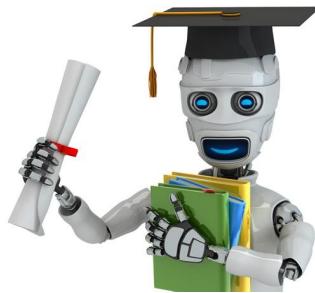
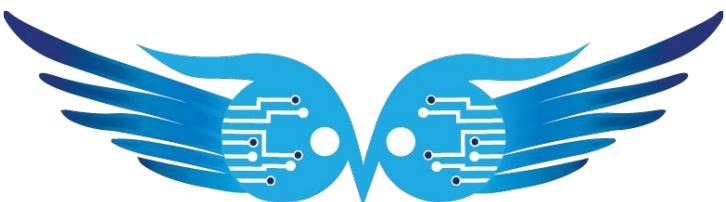


# Machine learning para problemas de hoy:

un enfoque para la industria.



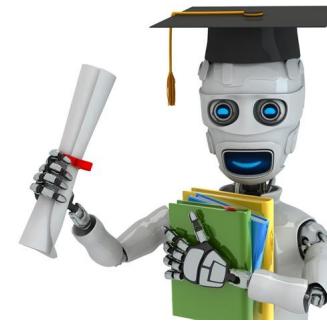
**Machine learning Meet-up  
Medellín**



**CIENTÍFICAS  
DE DATOS**



+ Meetups



Facebook:  
**Machine Learning Colombia**



## Juan Camilo Vásquez



Email:

**juancamilo0628@gmail.com**

Twitter: **@jcvasquezc1**

<http://jcvasquezc.wix.com/home>

## Sebastián Pineda



Email:

**sepia92@hotmail.com**

# Team

## Andres Velez

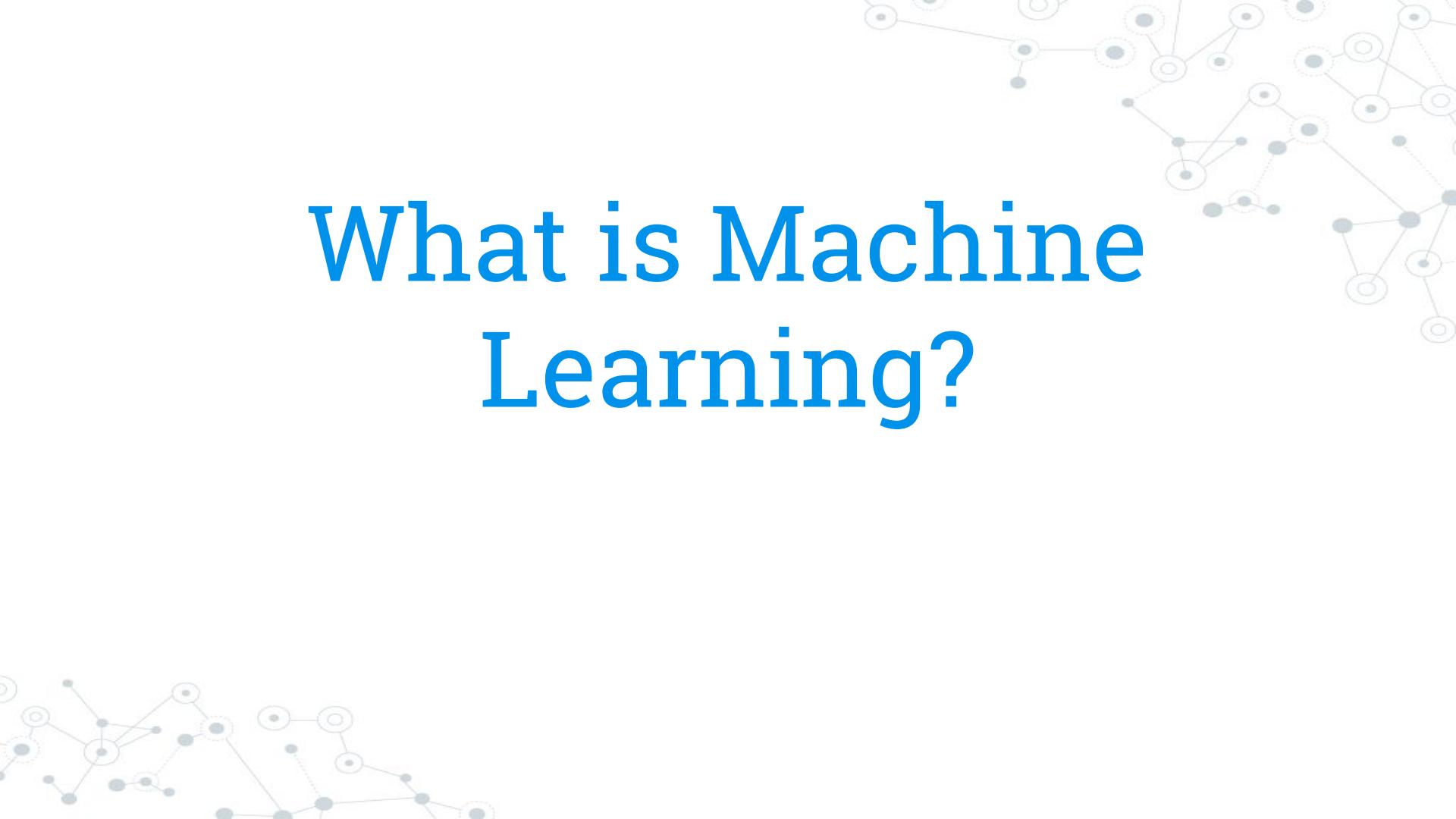


Email:

**andres.velez.e@gmail.com**

**linkedIn: anvelezec**

# What is Machine Learning?









Time, Money, Human Resources



# Machine learning: key concept

Available  
data



Predictions  
for new  
data



Machine learning  
algorithm

# PetApp

## My Pets



husky

Shadow



golden

Spike

# PetApp

## My Pets



husky

Shadow



golden

Spike



???

Nuke

## Traditional approach

```
class DogClassifier(object):

    def __init__(self, image):
        # ...

    def calculate_dog_size(self):
        # ...

    def estimate_dog_color(self):
        # ...

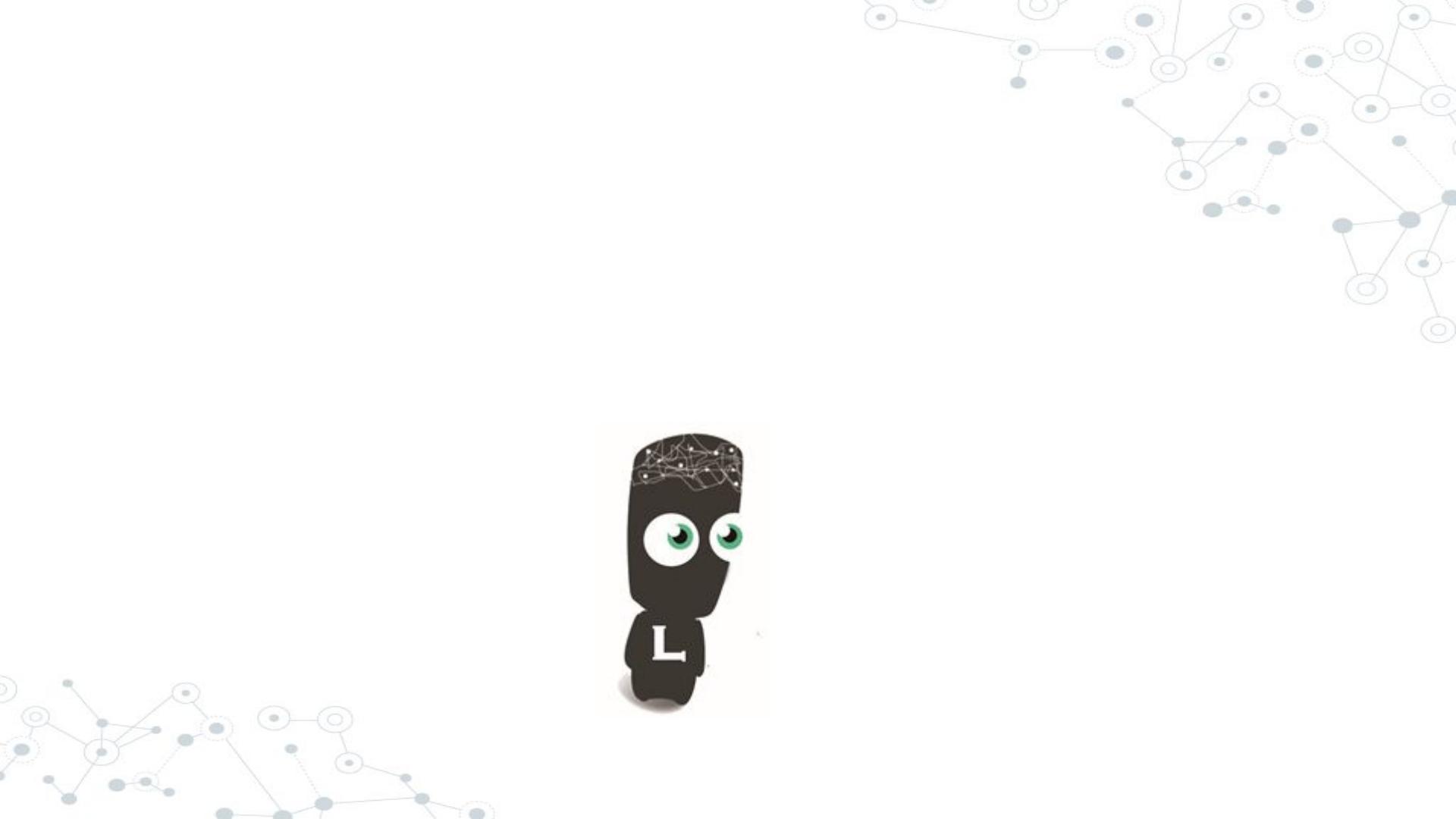
    def estimate_tail_length(self):
        # ...

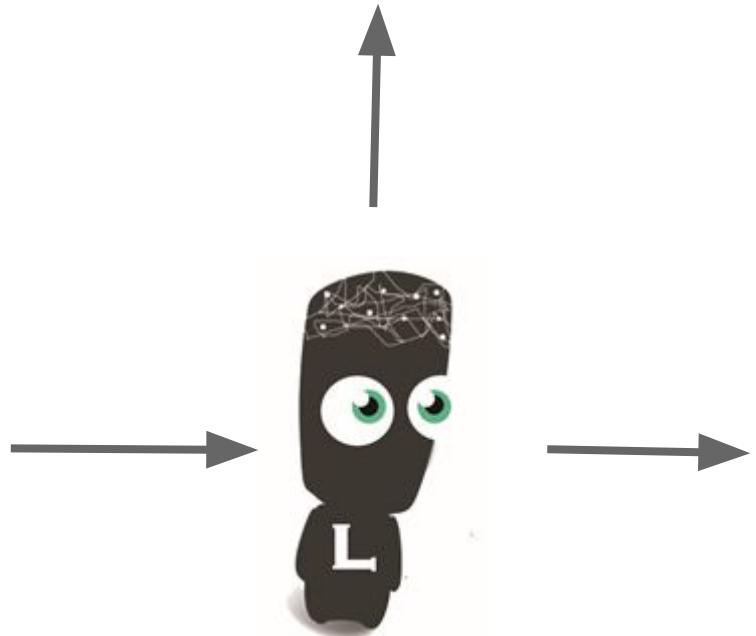
    # .....

    def is_there_even_a_dog_in_the_image(self):
        # ...
```

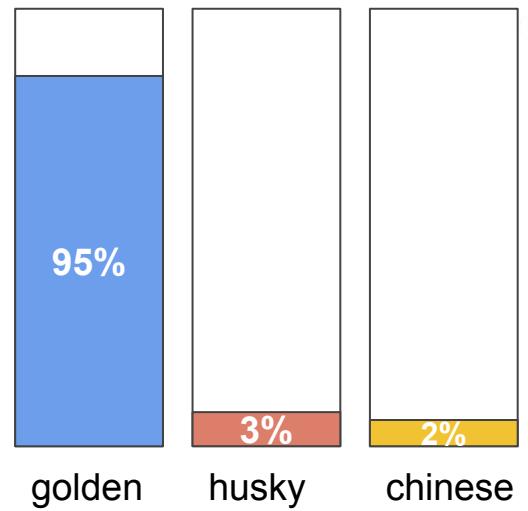
## Traditional approach

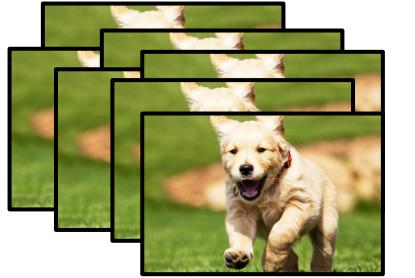
```
class DogClassifier(object):  
  
    def __init__(self, image):  
        self.image = image  
        dog_detector = cv2.CascadeClassifier('haarcascade_dogface.xml')  
  
    def estimate_dog_breed(self):  
        # ...  
  
    def is_there_even_a_dog_in_the_image(self):  
        # ...
```



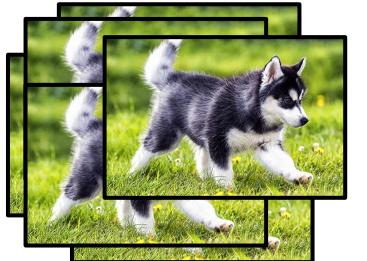


score=98%





golden



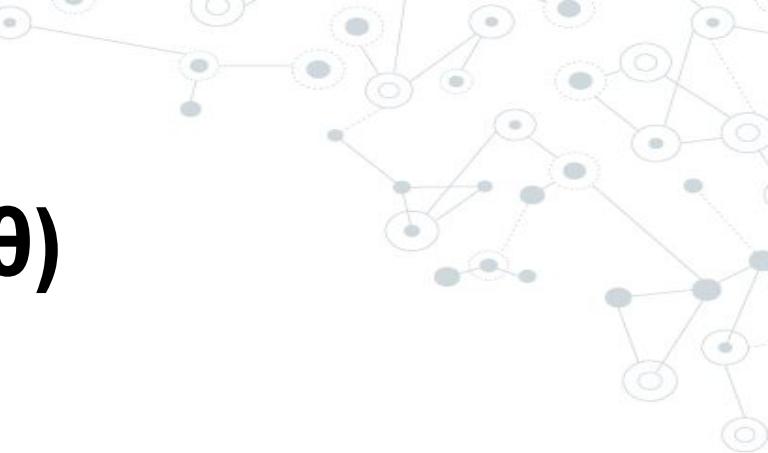
husky



chinese

**x**

**y**



$$y = f(x \mid \theta)$$



$f$   
→ golden

$x$

$y$



# Machine Learning as an Industry



50,000+  
employees



247

Companies/Products



\$23B  
funding



7

Billion Dollar Companies



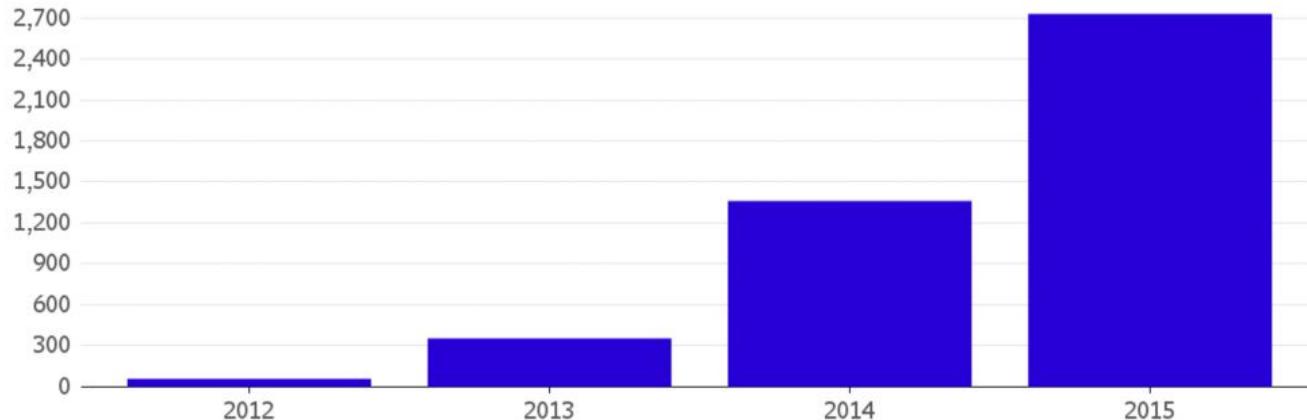
\$107B

Value Created

## Trends

### Artificial Intelligence Takes Off at Google

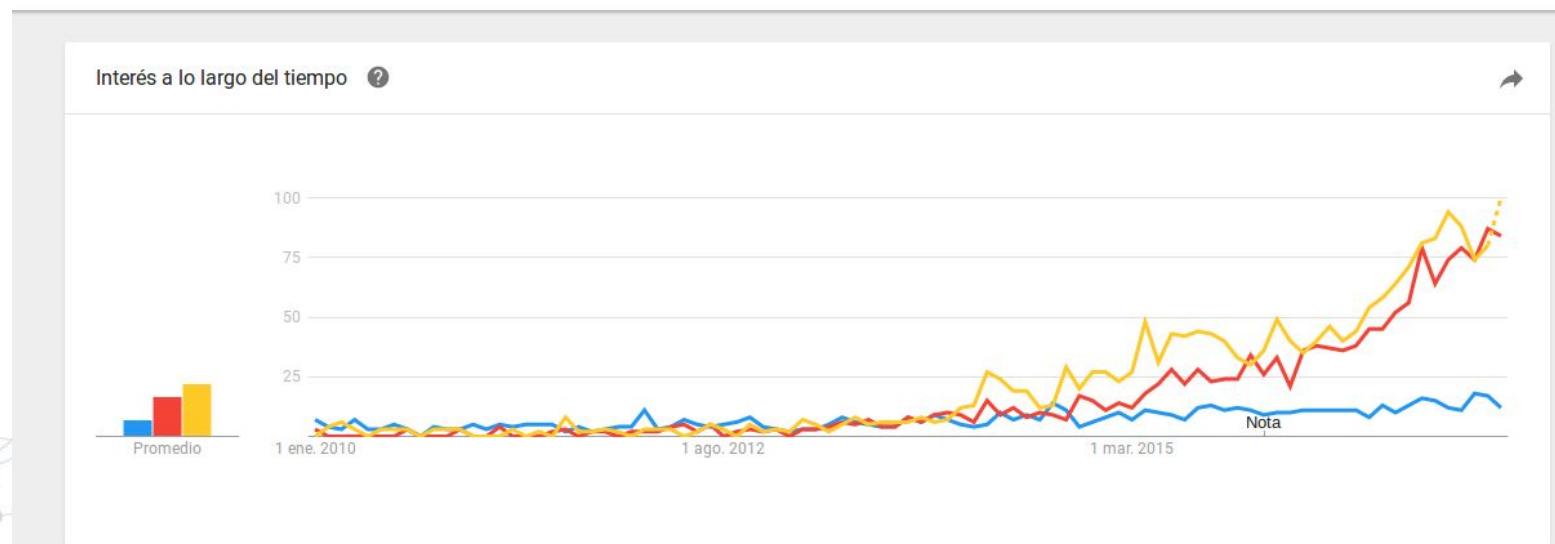
Number of software projects within Google that uses a key AI technology, called Deep Learning.



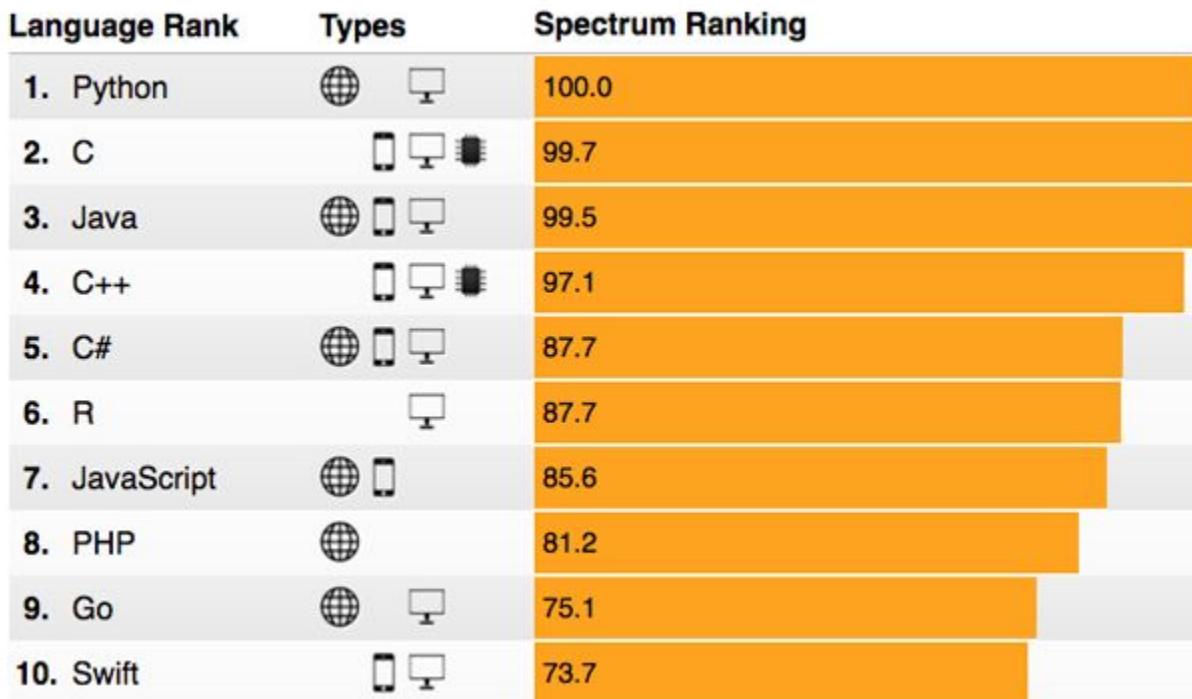
Source: Google

Note: 2015 data does not incorporate data from Q4

# Trends



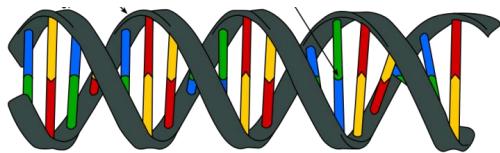
## Trends



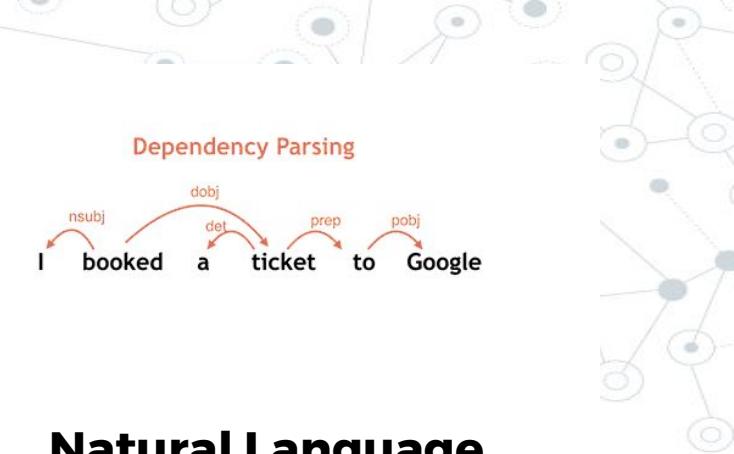
Source: IEEE Spectrum 2017:

<http://spectrum.ieee.org/computing/software/the-2017-top-programming-languages>

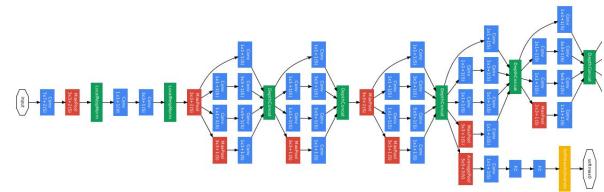
# Applications



**Genetics**



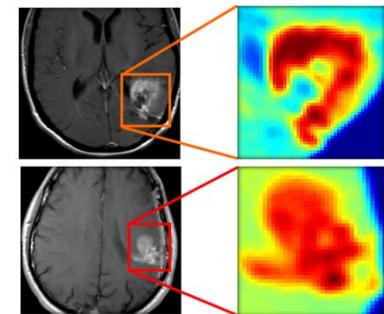
**Natural Language Processing**



**Deep Learning**



**Self Driving Cars**



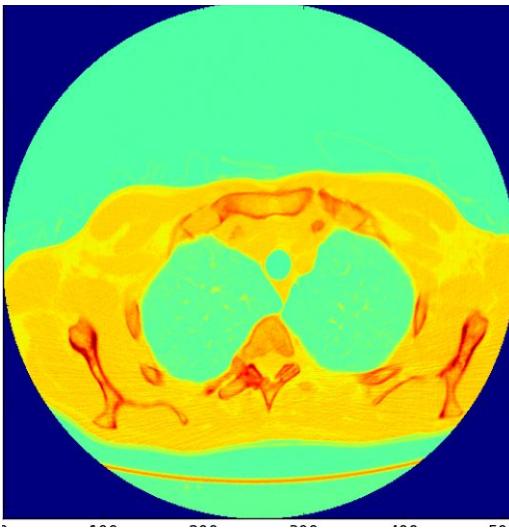
**Healthcare**

**2017 Worldwide**

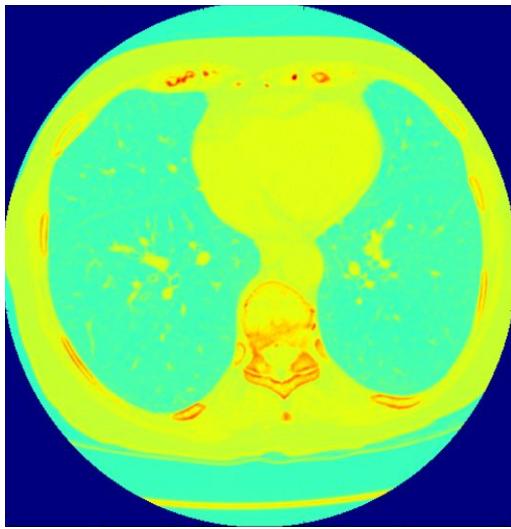
## Applications: human computer interaction



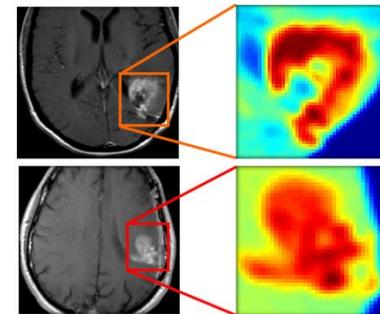
## Applications: healthcare



cancer



healthy



## Applications: business analytics



### Grupo Bimbo Inventory Demand

Maximize sales and minimize returns of bakery goods  
\$25,000 · 1,969 teams · a year ago



### Santander Customer Satisfaction

Which customers are happy customers?  
\$60,000 · 5,123 teams · a year ago

# Applications

The screenshot shows a Kaggle dataset page. At the top, there's a navigation bar with the 'kaggle' logo, a search bar containing 'Search kaggle', and links for 'Competitions' and 'Datasets'. Below the navigation bar is a large image showing a close-up of hands holding and counting banknotes. Overlaid on this image is the title 'Synthetic Financial Datasets For Fraud Detection' in bold white text. Below the title, a subtitle reads 'Synthetic datasets generated by the PaySim mobile money simulator'. In the bottom left corner of the image area, there's a small profile icon for 'TESTIMON @ NTNU' and the text 'last updated 3 months ago'.

kaggle Search kaggle Competitions Datasets

**Synthetic Financial Datasets For Fraud Detection**

Synthetic datasets generated by the PaySim mobile money simulator

TESTIMON @ NTNU • last updated 3 months ago

# Applications

The screenshot shows the Kaggle website interface. At the top, there are two navigation bars. The left bar has a "kaggle" logo and a search bar labeled "Search kaggle". The right bar has links for "Competitions", "Datasets", and "Kern". Below these, there are two more search bars, one for each navigation bar. The main content area features a large red "BOSCH" logo. To its right, the title "Bosch Production Line Performance" is displayed in bold black text. Below the title, the subtitle "Reduce manufacturing failures" and the competition statistics "\$30,000 · 1,373 teams · 8 months ago" are shown. On the far left, a dark sidebar contains the word "Synthetic" and a small logo with the letters "NTNU" and the word "TESTING".

kaggle Search kaggle Competitions Datasets

kaggle Search kaggle Competitions Datasets Kern

Synthetic TESTIN

**BOSCH**

**Bosch Production Line Performance**

Reduce manufacturing failures

\$30,000 · 1,373 teams · 8 months ago

# Applications

kaggle Search kaggle Competitions Datasets

kaggle Search kaggle Competitions Datasets Kern

kaggle Search kaggle Competitions Datasets Ke

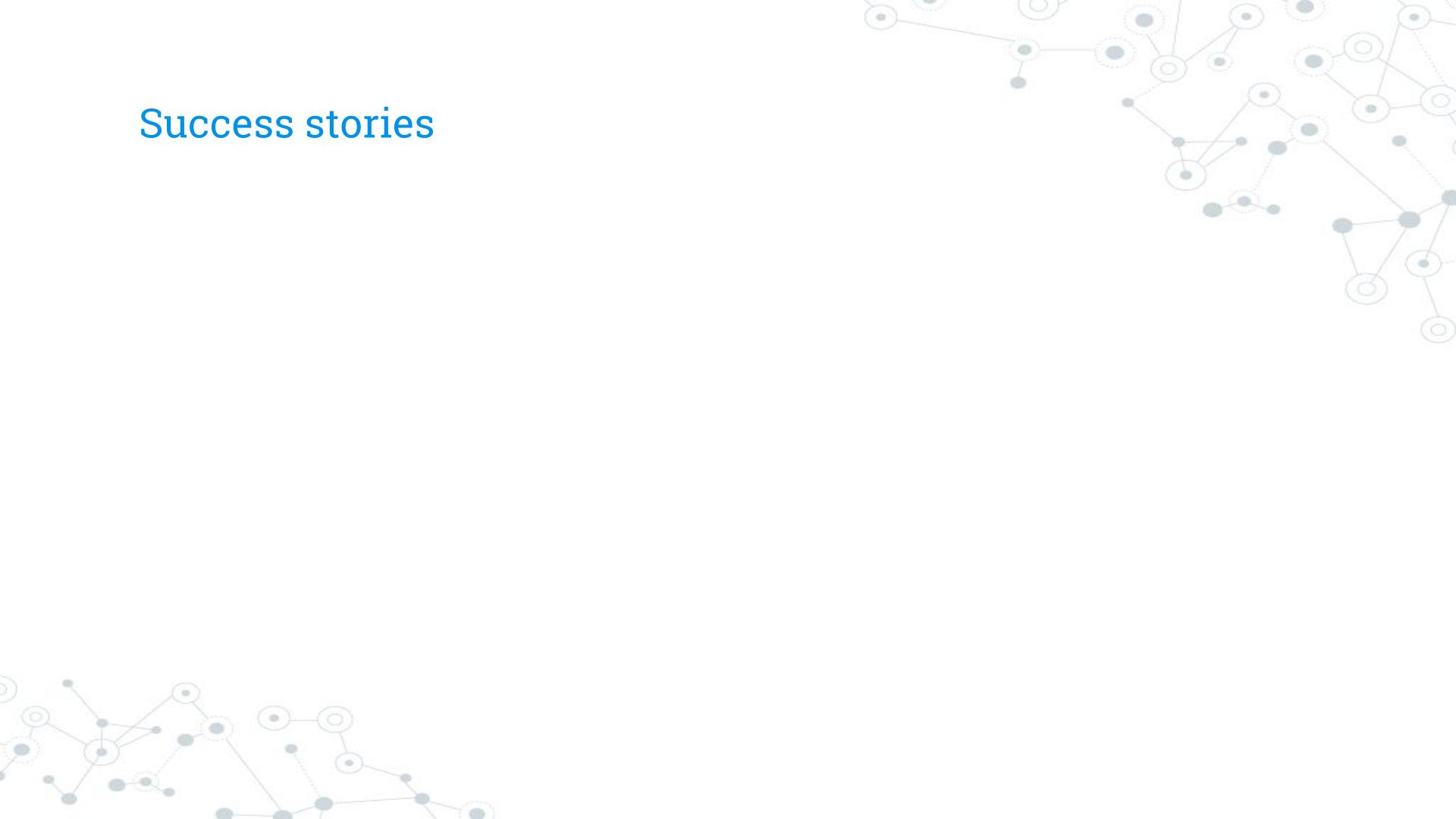
Synthetic

TESTING

E Human Activity Recognition with Smartphones  
Recordings of 30 study participants performing activities of daily living

UCI Machine Learning • last updated 9 months ago

## Success stories



# NETFLIX

Movie Recommendations

# Enlitic



Enlitic uses deep learning to make doctors faster



**Product Recommendations**

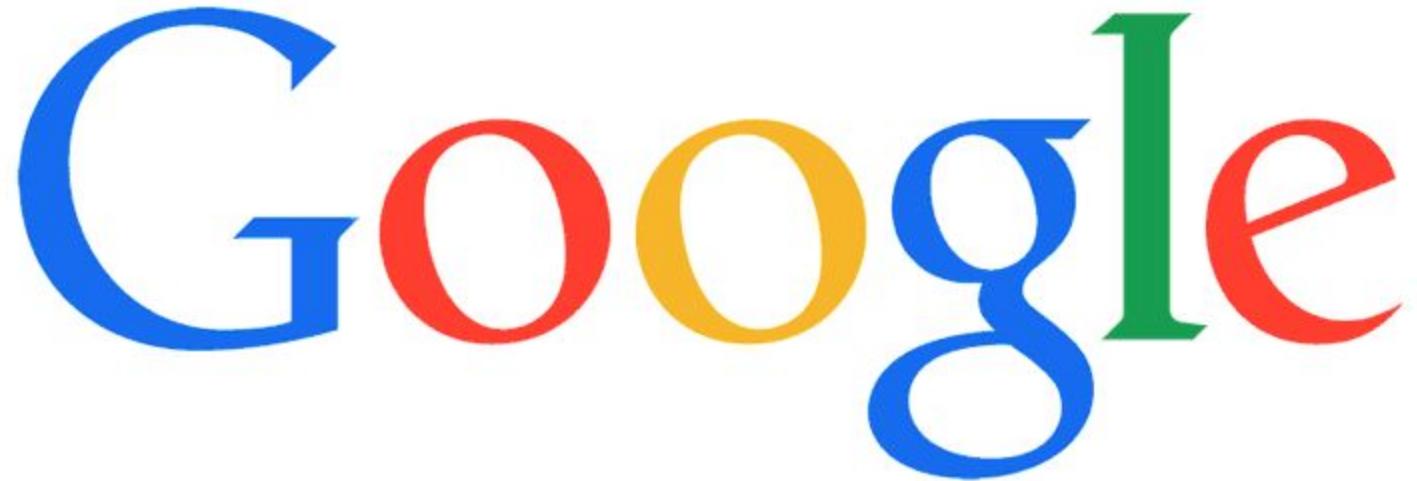
LEGAL ROBOT



# Artificial Intelligence for Legal Documents

---

Error checking, Correction Recommendations, Risk prediction

The Google logo is displayed prominently in the center of the slide. It consists of the word "Google" in a bold, sans-serif font, where each letter is a different color: G is blue, o is red, o is yellow, g is blue, l is green, and e is red.

**Search, Translation, Ads, Google Now, Google Images,  
Self Driving Cars,...**

## HOW IT WORKS



**Emotion recognition.** Our technology analyzes subtle facial expressions to identify human emotions.

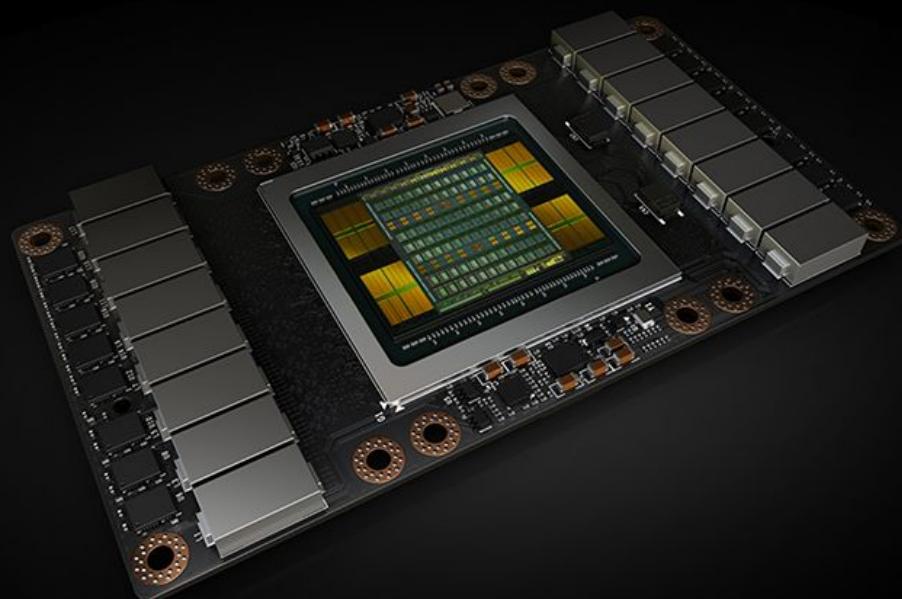
[LEARN MORE](#) [OUR PRODUCTS](#)



A screenshot of the AffdexMe mobile application. At the top, it shows the Affectiva logo and the app name "AffdexMe". Below that, it says "Affectiva Inc. Libraries &amp; Demo" and "Everyone". It also indicates compatibility with all devices. On the right, there are "Add to Wishlist" and "Install" buttons. The main area displays three frames of a man's face. Each frame has a different colored overlay (blue, green, red) highlighting specific features of his face, likely representing detected emotions. To the right of the frames, there is a vertical sidebar with various options like "Show Shapes", "Set To", "Show Emotions", "Show Speech", "Show Headpose", and "Enable Face Recognition".

## Emotion recognition from facial expressions and speech



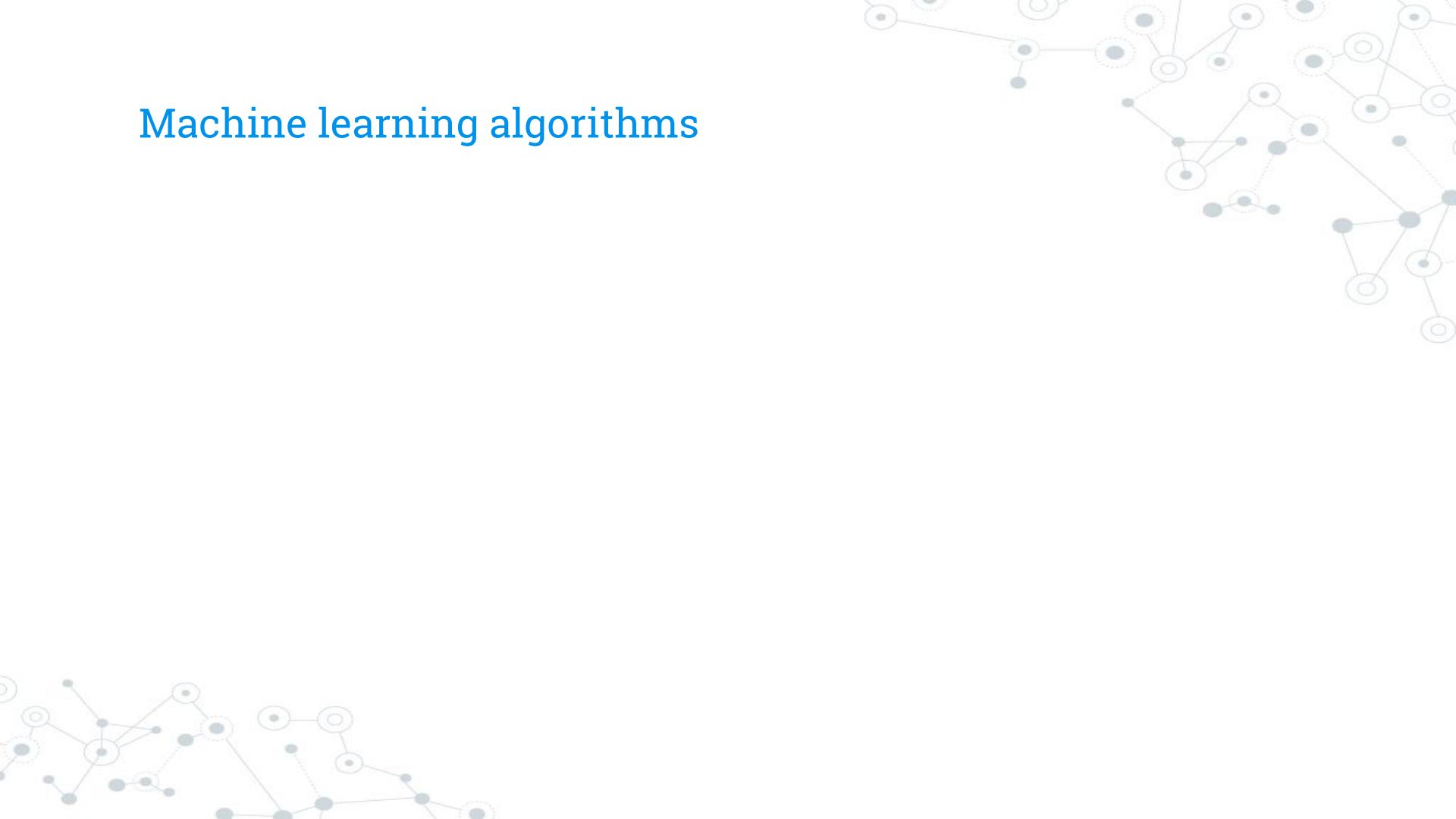


## THE CORE OF AI

NVIDIA Volta is the new driving force behind artificial intelligence.

[LEARN MORE](#)

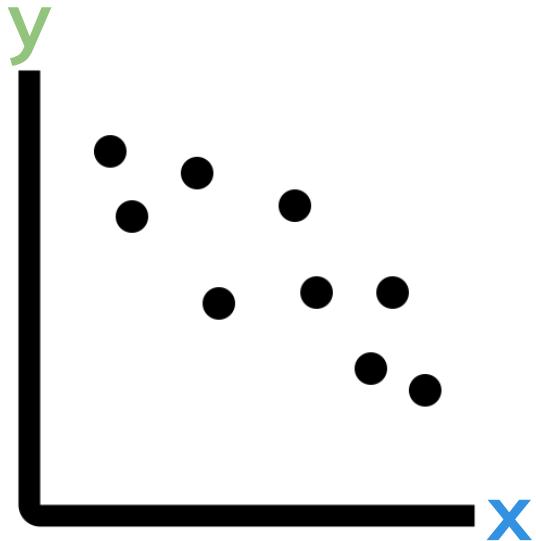
# Machine learning algorithms



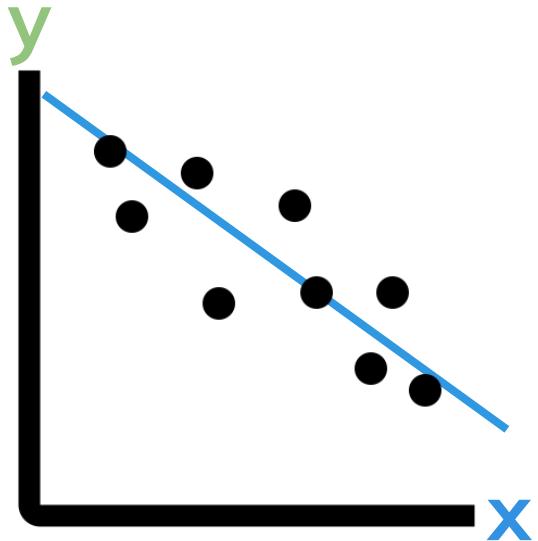
# Regression



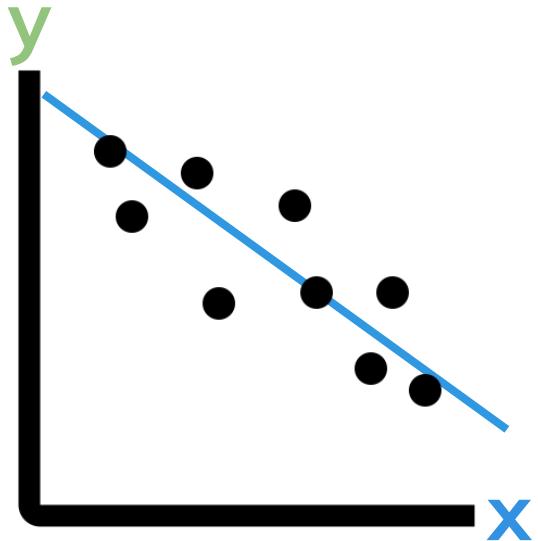
# Regression



# Regression



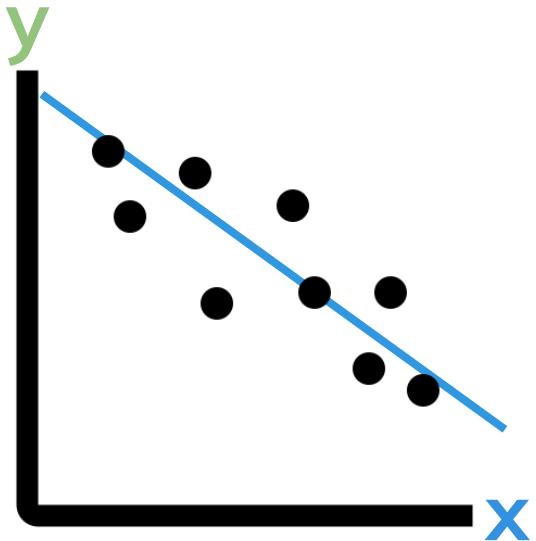
# Regression



# Classification



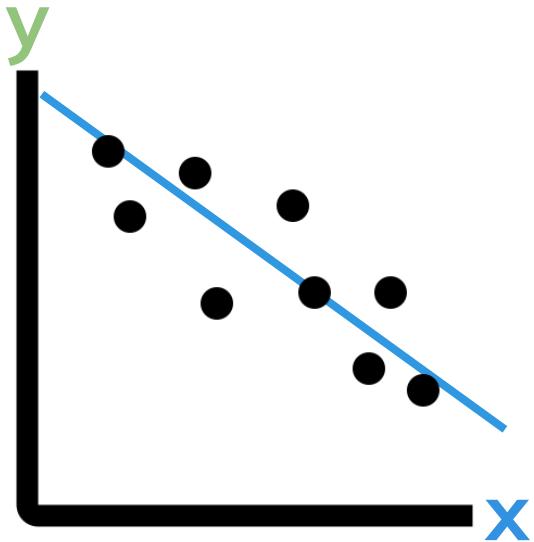
# Regression



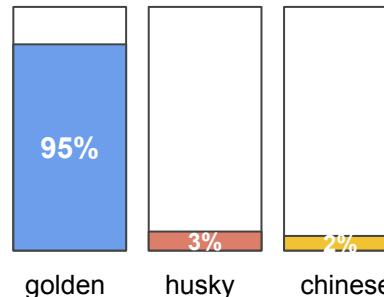
# Classification



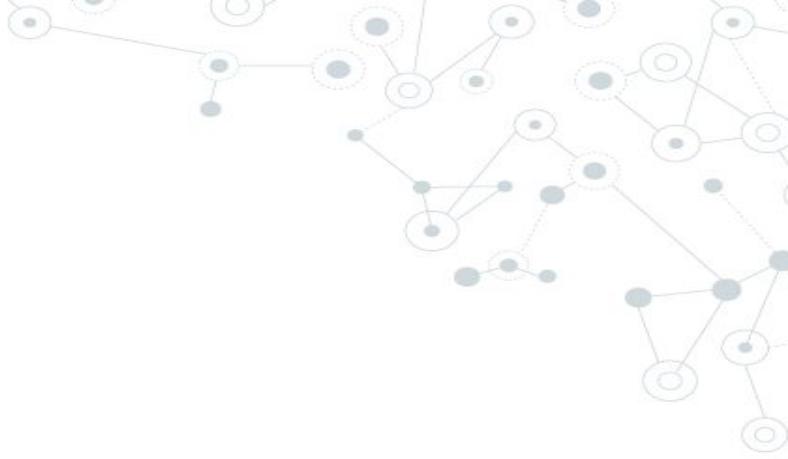
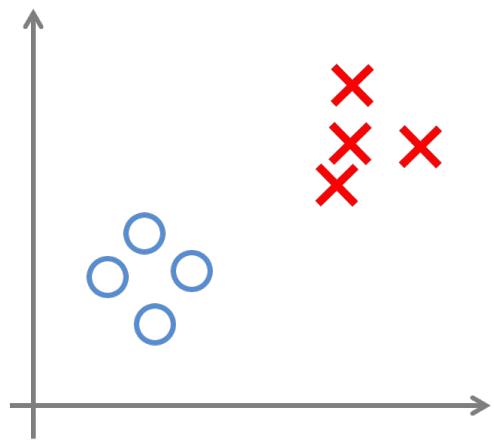
# Regression



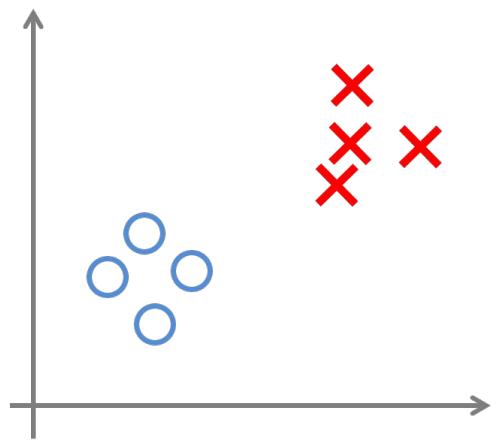
# Classification



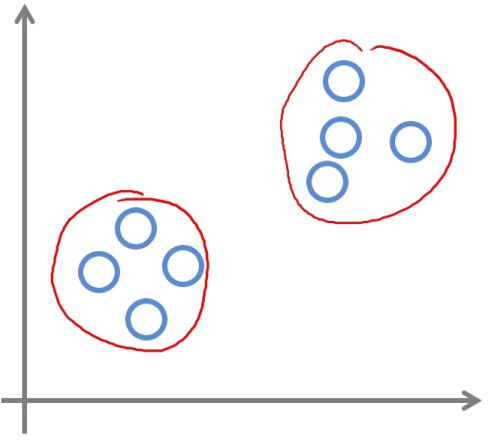
# Supervised



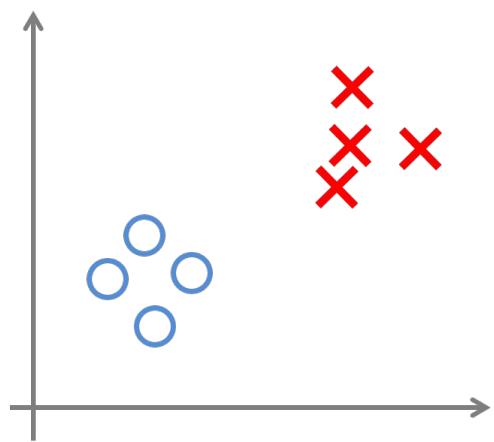
## Supervised



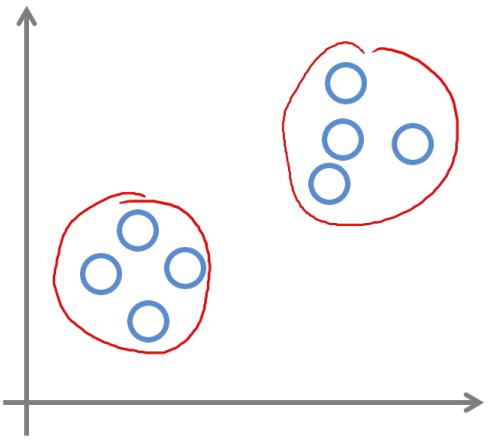
## Unsupervised



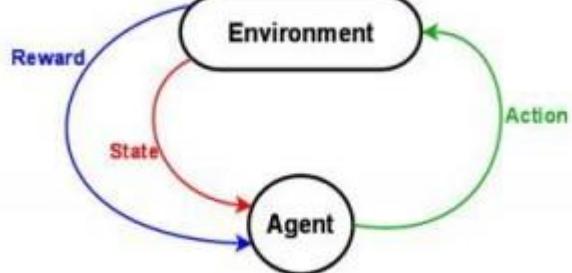
## Supervised



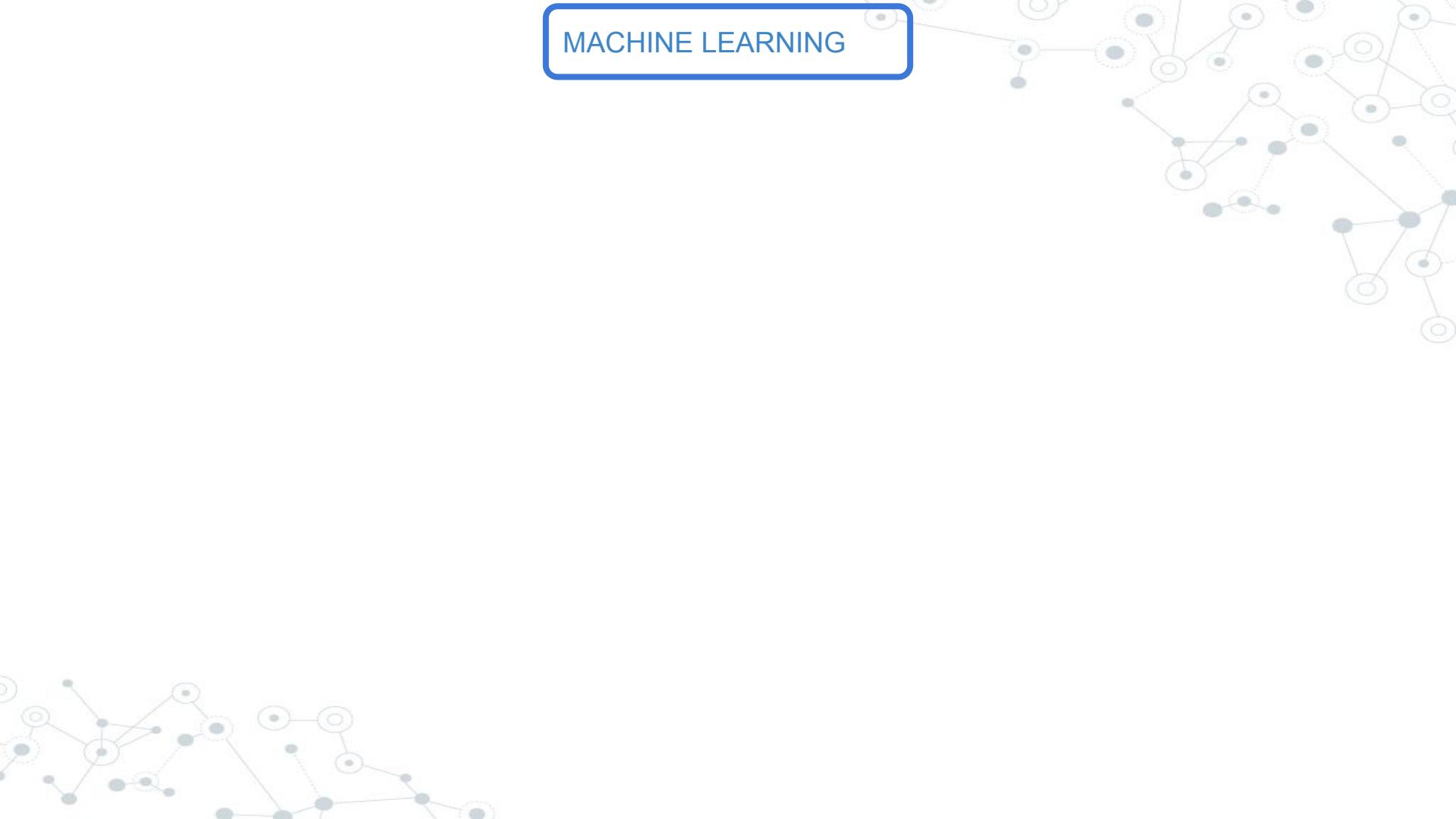
## Unsupervised

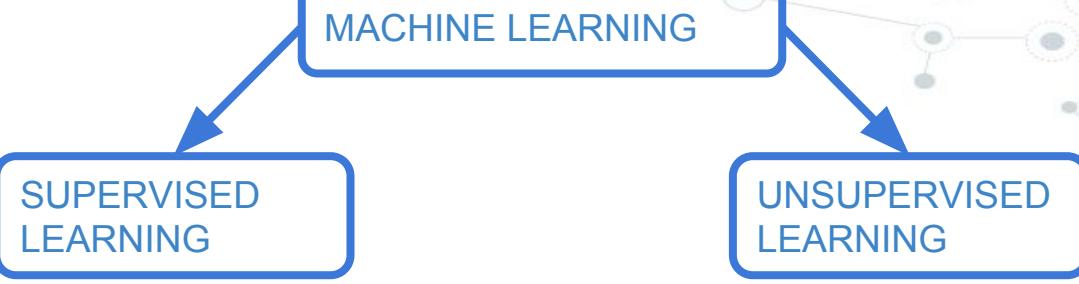


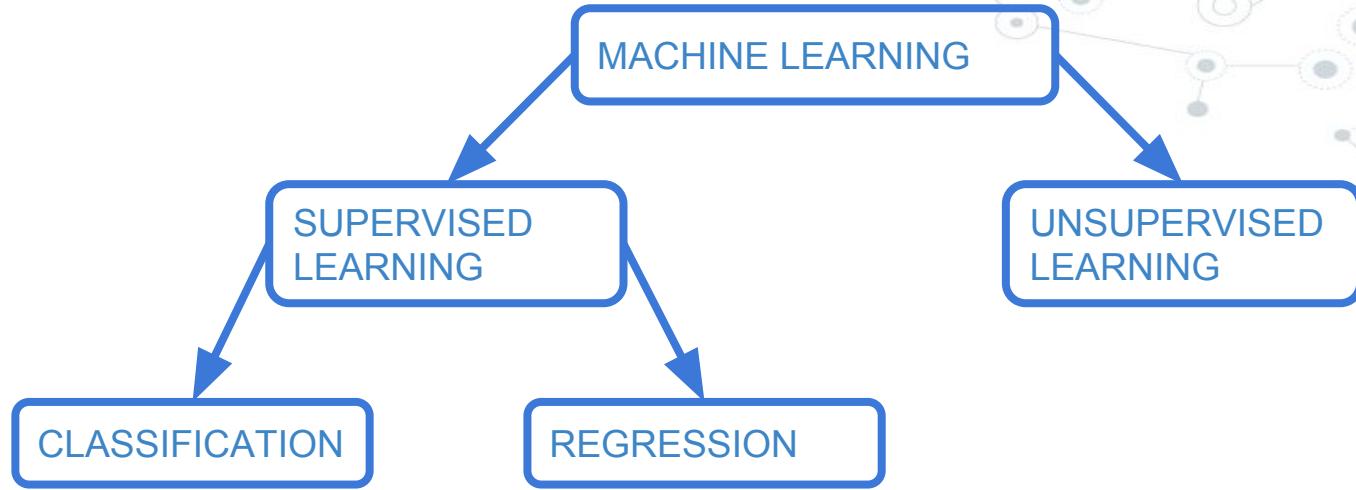
## Reinforcement

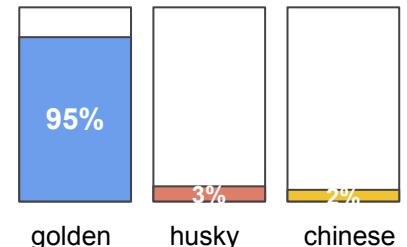
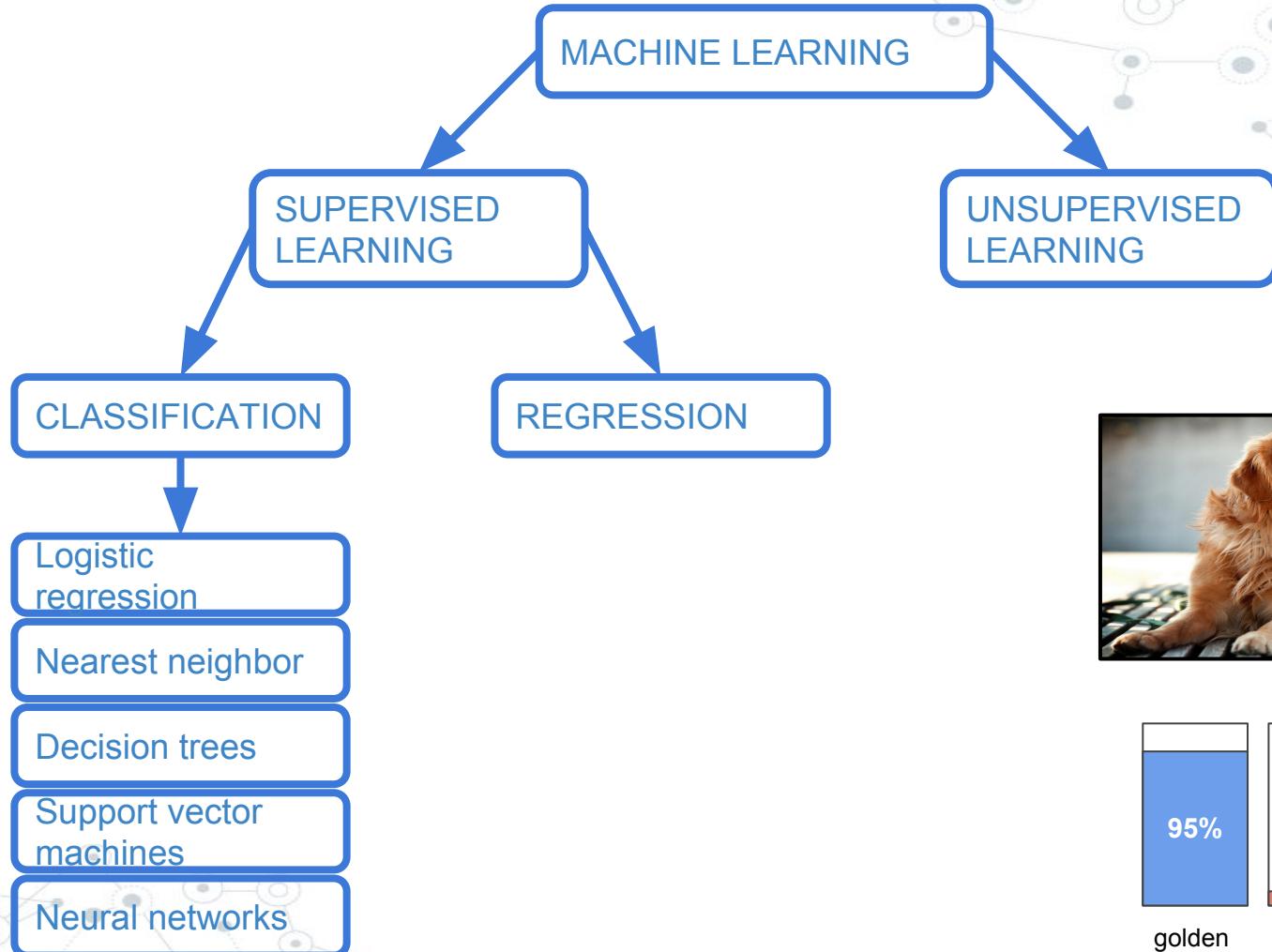


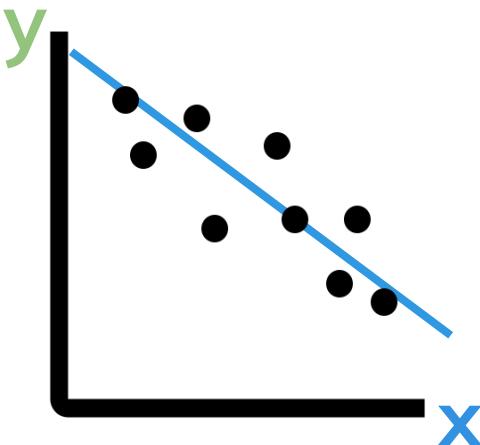
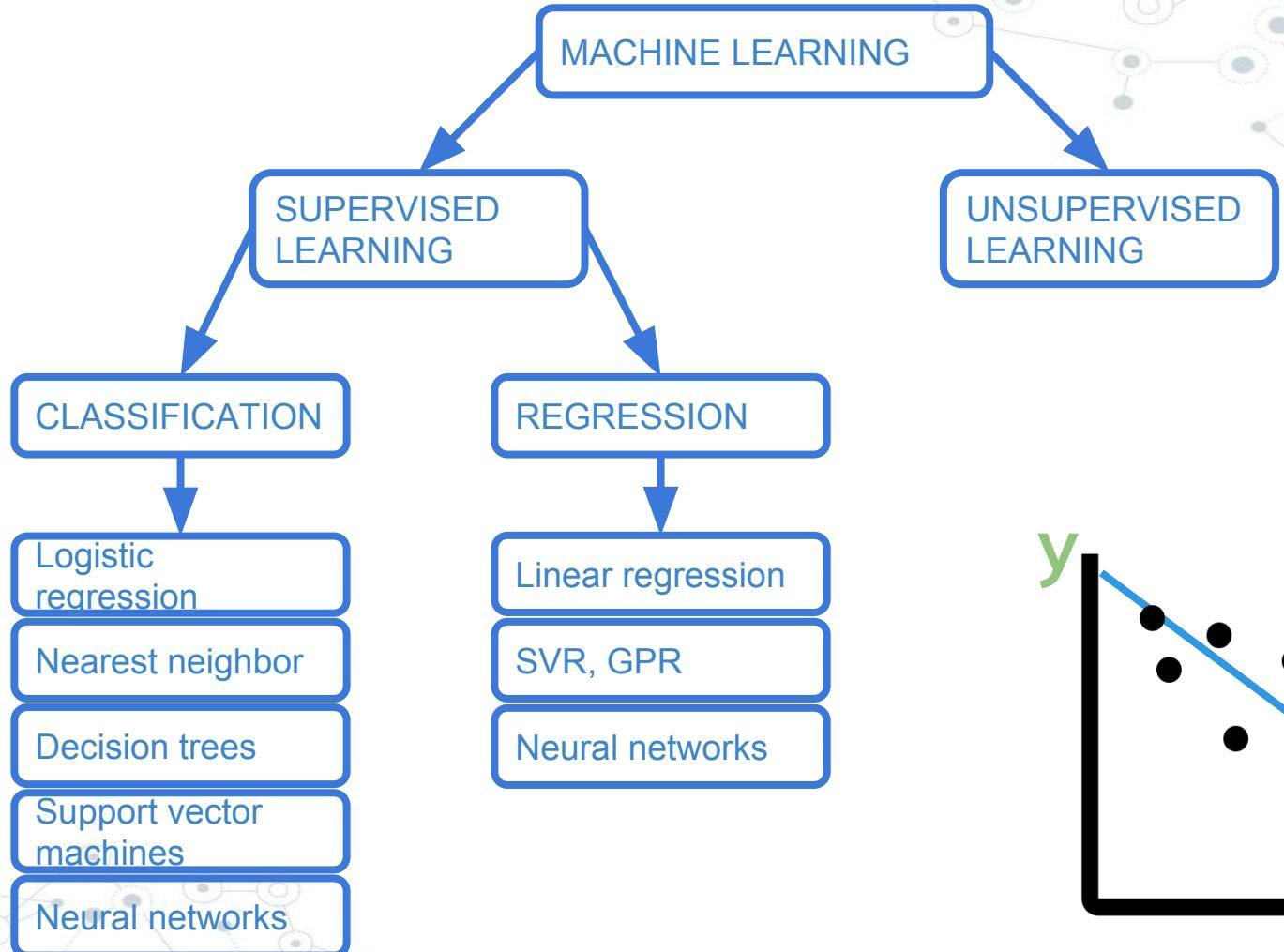
# MACHINE LEARNING

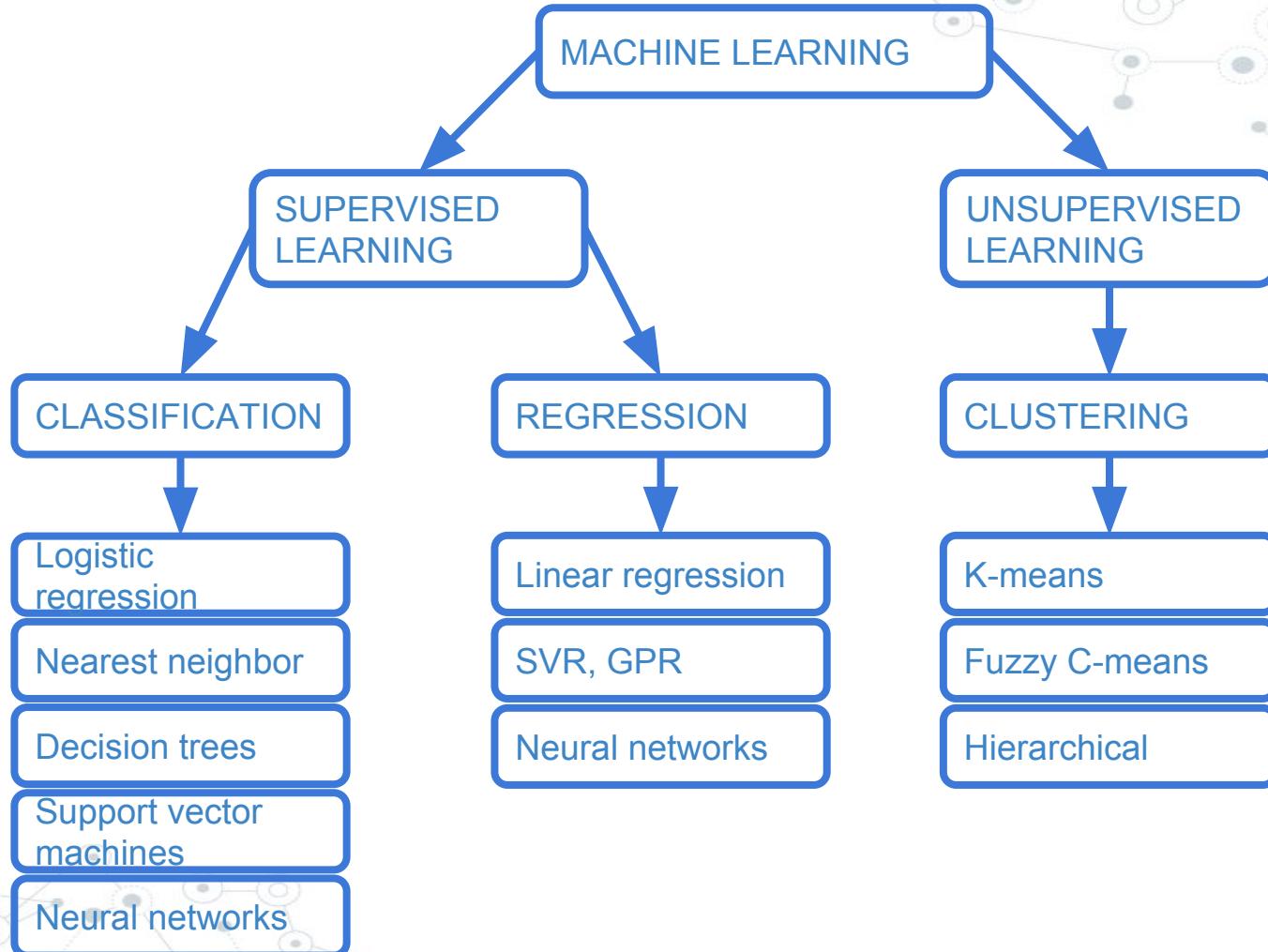












## What makes it possible

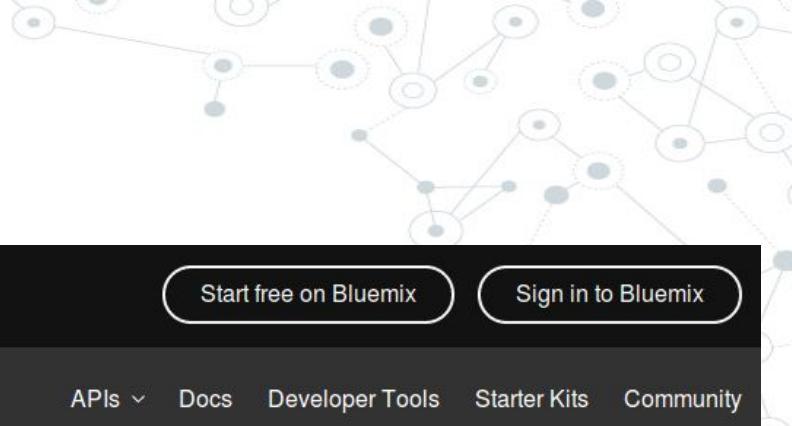








# Cloud machine learning tools



Watson Developer Cloud

[Start free on Bluemix](#)

[Sign in to Bluemix](#)

APIs ▾ Docs Developer Tools Starter Kits Community

## Build with Watson

Enable cognitive computing features in your app using IBM Watson's Language, Vision, Speech and Data APIs.

[Start free on Bluemix](#)

[See the services](#)



# Cloud machine learning tools

Watson Developer Cloud

Start free on Bluemix

Sign in to Bluemix

VENTAS 01-800-710-2238 ▾

MI CUENTA

PORTAL

BÚSQUEDA

## Microsoft Azure

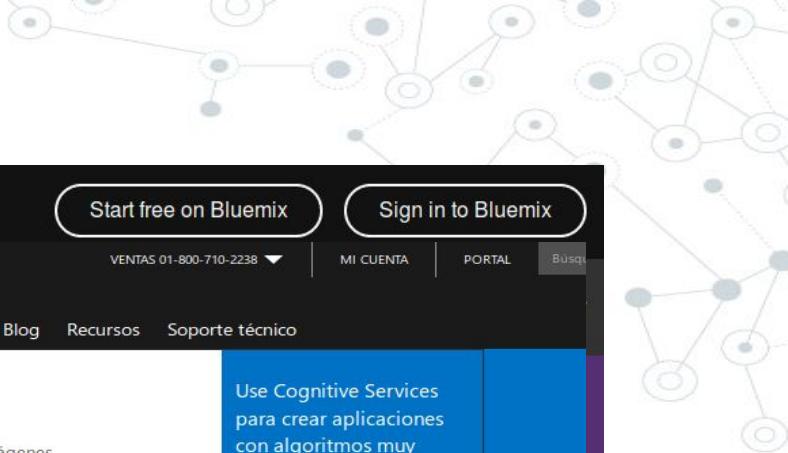
Por qué Azure Soluciones Productos Documentación Precios Formación Partners Blog Recursos Soporte técnico

- Proceso
- Redes
- Storage
- Web y móvil
- Contenedores
- Bases de datos
- Datos y análisis
- AI + Cognitive Services
- Internet de las cosas
- Integración empresarial
- Seguridad + Identidad
- Herramientas para desarrolladores
- Supervisión + Administración
- Microsoft Azure Stack
- [Ver todos los productos](#)

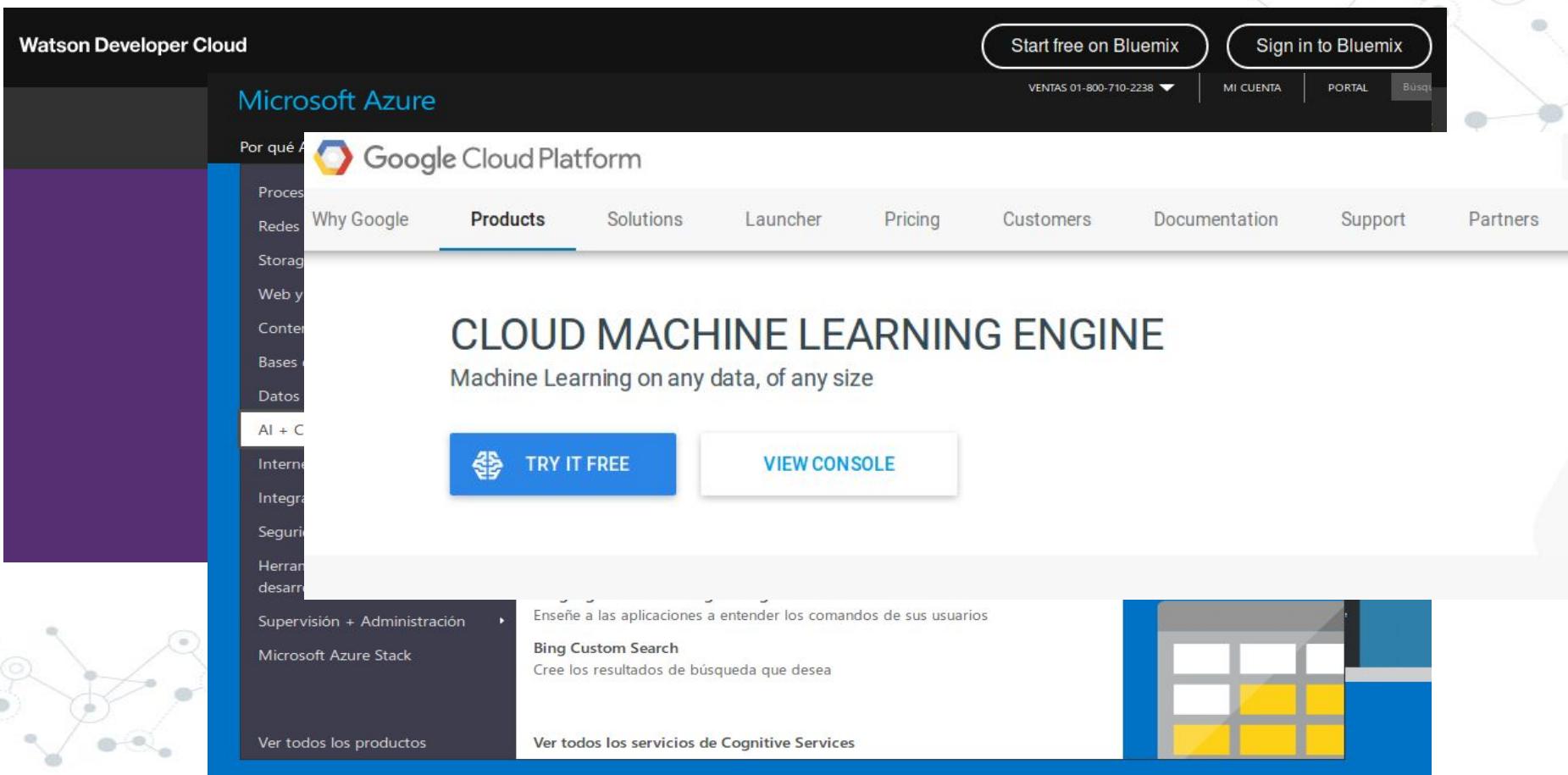
- AI + Cognitive Services
  - Computer Vision API**  
Condense información de aplicación práctica a partir de imágenes
  - API de reconocimiento facial**  
Detecte, identifique, analice, organice y etiquete caras en las fotos
  - Bing Web Search API**  
Conecte búsqueda en la Web de gran eficacia a sus aplicaciones
  - Custom Speech Service**  
Elimine las barreras del reconocimiento de voz, como el estilo de habla, el ruido de fondo y el vocabulario
- Custom Vision Service**  
Un servicio web personalizable que aprende a reconocer determinado contenido en imágenes
- Video Indexer**  
Haga búsquedas en sus videos, editelos, analicelos y obtenga conocimiento de ellos
- Language Understanding Intelligence Service**  
Enseñe a las aplicaciones a entender los comandos de sus usuarios
- Bing Custom Search**  
Cree los resultados de búsqueda que desea

Use Cognitive Services para crear aplicaciones con algoritmos muy eficaces con tan solo algunas líneas de código.

Pruebe Cognitive Services APIs gratis ►



# Cloud machine learning tools

A faint, light-gray network diagram consisting of numerous small circles connected by thin lines, resembling a neural network or a complex web of connections, serves as the background for the entire slide.

Watson Developer Cloud

Start free on Bluemix

Sign in to Bluemix

VENTAS 01-800-710-2238 ▾

MI CUENTA

PORTAL

Búsq...

Microsoft Azure

Por qué A...

Proces...

Redes

Storage

Web y...

Content...

Bases de...

Datos

AI + C...

Intelli...

Integrat...

Seguri...

Herram...

desarr...

Supervisión + Administración

Microsoft Azure Stack

Ver todos los productos

Google Cloud Platform

Why Google

Products

Solutions

Launcher

Pricing

Customers

Documentation

Support

Partners

## CLOUD MACHINE LEARNING ENGINE

Machine Learning on any data, of any size

TRY IT FREE

VIEW CONSOLE

Enseñe a las aplicaciones a entender los comandos de sus usuarios

Bing Custom Search

Cree los resultados de búsqueda que desea

Ver todos los servicios de Cognitive Services

# Cloud machine learning tools

The image features a central screenshot of a web-based machine learning platform. At the top, there's a navigation bar with the word "FLOYD" in blue on the left, and "FEATURES", "PRICING", and "DOCS" in white on the right. Below the navigation bar, the main content area has a light gray background. It features a large, semi-transparent watermark-like graphic in the center. This graphic contains various mathematical and engineering icons, such as a lightbulb, a calculator, a ruler, a triangle, a pencil, a gear, a cloud, a server, and a brain. Overlaid on this graphic is the text "Eliminate engineering bottlenecks in Deep Learning". In the bottom left corner of the screenshot, there's a cartoon illustration of a man with glasses and a whiteboard behind him. The whiteboard has some text and mathematical formulas like  $f(1)$  and  $\frac{1}{1+t}$ . Below the screenshot, there are two calls-to-action: "Ver todos los productos" and "Ver todos los servicios de Cognitive Services".

Elide engineering bottlenecks  
in Deep Learning

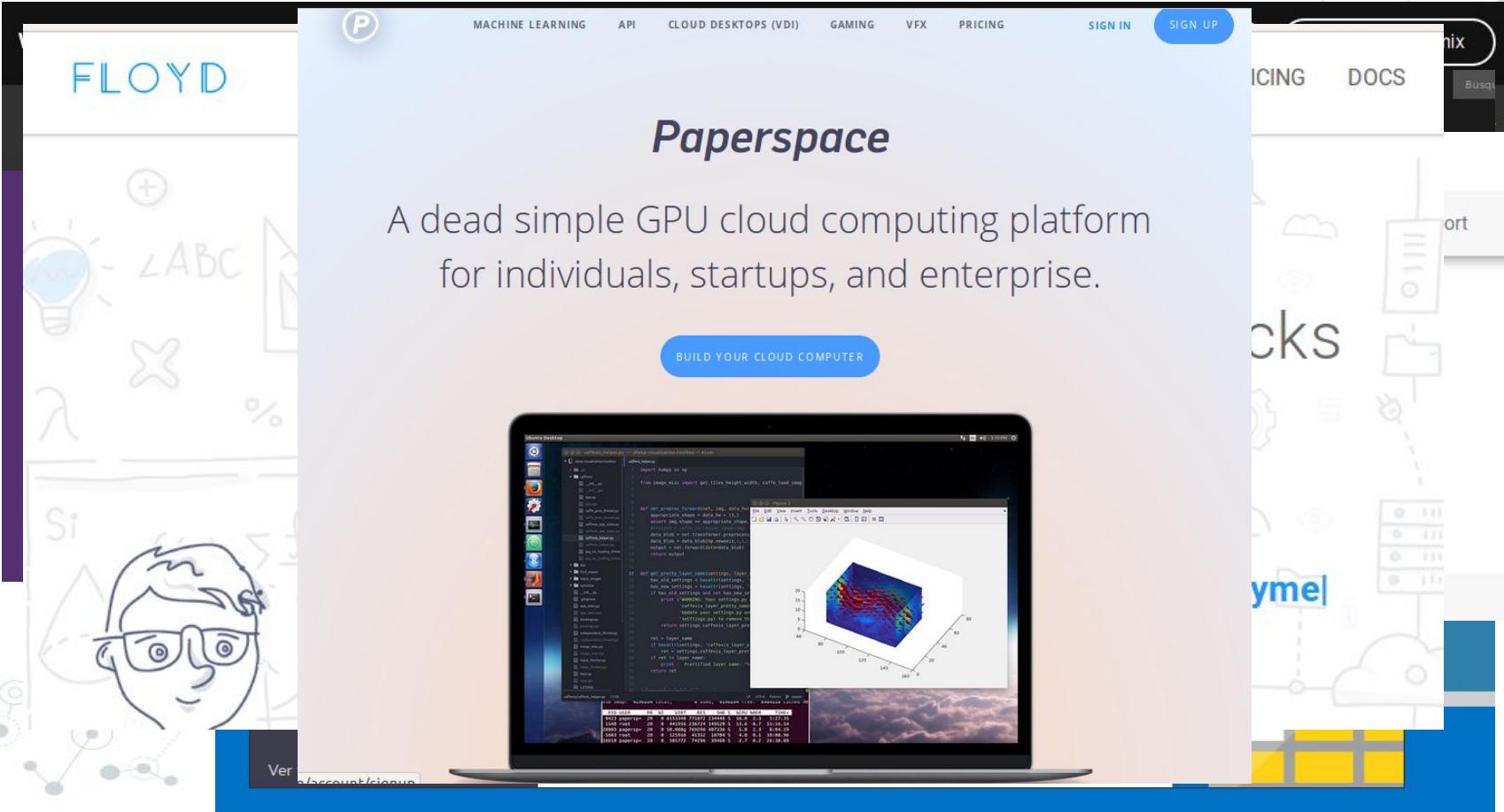
You focus on the **science**

We'll handle the **deployment**

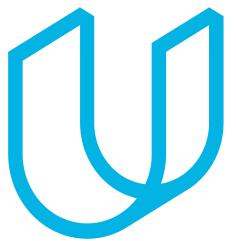
Ver todos los productos

Ver todos los servicios de Cognitive Services

## Cloud machine learning tools

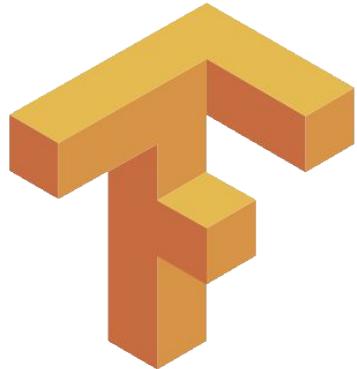


## Additional resources



UDACITY





# Important tools

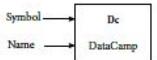
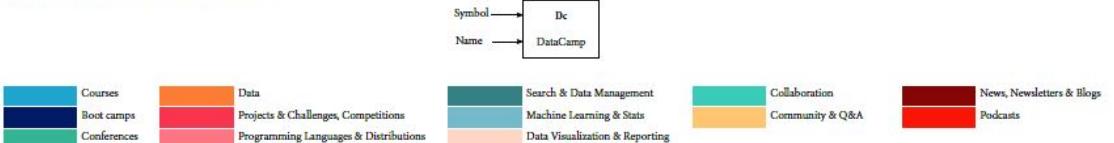
# Tools



## The Periodic Table of Data Science

An overview of key companies, resources and tools in data science (as of 4/12/2017)

Dc DataCamp	Ga General Assembly	Sd Strata Data
Sb SpringBoard	M Metis	OdSC
Ex Fdx Data Incubator	Di Tableau Conference	Tc Tableau Conference
C Coursera	In Insight	U UseR!
Uda Udacity	Da NYC Data Science Academy	Pd PyData
Ude Udemy	G Galvanize	Paw Predictive Analytics World
Ps Pluralsight	Dg Data Science for Social Good	Kdd ACM SIGKDD Conference
Ly Lynda	Dsy Data Society	Tpc Teradata Partners Conference
Tt TeamTreehouse	Djo Data Science Dojo	Icd IEEE International Conference on Data Mining
Bdu Big Data University		



Py Python	Js JavaScript	Vb Visual Basic	Pgs PostgreSQL	Sli SQLite	Ah Apache Hadoop	W Weka	Bml BigML	Kn KNIME	Sm Spark MLLib	Pb Power BI	Ohi Oracle BI	Shn Shiny	Ddl Domino Data Lab	De Data Science Experience
R R	Cp C++	Sc Scala	Ar Amazon Redshift	Bq Google BigQuery	Hw Hortonworks	O Oracle	Dar DataRobot	Lib LIBSVM	Ho H2O	Bo BusinessObjects	Alt Alteryx	Mpl Matplotlib	Nt Nteract	Rs RStudio
S SQL	Pl Perl	Ga Cassandra	Hb HBase	Td Teradata	Ci Cloudera	Mss Microsoft SQL server	Rm RapidMiner	Mat Mathematica	Th Theano	Sp Spotfire	Sav SAS Visual Analytics	Ply Plotly	Ro Rodeo	Be Beaker Notebook
B Bash	Mr Microsoft R Open	P Pig	Mdb Mongo DB	To Tod	Aem Amazon Elastic MapReduce	Spl Splunk	Cho Chorus	Mah Mahout	Aml Azure Machine Learning	Ql Qlikview	Po PowerPivot	Me Microsoft Excel	Spy Spyder	Ze Apache Zeppelin
Mtl Matlab	Cy Canopy	Im Impala	K Kafka	Ms MySQL	Mar MapR	Sr Solr	Tf Tensorflow	St Stata	D D3	Co Cognos	Gch Google Charts	Peh Pentaho	Dst Data Science Studio	Ju Jupyter
J Java	An Anaconda	Sp Spark	Hi Hive	Db IBM DB2	Lu Lucene	Ei ElasticSearch	Sk Scikit-Learn	Da Data/GraphLab	My Microstrategy	Aa Adobe Analytics	T Tableau	B Bokeh	Dh Databricks notebook	Gh GitHub

Dw Data.world	Q Quandl	Pte FiveThirtyEight	Ss Socrata	Gp Google Public	Dg Data.gov	K Kaggle	Re Reddit	So Stack Overflow	Cv Cross Validated	Qu Quora	Av Analytics Vidhya	Dse Data Science Stack Exchange
St Statista	Uci UCI Machine Learning Repository	Wb World Bank	At Academic Torrents	If Buzzfeed	Dk DataKind	Dd DrivenData	Mu Meetup	Rdm RDataMining				

Kdn KDnuggets	Ibd insideBIGDATA
Rb R-bloggers	Pp PlanetPython
Ha Hacker News	Dt DataTau
Dsc Data Science Central	Dr Data Science Roundup
Dsw Data Science Weekly	Or O'Reilly
Dr Data Blixir	Pw Python Weekly
Rw R Weekly	Pd Partially Derivative
Bds Becoming a Data Scientist	Tm Talking Machines
Ds Data Stories	Dsk Data Skeptic
Ld Linear Digressions	No Not So Standard Deviations



# Courses, bootcamps and conferences

Dc	Ga	Sd
DataCamp	General Assembly	Strata Data
Sb	M	Od
SpringBoard	Metis	ODSC
Ex	Di	Tc
Edx	Data Incubator	Tableau Conference
C	In	U
Coursera	Insight	UseR!
Uda	Dsa	Pd
Udacity	NYC Data Science Academy	PyData
Ude	G	Paw
Udemy	Galvanize	Predictive Analytics World
Ps	Dsg	Kdd
Pluralsight	Data Science for Social Good	ACM SIGKDD Conference
Ly	Dsy	Tpc
Lynda	Data Society	Teradata Partners Conference
Tt	Dsj	Icd
TeamTreeHouse	Data Science Dojo	IEEE International Conference on Data Mining
Bdu		
Big Data University		

# Programming languages, distributions and data management

Py	Js	Vb	Pgs	Sli	Ah	W
Python	JavaScript	Visual Basic	PostgreSQL	SQLite	Apache Hadoop	Weka
R	Cp	Sc	Ar Amazon Redshift	Bq Google BigQuery	Hw	O
R	C++	Scala			Hortonworks	Oracle
S	Pl	Ca	Hb	Td	Cl	Mss Microsoft SQL server
SQL	Perl	Cassandra	HBase	Teradata	Cloudera	
B	Mr Microsoft R Open	P	Mdb Mongo DB	To	Aem Amazon Elastic Mapreduce	Spl Splunk
Mtl	Cy	Im	K	Ms	Mar	Sr
Matlab	Canopy	Impala	Kafka	MySQL	MapR	Solr
J	An	Sp	Hi	Idb	Lu	El
Java	Anaconda	Spark	Hive	IBM DB2	Lucene	ElasticSearch

# Machine learning frameworks and visualization

Bml	Kn	Sm	Pb	Obi	Shn
BigML	Knime	Spark MLlib	Power BI	Oracle BI	Shiny
Dar	Lib	Ho	Bo	Alt	Mpl
DataRobot	LibSVM	H2O	BusinessObjects	Alteryx	Matplotlib
Rm	Mat	Ih	Sp	Sav SAS Visual Analytics	Ply
RapidMiner	Mathematica	Theano	Spotfire		Plotly
Cho	Mah	Aml Azure Machine Learning	Ql	Po	Me Microsoft Excel
Chorus	Mahout		Qlikview	PowerPivot	
Tf	St	D	Co	Gch	Pe
Tensorflow	Stata	D3	Cognos	Google Charts	Pentaho
Sk	Da	My	Aa Adobe Analytics	T	B
Scikit-Learn	Dato/Graphlab	Microstrategy		Tableau	Bokeh

# IDEs and collaborative tools

Ddl Domino Data Lab	De Data Science Experience
Nt Nteract	Rs Rstudio
Ro Rodeo	Be Beaker Notebook
Spy Spyder	Ze Apache Zeppelin
Dst Data Science Studio	Ju Jupyter
Db Databricks notebook	Gh Github

# Data sources and challenges

Dw	Q	Fte	Sa	Gp	Dg	K
Data.world	Quandl	FiveThirtyEight	Socrata	Google Public	Data.gov	Kaggle
St	Uci	Wb	At Academic Torrents	Bf	Dk	Dd
Statista	UCI Machine Learning Repository	World Bank		Buzzfeed	DataKind	DrivenData

# Community

Re Reddit	So Stack Overflow	Cv Cross Validated	Qu Quora	Av Analytics Vidhya	Dse Data Science Stack Exchange
Mu Meetup	Rdm RDataMining				

# Newsletters and podcasts

Kdn KDnuggets	Ibd insideBIGDATA	Dr Data Elixir	Pw Python Weekly
Rb R-Bloggers	Pp PlanetPython	Rw R Weekly	Pd Partially Derivative
Hn HackerNews	Dt DataTau	Bds Becoming a Data Scientist	Tm Talking Machines
Dsc Data Science Central	Dsr Data Science Roundup	Ds Data Stories	Dsk Data Skeptic
Dsw Data Science Weekly	Or O'Reilly	Ld Linear Digressions	Ns Not So Standard Deviations

# Tools used in this session

- Jupyter
- Numpy
- Pandas
- Scikit-learn
- Matplotlib

# Jupyter

Interactive environment to create documents, code, interactive widgets, graphs, texts and equations.



File Edit View Insert Cell Kernel Help

Python 3



Markdown ▾

Cell Toolbar: None ▾

## Simple spectral analysis

An illustration of the [Discrete Fourier Transform](#) using windowing, to reveal the frequency content of a sound signal.

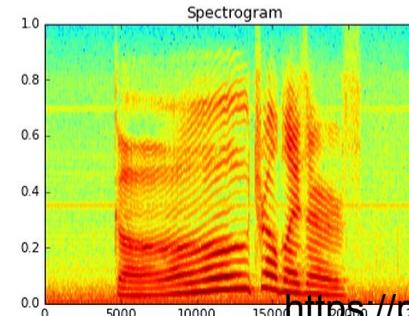
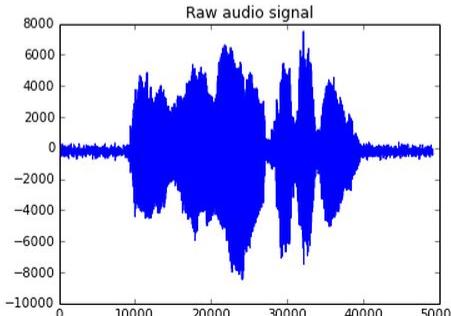
$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi}{N} kn} \quad k = 0, \dots, N-1$$

We begin by loading a datafile using SciPy's audio file support:

```
In [1]: from scipy.io import wavfile  
rate, x = wavfile.read('test_mono.wav')
```

And we can easily view its spectral structure using matplotlib's builtin specgram routine:

```
In [2]: %matplotlib inline  
from matplotlib import pyplot as plt  
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(12, 4))  
ax1.plot(x); ax1.set_title('Raw audio signal')  
ax2.specgram(x); ax2.set_title('Spectrogram');
```



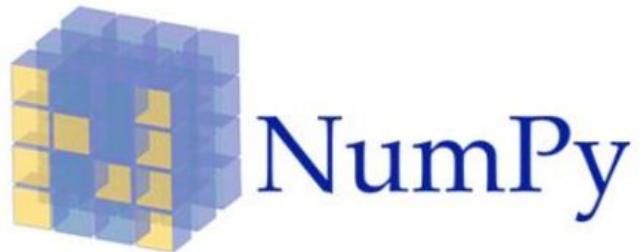
# Tools used in this session

- Jupyter
- Numpy
- Pandas
- Scikit-learn
- Matplotlib

# Numpy

- Array, matrix, vectorial operations.
- Import numpy as np

[0]	[1]	[2]	[3]	[4]
2	5	1	3	4



# Tools used in this session

- Jupyter
- Numpy
- Pandas
- Scikit-learn
- Matplotlib

# Pandas

- Data structure and tools for data analysis.
  - `DataFrames`
  - `GroupBy, merge, join`
  - Data reading and writing
- Import `pandas as pd`



# Tools used in this session

- Jupyter
- Numpy
- Pandas
- Scikit-learn
- Matplotlib

# Scikit-learn

- Machine learning library.
  - Linear regression
  - Logistic regression
  - Decision trees
  - Support vector machines
  - K-Nearest Neighbors
  - Other methods for regression and classification
  - Unsupervised methods
- `Import sklearn`

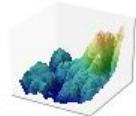
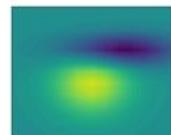
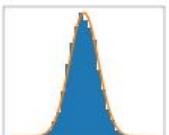


# Tools used in this session

- Jupyter
- Numpy
- Pandas
- Scikit-learn
- Matplotlib

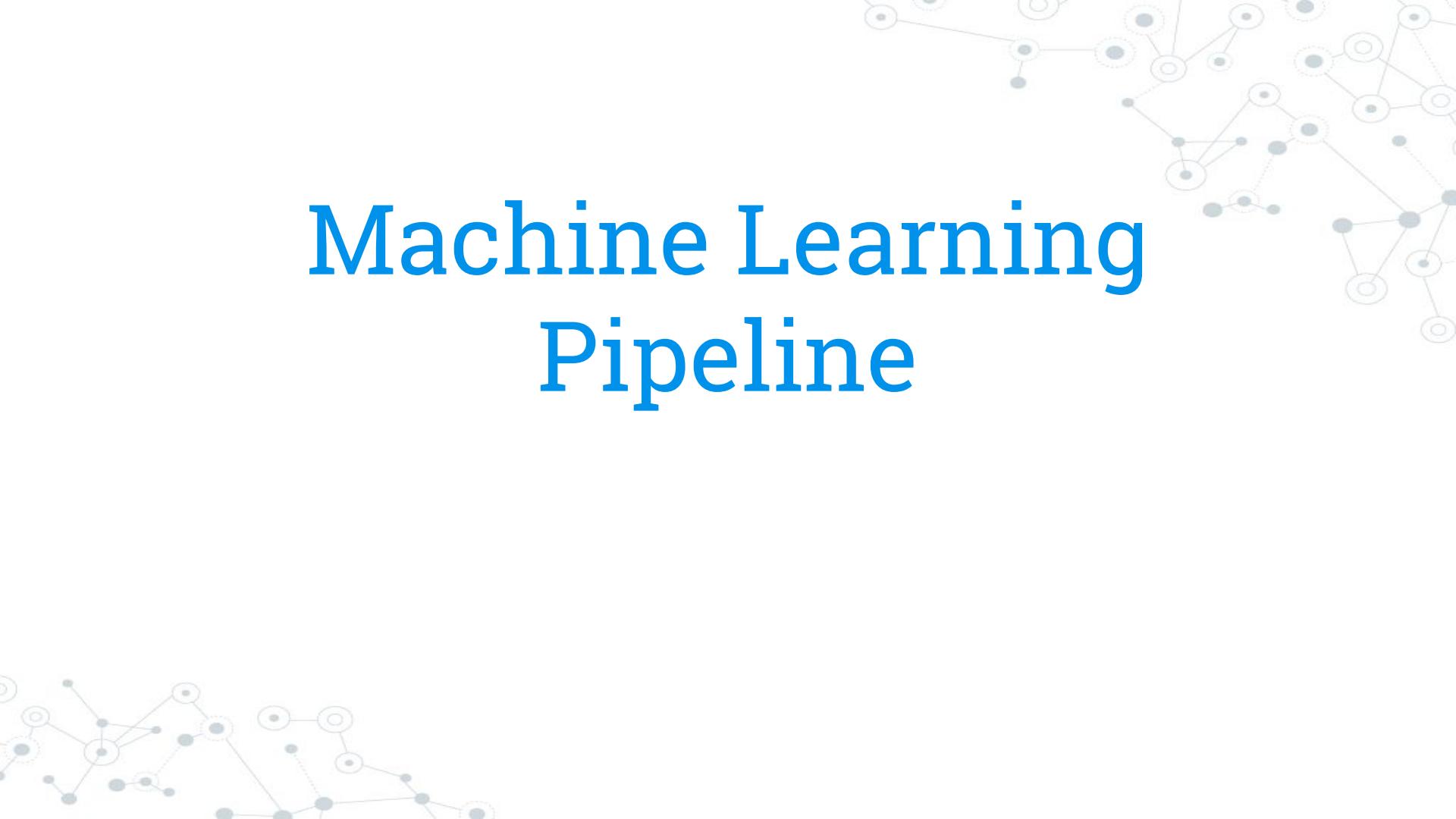
# Matplotlib

- Graphics in Python (Frequency diagram, scatter-plot, 3-D graphics, etc.)
- `Import matplotlib`



**matplotlib**

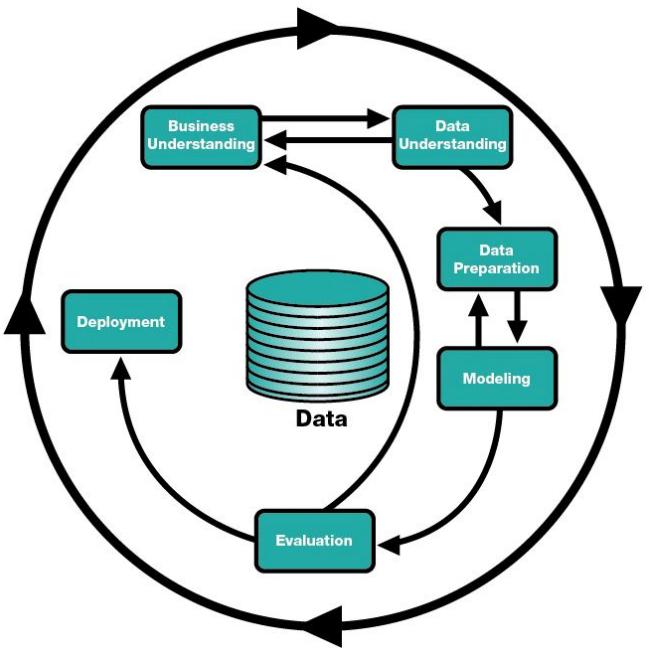
# Machine Learning Pipeline



# Pipeline

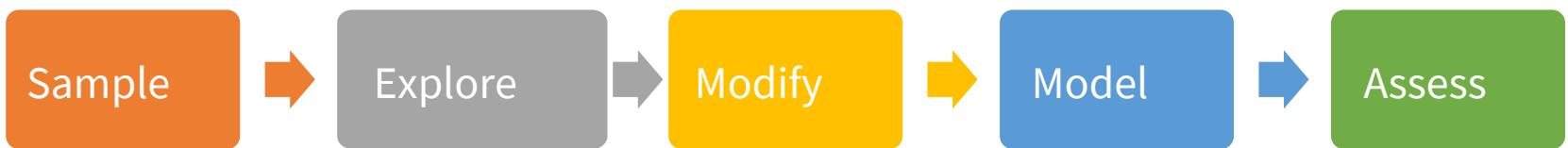


# CRISP-DM Methodology



Source: <http://crisp-dm.eu/>

# SEMMA



# Data acquisition

Raw material: Data



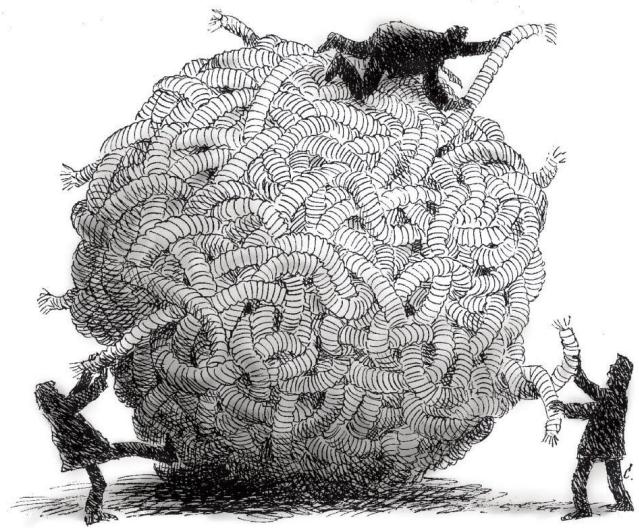
Data come in several formats:

- Web pages
- Video
- Audio
- CSV/TXT/Excel
- Databases

Source:

<https://cdn.sisense.com/wp-content/uploads/messy-data.png>

# Preprocessing



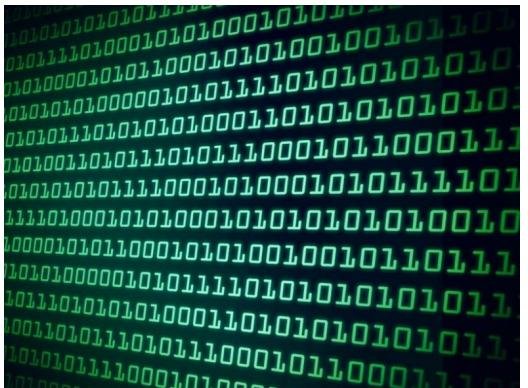
Get data ordered

- Scaling
- Missing values
- Feature extraction
- Feature transformation

source:

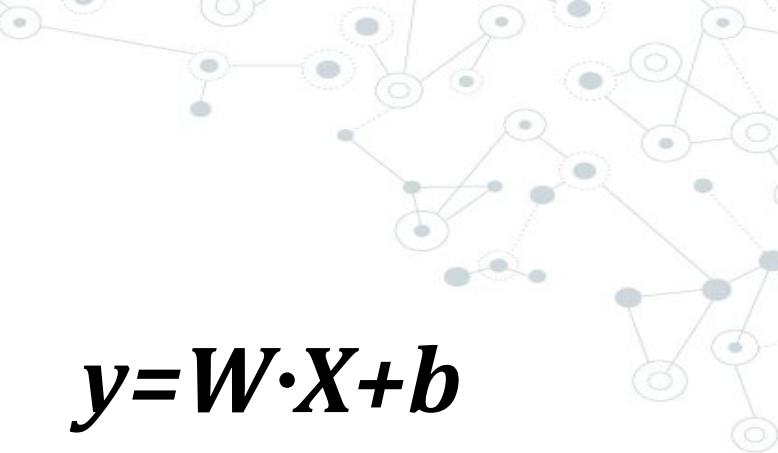
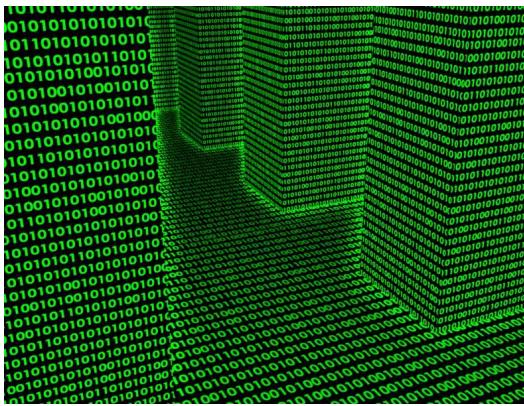
<https://ghotchkiss.files.wordpress.com/2015/04/messymarketing.jpeg>

# Modeling



$$A = \pi r^2$$

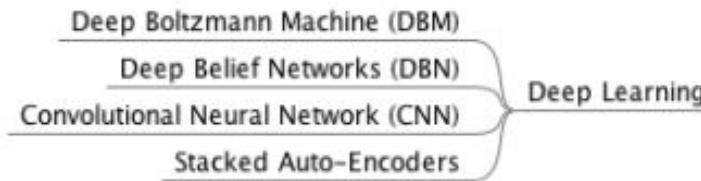
$$y = W \cdot X + b$$



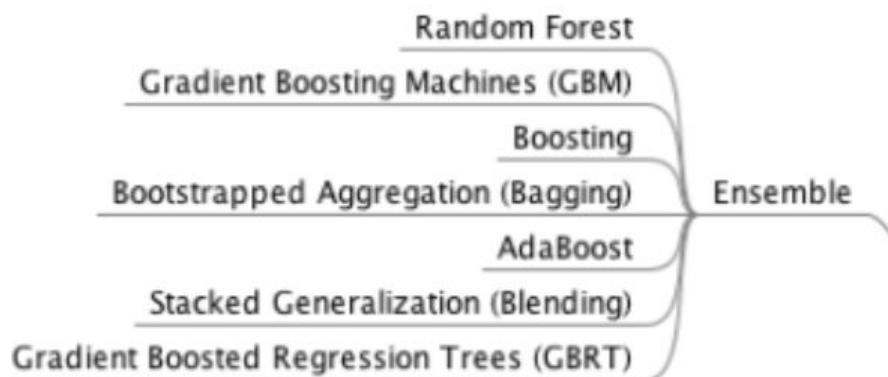


source: <https://jixta.wordpress.com/2015/07/17/machine-learning-algorithms-mindmap/>

# Deep Learning



# Ensemble



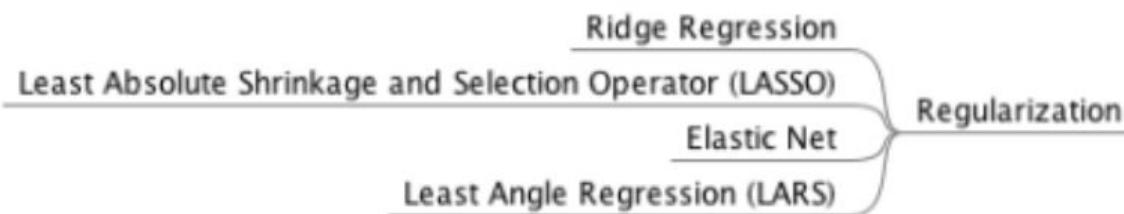
# Neural Networks

Radial Basis Function Network (RBFN)

Perceptron  
Back-Propagation  
Hopfield Network

Neural Networks

# Regularization



# Regression

Linear Regression

Ordinary Least Squares Regression (OLSR)

Stepwise Regression

Multivariate Adaptive Regression Splines (MARS)

Locally Estimated Scatterplot Smoothing (LOESS)

Logistic Regression

Regression

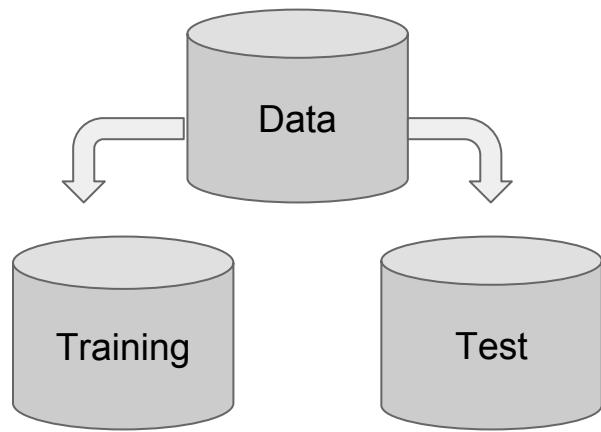
# Validation

Find performance of the modeling stage

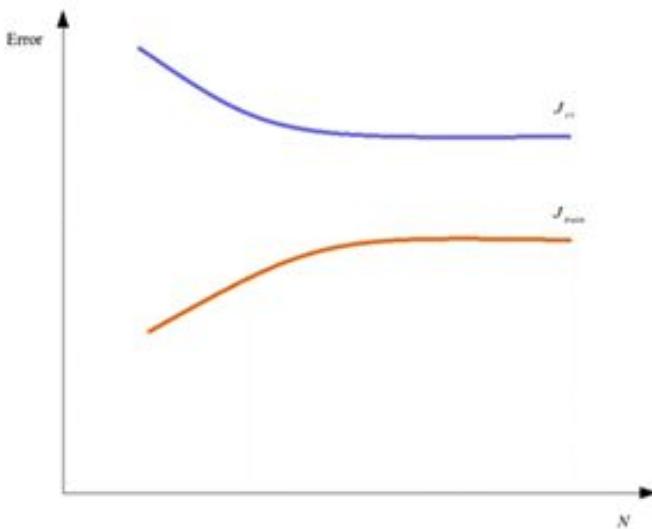
- Mean square error
- Precision/Recall
- F1-score
- ROC Curve



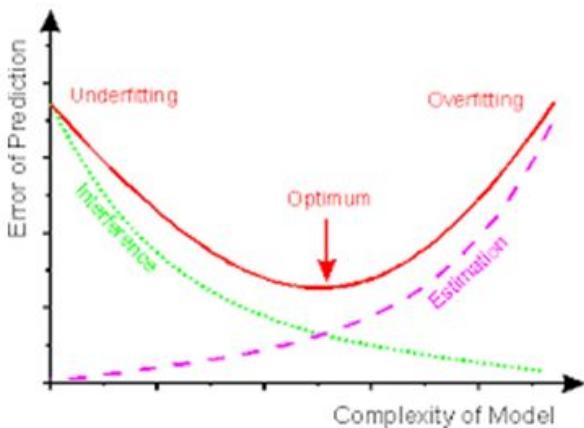
# Generalization



Generalization is the main goal of machine learning:  
predict on new, unseen data.



# Overfitting and underfitting



# Deployment

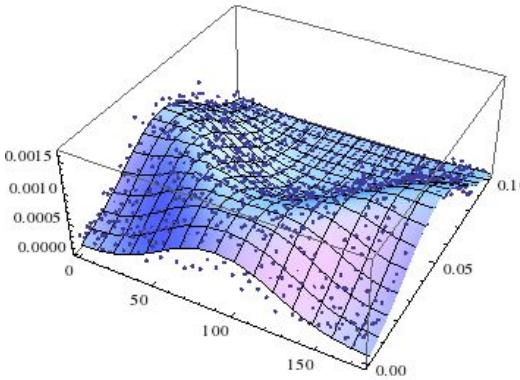
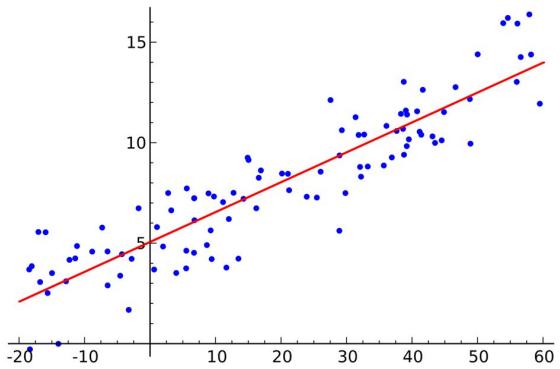


source: <https://marketing4ecommerce.net/como-crear-un-dashboard-para-tu-ecommerce/>

# Dimension Reduction and Data Visualization Demo

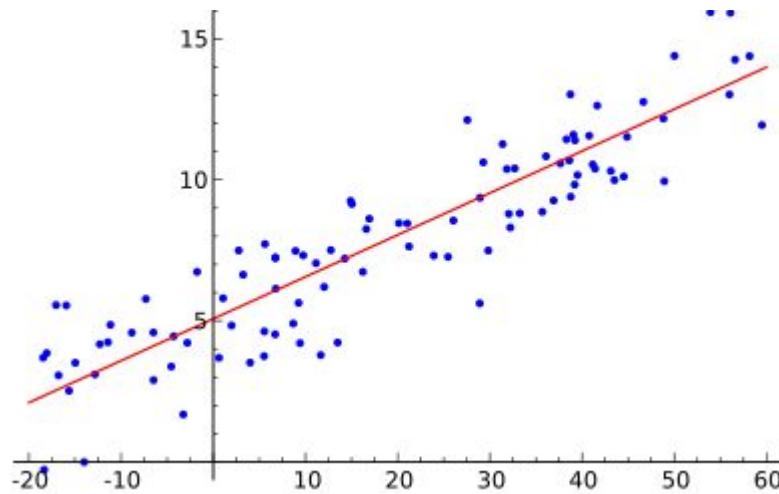
<https://github.com/neuraldevs/ML-ND-CD/blob/master/PythonLibraries/Librer%C3%A3Das.ipynb>

# Regression



## Regression

$$Y = XB + \varepsilon$$



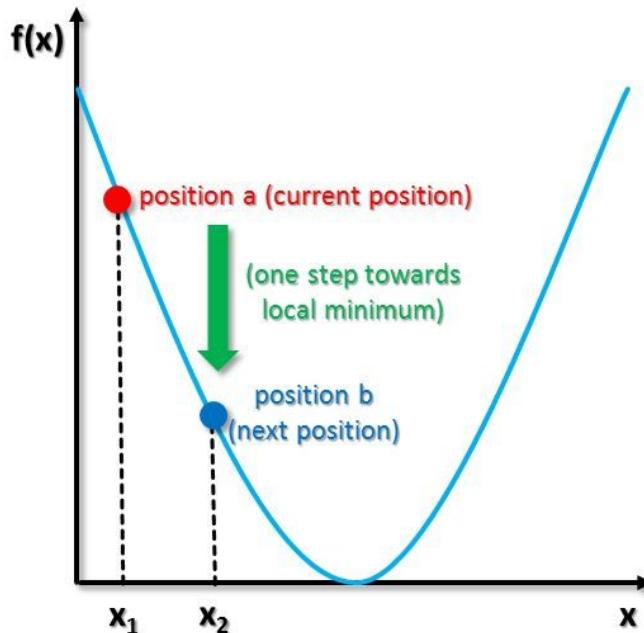
# Regression

(minimization: subtract gradient term because we move towards local minima)

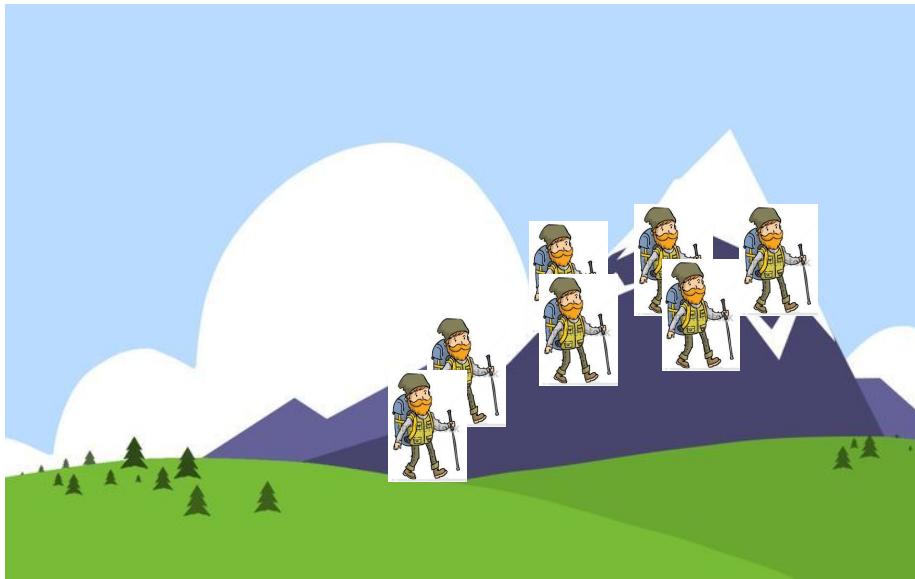
$$b = a - \gamma \nabla f(a)$$

↑   ↑  
new position after the step      old position before the step  
↑  
(weighting factor known as step-size, can change at every iteration, also called learning rate)

(the derivative of  $f$  with respect to  $a$ )  
(gradient term is steepest ascent)



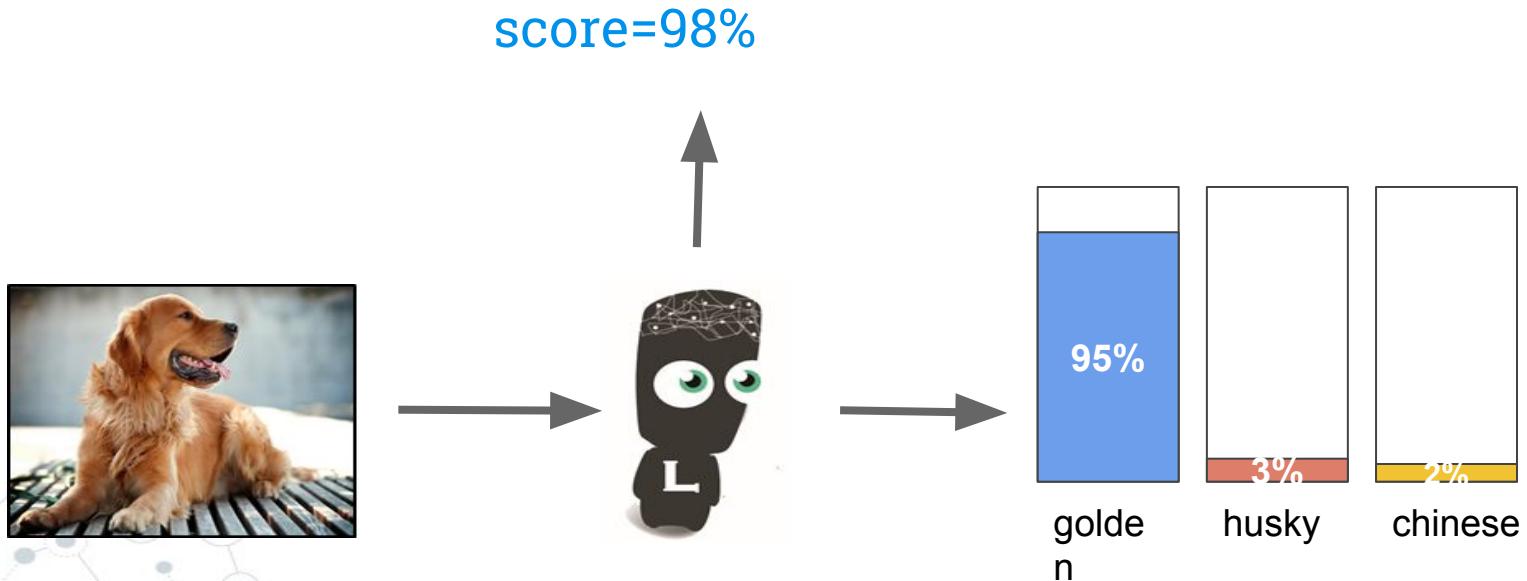
# Gradient Descendent



# Regression Demo

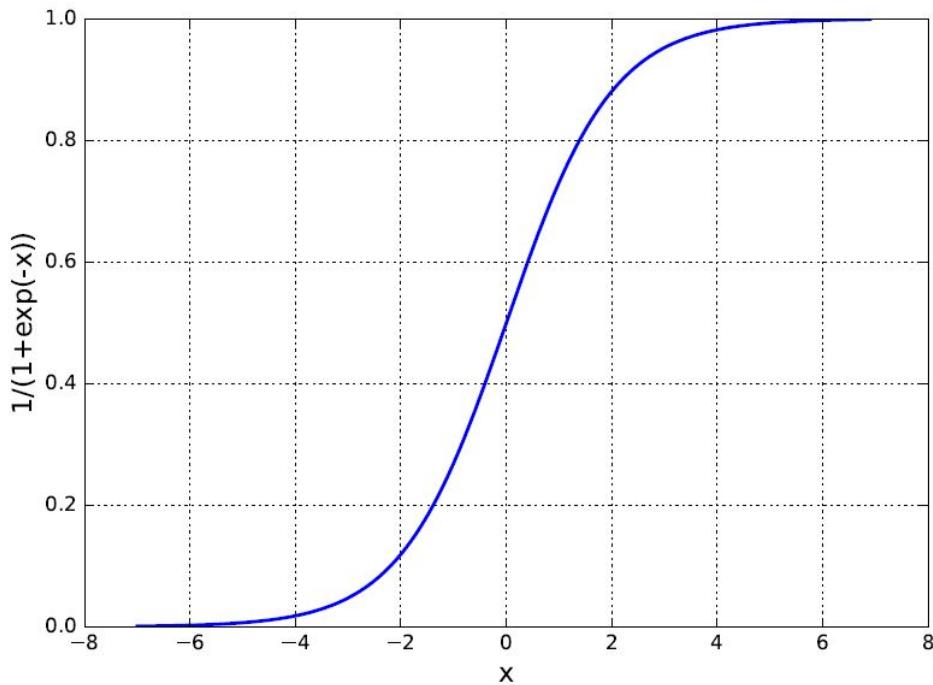
- <https://github.com/neuraldevs/ML-ND-CD/blob/master/Regressi on/RegressionBasic.ipynb>
- <https://github.com/neuraldevs/ML-ND-CD/blob/master/Regressi on/Regression.ipynb>

# Supervised classification



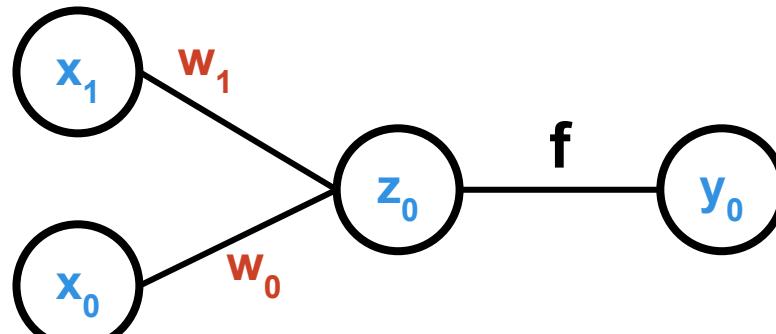
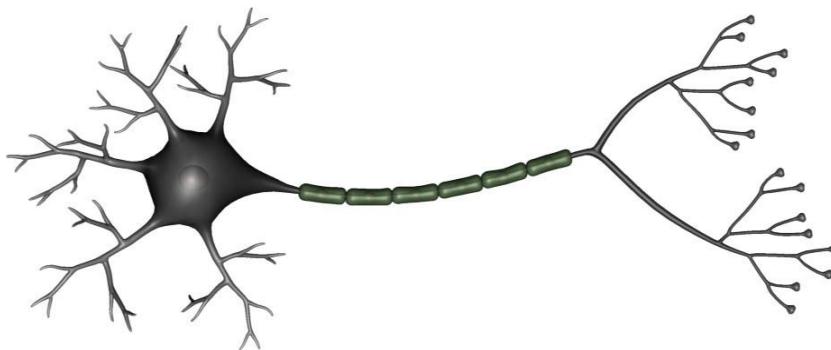
# Methods

## ◎ Logistic regression



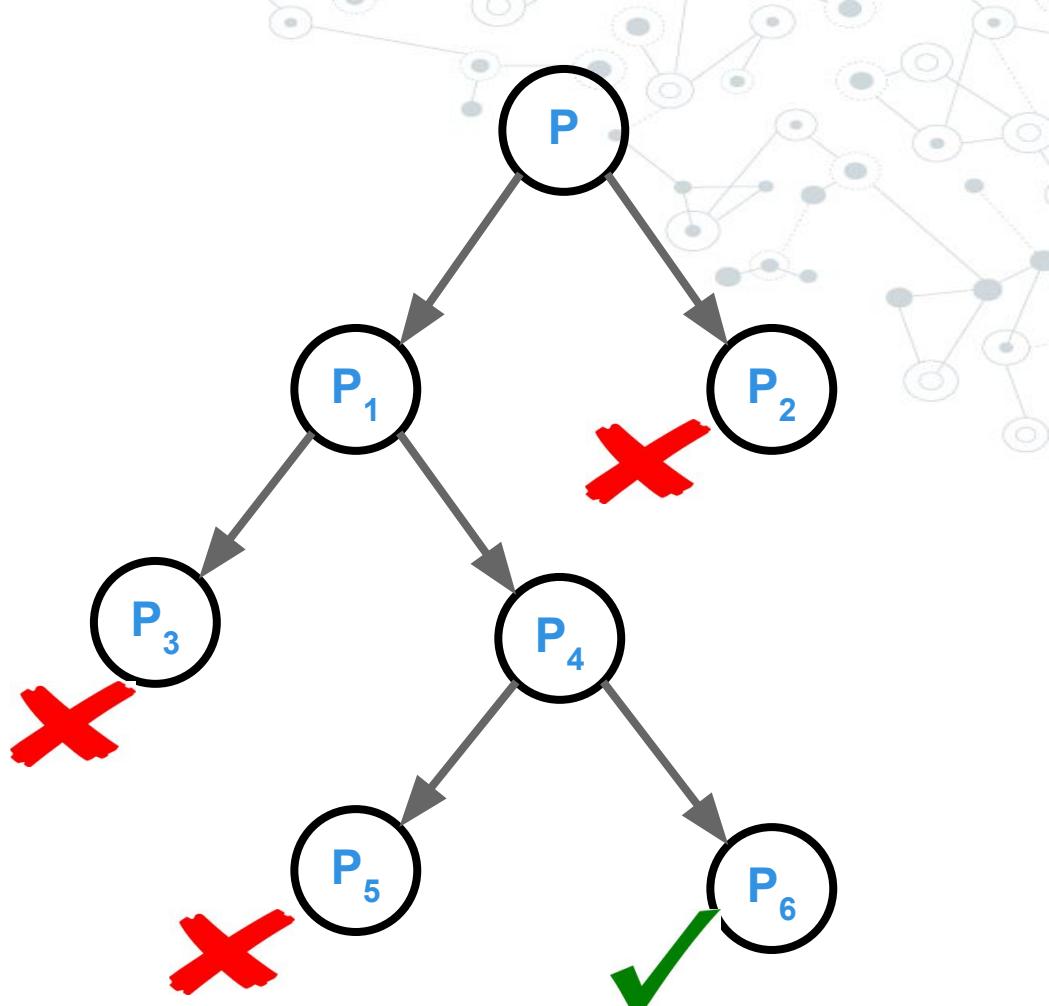
# Methods

## ◎ Neural networks



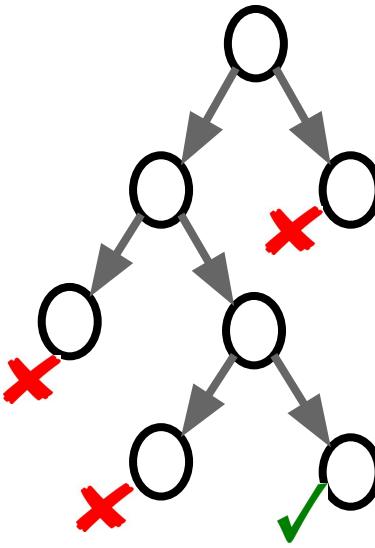
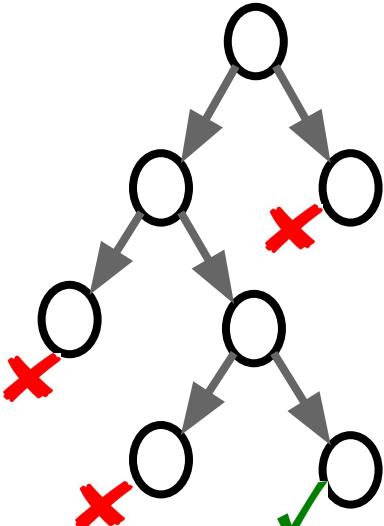
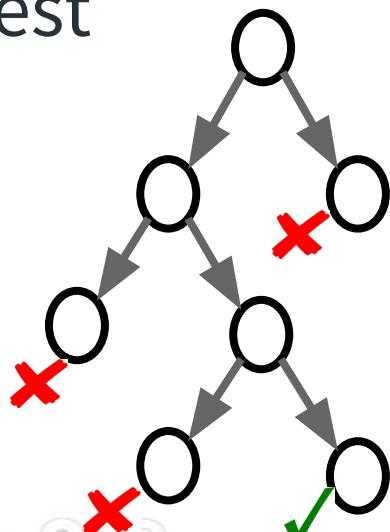
# Methods

## ◎ Decision trees



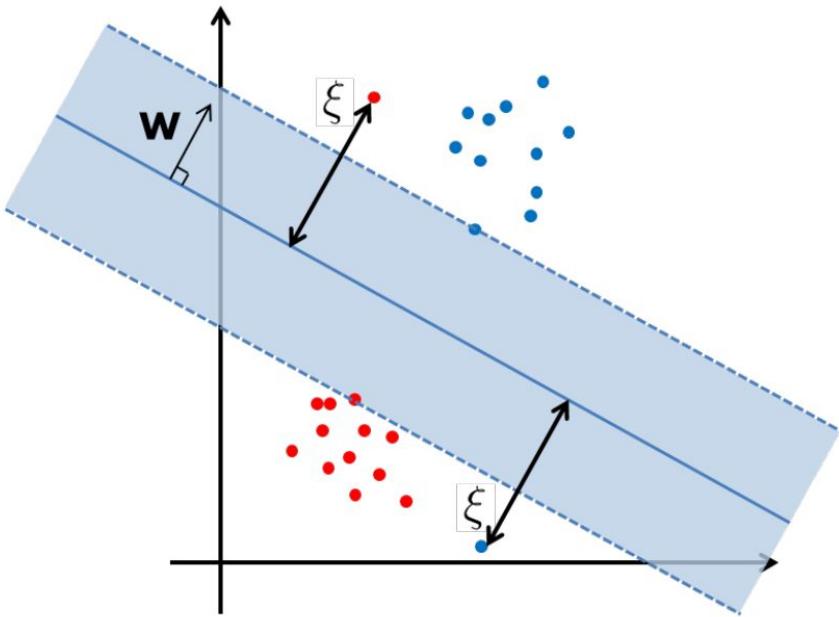
# Methods

# Random Forest



# Methods

## ○ Support vector machines



# Logistic regression

$$W \cdot X + b = y$$

# Logistic regression



$$W \cdot X + b = y$$

# Logistic regression



$$W \cdot X + b = y$$

golden

chinese

# Logistic regression



$$W \cdot X + b = y$$

Weights

bias

golden

chinese

# Logistic regression



$$W \cdot X + b = y = \begin{pmatrix} 2.0 \\ 0.5 \end{pmatrix}$$

The equation  $W \cdot X + b = y$  represents a logistic regression model. An arrow points from the image of the dog to this equation. To the right of the equation, two arrows point outwards: one pointing upwards labeled "golden" and one pointing downwards labeled "chinese", indicating the two classes being predicted by the model.

# Logistic regression



$$W \cdot X + b = y = \begin{pmatrix} 2.0 \\ 0.5 \end{pmatrix}$$

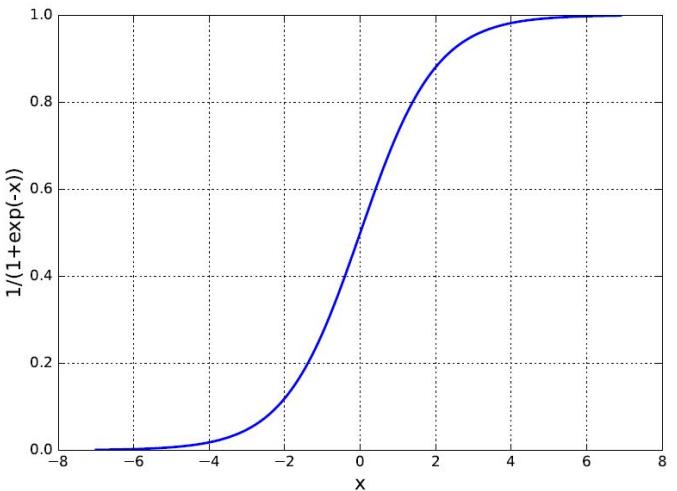
golden ✓  
 $P(X)=0.99$

chinese ✗  
 $P(X)=0.05$

# Logistic regression

$$y = \begin{bmatrix} 2.0 \\ 0.1 \end{bmatrix}$$

$$p(y) = \frac{1}{1+e^{-y}}$$



$$p = \begin{bmatrix} 0.99 \\ 0.05 \end{bmatrix}$$

# Logistic regression

- + Easy to train
- + Free adjustable hyper-parameters
- Could be very simple
- Only for linearly-separable classes

## How to find $W$ and $b$ ?

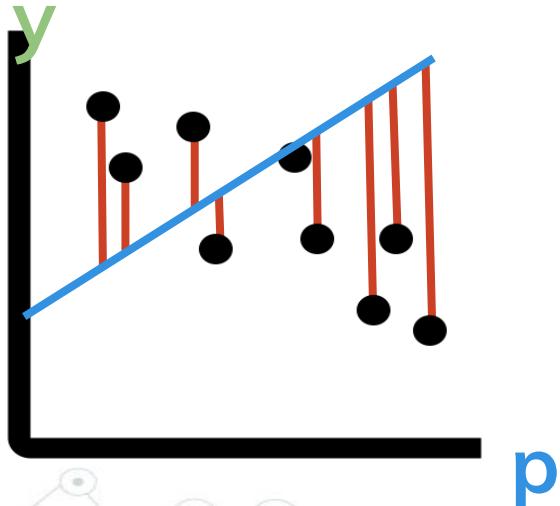
Answer:

Gradient descendent algorithm

1.  $p = f(\mathbf{x})$  #logistic function
2.  $e = E(p, \mathbf{y}; \theta)$  #get error
3.  $\Delta = \nabla_{\theta} e$  #error's gradient / derivative
4.  $\theta := \theta - \alpha \Delta$  #lower error moving against gradient
5. repeat

# How to find $W$ and $b$ ?

Error function: Mean square error



$$E = \frac{\sum (p - y)^2}{N}$$

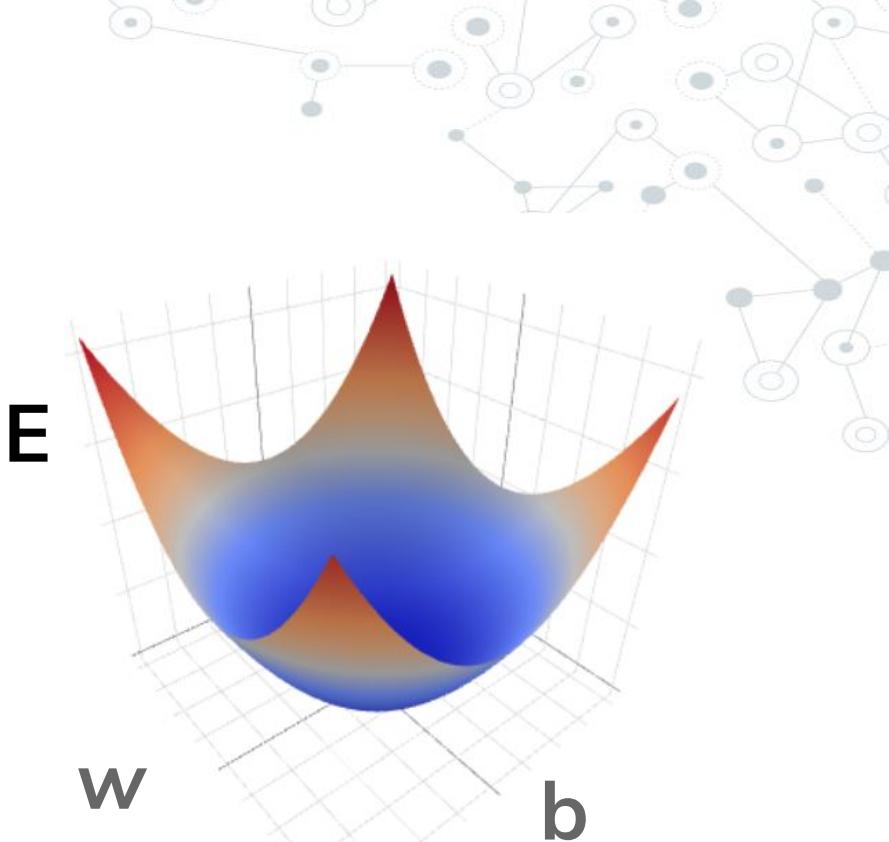
## How to find $W$ and $b$ ?

Error function 2:

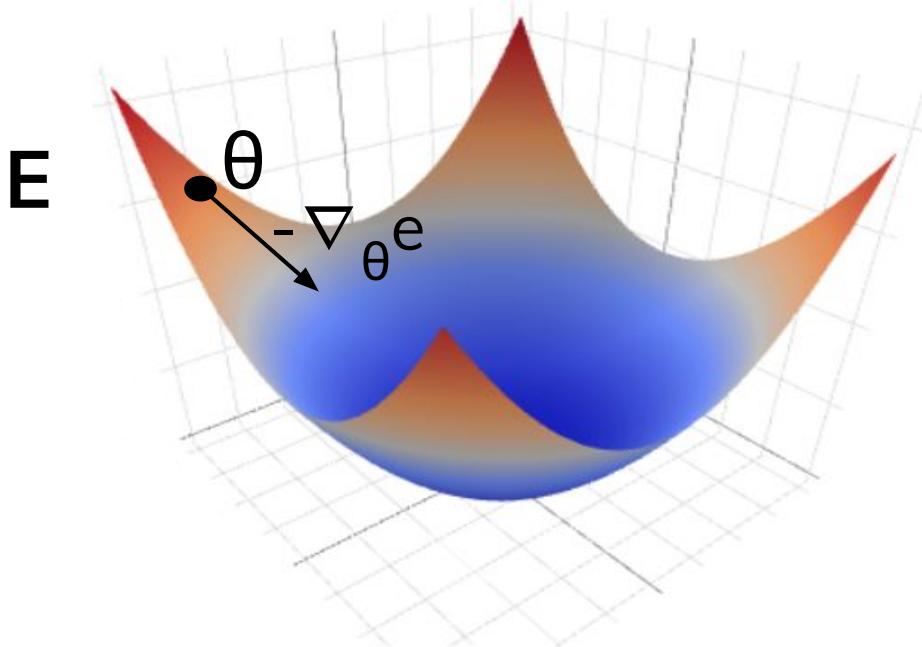
cross-entropy

Distance between two probability  
vectors

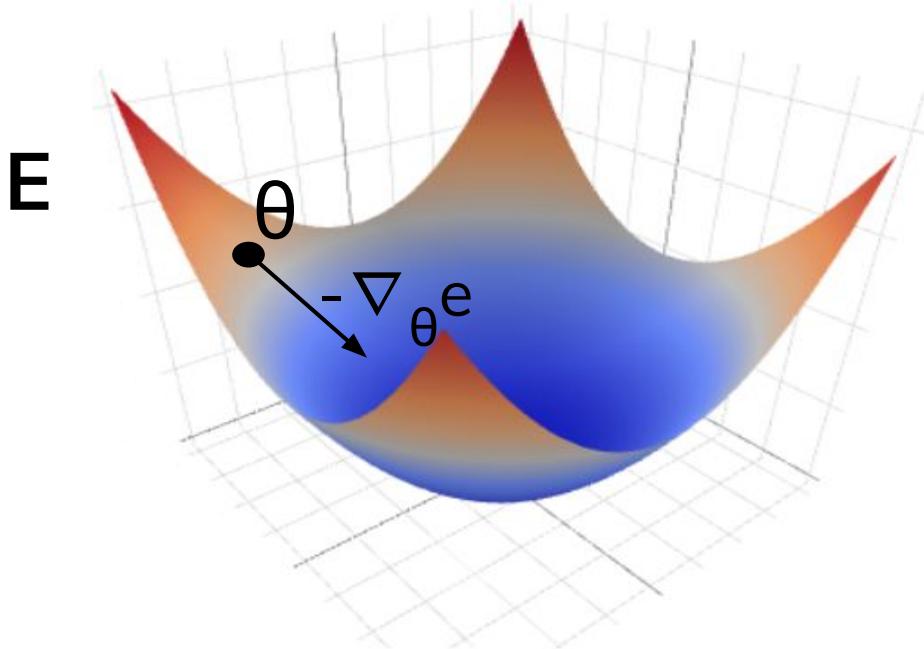
$$E = \frac{\sum (y \log(p))}{N}$$



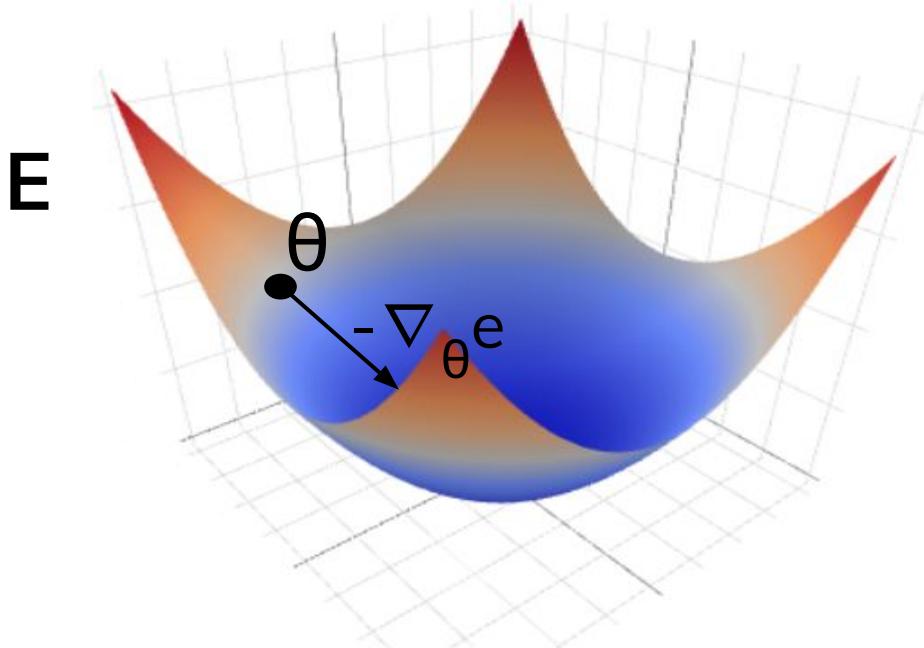
# How to find $W$ and $b$ ?



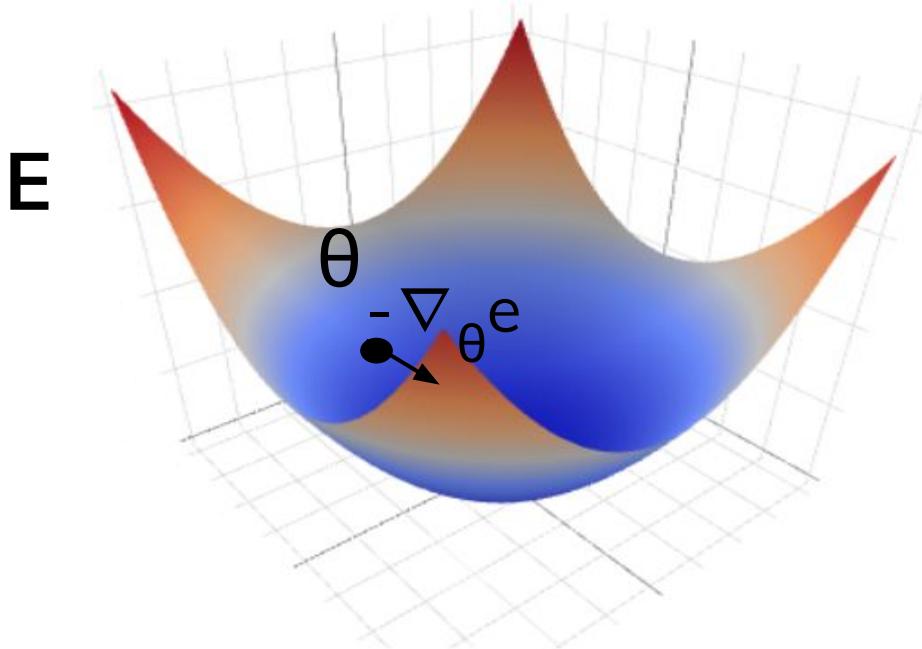
# How to find $W$ and $b$ ?



# How to find $W$ and $b$ ?

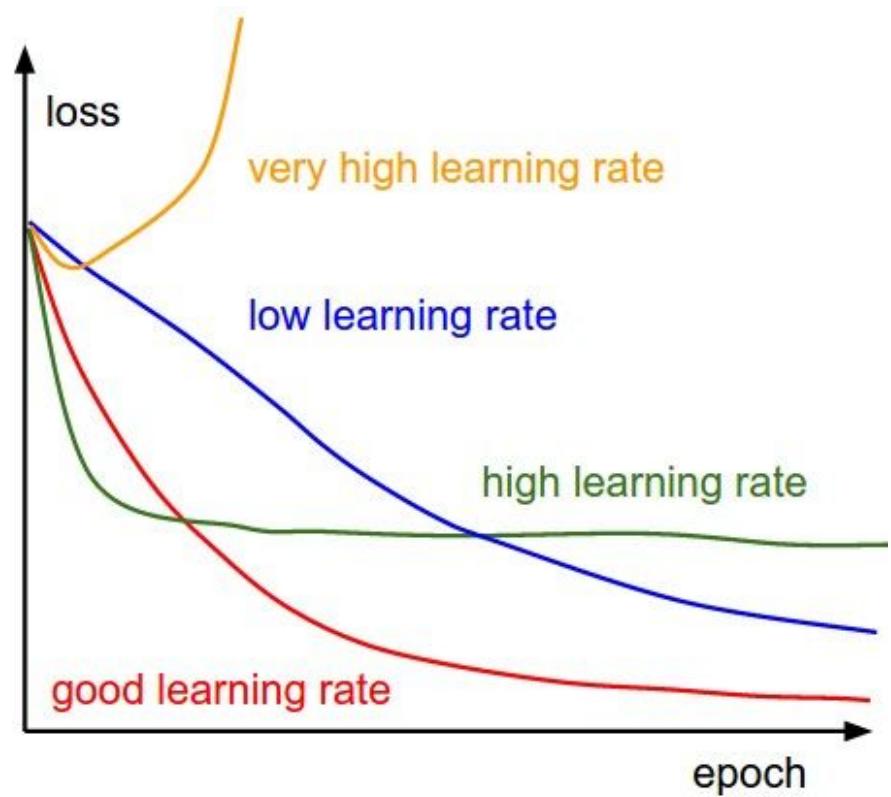


# How to find $W$ and $b$ ?



And so on...

# How to find $W$ and $b$ ?

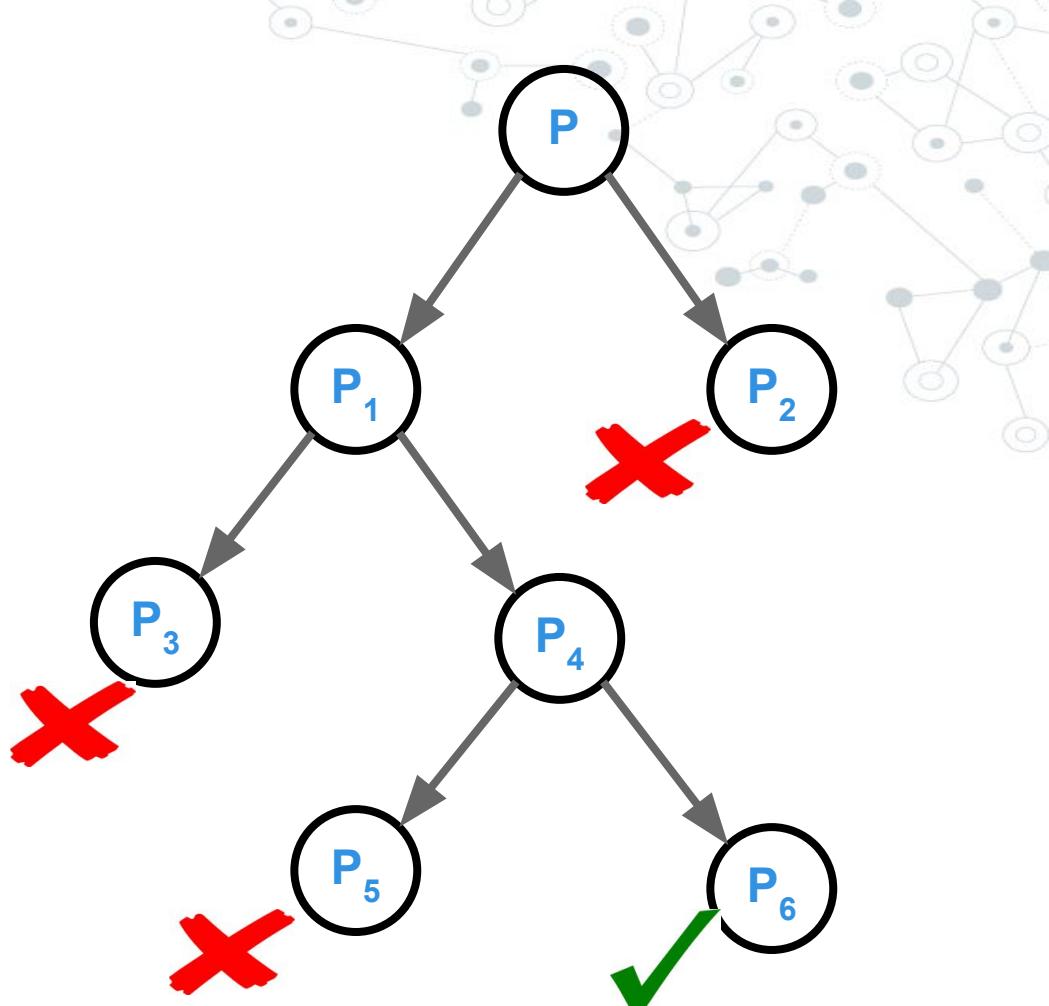


# Logistic regression Demo

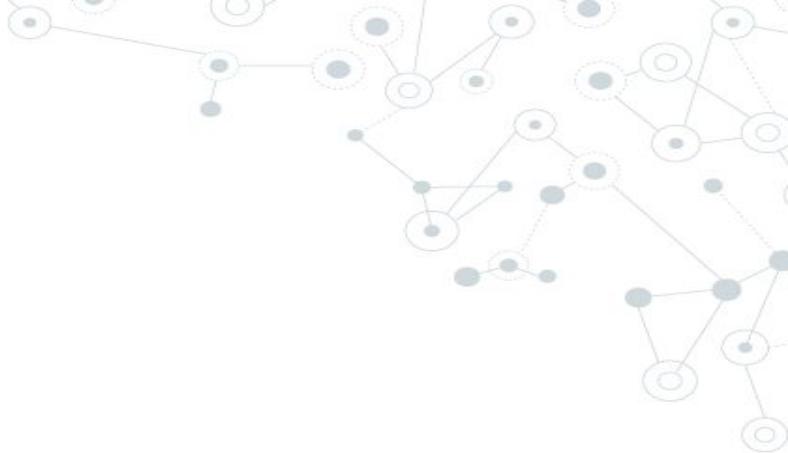
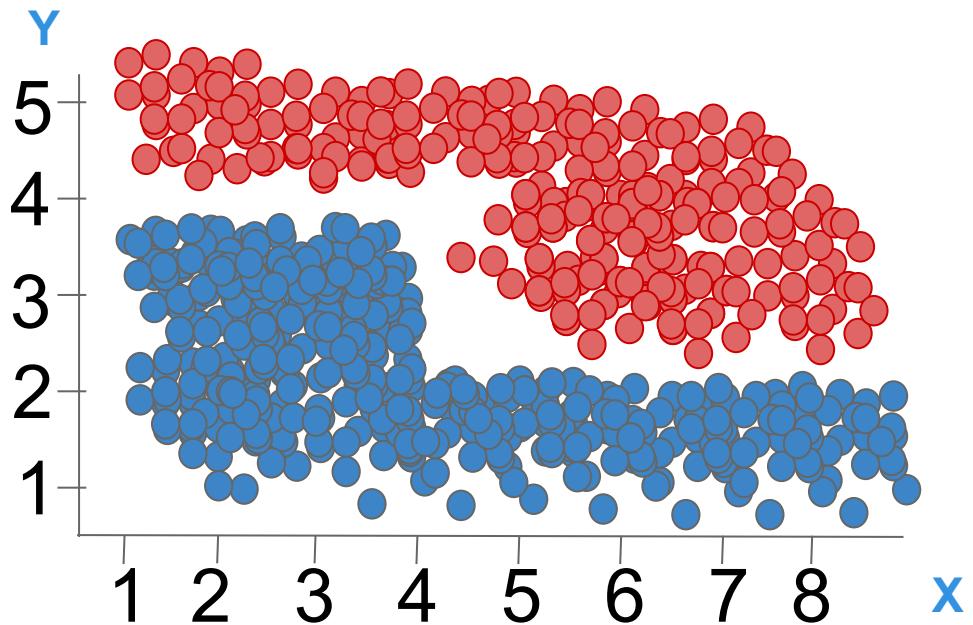
<https://github.com/neuraldevs/ML-ND-CD/tree/master/cancer>

# Methods

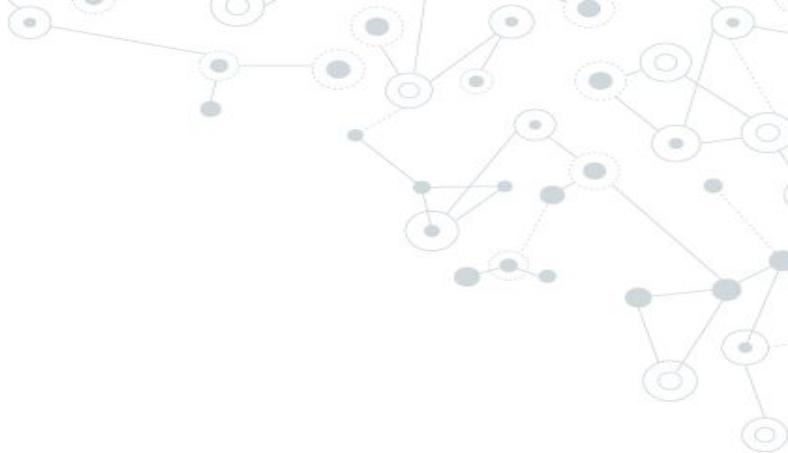
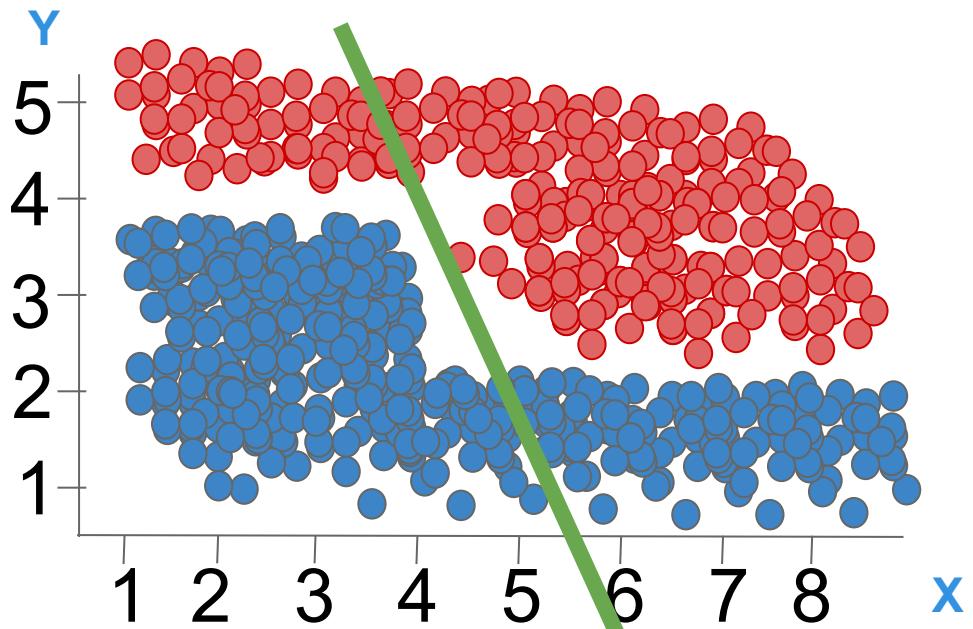
- Decision trees



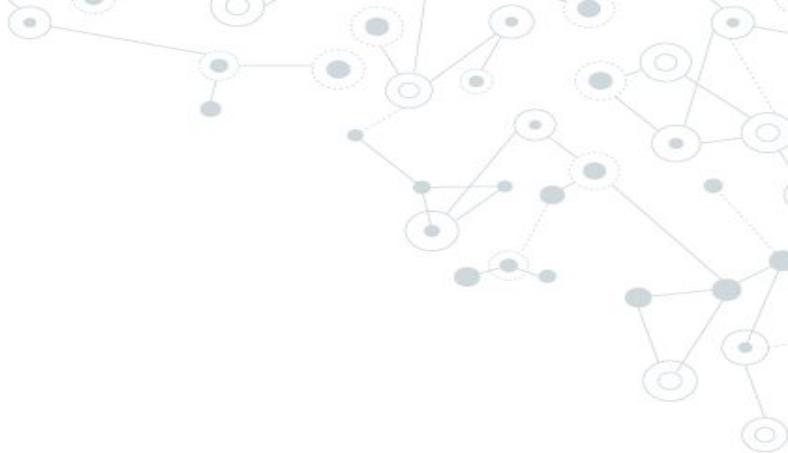
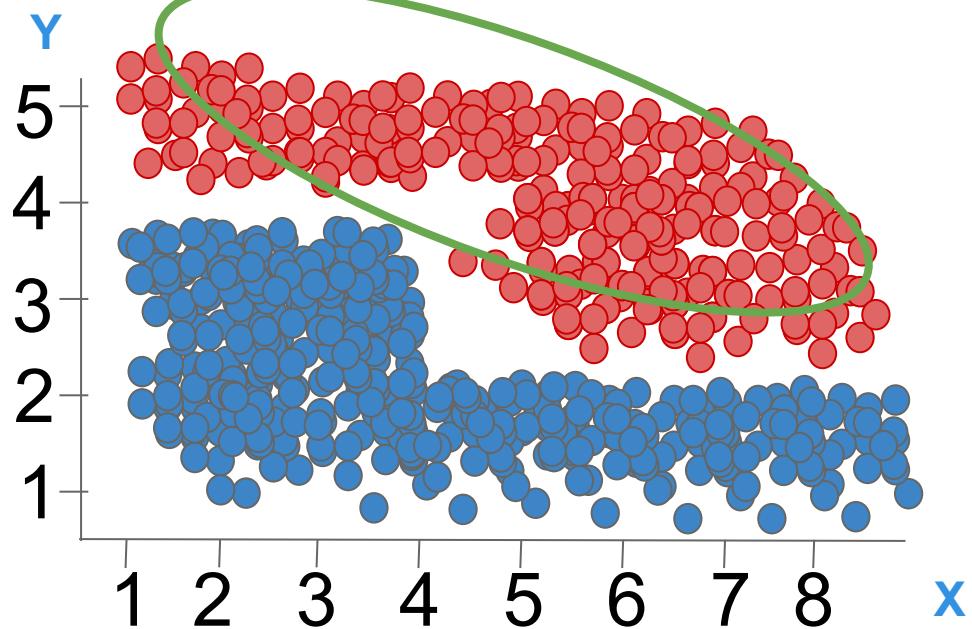
# Decision trees



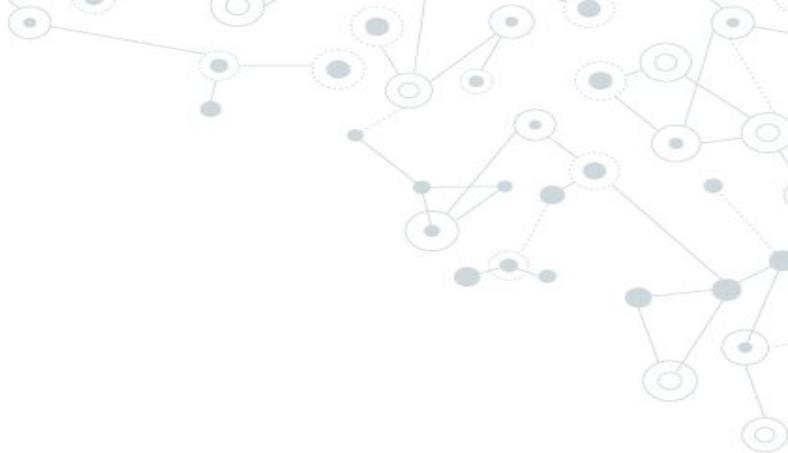
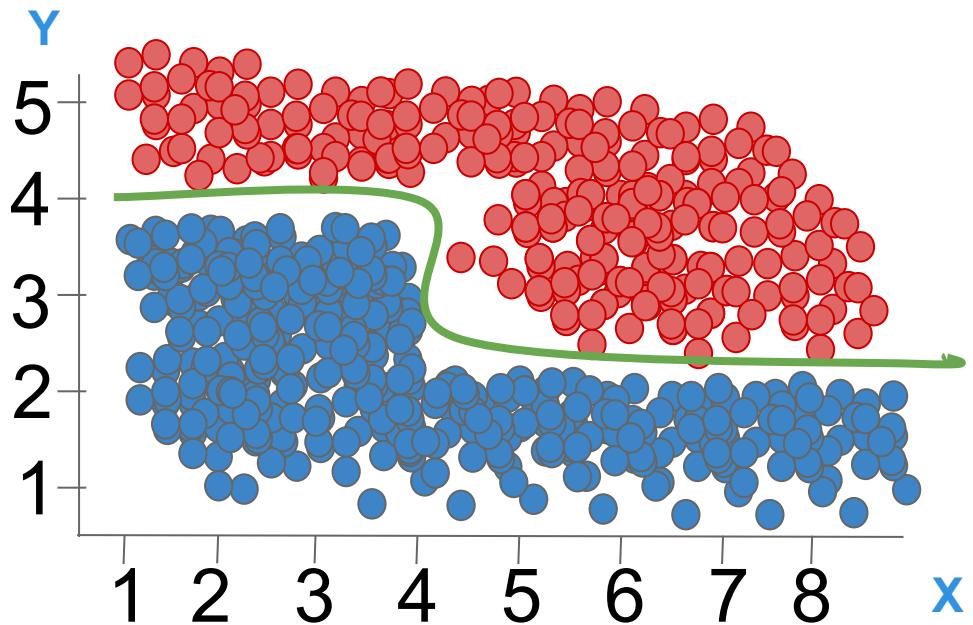
# Decision trees



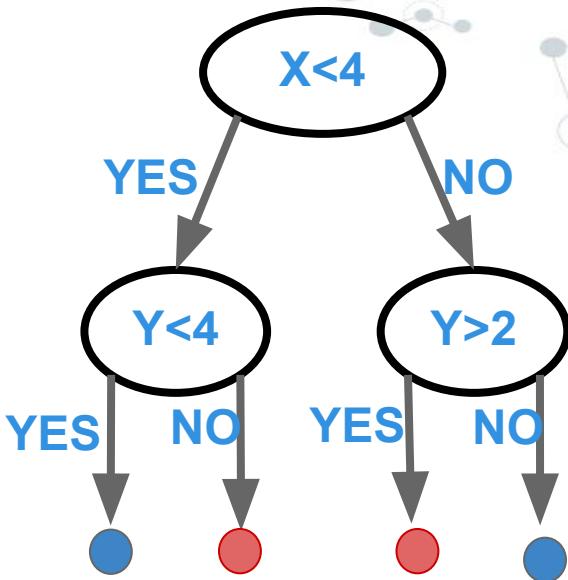
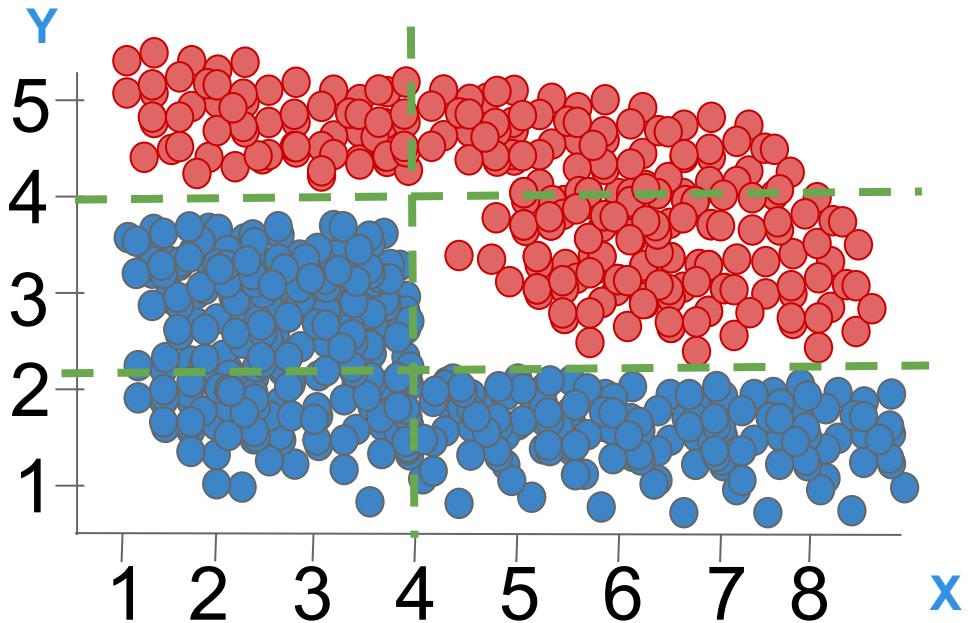
# Decision trees



# Decision trees



# Decision trees



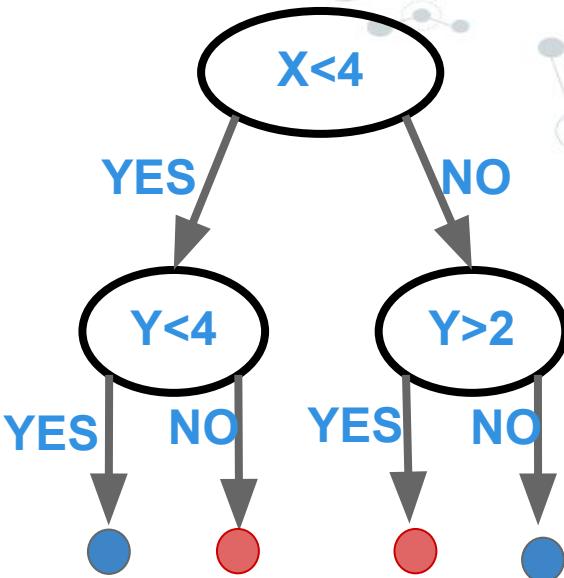
# Decision trees

## Goal:

- Achieve perfect classification with minimal number of decisions

## Advantages:

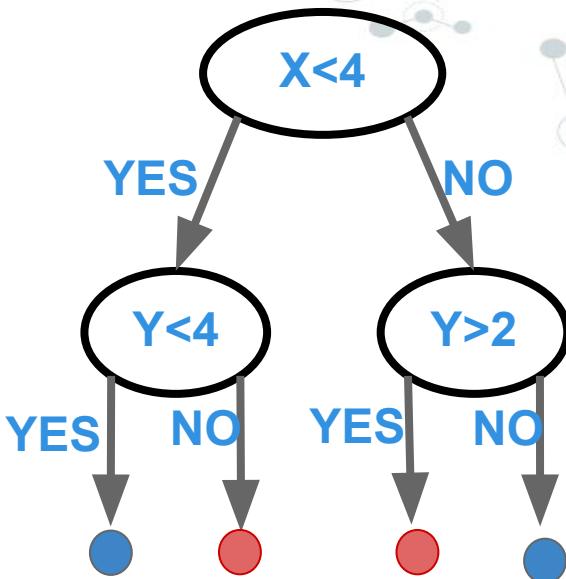
- Simple to understand
- Requires little data preparation
- Trees can be visualized
- Only one adjustable hyper-parameter (number of leafs)



# Decision trees

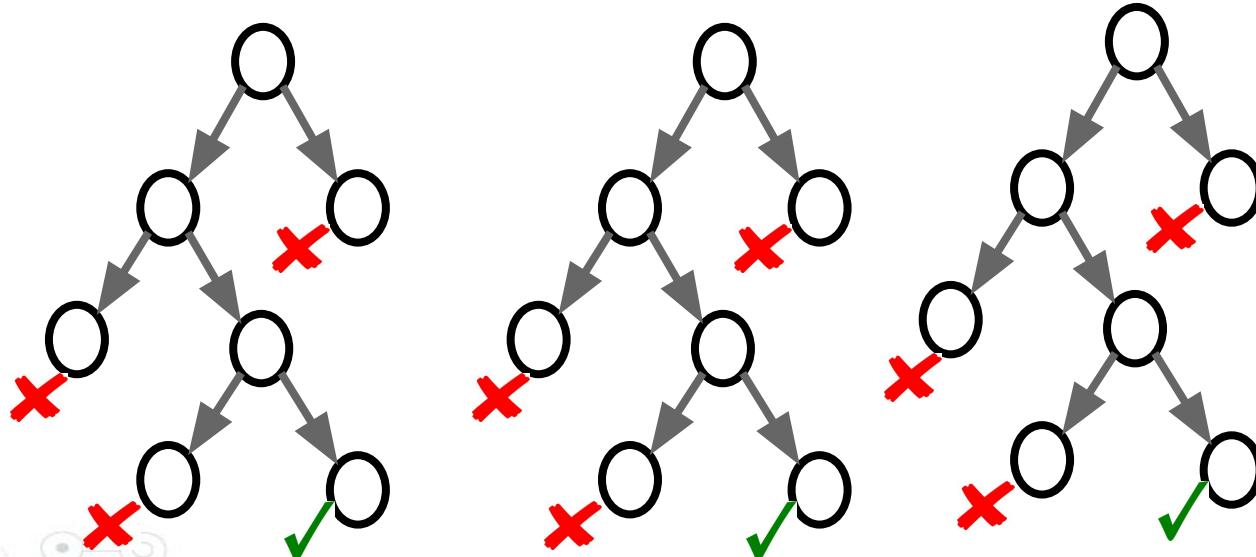
## Disadvantages:

- Over-complex trees that not generalize well
- Unstable due to small variations in training data



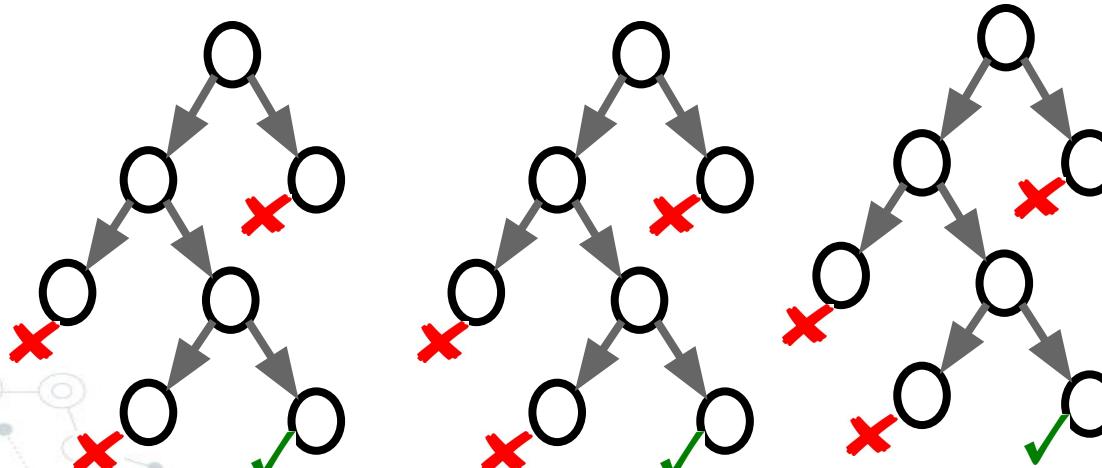
# Random Forest

Ensemble method formed with several decision trees

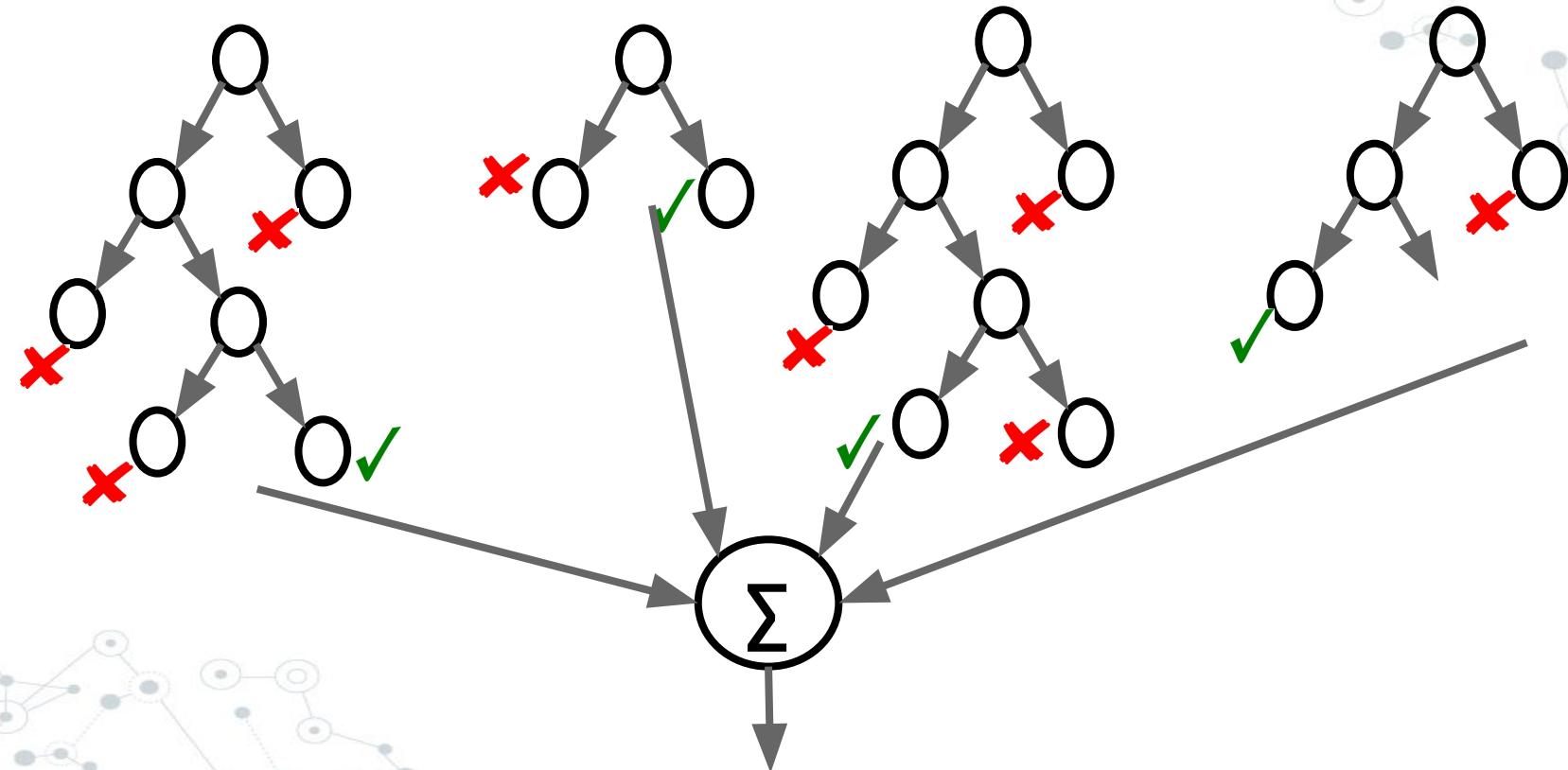


# Random Forest

- ◎ Collection of tree-structured classifiers  $\{h(\mathbf{x}, \theta), k=1, \dots\}$
- ◎  $\theta_k$  are i.i.d. random trees
- ◎ Each tree casts a unit vote for the final classification of input  $\mathbf{X}$
- ◎ The final class of the random forest is chosen is voted by weighted values of each tree



# Random Forest



# Decision tree

- + Trees yield insight into decision rules
- + Rather fast
- + Easy to tune parameters
- Prediction of trees tend to have a high variance

vs.

# Random forest

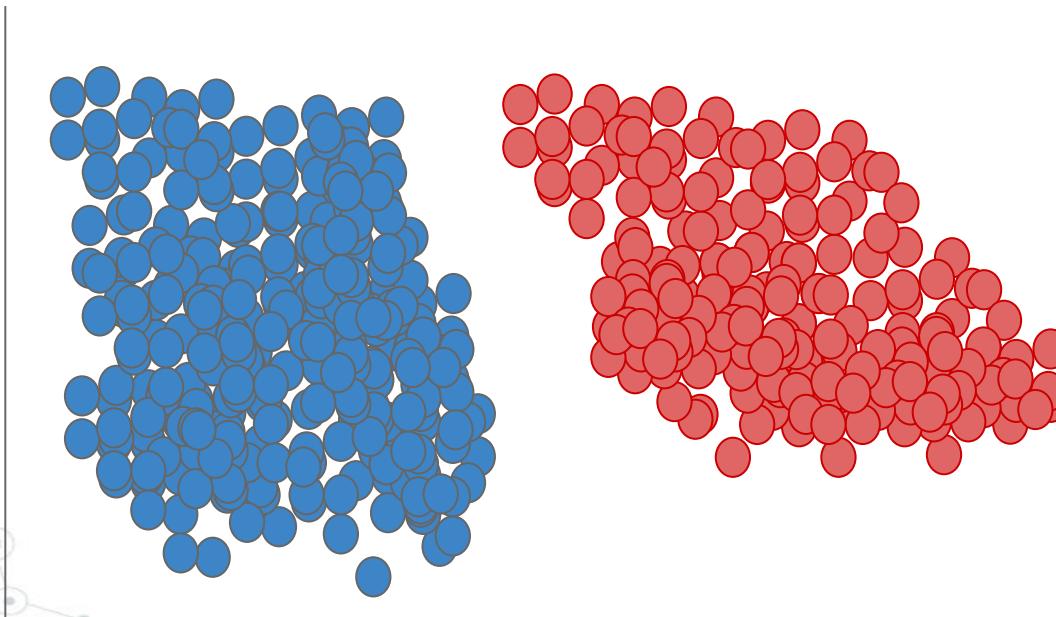
- + Usually a better general performance
- + Easy to tune parameters
- Rather slow
- “Black Box”: Rather hard to get insights into decision rules

# Decision Tree Demo

<https://github.com/neuraldevs/ML-ND-CD/tree/master/cancer>

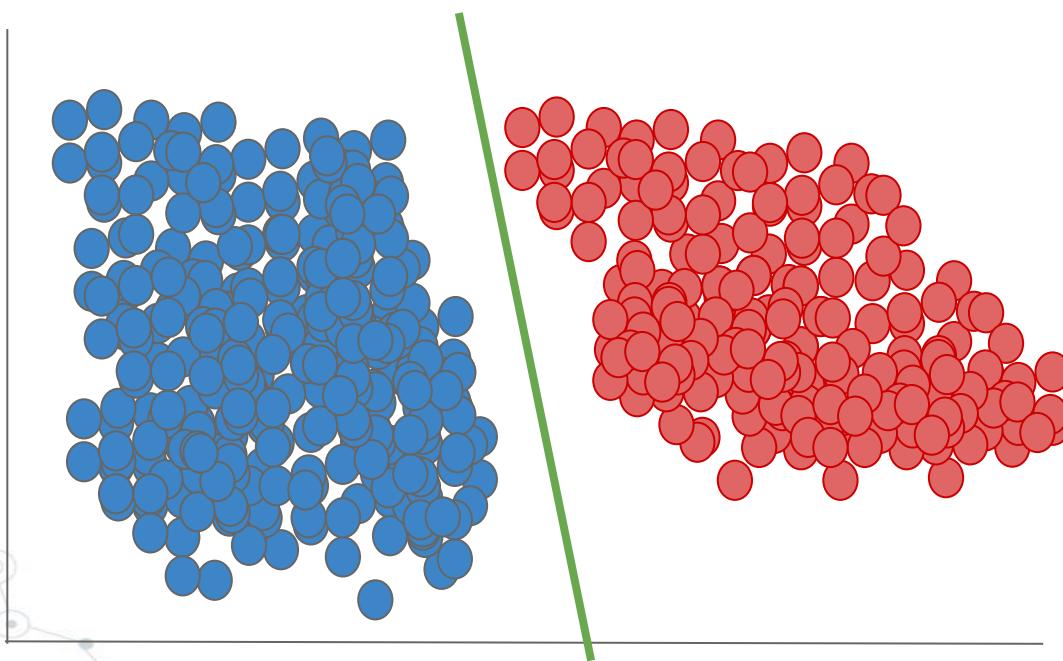
# Support vector machines

Which of the linear separators is optimal?



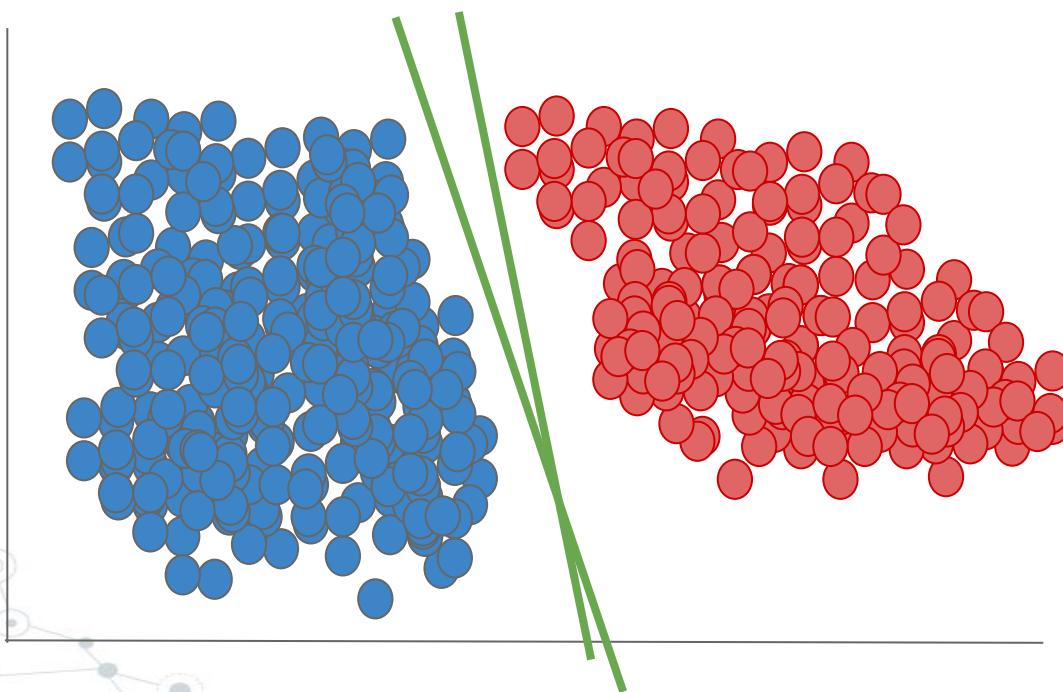
# Support vector machines

Which of the linear separators is optimal?



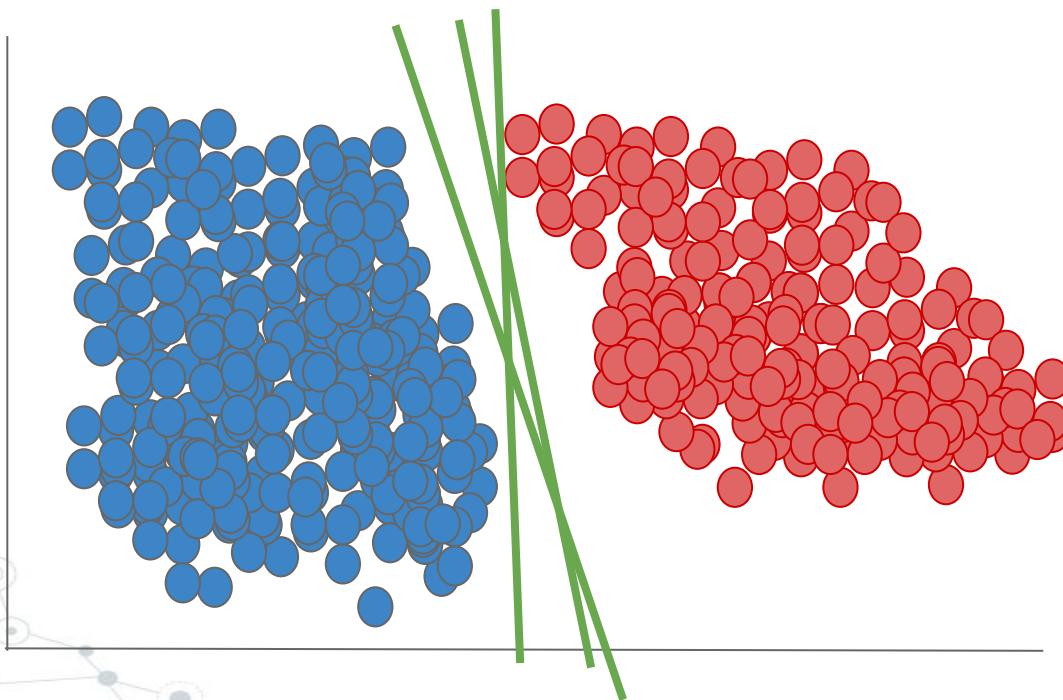
# Support vector machines

Which of the linear separators is optimal?



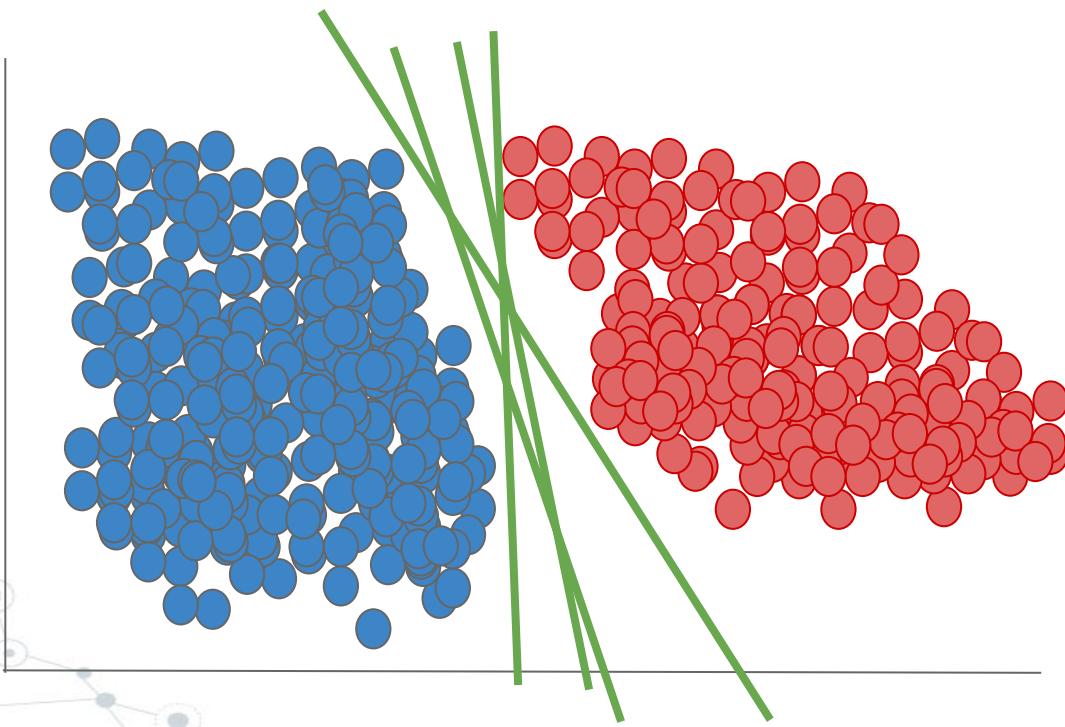
# Support vector machines

Which of the linear separators is optimal?



# Support vector machines

Which of the linear separators is optimal?

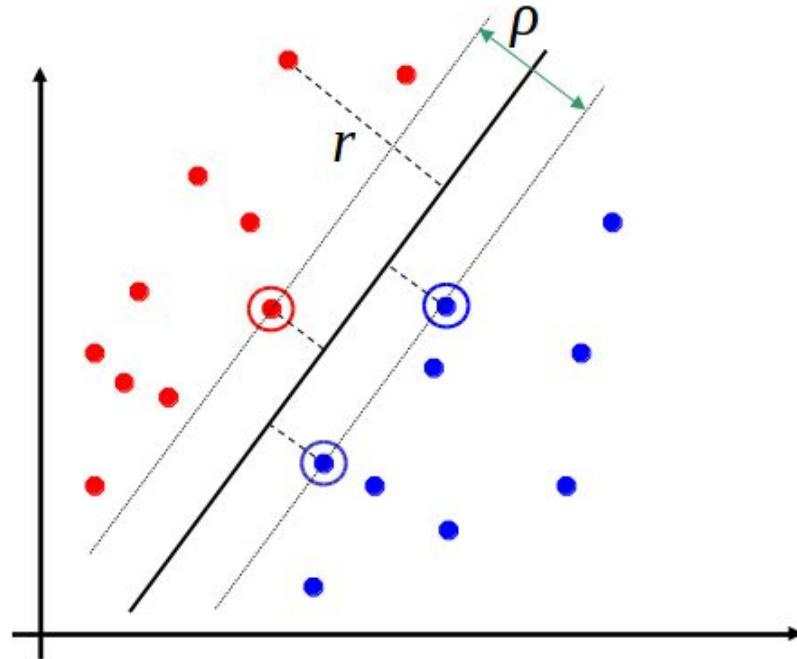


# Support vector machines

Which of the linear separators is optimal?

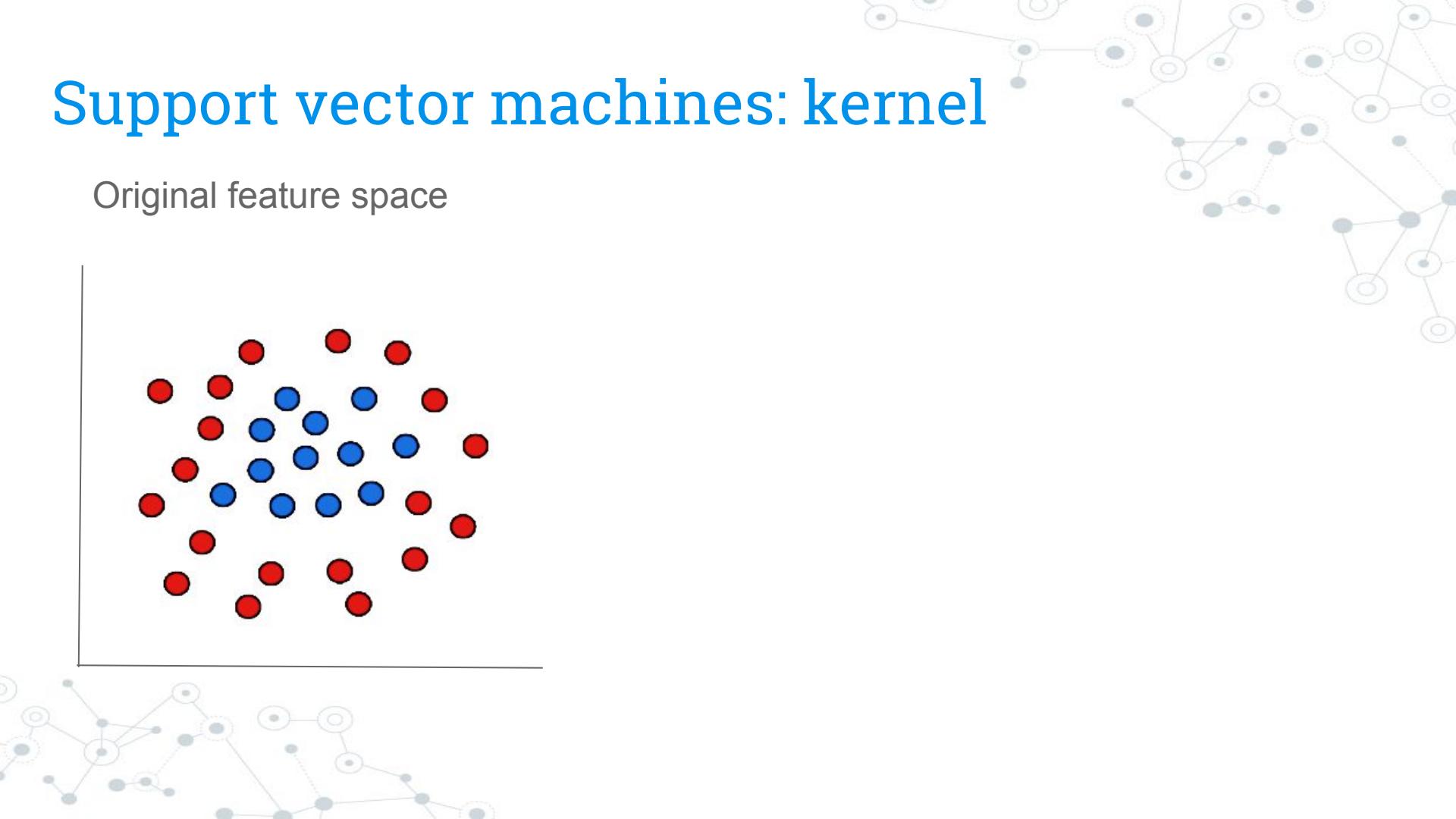
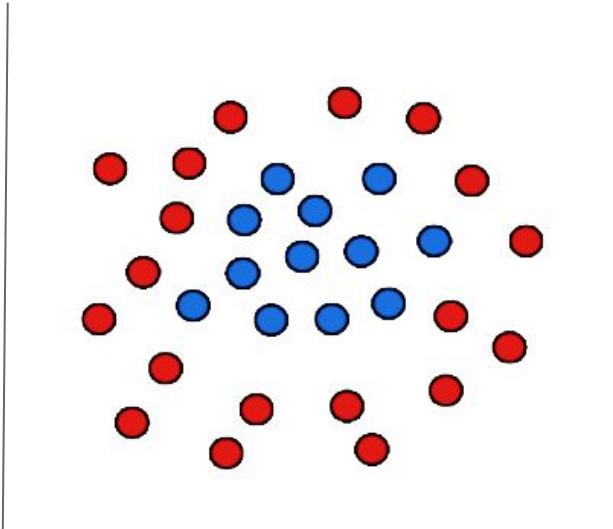
Examples closest to the hyperplane are **support vectors**.

Margin  $\rho$  of the separator is the distance between support vectors.



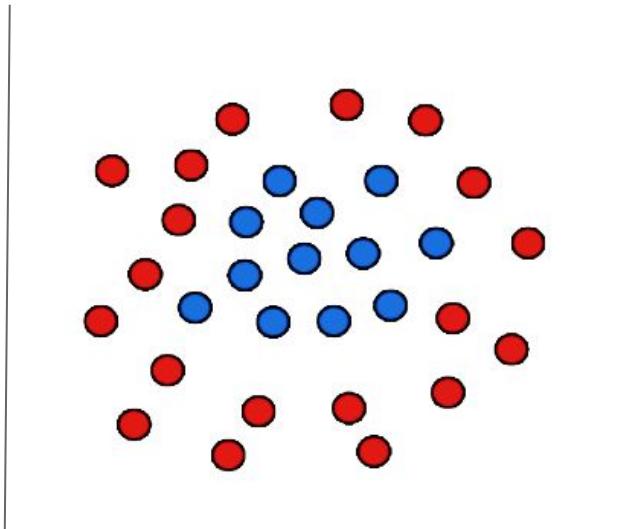
# Support vector machines: kernel

Original feature space



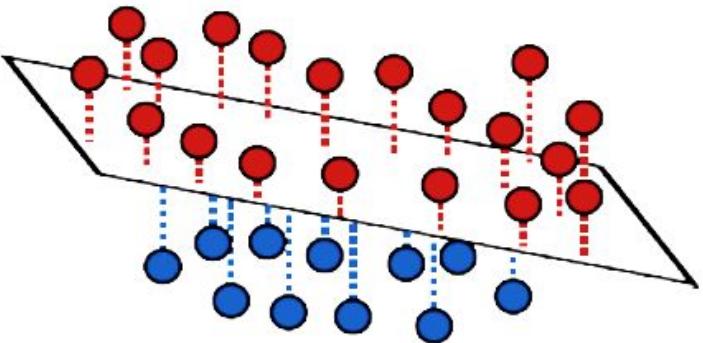
# Support vector machines: kernel

Original feature space



kernel  
function

Transformed feature space

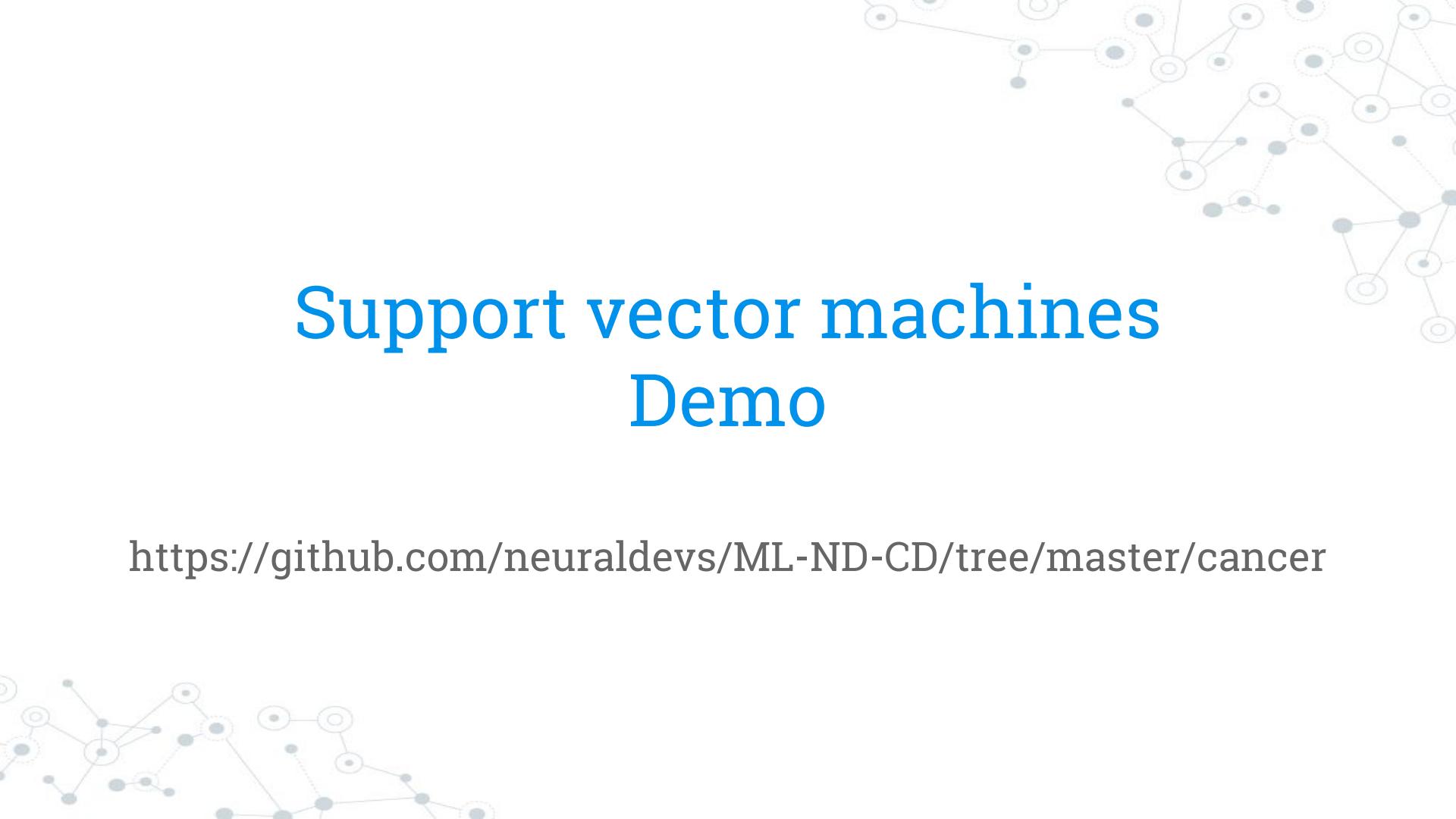


# Support vector machines

- + Effective in high dimensional spaces: high number of features
- + Memory efficient.
- + Kernel functions allow non-linear estimation.
- + Suitable for low number of samples

# Support vector machines

- + Effective in high dimensional spaces: high number of features
- + Memory efficient.
- + Kernel functions allow non-linear estimation.
- + Suitable for low number of samples
- Very slow for data with high number of samples
- Totally discriminative method.
- Tricky hyper-parameter tuning: complexity C and kernel parameters (d, gamma)



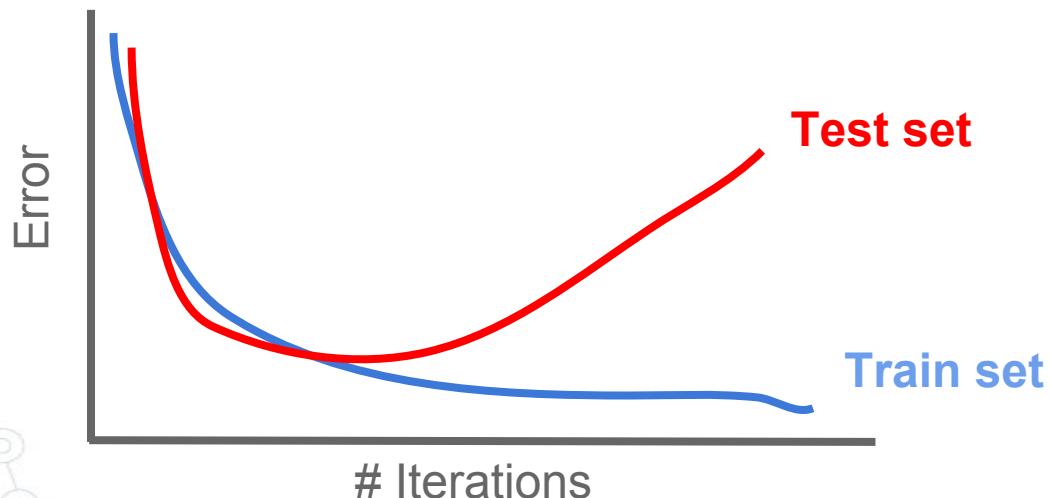
# Support vector machines Demo

<https://github.com/neuraldevs/ML-ND-CD/tree/master/cancer>

# Validation methods

Evaluating estimator performance

- Learning the parameters of a prediction function and testing it on the same data is a **methodological mistake (overfitting)**
- Hold out part of the available data as a **test set**



# Validation methods

Evaluating estimator performance

- ◎ When evaluating different settings (“**hyperparameters**”) for estimators such as **maximum depth** of a decision tree there is still a risk of **overfitting** on the **test set**.
- ◎ Knowledge about the **test set** can “**leak**” into the model.

# Validation methods

Evaluating estimator performance

- ◎ When evaluating different settings (“**hyperparameters**”) for estimators such as **maximum depth** of a decision tree there is still a risk of **overfitting** on the **test set**.
- ◎ Knowledge about the **test set** can “**leak**” into the model.
- ◎ Another part of the dataset can be held out as a so-called “**validation set**”
- ◎ Training proceeds on the **training set**, after which evaluation is done on the **validation set**, and when the experiment seems to be successful, final evaluation can be done on the **test set**.

# Validation methods

## Performance Measures

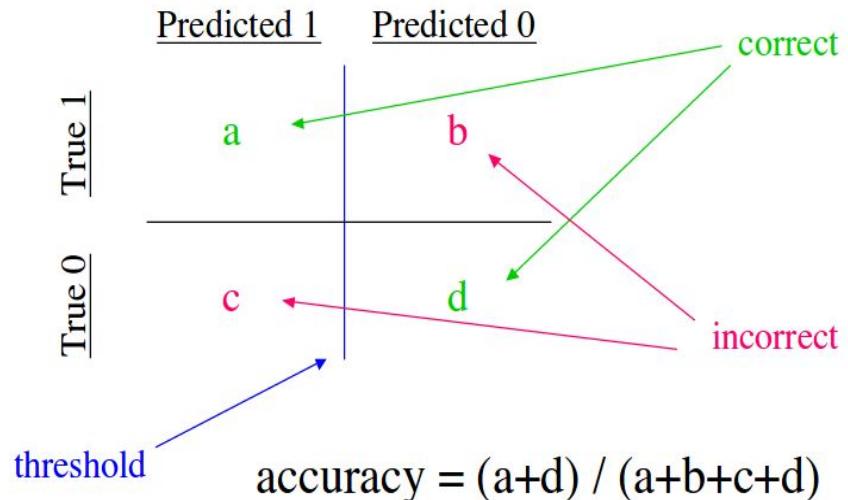
- ◎ Accuracy

$$\text{accuracy}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} 1(\hat{y}_i = y_i)$$

# Validation methods

## Performance Measures

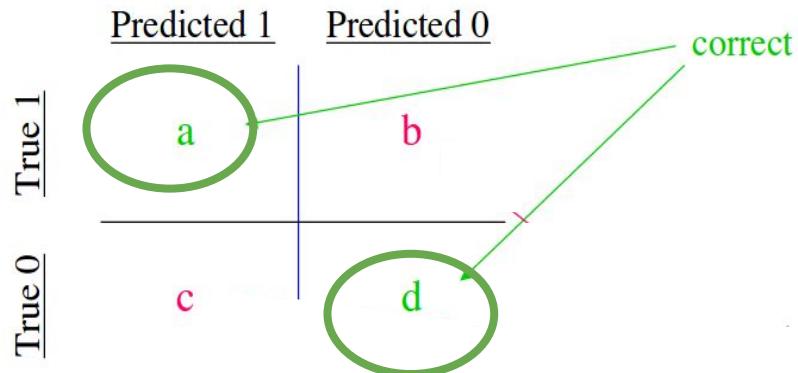
### ◎ Confusion Matrix



# Validation methods

## Performance Measures

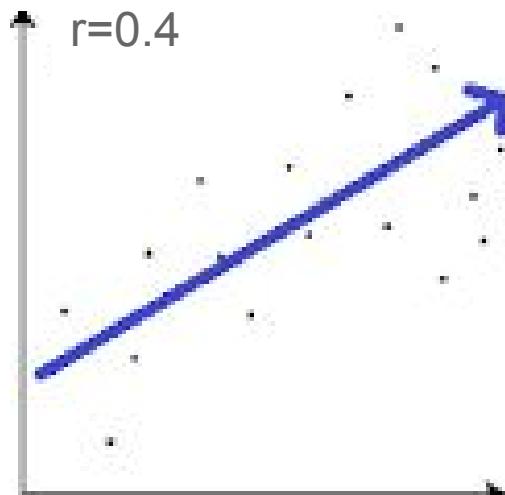
### ◎ Specificity / Sensitivity



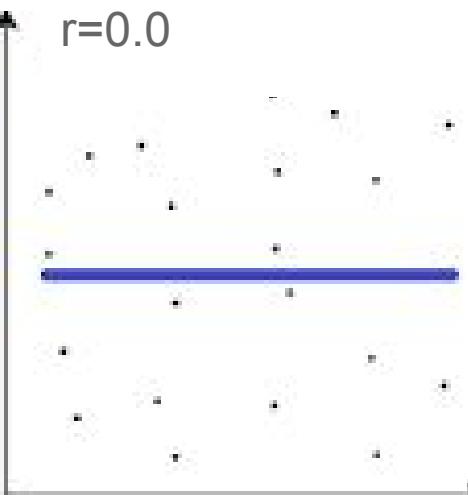
# Validation methods

## Performance Measures

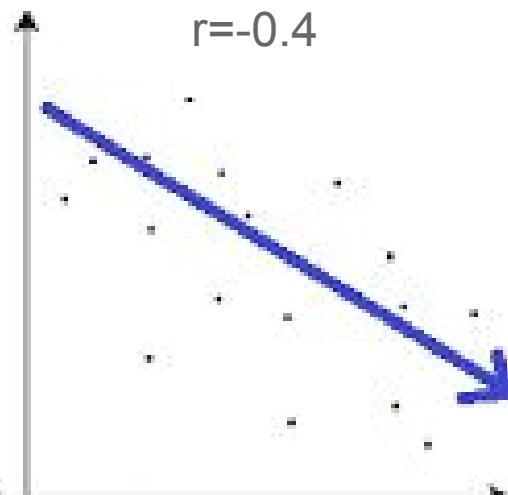
- ◎ Pearson correlation coefficient



Positive correlation



No correlation

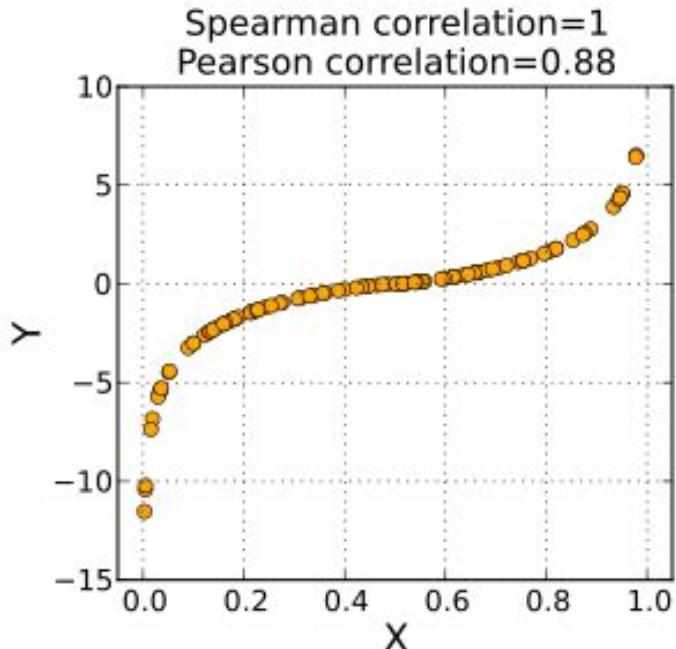


Negative correlation

# Validation methods

## Performance Measures

- ◎ Spearman correlation coefficient



# SIATA air quality in Medellín: complete use case

## Demo

<https://github.com/neuraldevs/ML-ND-CD/tree/master/Siata>

# Other technologies



Spark + H<sub>2</sub>O  
SPARKLING  
**WATER**



# TensorFlow

- Open source software library for numerical computation using data flow graphs.
  - Nodes = represent mathematical operations,
  - tensors = The central unit of data.

```
node1 = tf.constant(3.0, dtype=tf.float32)  
node2 = tf.constant(4.0) # also tf.float32 implicitly
```

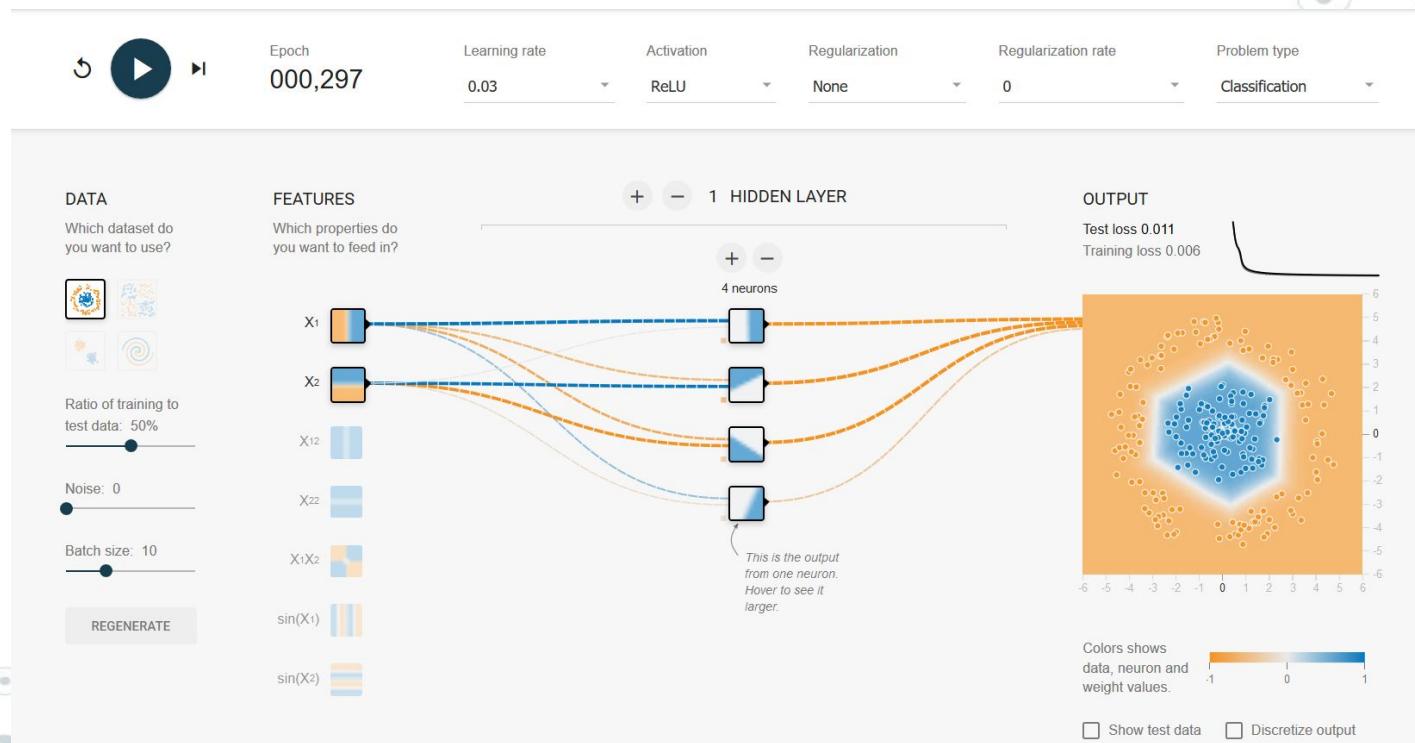
```
node3 = tf.add(node1, node2)  
sess.run(node3)
```



- CPUs
- GPUs
- desktop,  
Server
- Mobile device with a single API.

# TensorFlow

(<http://playground.tensorflow.org>)



# Spark

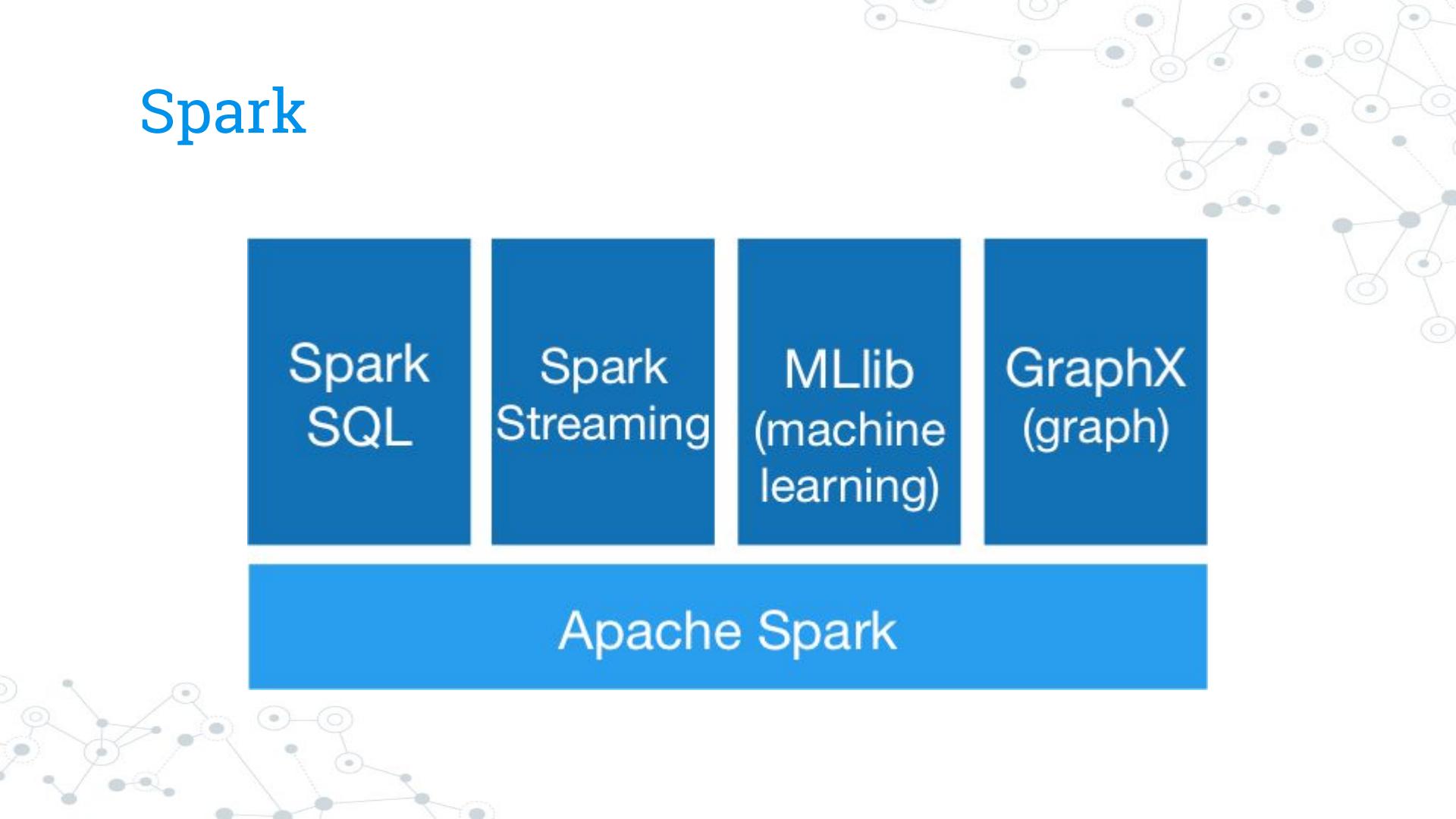
Spark  
SQL

Spark  
Streaming

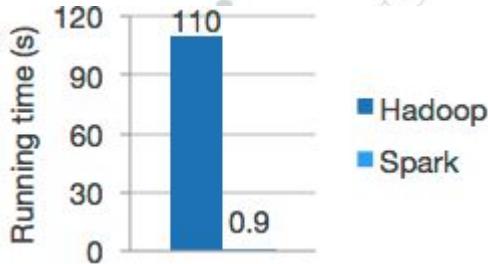
MLlib  
(machine  
learning)

GraphX  
(graph)

Apache Spark



# Spark MLlib



- Run programs up to 100x faster than Hadoop MapReduce in memory, or 10x faster on disk (acyclic data flow).
- Write applications quickly in Java, Scala, Python, R.
- MLlib contains many algorithms and utilities.
- Sparkling Water (spark + h2o)

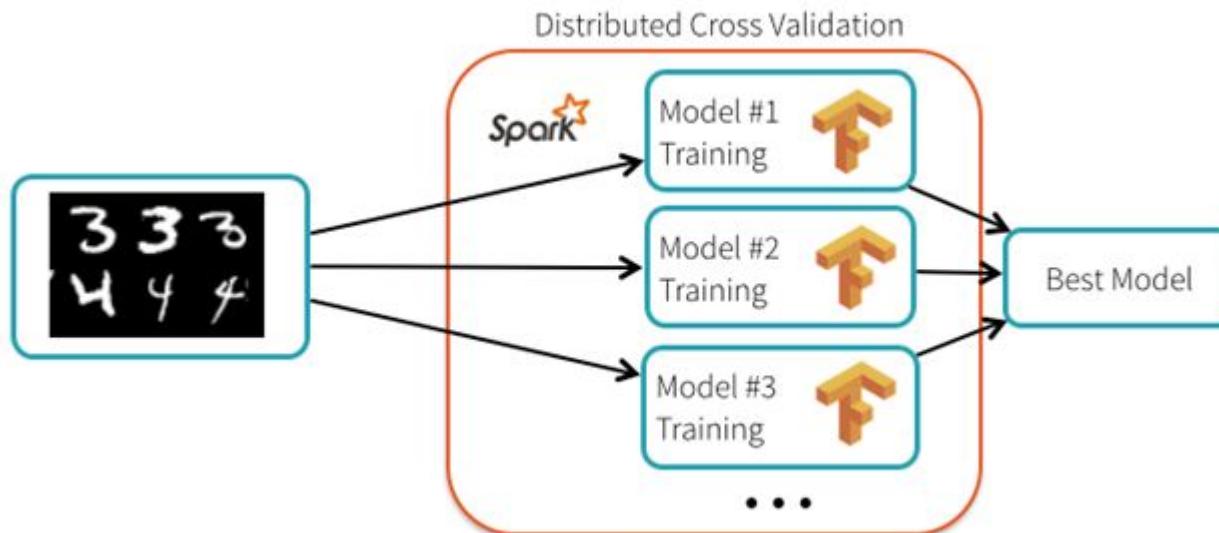
# H2O

**H<sub>2</sub>O.ai**

- In an open source, in-memory, distributed, fast, and scalable machine learning and predictive analytics platform that allows you to build machine learning models on big data and provides easy productionalization of those models in an enterprise environment.

# Spark + TensorFlow

## Hyperparameter Tuning



# R Language

- Open source language.
- provides a wide variety of techniques (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, ...) and graphical techniques
- Highly extensible.
- More than 10000 packages.
- Easily integration with Spark and TensorFlow.



# References

- <http://blog.revolutionanalytics.com/2017/01/cran-10000.html>
- <https://spark.apache.org/>
- <http://docs.h2o.ai/h2o/latest-stable/h2o-docs/welcome.html>
- <https://databricks.com/blog/2016/01/25/deep-learning-with-apache-spark-and-tensorflow.html>
- <https://www.tensorflow.org/>

# Exercises and practice

<https://github.com/colomb-ia/retos>