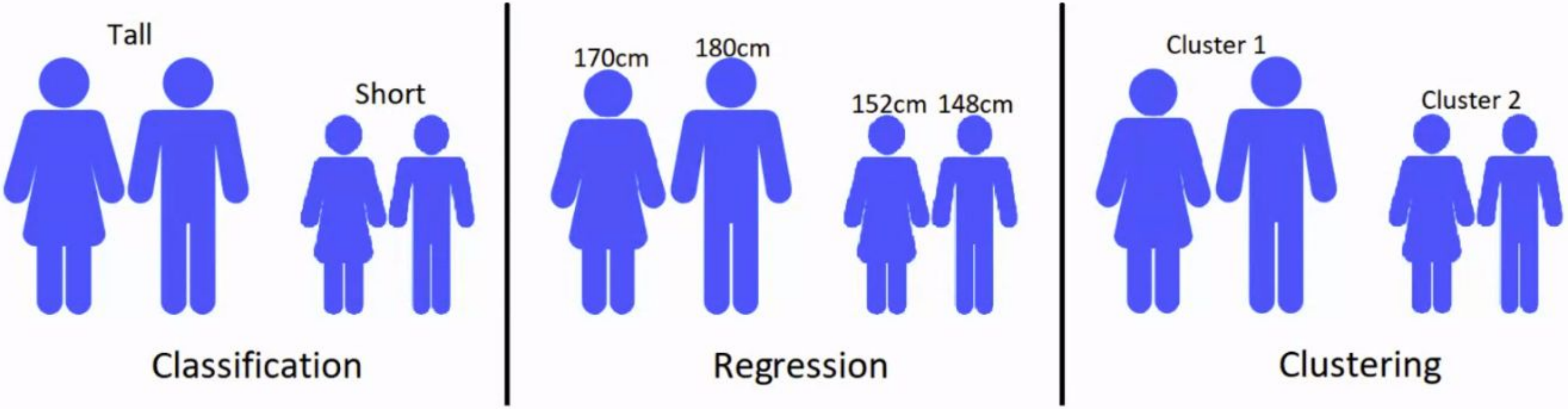
Machine Learning - Unsupervised

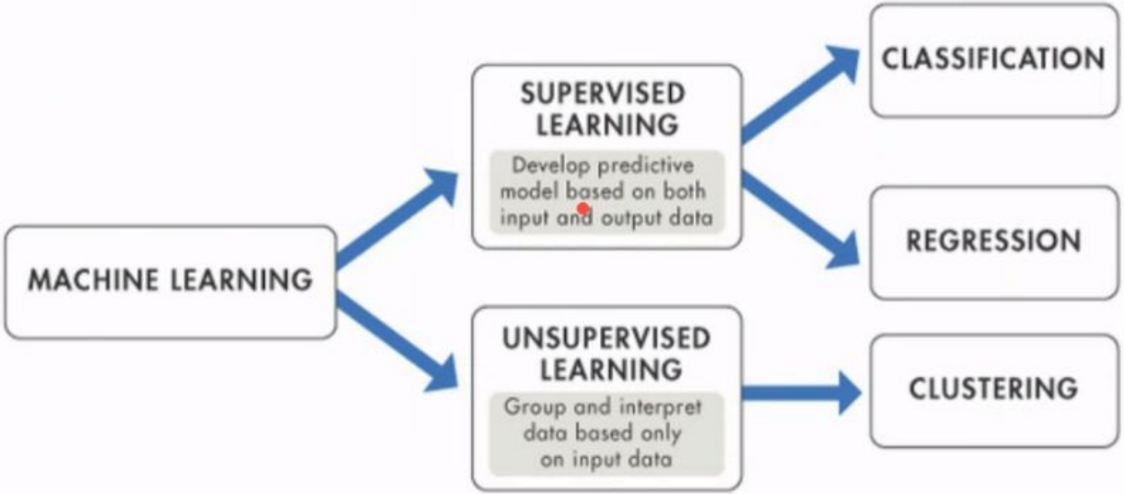
Soluciones de ML

mojix



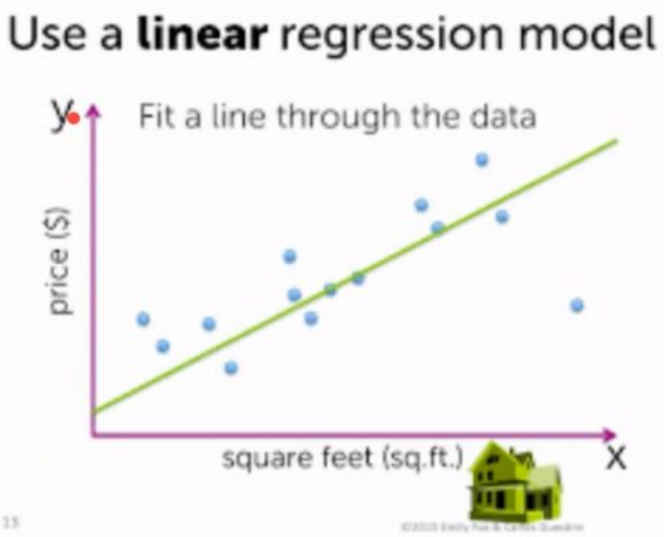
Data etiquetada o no

mojix



Tipos

Regresión



Clasificación

Tipos

mojix

Supervised Learning

CAT

UnSupervised Learning

Dataset

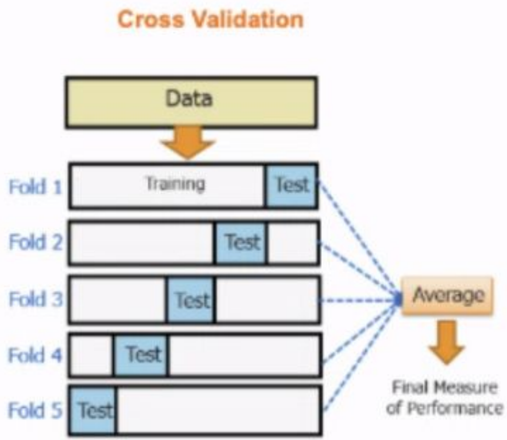
	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5.0	3.4	1.5	0.2	setosa

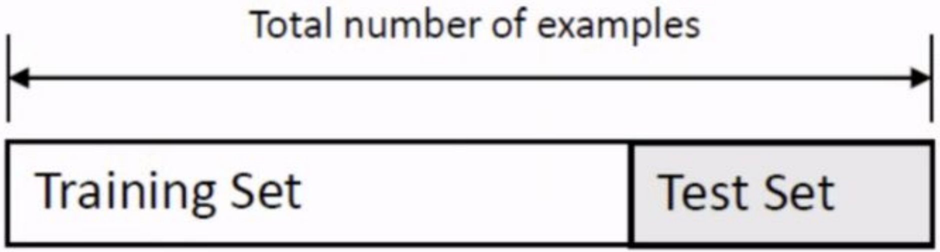
Training

Validation

Test

m x n





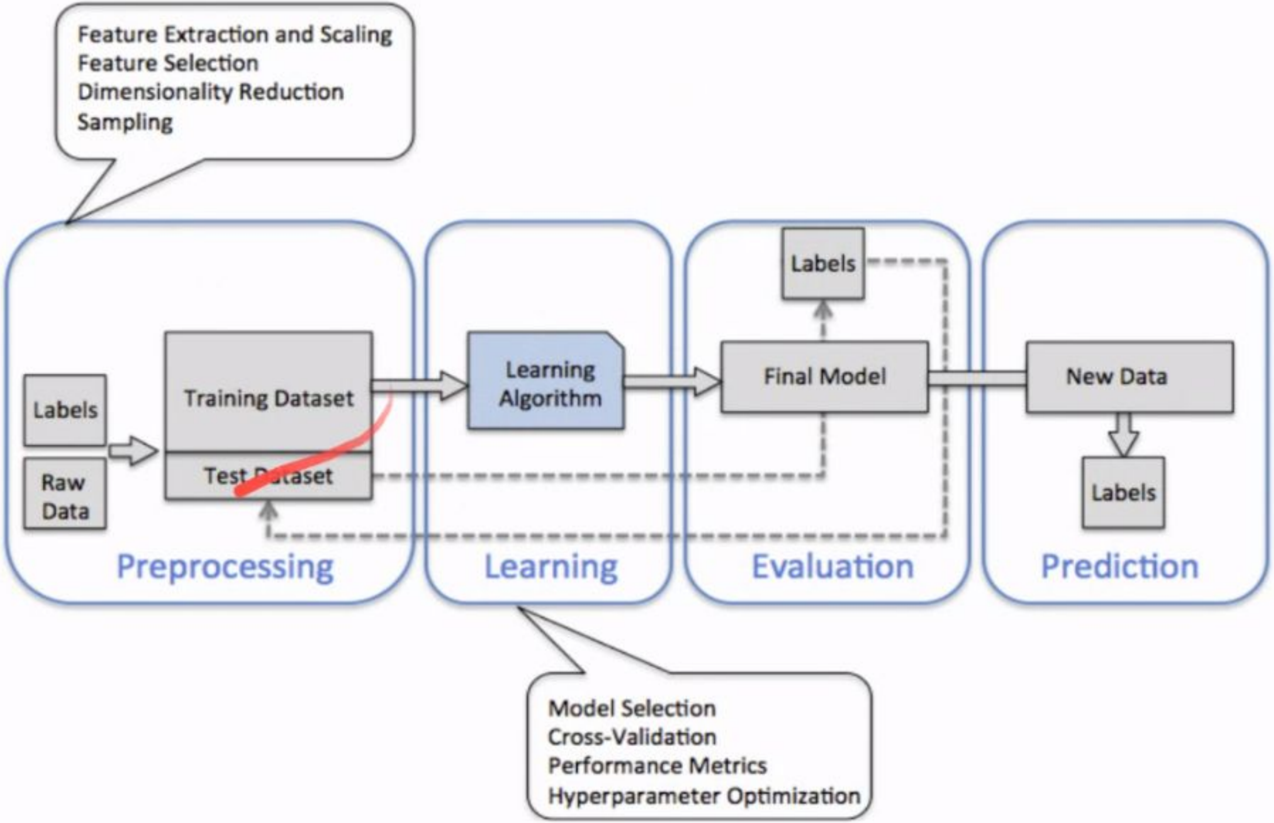
Usual data distributions

Regression: 80%-20%, 70%-30%

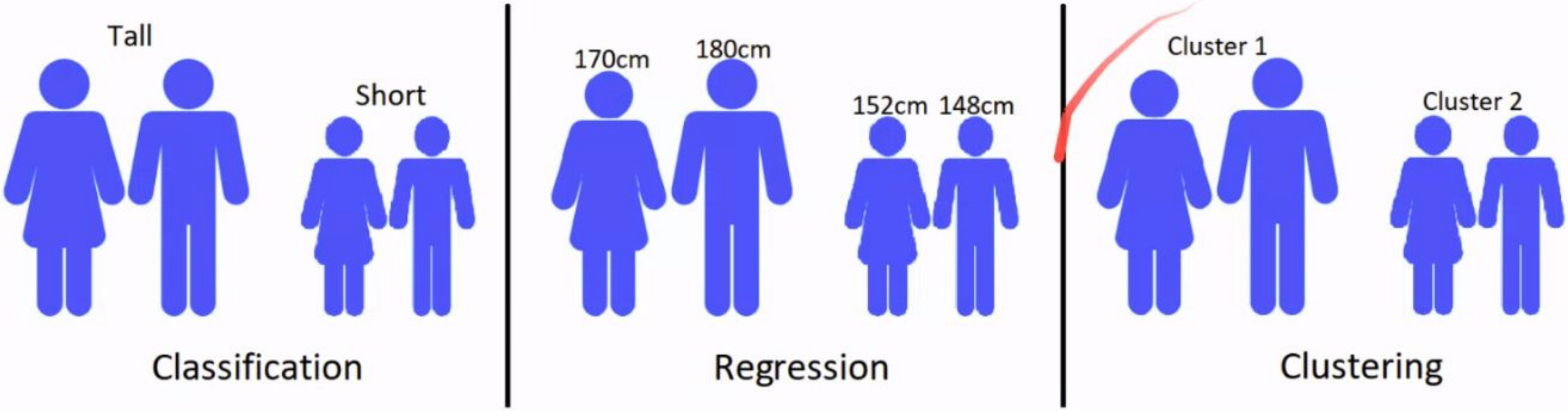
Classification: 90%-10%

El proceso de Machine Learning

mojix

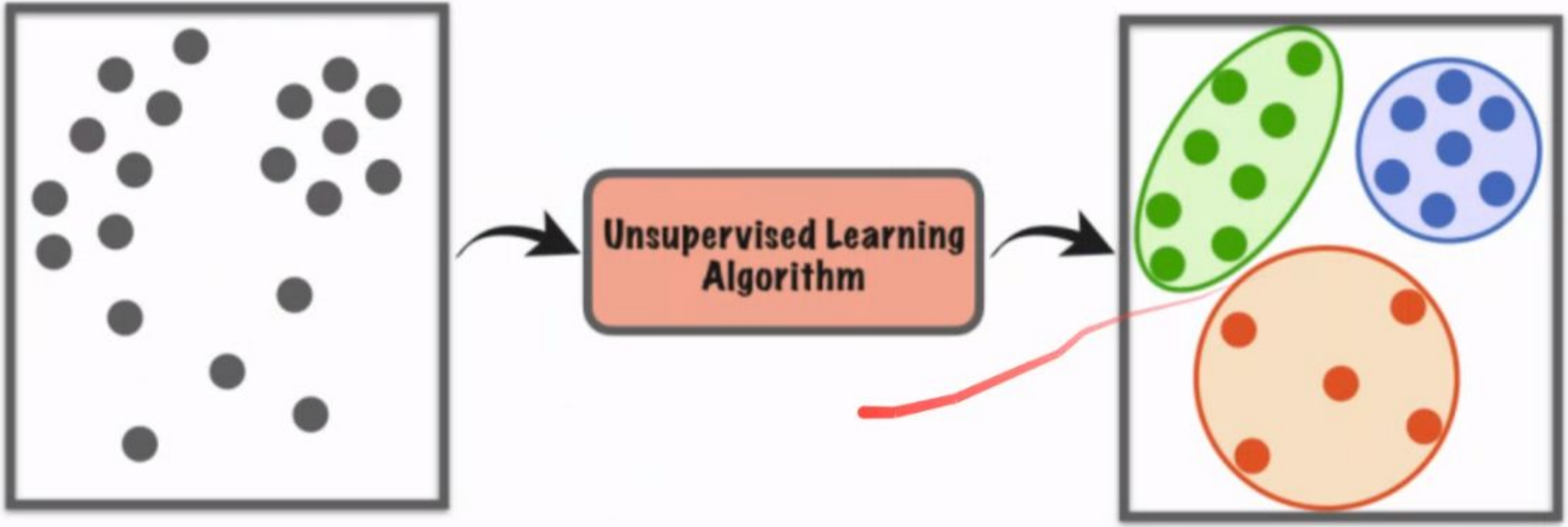


Soluciones de ML



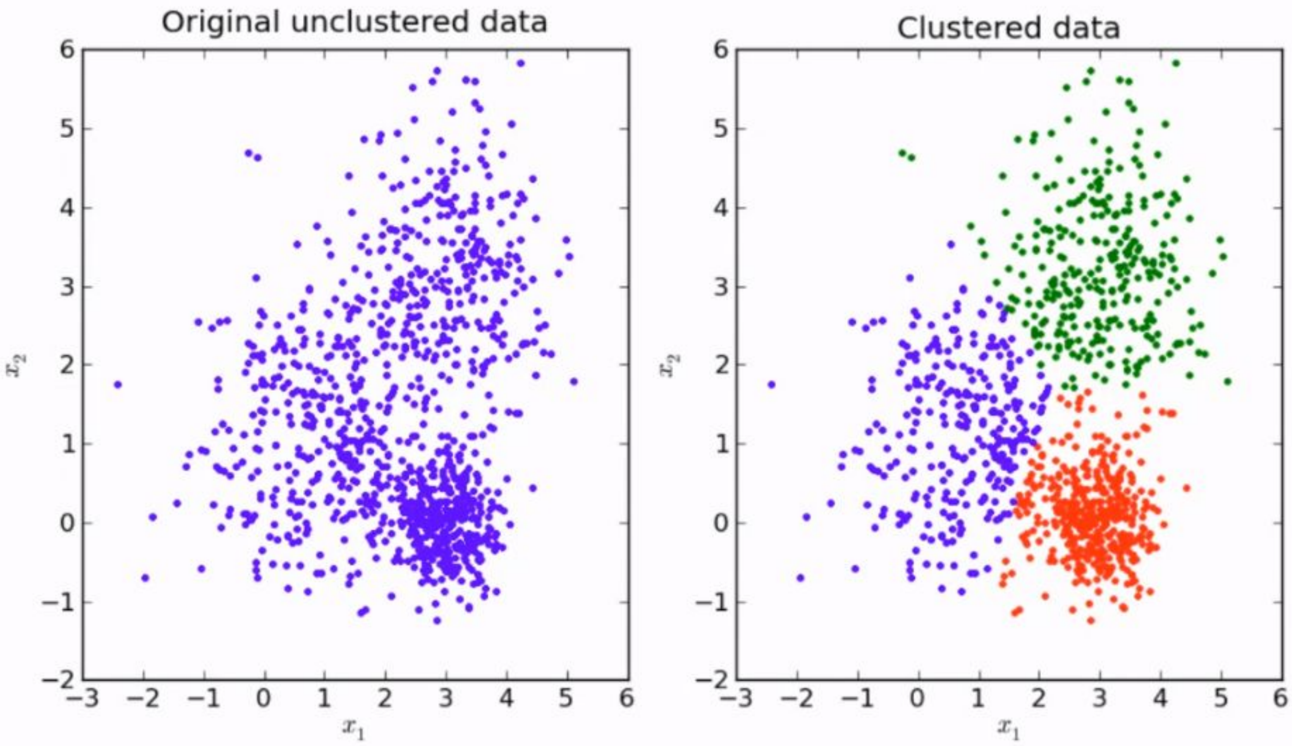
Aprendizaje No Supervisado

mojix



Clustering

mojix



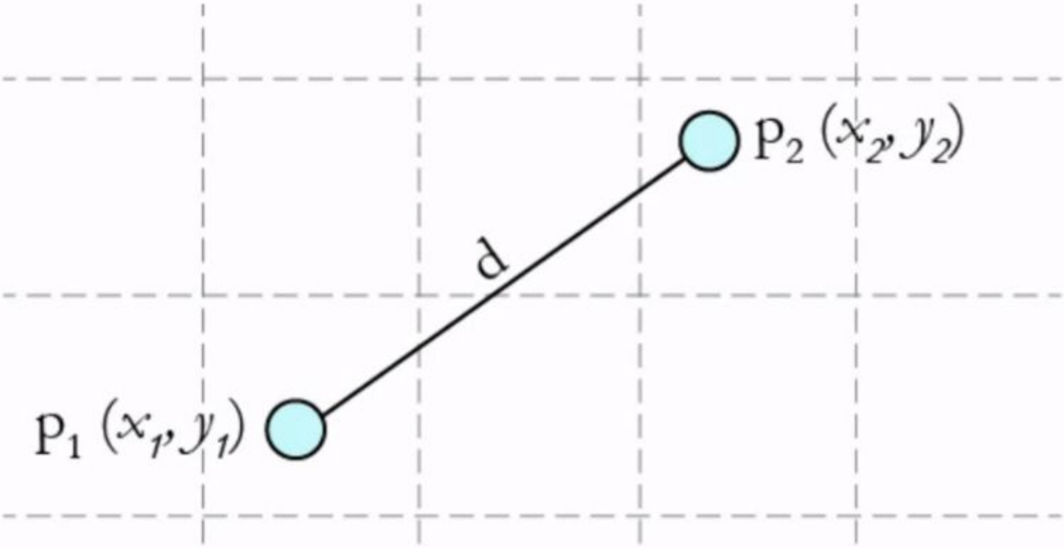
https://scikit-learn.org/stable/modules/clustering.html

K-Means Algorithm

Iterative, hard, flat clustering algorithm based on Euclidean distance

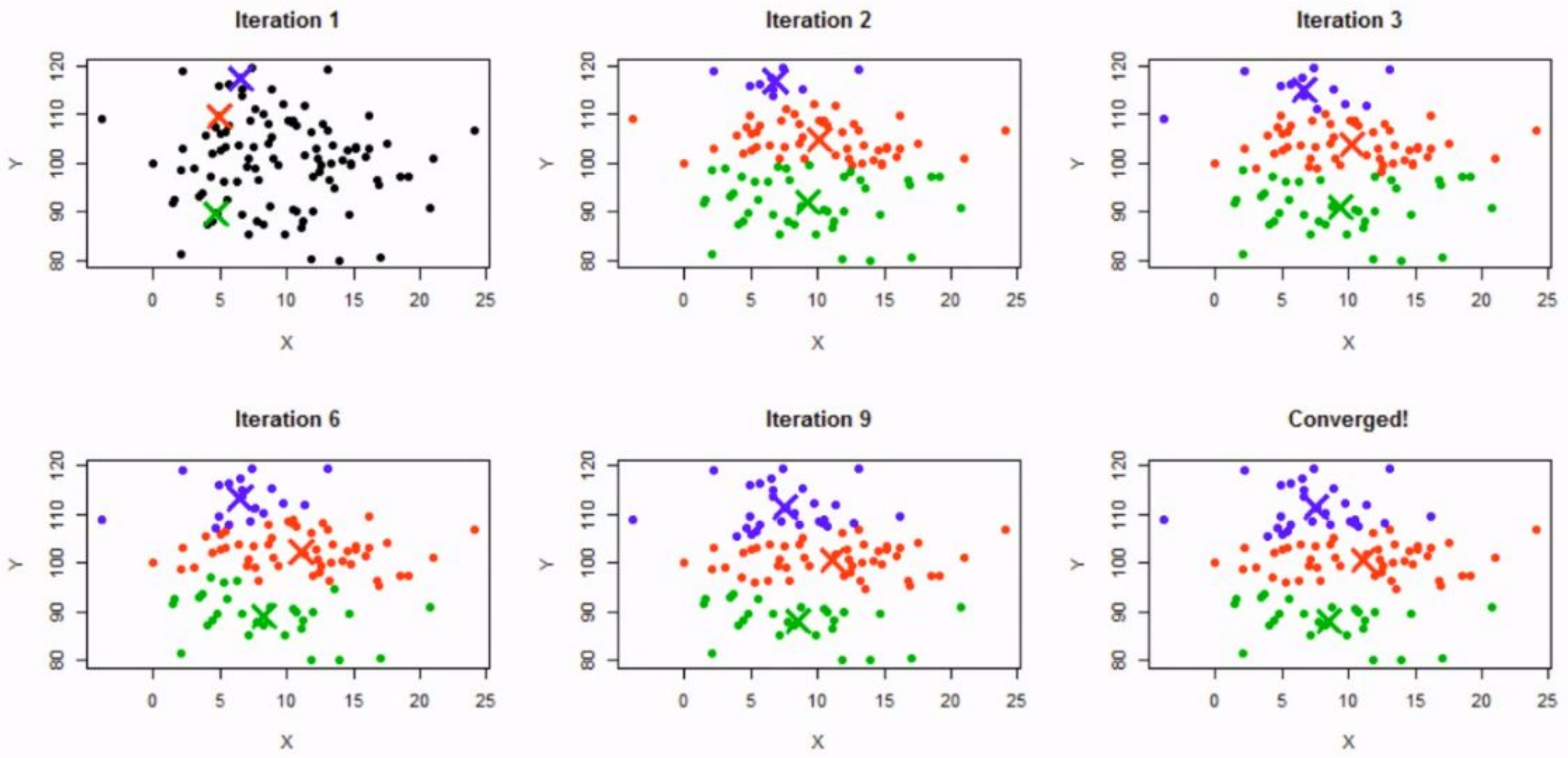
- Specify k, the number of clusters to be generated
- Choose k points at random as cluster centers
- Assign each instance to its closest cluster center using Euclidean distance
- Calculate the centroid (mean) for each cluster, use it as a new cluster center
- Reassign all instances to the closest cluster center
- Iterate until the cluster centers don't change anymore

Distancia



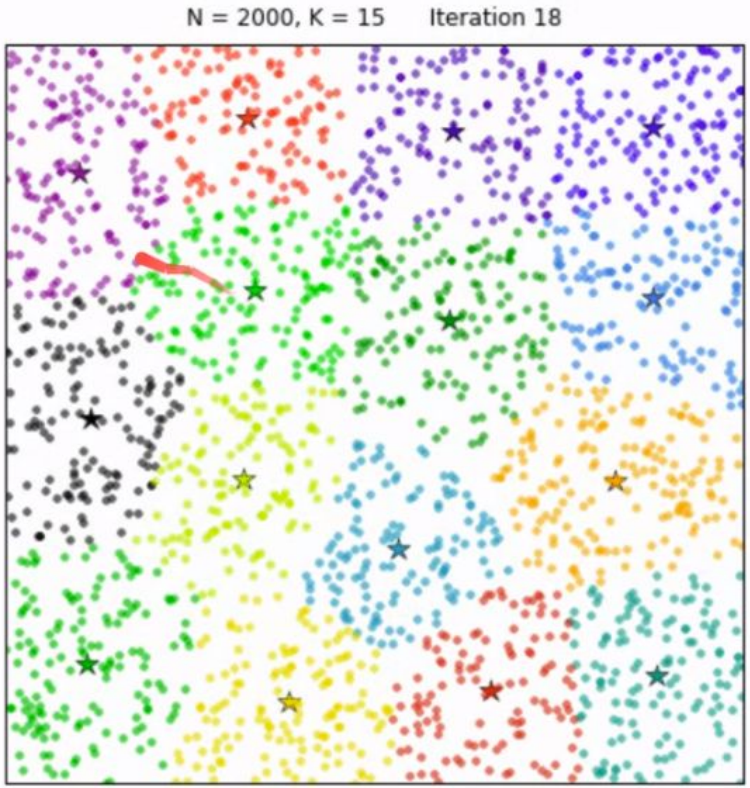
$$\text{Euclidean distance (d)} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

K-means

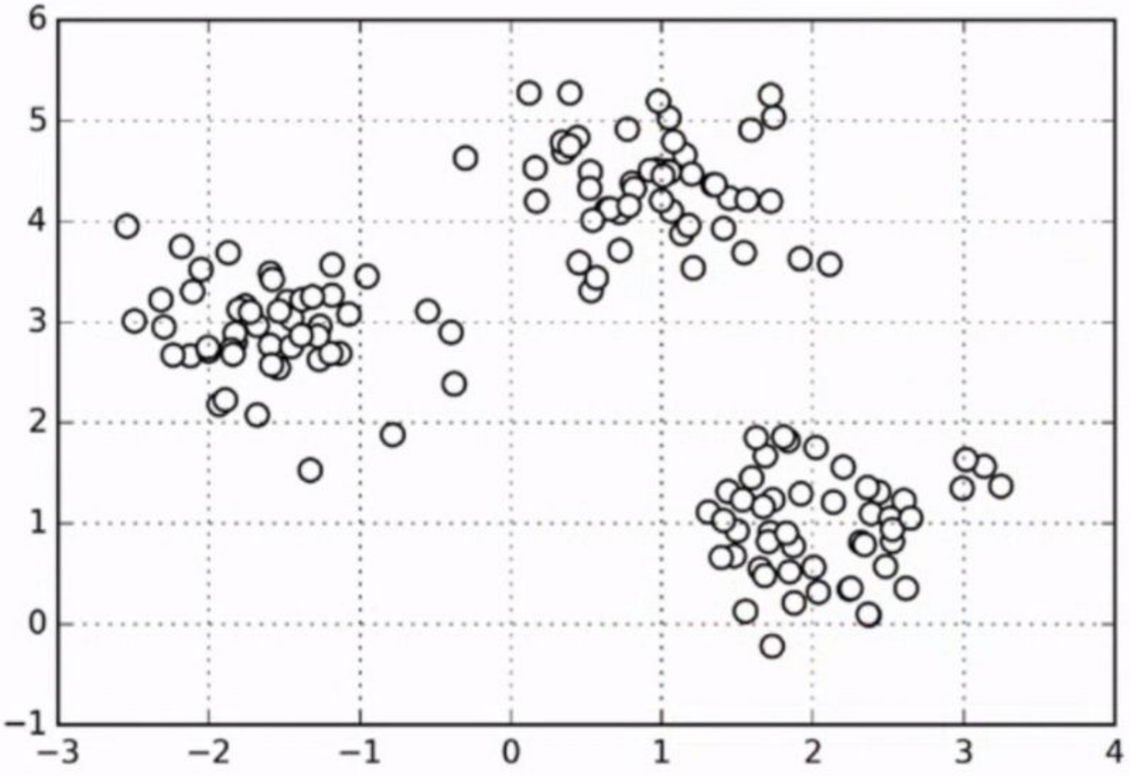

Iteration 1
Iteration 2
Iteration 3
Iteration 6
Iteration 9
Converged!
X
Y
0 5 10 15 20 25
80 90 100 110 120

K-means

mojix



K means



- Definir K
- No tenemos etiquetas (Ground Truth) para comparar

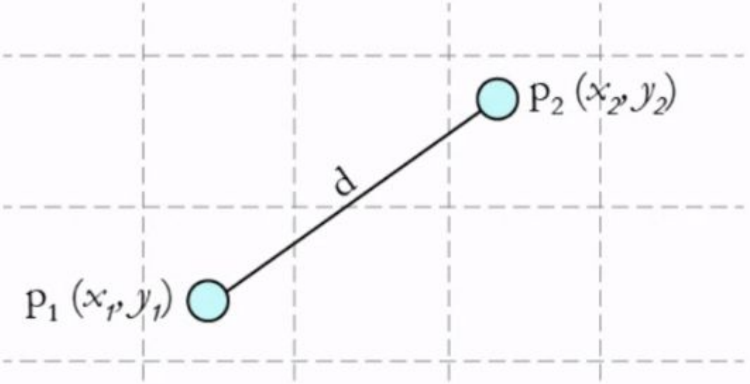
K-means

mojix

K-Means Algorithm

Iterative, hard, flat clustering algorithm based on Euclidean distance

- Specify k, the number of clusters to be generated
- Choose k points at random as cluster centers
- Assign each instance to its closest cluster center using Euclidean distance
- Calculate the centroid (mean) for each cluster, use it as a new cluster center
- Reassign all instances to the closest cluster center
- Iterate until the cluster centers don't change anymore



$$\text{Euclidean distance (d)} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

COLAB

https://colab.research.google.com/drive/199MZvaBiKl3zrn0-gyNcAS8GCU96IP26?usp=sharing

- El problema radica el nro de clusters

  IMPORTAR DATASET DE SCIKTLEARN

  BUSCAR

  centros=centroides = 3

  classes

- Blobs =>
- classes => grupos

kmeans = KMeans(n_clusters=3)

kmean.fit(blobs)

K-means as an Optimization Problem

mojix

Based on this Euclidean distance metric, we can describe the k-means algorithm as a simple optimization problem, an iterative approach for minimizing the **within-cluster sum of squared errors (SSE)**, which is sometimes also called **cluster inertia**:

$$SSE = \sum_{i=1}^{n} \sum_{j=1}^{k} w^{(i,j)} \left\| \boldsymbol{x}^{(i)} - \boldsymbol{\mu}^{(j)} \right\|_2^2$$

Here, $\boldsymbol{\mu}^{(j)}$ is the representative point (centroid) for cluster j, and $w^{(i,j)} = 1$ if the sample $\boldsymbol{x}^{(i)}$ is in cluster j; $w^{(i,j)} = 0$ otherwise.

Hard vs Fuzzy clustering

mojix

HARD: Un elemento pertenece a un solo cluster

$$\begin{bmatrix} \mu^{(1)} \to 0 \\ \mu^{(2)} \to 1 \\ \mu^{(3)} \to 0 \end{bmatrix}$$

FUZZY: Probability membership

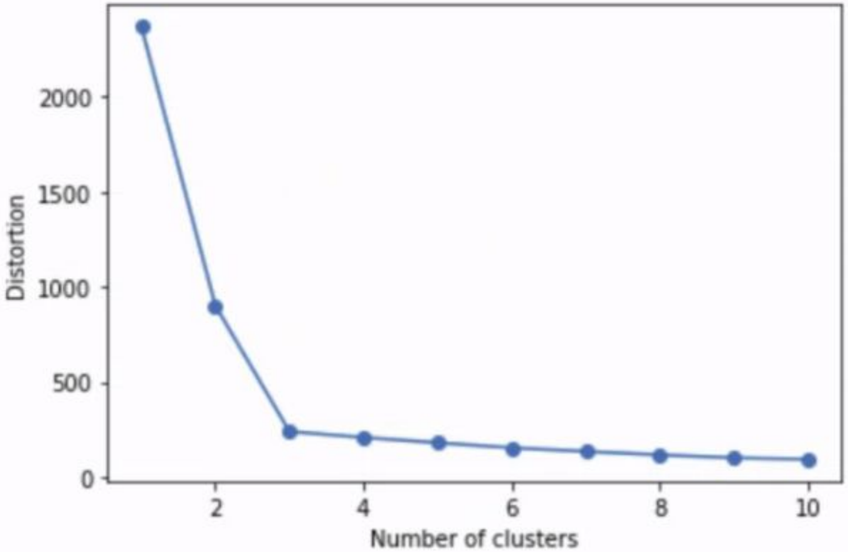
$$\begin{bmatrix} \mu^{(1)} \to 0.1 \\ \mu^{(2)} \to 0.85 \\ \mu^{(3)} \to 0.05 \end{bmatrix}$$

Fuzzy C-means

μ Centroides

K óptimo → Elbow Method

mojix



- Within-cluster SSE (distortion)
- If K increases distortion will decrease, because samples will be closer.

Goal:

Identify the value of k where the distortion begins to increase

Silhouette Coefficient

mojix

Silhouette Coefficient or silhouette score is a metric used to calculate the goodness of a clustering technique. Its value ranges from -1 to 1.

1: Means clusters are well apart from each other and clearly distinguished.

0: Means clusters are indifferent, or we can say that the distance between clusters is not significant.

-1: Means clusters are assigned in the wrong way.

https://colab.research.google.com/drive/1wPTr22hDWloyU1Nn0bI2oxfmWwAjrARo?usp=sharing

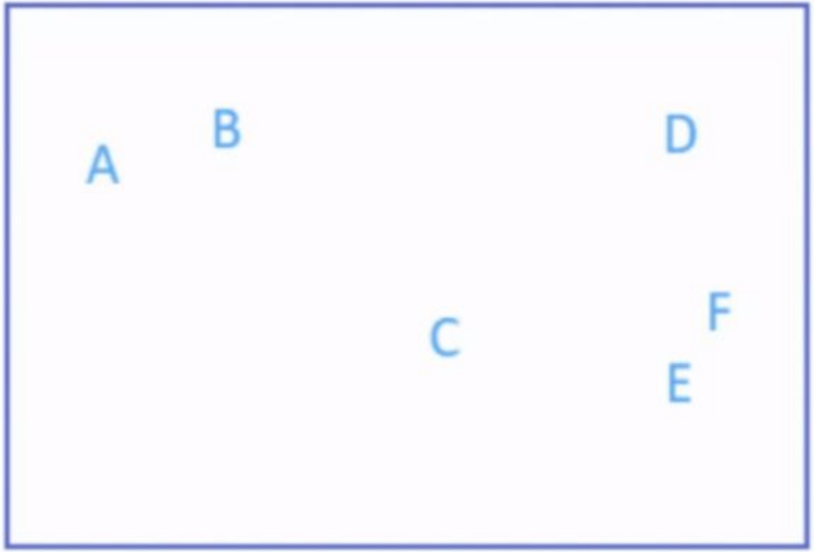
kmeans

https://realpython.com/k-means-clustering-python/

What is Hierarchical Clustering?

https://www.displayr.com/what-is-hierarchical-clustering/

Hierarchical Clustering

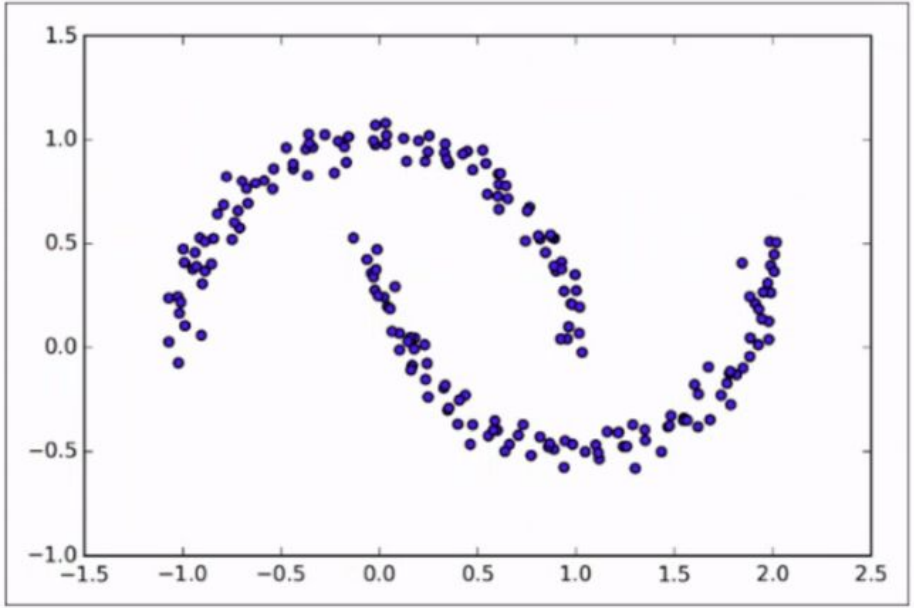


Dendrogram

DBSCAN

mojix

Density-based Spatial Clustering of Applications with Noise (DBSCAN). The notion of density in DBSCAN is defined as the number of points within a specified radius ε .



Unsupervised Learning

mojix

Unsupervised Learning

- Unsupervised learning: a set of statistical tools for data for which only features/inputs are available

– We have X's but no associated labels Y

– Goal: discover interesting patterns/properties of the data

- e.g. for visualizing or interpreting high-dimensional data

Unsupervised Learning

Why is unsupervised learning challenging?

- Exploratory data analysis - goal is not as clearly defined
- Difficult to assess performance - “right answer” unknown
- Working with high-dimensional data

Unsupervised Learning

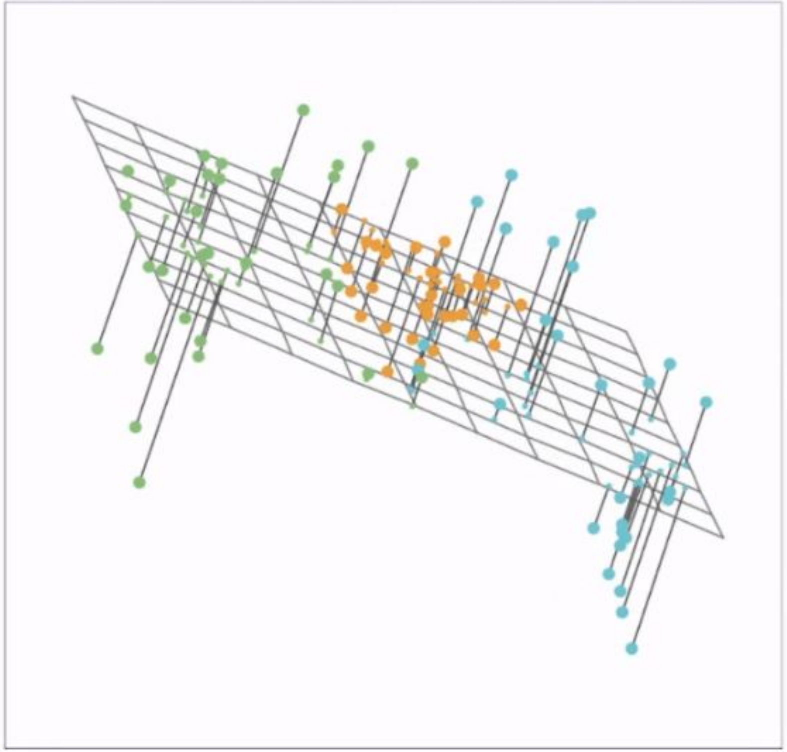
mojix

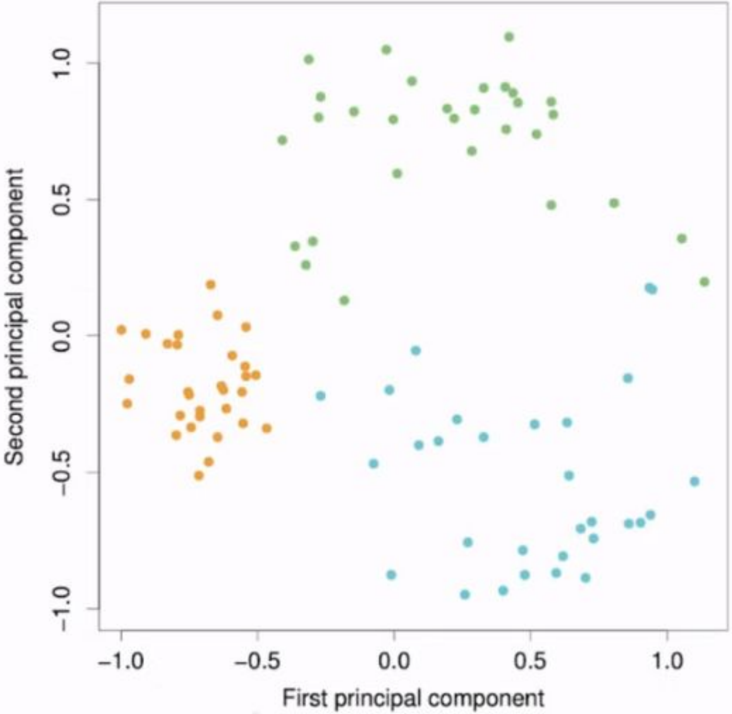
Two approaches:

- Cluster Analysis
  - For identifying homogeneous subgroups of samples
- Dimensionality Reduction
  - For finding a low-dimensional representation to characterize and visualize the data

Dimensionality Reduction

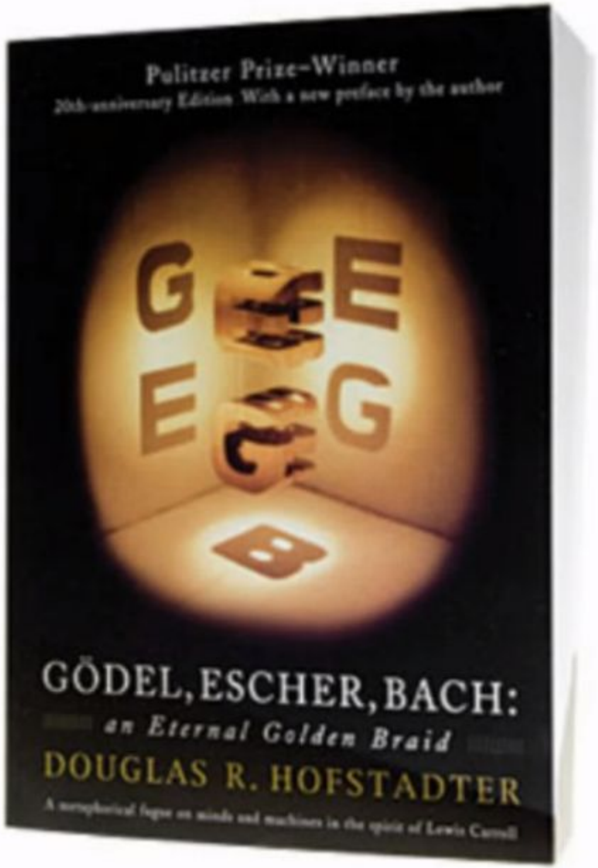
mojix





Projections

mojix

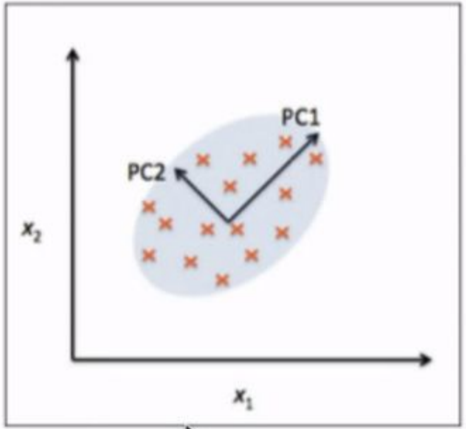


Principal Component Análisis

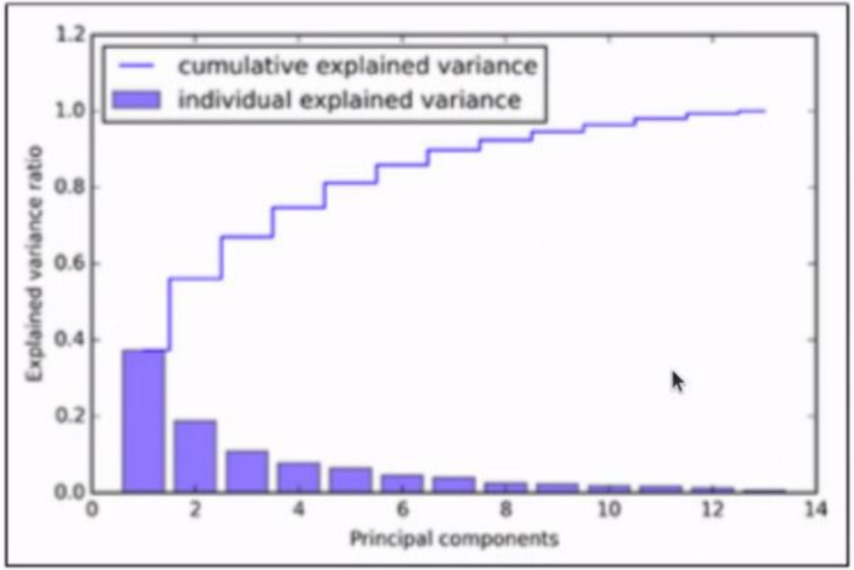
mojix

Before looking at the PCA algorithm for dimensionality reduction in more detail, let's summarize the approach in a few simple steps:

1. Standardize the d-dimensional dataset.
2. Construct the covariance matrix.
3. Decompose the covariance matrix into its eigenvectors and eigenvalues.
4. Select k eigenvectors that correspond to the k largest eigenvalues, where k is the dimensionality of the new feature subspace ($k \leq d$).
5. Construct a projection matrix $\boldsymbol{W}$ from the "top" k eigenvectors.
6. Transform the d-dimensional input dataset $\boldsymbol{X}$ using the projection matrix $\boldsymbol{W}$ to obtain the new k-dimensional feature subspace.



Cumulative explained variance



PCA

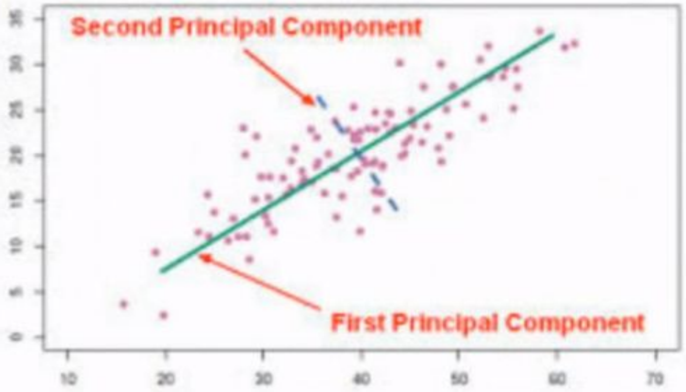
https://colab.research.google.com/drive/12VL_fYaT04EGNKfHsQP-CrSvIOKwCPV4?usp=sharing

https://www.kaggle.com/uciml/pima-indians-diabetes-database

[INTERNAL] ML_Bootcamp_2021

PCA

mojix



$$Av = \lambda v$$

Matrix
Eigenvector
Eigenvalue

Descomposición de matrices

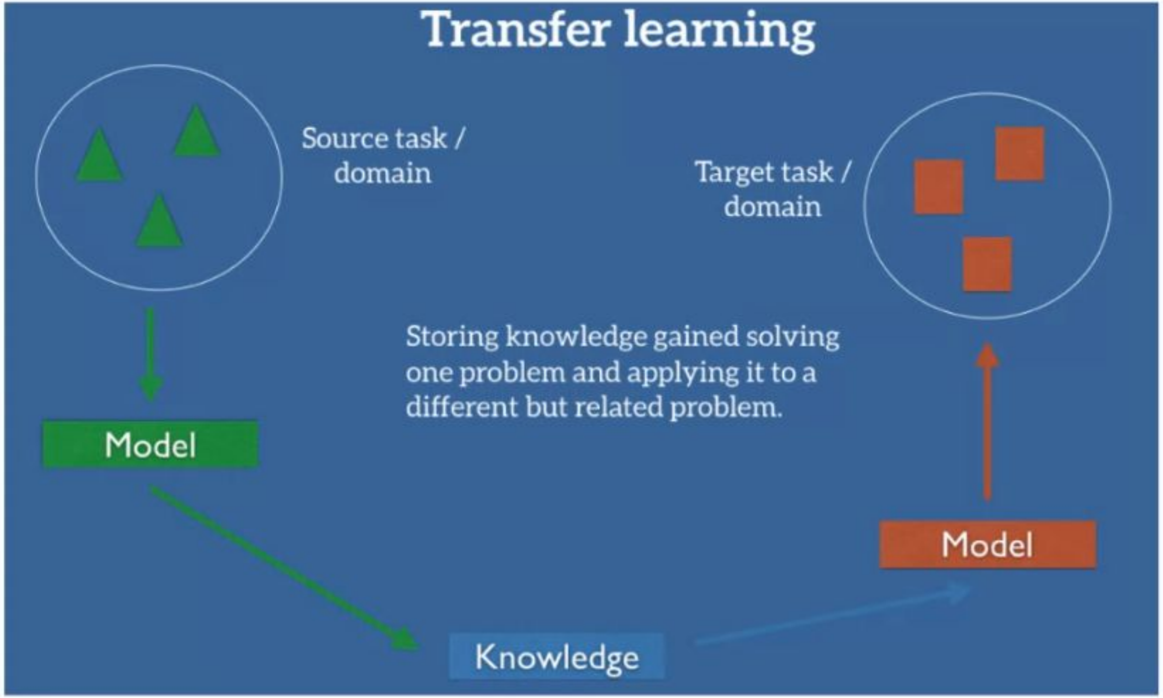
- Eigen Vectors
- Eigen Values

Transfer Learning

mojix

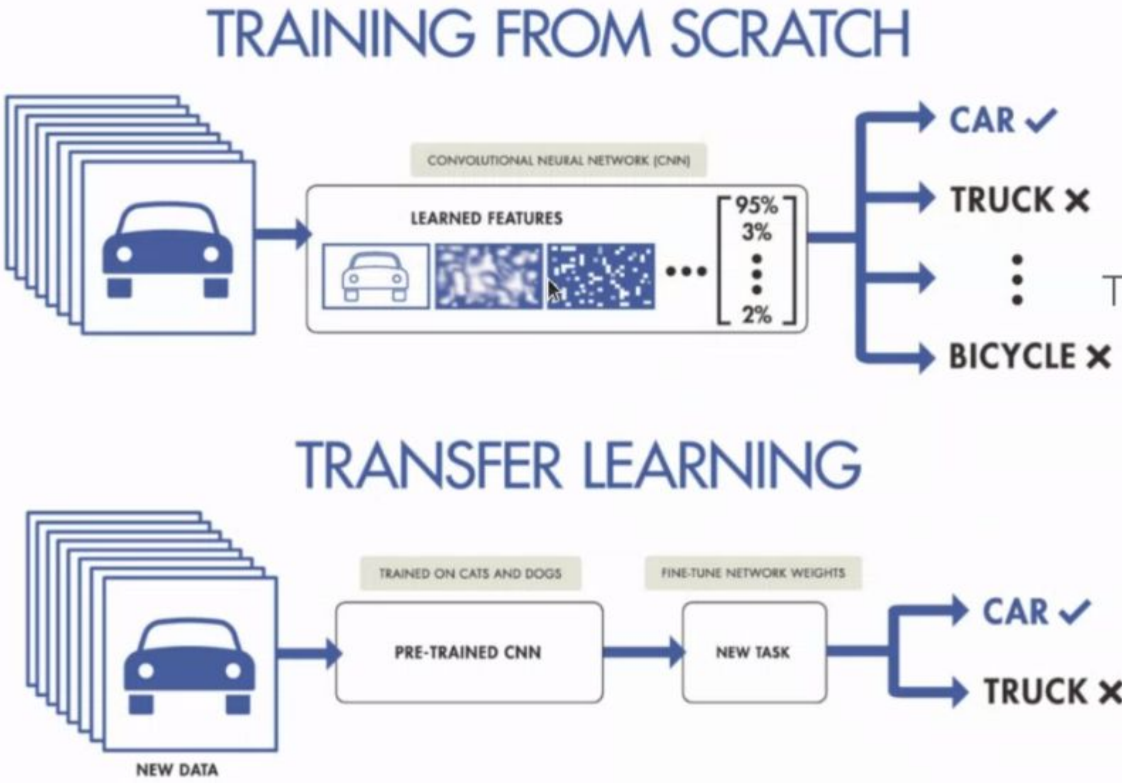
Transfer Learning is the process of applying an existing trained ML model to a new, but related, problem.

-



Transfer Learning

mojix

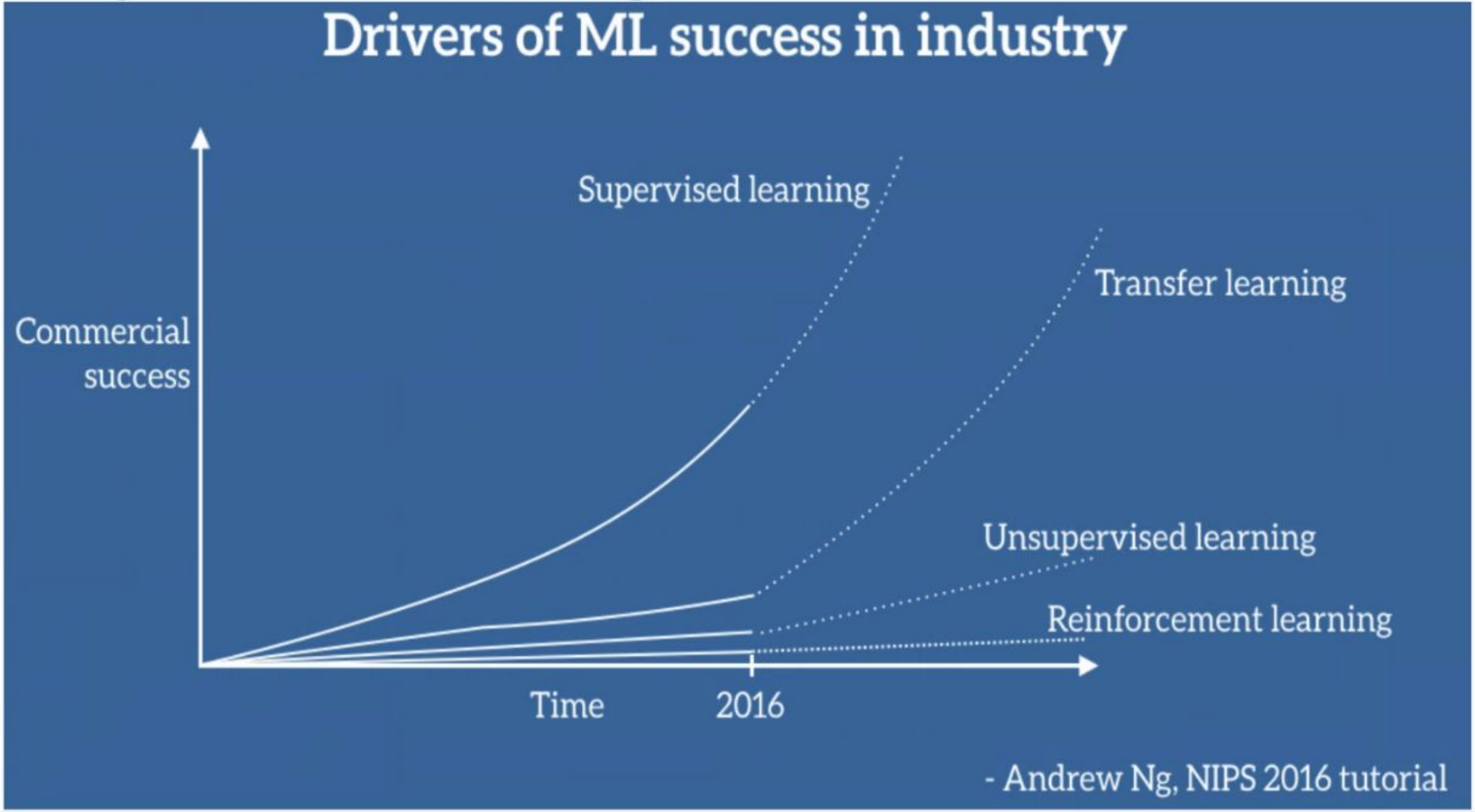


Training Options:

- Use the original part of the network as feature extractor.
- Fine tuning the whole network.

Why Transfer Learning?

mojix



El Reto


BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

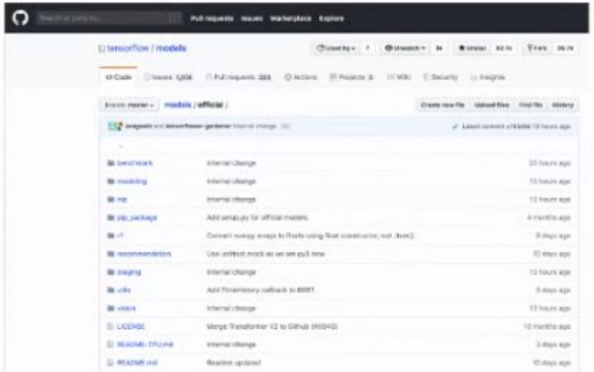
Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova

Google AI Language

Abstract

1 Introduction

arXiv:1810.04805v2 [cs.CL] 24 May 2019



¿Cómo lo uso?

¿Es seguro?

¿Cuento con todos los datos?

¿Es la última versión?

TensorFlow Hub es una biblioteca de módulos de aprendizaje automático reutilizables.

TensorFlow Hub es una biblioteca para la publicación, el descubrimiento y el consumo de módulos reutilizables de modelos de aprendizaje automático. Un *módulo* es una pieza autónoma de un grafo de TensorFlow, junto con sus pesos y recursos, que se puede reutilizar para diferentes tareas en procesos de aprendizaje por transferencia. Con el aprendizaje por transferencia, se puede realizar lo siguiente:

- Entrenar un modelo con un conjunto de datos más pequeño
- Mejorar la generalización
- Acelerar el entrenamiento

Explorar módulos en tfhub.dev

```
!pip install "tensorflow_hub>=0.6.0"
!pip install "tensorflow>=2.0.0"

import tensorflow as tf
import tensorflow_hub as hub

module_url = "https://tfhub.dev/google/nnlm-en-dim128/2"
embed = hub.KerasLayer(module_url)
embeddings = embed(["A long sentence.", "single-word",
                    "http://example.com"])
print(embeddings.shape)  #(3,128)
```

TensorFlow Hub

tfhub.dev

Explorar módulos

Introducción a TensorFlow Hub

Leer en el blog de TensorFlow

TensorFlow Hub en la Dev Summit

Ver el video

TensorFlow Hub en GitHub

Ver en GitHub

TensorFlow Hub

#TensorFlowCommunityTraining

Una colección completa de modelos

Image Text Video Audio

Modelos Listos para usar

#TensorFlowCommunityTraining

Modelos pre entrenados listos para transmitir el aprendizaje a tus propios datasets y además son deployables en varios tipos de dispositivos

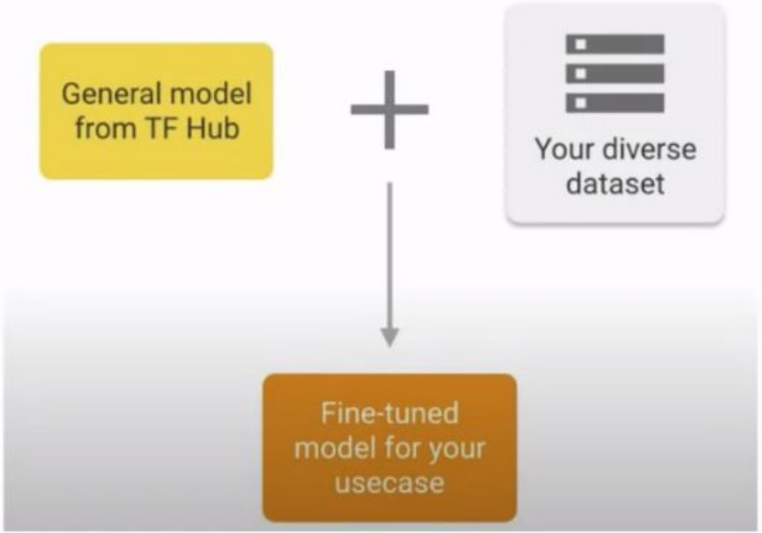
TensorFlow Extended

TensorFlow JS

TensorFlow Lite

Coral

Transfer Learning



Productive Machine Learning

https://www.youtube.com/watch?v=SJ1LGUyw-Xg

From researchers to you

Universal Sentence Encodings

Daniel Cer[a], Yinfei Yang[a], Angus Kong[a], Nan Hua[a], Nicole Limtiaco[b], Rhomni St. John[a], Noah Constant[a], Mario Guajardo-Céspedes[a], Steve Yuan[c], Chris Tar[a], Yun-Hsuan Sung[a], Brian Strope[a], Ray Kurzweil[a]

[a]Google Research, Mountain View, CA
[b]Google Research, New York, NY
[c]Google, Cambridge, MA

Abstract

The pre-trained sentence encoding models reported in the paper are made freely available for download.

nington et al., 2014). Recent work has demon-
transfer results using pre-trained
embeddings (Conneau et al., 2017).
r, we present a collection of models
sentence embeddings that demon-
nsfer learning to a number of other
tasks. The sentence encoding mod-
are made publicly available. We in-
ents with varying amounts of trans-
ing data to illustrate the interac-
the impact of transfer learning and
ize. Engineering characteristics of
for transfer learning are important.
odeling trade-offs regarding mem-
nts as well as compute time on CPU
ource consumption comparisons are
ences of varying lengths.

oolkit

lable two new models for encoding
embedding vectors. One makes use
former (Vaswani et al., 2017) architec-
ture, while the other is formulated as a Deep Aver-
aging Network (DAN) (Iyyer et al., 2015). Both
models are implemented in TensorFlow (Abadi

models are implemented in TensorFlow (Abadi et al., 2016) and are available to download as pre-trained models here:

https://tfhub.dev/google/universal-sentence-encoder/1

The models take as input English strings and produce as output a fixed dimensional embedding representation of the string. The encoding models

Colab

mo

https://colab.research.google.com/drive/1slkBSZGVi9BIkjD4JiILlV94PBw5AOO8?usp=sharing

TENSORFLOW HUB

https://colab.research.google.com/drive/1slkBSZGVi9BIkjD4JiILlV94PBw5AOO8?usp=sharing

Copia de TensorFlowHub_TheBasics_[MOJIX].ipynb

https://www.tensorflow.org/hub/hosting

```python
# Vamos a crear una variable con la url del modelo
# https://tfhub.dev/<publisher>/<model_name>/<version>

modelo_tfhub = "https://tfhub.dev/google/imagenet/mobilenet_v2_100_224/classification/4"
```

Paso 3: Define la arquitectura de tu modelo

Revisa la documentación del modelo que escogiste

```python
IMAGE_SHAPE = (224, 224)

modelo = Sequential([
    hub.KerasLayer(modelo_tfhub, input_shape=IMAGE_SHAPE+(3,))
])
```

Paso 4: Escoge una imagen para realizar las predicciones

```python
imagen_test_url = 'https://live.staticflickr.com/5449/7440792338_eb509ba9da_b.jpg'
```

✓ 0 s completado a las 21:01

Activities Google Chrome Thu Oct 28 9:01 PM es 100 %

[INT Clus Copi Clus Copi Copi PCA Copi Pima Tens Copi Tens Tens

tfhub.dev/google/imagenet/mobilenet_v2_100_224/classification/4

TensorFlow Hub

Search for models, collections & publishers

Send feedback

Back

Architecture: Dataset:

MobileNet V2 ImageNet (ILSVRC-2012-CLS)

Overall usage data

501.5k Downloads

Model formats

TF	.JS (v1, default)	.JS (v2, default)	.JS (v3, default)

Warning: An updated version of this module is available at https://tfhub.dev/google/imagenet/mobilenet_v2_100_224/classification/5

TF2.0 Saved Model (v4)

Usage data: 2.5k Downloads V4

archive (2).zip

Show all

Activities Google Chrome Thu Oct 28 9:02 PM es 100 %

colab.research.google.com/drive/19YFy6QQ6QDFKEhVSpeUprPvao8hnV3F1#scrollTo=nuERg7MGz2OC

Copia de TensorFlowHub_TheBasics_[MOJIX].ipynb

Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda Se han guardado todos los cambios

Comentario Compartir

+ Código + Texto RAM Disco Editar

Paso 3: Define la arquitectura de tu modelo

Revisa la documentación del modelo que escogiste

[3]
```
IMAGE_SHAPE = (224, 224)

modelo = Sequential([
    hub.KerasLayer(modelo_tfhub, input_shape=IMAGE_SHAPE+(3,))
])
```

Paso 4: Escoge una imagen para realizar las predicciones

```
imagen_test_url = 'https://live.staticflickr.com/5449/7440792338_eb509ba9da_b.jpg'
# "https://boliviatravelsite.com/Images/Attractionphotos/illimani-02.jpg"
imagen_test_url
```

Vamos a descargar la imagen y vamos a ajustar sus dimensiones

[ ]
```
imagen_test = tf.keras.utils.get_file("image"+str(np.random.randint(100))+'.jpg',imagen_test_url,  )
```

✓ 3 s completado a las 21:02

archive (2).zip Show all

Copia de TensorFlowHub_TheBasics_[MOJIX].ipynb

Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda Se han guardado todos los cambios

```
    hub.KerasLayer(modelo_tfhub, input_shape=IMAGE_SHAPE+(3,))
])
```

Paso 4: Escoge una imagen para realizar las predicciones

```
imagen_test_url = 'https://live.staticflickr.com/5449/7440792338_eb509ba9da_b.jpg'
# "https://boliviatravelsite.com/Images/Attractionphotos/illimani-02.jpg"
imagen_test_url
```

Vamos a descargar la imagen y vamos a ajustar sus dimensiones

```
imagen_test = tf.keras.utils.get_file("image"+str(np.random.randint(100))+'.jpg',imagen_test_url,  )
imagen_test = Image.open(imagen_test).resize(IMAGE_SHAPE)
imagen_test
```

Vamos a ajustar los valores de la imagen

```
imagen_test = np.array(imagen_test)/255.0
imagen_test.shape
```

3 s completado a las 21:02

colab.research.google.com/drive/19YFy6QQ6QDFKEhVSpeUprPvao8hnV3F1#scrollTo=6r7Elzg_17Xe

Copia de TensorFlowHub_TheBasics_[MOJIX].ipynb

Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda

+ Código + Texto

Vamos a ajustar los valores de la imagen

```
imagen_test = np.array(imagen_test)/255.0
imagen_test.shape
```

Paso 5: Realizamos predicciones

Obtenemos la predicción bajo la estructura de las clases con las que ha sido entrenado el modelo

https://tfhub.dev/google/imagenet/mobilenet_v2_130_224/classification/4

✓ 0 s completado a las 21:03

archive (2).zip

Paso 5: Realizamos predicciones

Obtenemos la predicción bajo la estructura de las clases con las que ha sido entrenado el modelo

https://tfhub.dev/google/imagenet/mobilenet_v2_130_224/classification/4

```
[7] prediccion = modelo.predict(imagen_test[np.newaxis, ...])
    prediccion.shape
```

```
(1, 1001)
```

```
[8] prediccion
```

```
array([[ 0.24092045, -0.31635118,  1.8853828 , ..., -1.0570533 ,
         1.0253358 , -0.7496758 ]], dtype=float32)
```

```
clase_prediccion = np.argmax(prediccion)
clase_prediccion
```

```
664
```

Descargamos las clases en formato de texto

```
[ ] labels_path = tf.keras.utils.get_file('ImageNetLabels.txt','https://storage.googleapis.com/download.tensorflow.org/data/ImageNetLabels.txt')
```

colab.research.google.com/drive/19YFy6QQ6QDFKEhVSpeUprPvao8hnV3F1#scrollTo=Jx9rRt2Q6OP6

Copia de TensorFlowHub_TheBasics_[MOJIX].ipynb

Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda

Comentario Compartir

+ Código + Texto

RAM Disco Editar

```
1.0253358 , -0.7496758 ]], dtype=float32)
```

```
[9] clase_prediccion = np.argmax(prediccion)
    clase_prediccion
```

```
664
```

Descargamos las clases en formato de texto

```
[10] labels_path = tf.keras.utils.get_file('ImageNetLabels.txt','https://storage.googleapis.com/download.tensorflow.org/data/ImageNetLabels.txt')
     imagenet_labels = np.array(open(labels_path).read().splitlines())
```

```
Downloading data from https://storage.googleapis.com/download.tensorflow.org/data/ImageNetLabels.txt
16384/10484 [=============================================] - 0s 0us/step
24576/10484 [======================================================================] - 0s 0us/step
```

Vamos a obtener el nombre de la clase de nuestra imagen

```
imagenet_labels[clase_prediccion]
```

```
'monastery'
```

0 s completado a las 21:05

archive (2).zip

Show all

Ruth Chirinos7:11 PM
Podemos grabar la sesión?

Nathaly Alarcon7:39 PM
https://colab.research.google.com/drive/199MZvaBiKl3zrn0-gyNcAS8GCU96IP26?usp=sharing

Nathaly Alarcon8:04 PM
https://colab.research.google.com/drive/1wPTr22hDWloyU1Nn0bI2oxfmWwAjrARo?usp=sharing

Nathaly Alarcon8:19 PM
https://www.displayr.com/what-is-hierarchical-clustering/#:~:text=Hierarchical%20clustering%2C%20also%20known%20as,broadly%20similar%20to%20each%20other.

Nathaly Alarcon8:22 PM
https://realpython.com/k-means-clustering-python/

https://colab.research.google.com/drive/1LZ86ki2A0naIWHDnwuV2wBU1LcfV7JIg?usp=sharing

Nathaly Alarcon8:34 PM
https://colab.research.google.com/drive/12VL_fYaT04EGNKfHsQP-CrSvIOKwCPV4?usp=sharing

https://www.kaggle.com/uciml/pima-indians-diabetes-database

You8:46 PM
se puede saber cuales son esos componentes en el dataset (los nombres de las columnas)?

Aldo Gutierrez8:47 PM
osea son planos entre componentes Xs?