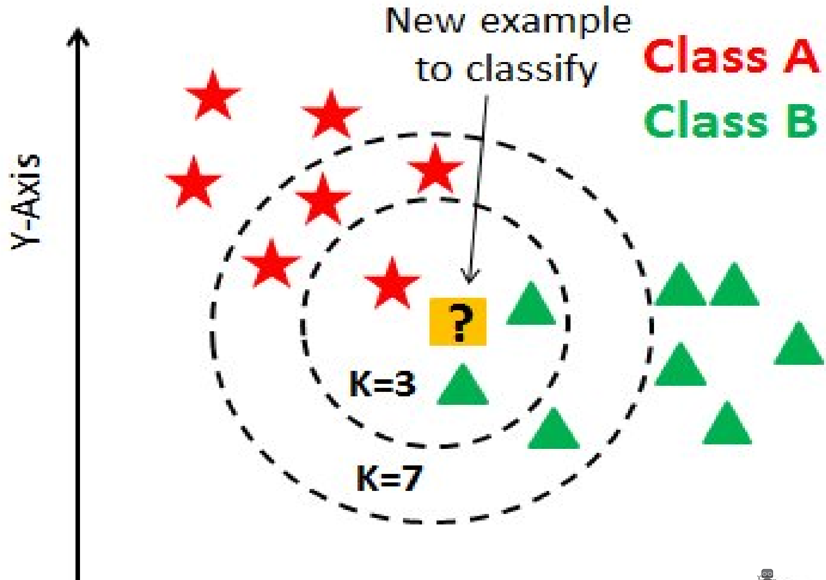
QUAL A TAG A APLICAR?

?

?

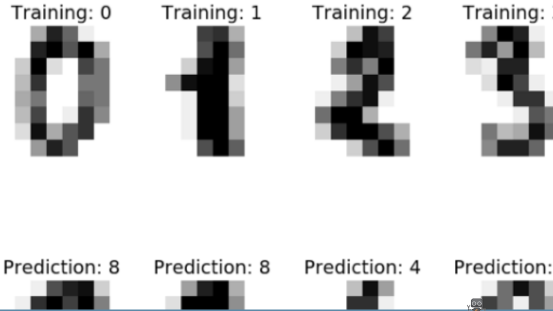
QUAL A TAG A APLICAR?

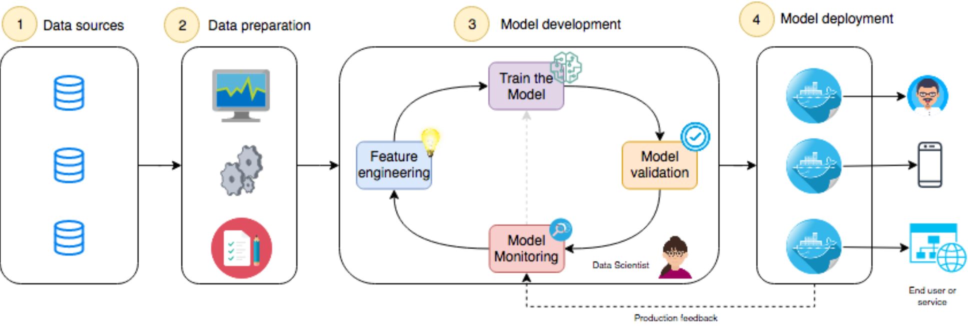


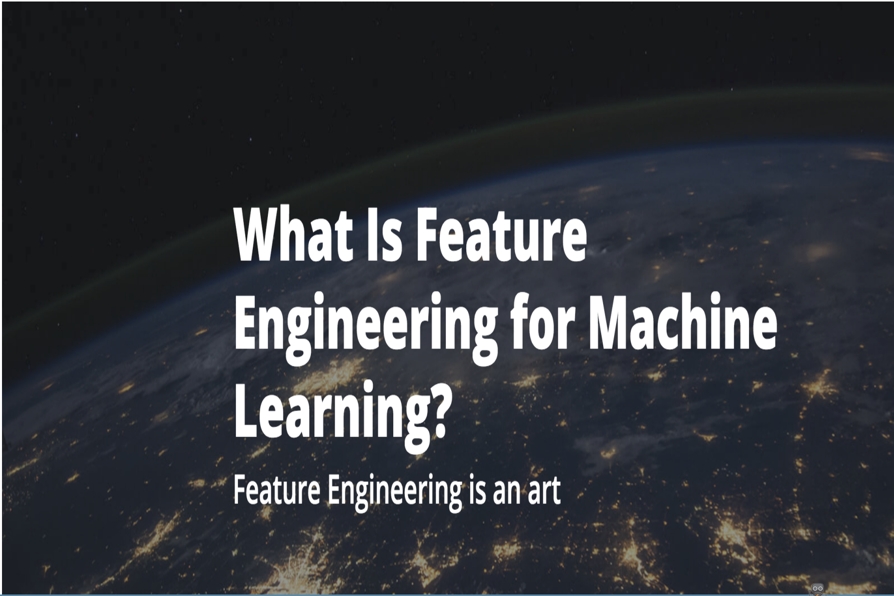

New example
to classify
Class A
Class B
Y-Axis
?
K=3
K=7

CLASSIFICAÇÃO NO SKLEARN

```
# Create a classifier: a support vector classifier
classifier = svm.SVC(gamma=0.001)

# We learn the digits on the first half of the digits
classifier.fit(data[:n_samples / 2], digits.target[:n_sample
2])
```




1
Data sources
2
Data preparation
3
Model development
4
Model deployment
Train the Model
Feature engineering
Model validation
Model Monitoring
Data Scientist
End user or service
Production feedback


What Is Feature Engineering for Machine Learning?
Feature Engineering is an art

EXEMPLO INTERVENÇÃO DE ESTUDANTES

INTERVENÇÃO DE ESTUDANTES

```
Feature values:
  school sex  age address famsize Pstatus  Medu  Fedu     Mjob      Fjob  \
0     GP   F   18       U     GT3       A     4     4  at_home   teacher
1     GP   F   17       U     GT3       T     1     1  at_home     other
2     GP   F   15       U     LE3       T     1     1  at_home     other
3     GP   F   15       U     GT3       T     4     2   health  services
4     GP   F   16       U     GT3       T     3     3    other     other

    ...    higher internet  romantic  famrel  freetime goout Dalc Walc health  \
0   ...       yes       no        no       4         3     4    1    1      3
1   ...       yes      yes        no       5         3     3    1    1      3
2   ...       yes      yes        no       4         3     2    2    3      3
3   ...       yes      yes       yes       3         2     2    1    1      5
4   ...       yes       no        no       4         3     2    1    2      5

  absences
0        6
1        4
2       10
3        2
4        4
```

PREPARANDO OS DADOS

```
# If data type is non-numeric, replace all yes/no values with 1/0
  if col_data.dtype == object:
      col_data = col_data.replace(['yes', 'no'], [1, 0])
```

SEPARANDO DADOS DE TREINO E TESTE

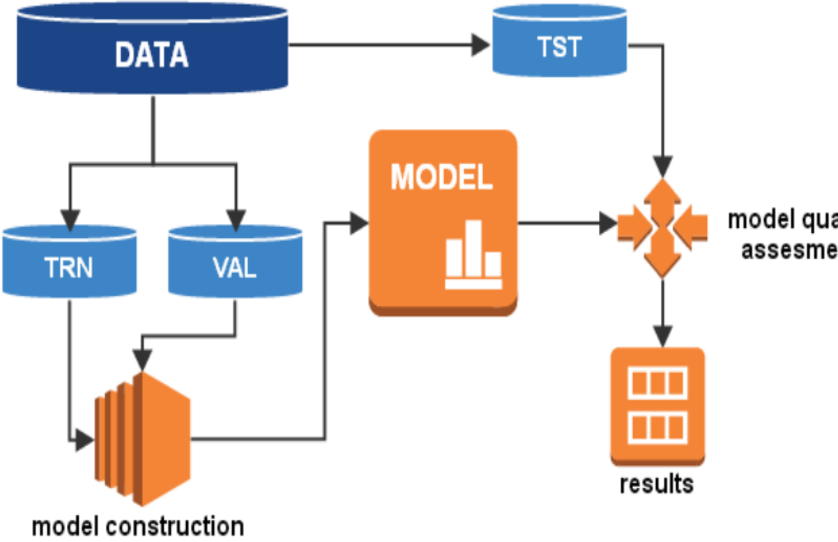
```
X_train, X_test, y_train, y_test = train_test_split(X_all, y_all,
stratify=y_all, train_size=train_size,test_size=0.24)
```

DADOS DE TREINO VS DADOS DE TESTE

- ***Dados de Treino***
  - Usados para treinar um modelo
  - Exemplos

- ***Dados de Teste***
  - Usados para testar a performance do modelo
  - Dados de validação.

e.g. facial gender classification

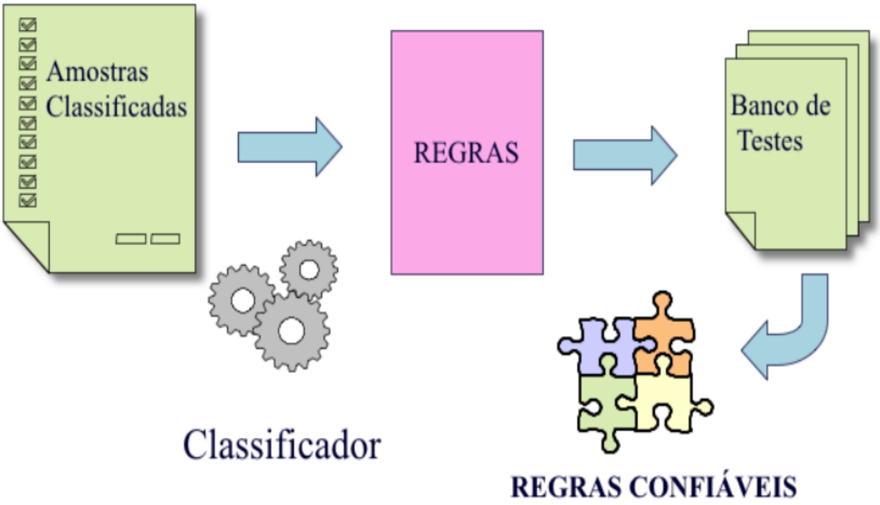

DATA
TST
MODEL
TRN
VAL
model qua
asseme
results
model construction

Classificação

Nome	Idade	Renda	Profissão	Classe
Daniel	≤ 30	Média	Estudante	Sim
João	31..50	Média-Alta	Professor	Sim
Carlos	31..50	Média-Alta	Engenheiro	Sim
Maria	31..50	Baixa	Vendedora	Não
Paulo	≤ 30	Baixa	Porteiro	Não
Otavio	> 60	Média-Alta	Aposentado	Não

SE. Idade ≤ 30 E Renda é Média ENTÃO Compra-Produto-Eletrônico = SIM.

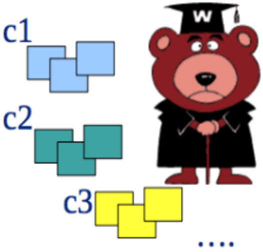
Etapas do Processo



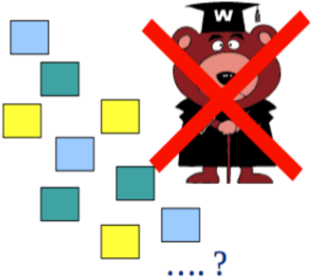
APRENDIZADO SUPERVISIONADO VS NÃO SUPERVISIONADO

APRENDIZADO SUPERVISIONADO VS NÃO SUPERVISIONADO

- ***Supervisionado***
  - Conhecimento das entradas e saídas de dados
  - Os dados possuem um label
  - O objetivo é predizer a classe ou o label do dado



- ***Não Supervisionado***
  - Sem conhecimento prévio dos dados
  - O objetivo é determinar padrões



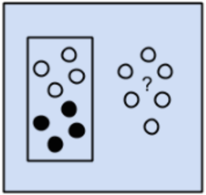
1. Supervised Learning

Input data is called training data and has a known label or result such as spam/not-spam or a stock price at a time.

A model is prepared through a training process in which it is required to make predictions and is corrected when those predictions are wrong. The training process continues until the model achieves a desired level of accuracy on the training data.

Example problems are classification and regression.

Example algorithms include: Logistic Regression and the Back Propagation Neural Network.



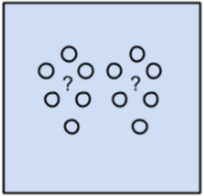
Supervised Learning Algorithms

2. Unsupervised Learning

Input data is not labeled and does not have a known result.

A model is prepared by deducing structures present in the input data. This may be to extract general rules. It may be through a mathematical process to systematically reduce redundancy, or it may be to organize data by similarity.

Example problems are clustering, dimensionality reduction and association rule learning.

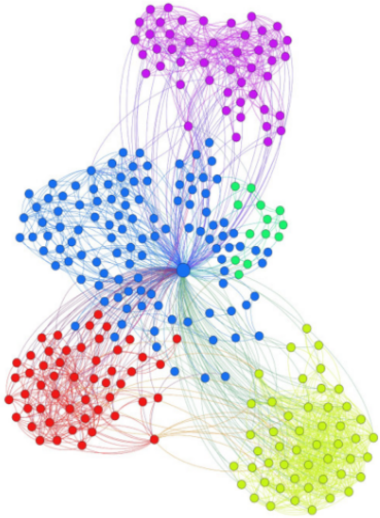


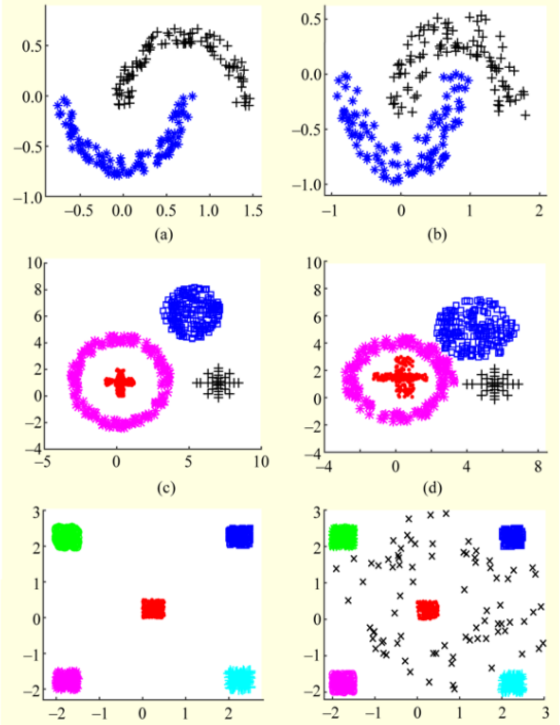
Unsupervised Learning Algorithms

Example algorithms include: the Apriori algorithm and K-Means.

https://machinelearningmastery.com/a-tour-of-machine-learning-algorithms/

O QUE É AGRUPAMENTO ?



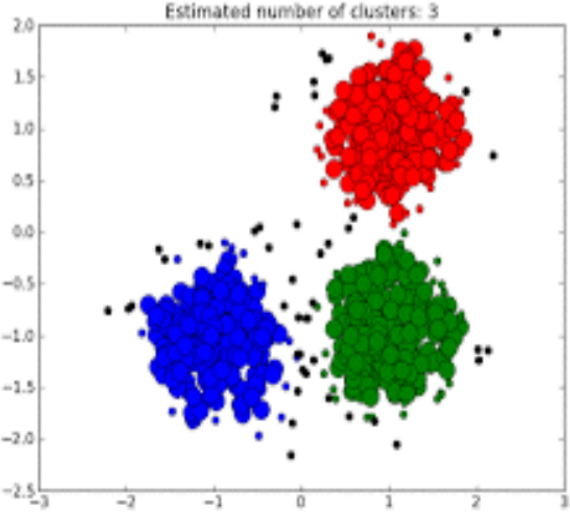

(a)
(b)
(c)
(d)

AGGLOMERATIVE
CLUSTERING

All observations start as their own cluster. Clusters meeting some criteria are merged. This process is repeated, growing clusters until some end point is reached.

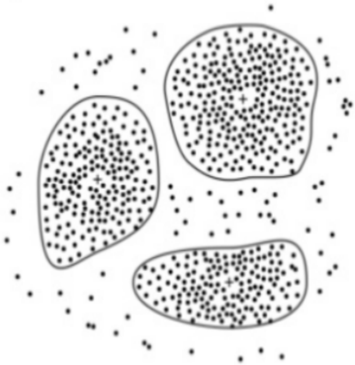
AGRUPAMENTO

Processo de agrupar objetos com características semelhantes



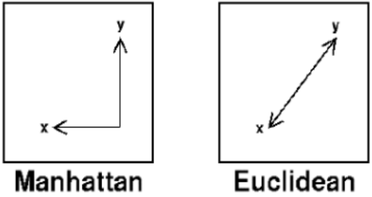
Cluster

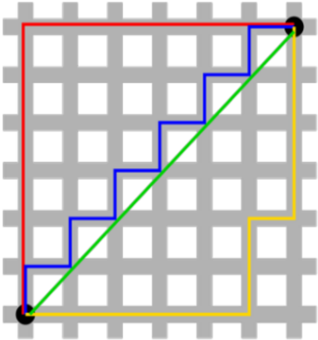
- Uma coleção de objetos que são similares entre si, e diferentes dos objetos pertencentes a outros clusters.



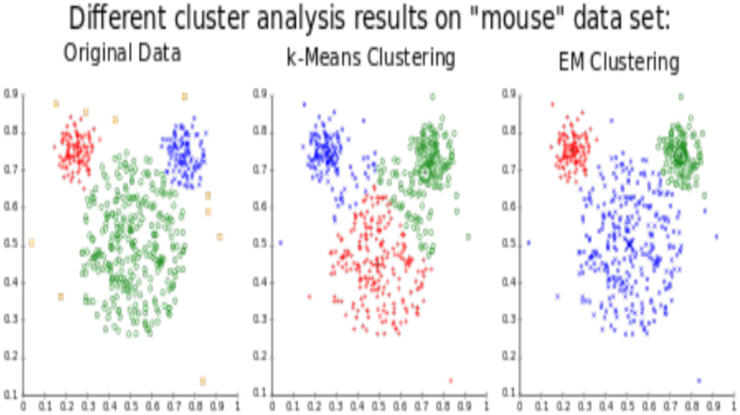
COMO AGRUPAR?

DISTÂNCIA

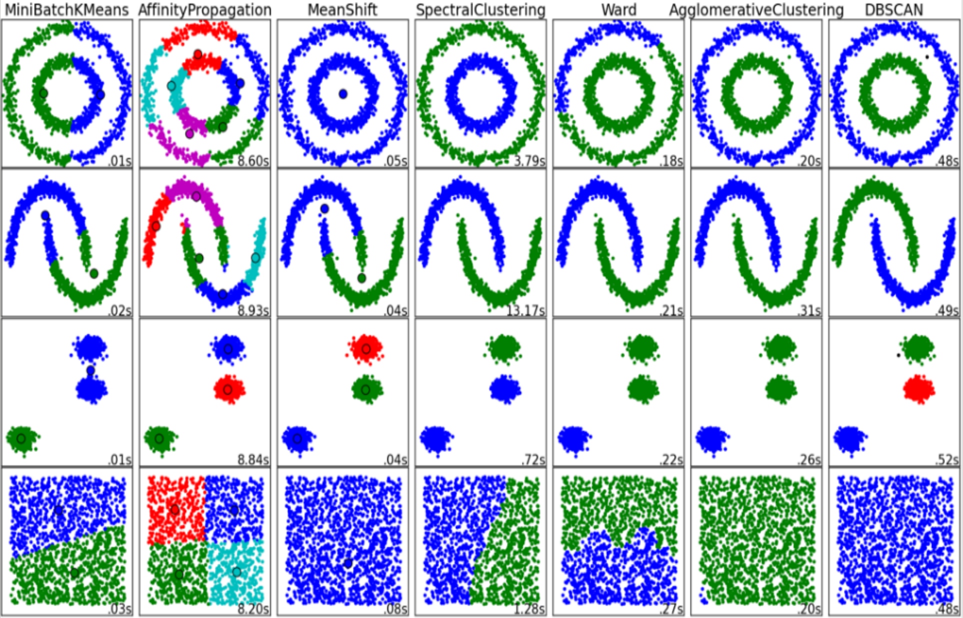




MODELOS DE AGRUPAMENTO



ALGORITMOS DE AGRUPAMENTO DO SKLEARN




CUSTOMER
SEGMENTANDO
FORNECEDORE

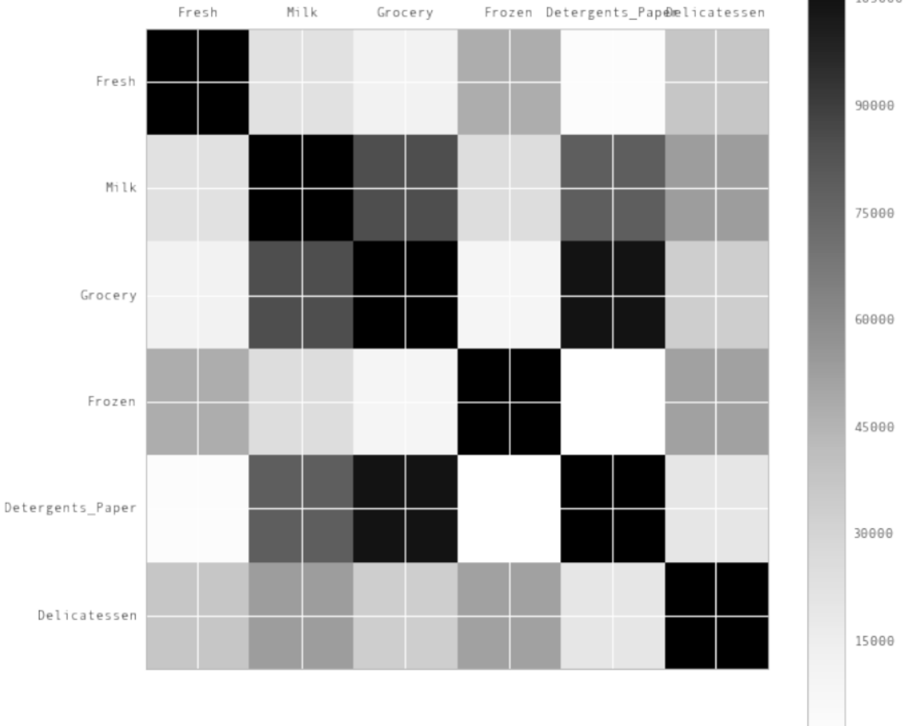
BANCO DE DADOS DE PRODUTOS

	Fresh	Milk	Grocery	Frozen	Detergents_Paper	Delicatessen
count	440.000000	440.000000	440.000000	440.000000	440.000000	440.000000
mean	12000.297727	5796.265909	7951.277273	3071.931818	2881.493182	1524.870455
std	12647.328865	7380.377175	9503.162829	4854.673333	4767.854448	2820.105937
min	3.000000	55.000000	3.000000	25.000000	3.000000	3.000000
25%	3127.750000	1533.000000	2153.000000	742.250000	256.750000	408.250000
50%	8504.000000	3627.000000	4755.500000	1526.000000	816.500000	965.500000
75%	16933.750000	7190.250000	10655.750000	3554.250000	3922.000000	1820.250000
max	112151.000000	73498.000000	92780.000000	60869.000000	40827.000000	47943.000000

VERIFICANDO CORRELACIONAMENTO

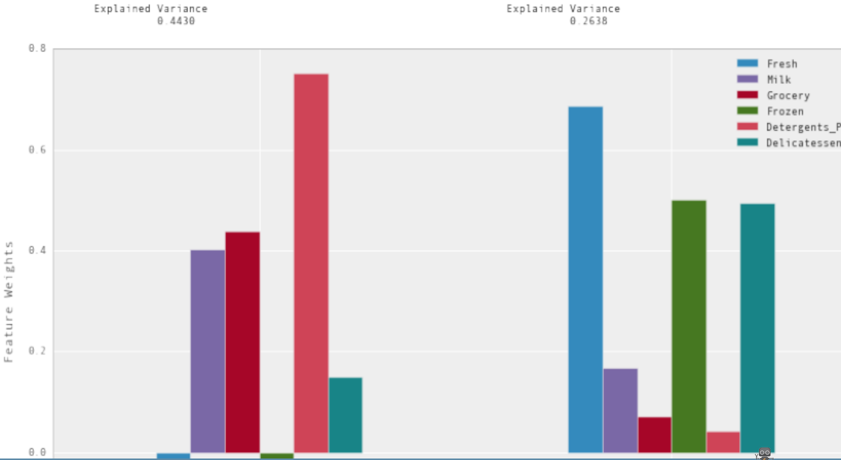
```
for product in products:

    y_array=data[product]
    new_data = data.drop([product],axis=1)


    X_train, X_test, y_train, y_test = train_test_split( new_data, y_array, test_size=0.25, random_state

    for reg in regressors:
        reg.fit(X_train,y_train)
        score = reg.score(X_test, y_test)
        print('Product '+product+' Score is '+ str(score))
        parameters=regressors_parameters.get(reg)
        clf = GridSearchCV(reg, parameters)
        clf.fit(X_train,y_train)
        score = clf.score(X_test, y_test)
        print('Product '+product+' Score is '+ str(score)+' after GridSearchCV ')
```

```
Product Fresh Score is -0.333070533605
Product Fresh Score is -0.329449950604 after GridSearchCV
Product Milk Score is 0.173438009379
Product Milk Score is 0.205871721893 after GridSearchCV
Product Grocery Score is 0.699248196675
Product Grocery Score is 0.699248196675 after GridSearchCV
Product Detergents_Paper Score is 0.348777454691
Product Detergents_Paper Score is 0.348777454691 after GridSearchCV
Product Frozen Score is -0.278249148824
Product Frozen Score is -1.30732144534 after GridSearchCV
Product Delicatessen Score is -11.0236279005
Product Delicatessen Score is -9.55743305081 after GridSearchCV
```



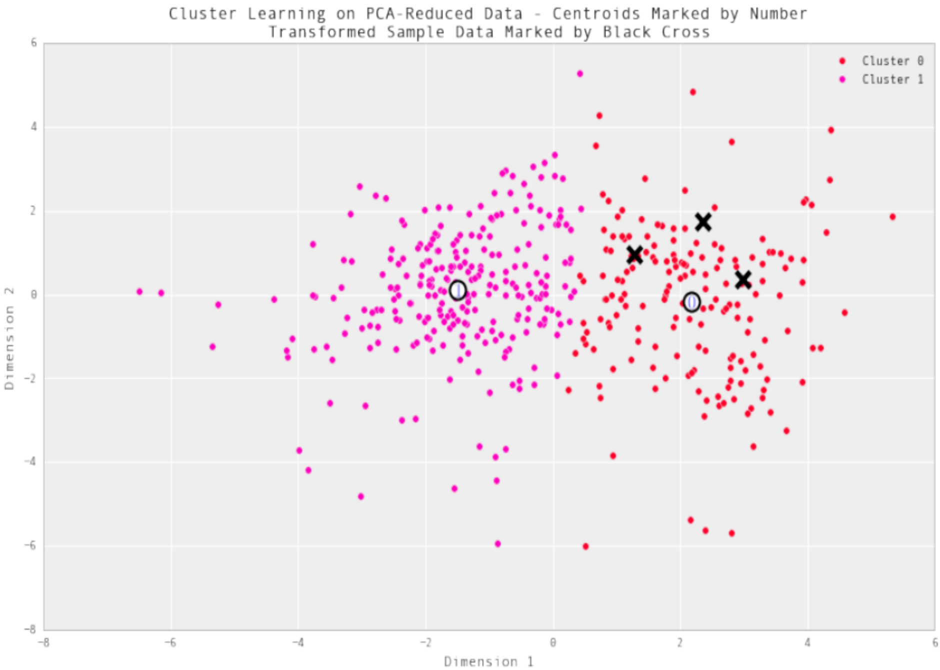
Fresh
Milk
Grocery
Frozen
Detergents_Paper
Delicatessen
Fresh
Milk
Grocery
Frozen
Detergents_Paper
Delicatessen
90000
75000
60000
45000
30000
15000

PCA –REDUZINDO DIMENSÕES



GERANDO OS CLUSTERS

```
clusterer = KMeans(n_clusters=i,
random_state=29).fit(reduced_data)

preds = clusterer.predict(reduced_data)
```

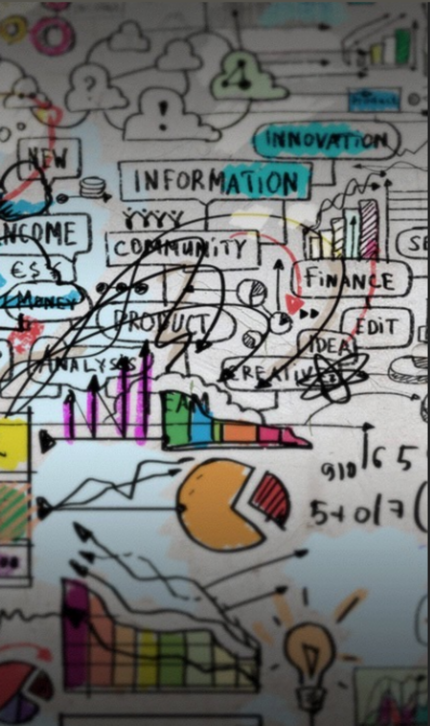

Cluster Learning on PCA-Reduced Data - Centroids Marked by Number
Transformed Sample Data Marked by Black Cross
Cluster 0
Cluster 1
Dimension 1
Dimension 2
6
4
2
0
-2
-4
-6
-8
-8
-6
-4
-2
0
2
4
6
1
0

	Predicted 0	Predicted 1
Actual 0	TN	FP
Actual 1	FN	TP

F1 SCORE

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

F1 score is the harmonic mean of precision and recall. Values range from 0 (bad) to 1 (good).



ANALISANDO A PERFORMANCE DE UM MODELO

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

Outras medidas mais precisas

- **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 classificado como "bom classificador" mesmo com acurácia alta ?

- **Sensitividade** $= \dfrac{\text{true-pos}}{\text{pos}}$

% pacientes classificados **corretamente** **com câncer** dentre todos os que **realmente tem câncer**

- **Especificidade** $= \dfrac{\text{true-neg}}{\text{neg}}$

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

% pacientes classificados **corretamente** **com câncer** dentre todos os que foram classificados **com câncer**

PERFORMANCE DE UM CLASSIFICADOR

- Acuracia = classificados corretamente /total de exemplos
- Erro = 1-Acuracia

PERFORMANCE DE UMA REGRESSÃO

Uma das formas de avaliar a qualidade do ajuste do modelo é através do coeficiente de determinação. Basicamente, este coeficiente indica quanto o modelo foi capaz de explicar os dados coletados. O coeficiente de determinação é dado pela expressão

$$R^2 = \frac{SQR}{SQT} = 1 - \frac{SQE}{SQT} = \frac{\widehat{\beta}_1 \sum_{i=1}^{n}(x_i - \bar{x})Y_i}{\sum_{i=1}^{n}(Y_i - \bar{Y})^2},$$

Razão entre a soma de quadrados da regressão e a soma de quadrados total.

$$0 \leq R^2 \leq 1.$$

TABELA CONFUSÃO

		Classe real	
		p	n
Classe predita	p	Verdadeiro Positivo	Falso Positivo
	n	Falso Negativo	Verdadeiro Negativo

MEAN SQUARE ERROR

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \tilde{y}_i\right)^2$$

FERRAMENTAS E LINGUAGENS

- NUMPY
- PANDAS
- SKLEARN
- R STUDIO

NUMPY

Biblioteca em python para manipulação de arrays e matrizes

PANDAS

Biblioteca de Manipulação de dados e analise em python

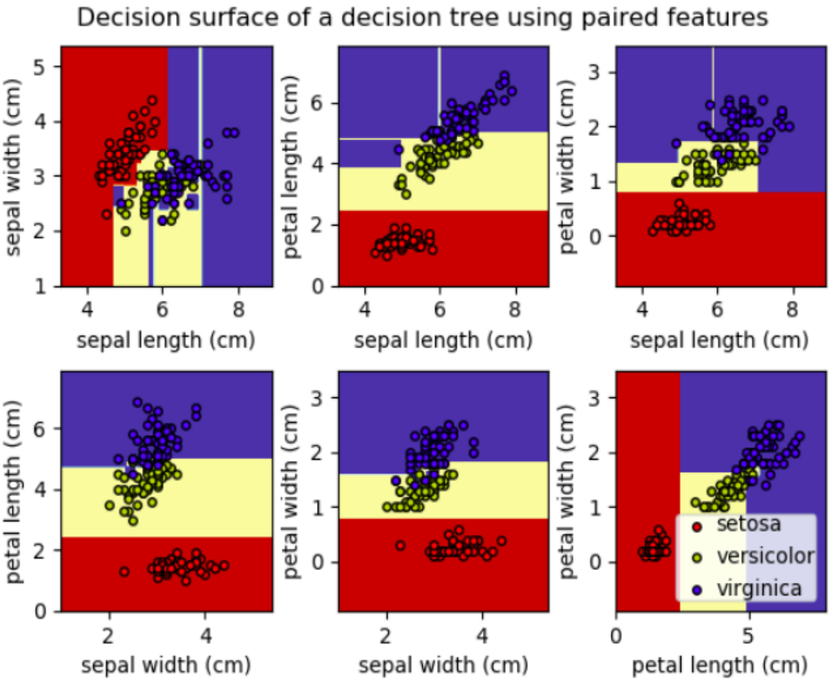
NUMPY E PANDAS- LENDO ARQUIVO CSV

```
import numpy as np
import pandas as pd
import visuals as vs # Supplementary code
from sklearn.cross_validation import ShuffleSplit
```

```
# Load the Boston housing dataset
data = pd.read_csv('housing.csv')
prices = data['MEDV']
features = data.drop('MEDV', axis = 1)
```

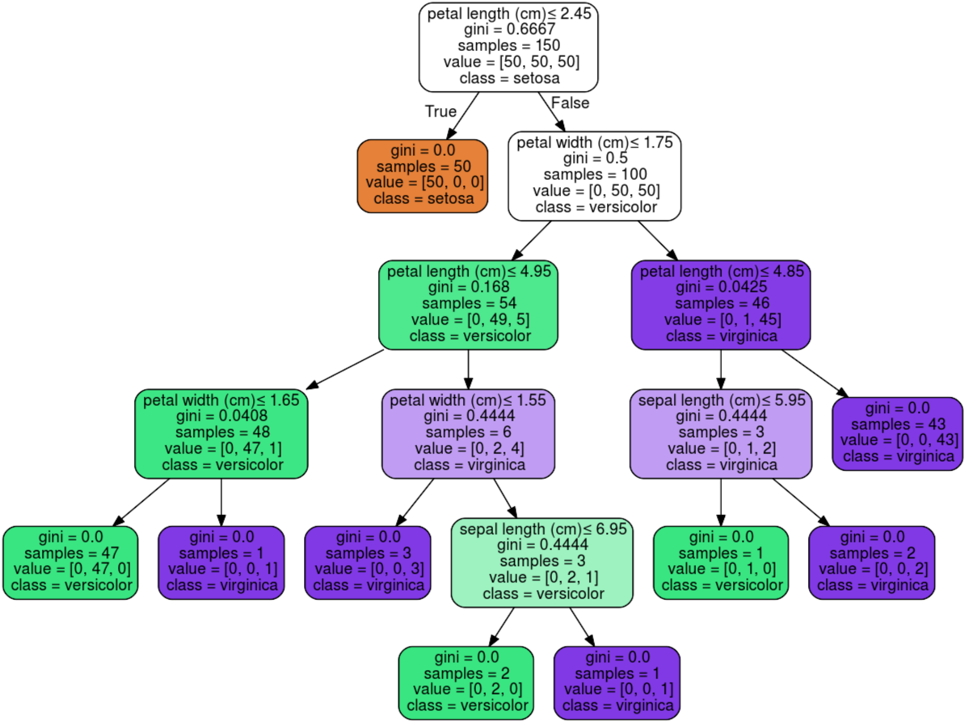
SKLEARN

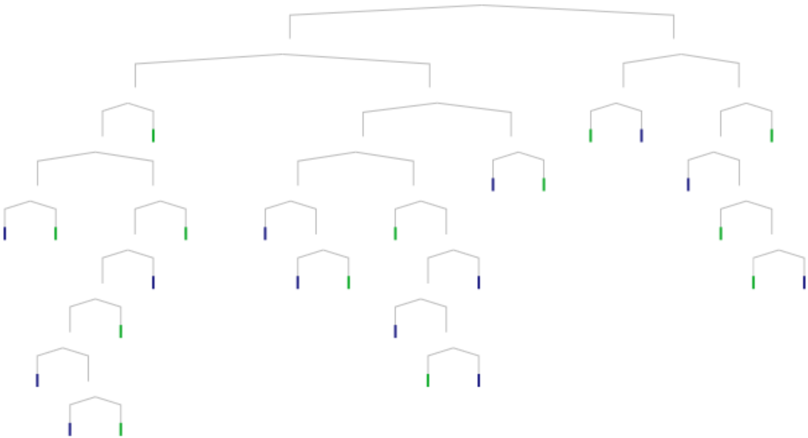
- Aplicação simples e eficiente para data mining e data analysis
- Feito com NumPy, SciPy, e matplotlib
- Open source, commercially usable – BSD license


Decision surface of a decision tree using paired features
sepal width (cm)
sepal length (cm)
petal length (cm)
sepal length (cm)
petal width (cm)
sepal length (cm)
petal length (cm)
sepal width (cm)
petal width (cm)
sepal width (cm)
petal width (cm)
petal length (cm)
setosa
versicolor
virginica

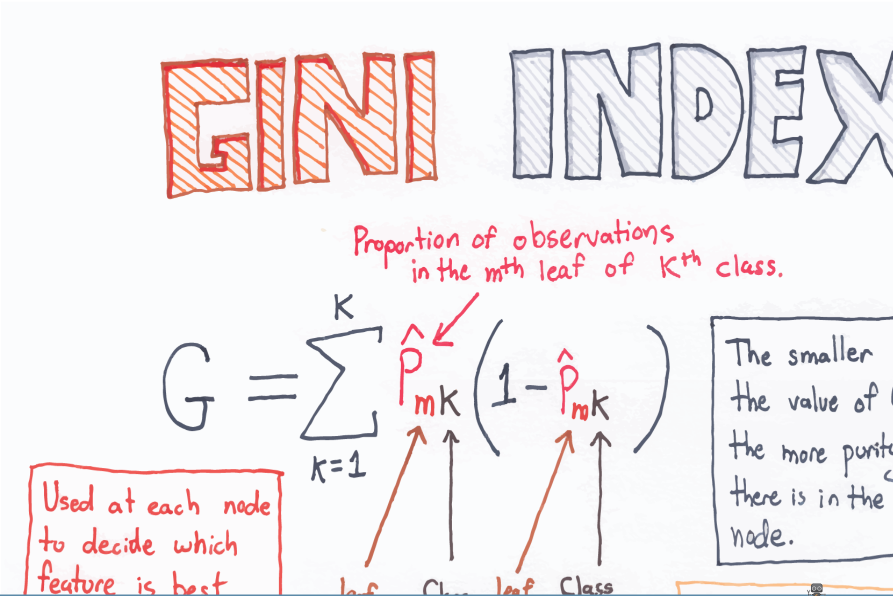
DT Advantages/Disadvantages

- Advantages:
  - Easy to understand.
  - Easy to generate rules
- Disadvantages:
  - May suffer from overfitting.
  - Classifies by rectangular partitioning.
  - Does not easily handle nonnumeric data.
  - Can be quite large – pruning is necessary.


petal length (cm)≤ 2.45
gini = 0.6667
samples = 150
value = [50, 50, 50]
class = setosa
True
False
gini = 0.0
samples = 50
value = [50, 0, 0]
class = setosa
petal width (cm)≤ 1.75
gini = 0.5
samples = 100
value = [0, 50, 50]
class = versicolor
petal length (cm)≤ 4.95
gini = 0.168
samples = 54
value = [0, 49, 5]
class = versicolor
petal length (cm)≤ 4.85
gini = 0.0425
samples = 46
value = [0, 1, 45]
class = virginica
petal width (cm)≤ 1.65
gini = 0.0408
samples = 48
value = [0, 47, 1]
class = versicolor
petal width (cm)≤ 1.55
gini = 0.4444
samples = 6
value = [0, 2, 4]
class = virginica
sepal length (cm)≤ 5.95
gini = 0.4444
samples = 3
value = [0, 1, 2]
class = virginica
gini = 0.0
samples = 43
value = [0, 0, 43]
class = virginica
gini = 0.0
samples = 47
value = [0, 47, 0]
class = versicolor
gini = 0.0
samples = 1
value = [0, 0, 1]
class = virginica
gini = 0.0
samples = 3
value = [0, 0, 3]
class = virginica
sepal length (cm)≤ 6.95
gini = 0.4444
samples = 3
value = [0, 2, 1]
class = versicolor
gini = 0.0
samples = 1
value = [0, 1, 0]
class = versicolor
gini = 0.0
samples = 2
value = [0, 0, 2]
class = virginica
gini = 0.0
samples = 2
value = [0, 2, 0]
class = versicolor
gini = 0.0
samples = 1
value = [0, 0, 1]
class = virginica



Test Accuracy


GINI INDEX
Proportion of observations in the mth leaf of Kth class.
$G = \sum_{k=1}^{K} \hat{p}_{mk}(1-\hat{p}_{mk})$
Used at each node to decide which feature is best
The smaller
the value of
the more purit
there is in the
node.
Class

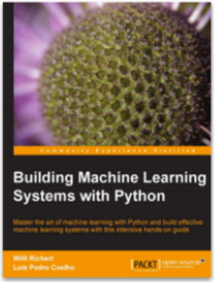
Iniciando a Trabalhar com modelos de Regressão

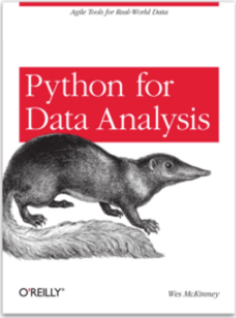
Modelo de Regressão Linear

```
>>> from sklearn import tree
>>> X = [[0, 0], [2, 2]]
>>> y = [0.5, 2.5]
>>> clf = tree.DecisionTreeRegressor()
>>> clf = clf.fit(X, y)
>>> clf.predict([[1, 1]])
array([0.5])
```

LIVROS


O'REILLY
Agile Data Science
BUILDING DATA ANALYTICS APPLICATIONS WITH HADOOP
Russell Jurney


Community Experience Distilled
Building Machine Learning Systems with Python
Master the art of machine learning with Python and build effective machine learning systems with this intensive hands-on guide
Willi Richert
Luis Pedro Coelho
PACKT open source


Agile Tools for Real-World Data
Python for Data Analysis
O'REILLY
Wes McKinney


O'REILLY
Doing Data Science
STRAIGHT TALK FROM THE FRONTLINE
Rachel Schutt & Cathy O'Neil


Quick answers to common problems
Python Data Visualization Cookbook
Over 60 recipes that will enable you to learn how to create attractive visualizations using Python's most popular libraries
Igor Milovanović
PACKT open source

A Tour of Machine Learning Algorithms

by **Jason Brownlee** on August 12, 2019 in **Machine Learning Algorithms**

Last Updated on December 5, 2019

In this post, we will take a tour of ***the most popular machine learning algorithms***.

It is useful to tour the main algorithms in the field to get a feeling of what methods are available.

There are so many algorithms that it can feel overwhelming when algorithm names are thrown around and you are expected to just know what they are and where they fit.

I want to give you two ways to think about and categorize the algorithms you may come across in the field.

- The first is a grouping of algorithms by their **learning style**.
- The second is a grouping of algorithms by their **similarity** in form or function (like grouping similar animals together).

Both approaches are useful, but we will focus in on the grouping of algorithms by similarity

Install User Guide API Examples More ▾

Go

scikit-learn

Machine Learning in Python

Getting Started | What's New in 0.22.1 | GitHub

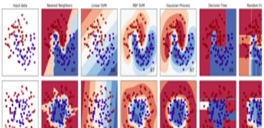
- Simple and efficient tools for predictive data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

Classification

Identifying which category an object belongs to.

Applications: Spam detection, image recognition.

Algorithms: SVM, nearest neighbors, random forest, and more...

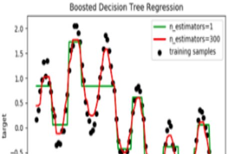


Regression

Predicting a continuous-valued attribute associated with an object.

Applications: Drug response, Stock prices.

Algorithms: SVR, nearest neighbors, random forest, and more...



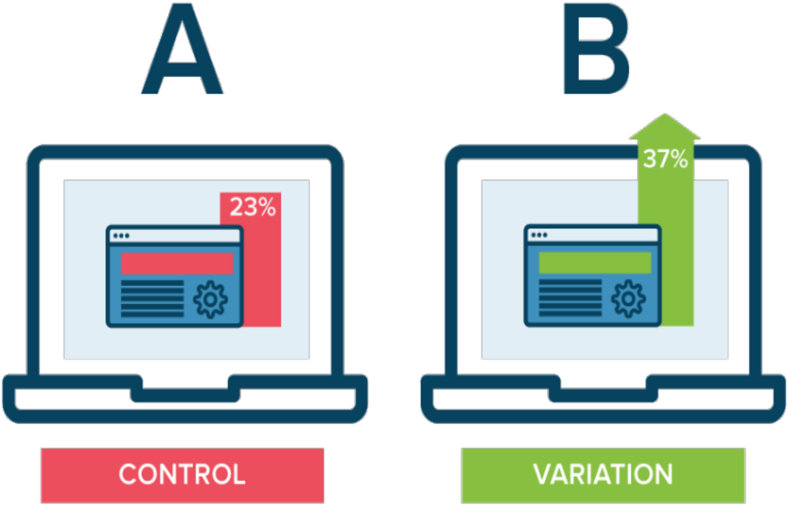
Clustering

Automatic grouping of similar objects into sets.

Applications: Customer segmentation, Grouping experiment outcomes

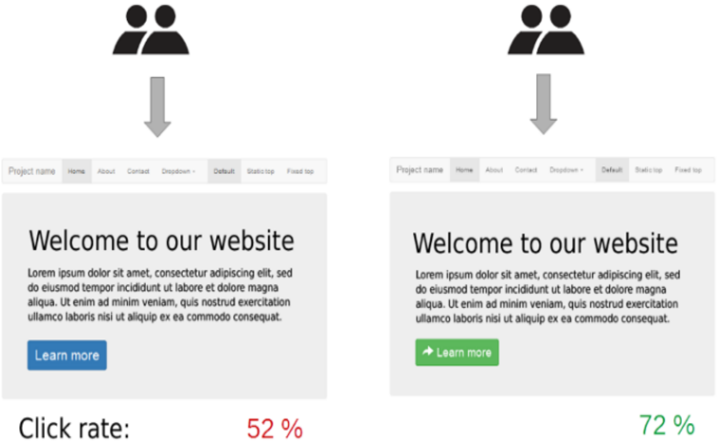
Algorithms: k-Means, spectral clustering, mean-shift, and more...

TESTE A/B



TESTE A/B

Teste A/B é um método de teste onde se comparam duas práticas, A e B, em que estes são o controle e o tratamento de uma experiência controlada, com o objetivo de melhorar a percentagem de aprovação.



LINGUAGEM R

R é uma linguagem e também um ambiente de desenvolvimento integrado para cálculos estatísticos e gráficos.

Foi criada originalmente por Ross Ihaka e por Robert Gentleman no departamento de Estatística da universidade de Auckland, Nova Zelândia.

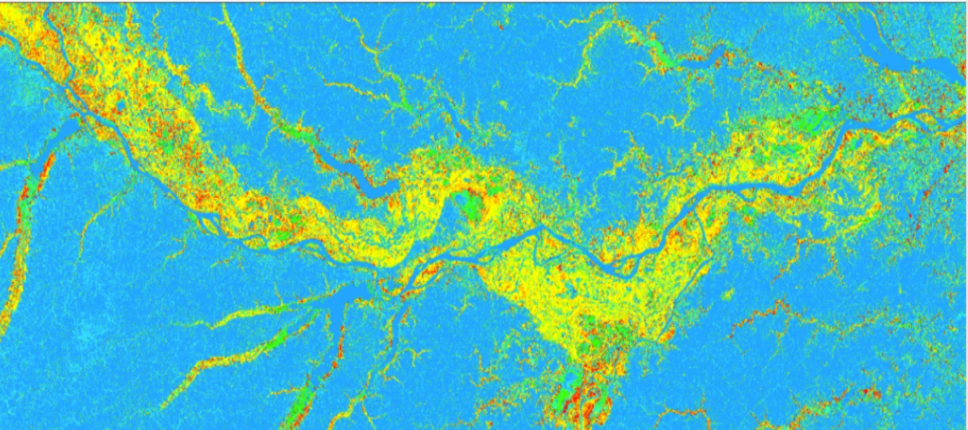
SITES ONDE CONSEGUIR INFORMAÇÃO

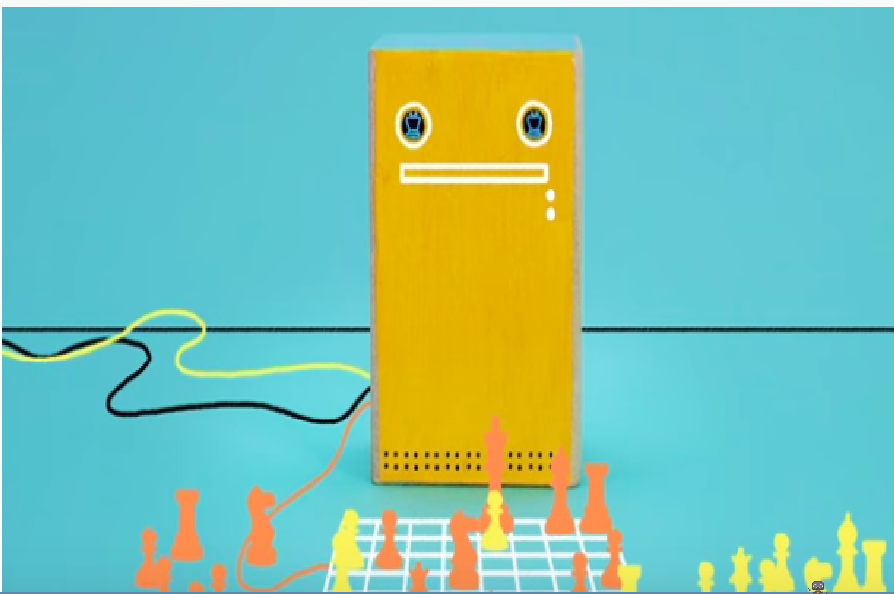
HTTPS://ENSINANDOMAQUINASBLOG.WORDPRESS.COM

INTELIGÊNCIA COMPUTACIONAL PARA MINERAÇÃO DE DADOS

ENSINANDO MÁQUINAS

SOBRE





Conclusão

Você esta apto a:

Compreender o que é Aprendizado de Máquina e DataScience
Entender o que é classificação, regressão e clustering

Bibliografia I