

Redes Neuronales en Scikit-Learn

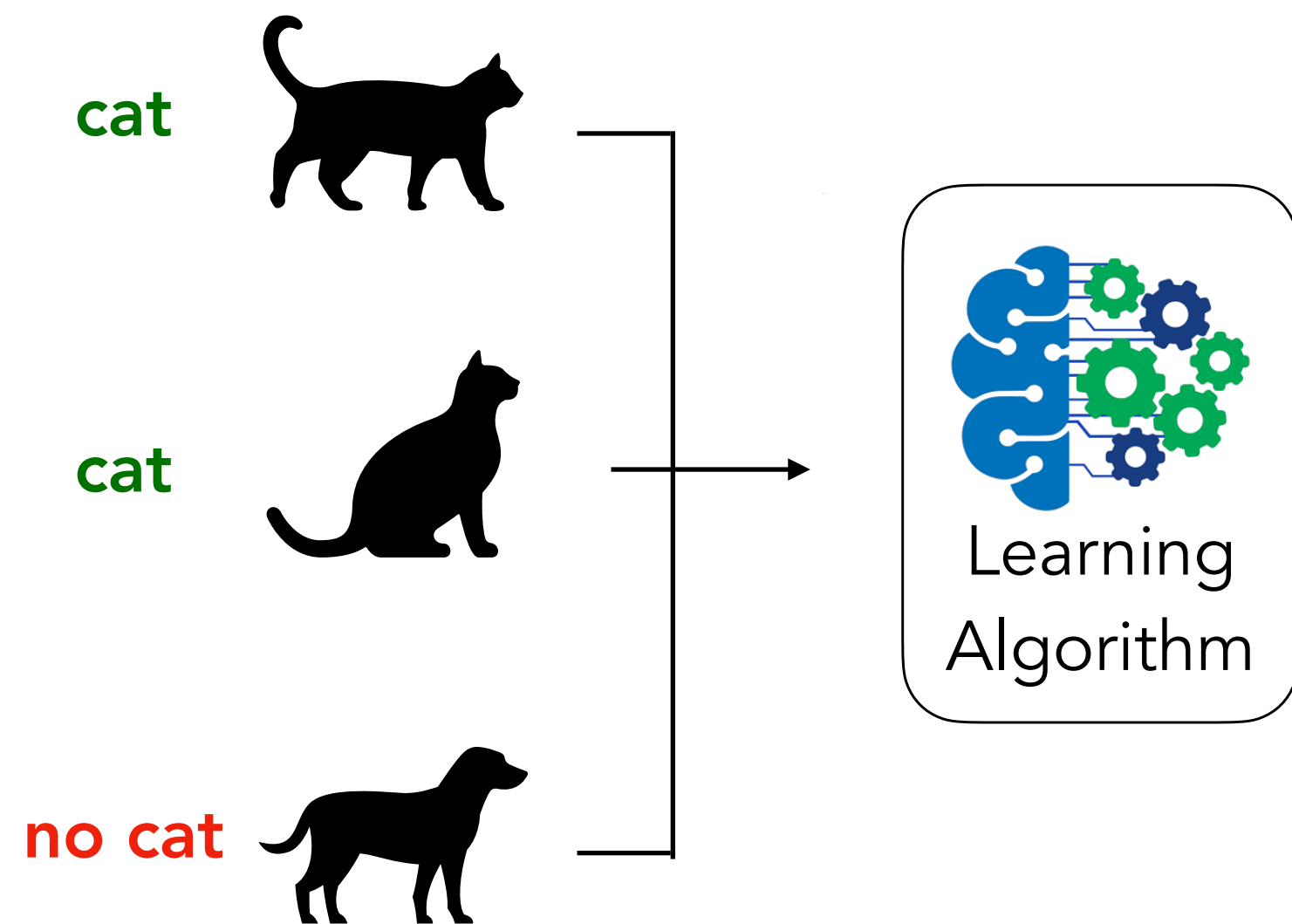
Facundo Molina

Departamento de Computación - FCEFQyN
Universidad Nacional de Río Cuarto, Argentina

Aprendizaje Supervisado

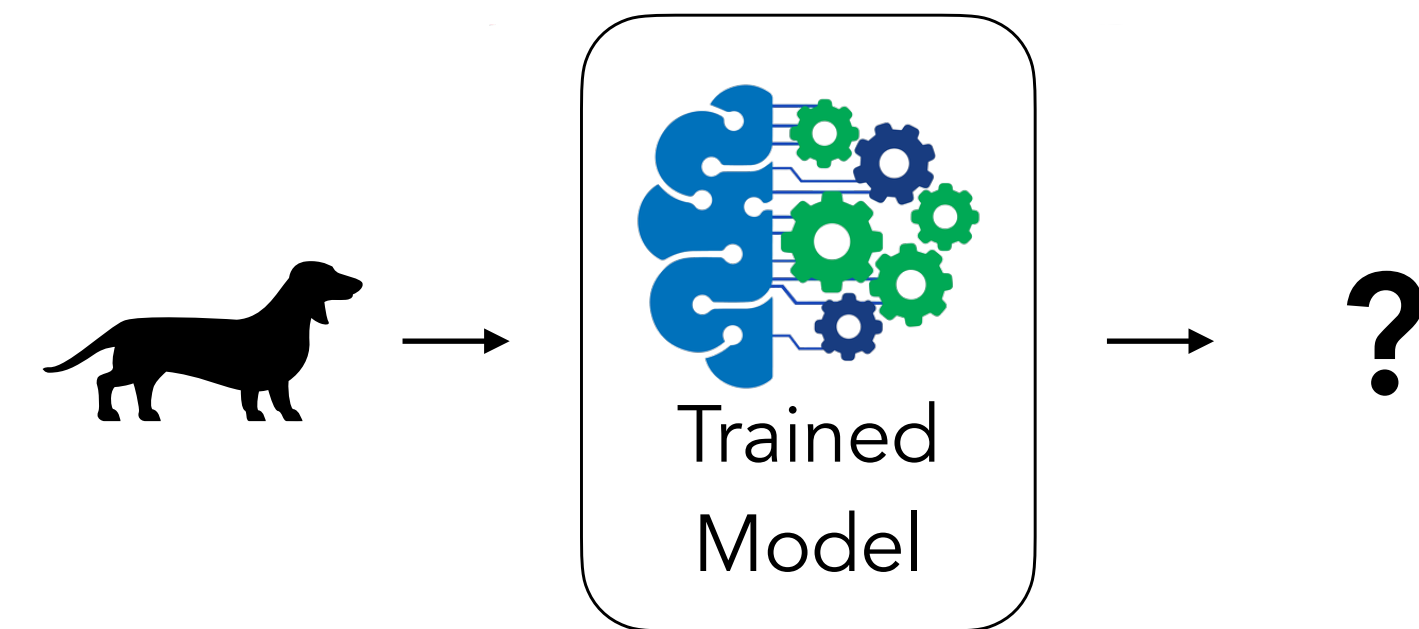
Fase 1

Proveer al algoritmo datos de **entrenamiento** etiquetados



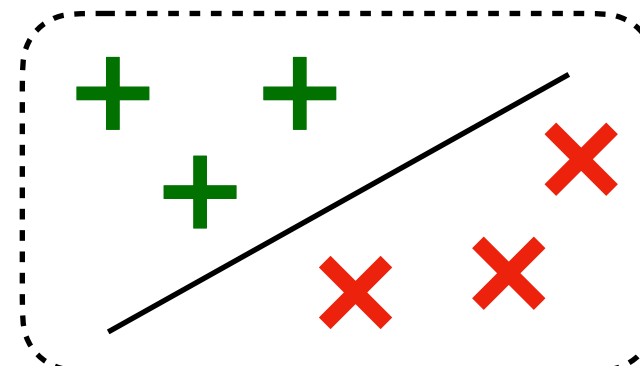
Fase 2

Utilizar el algoritmo para **predecir** la etiqueta de nuevos datos

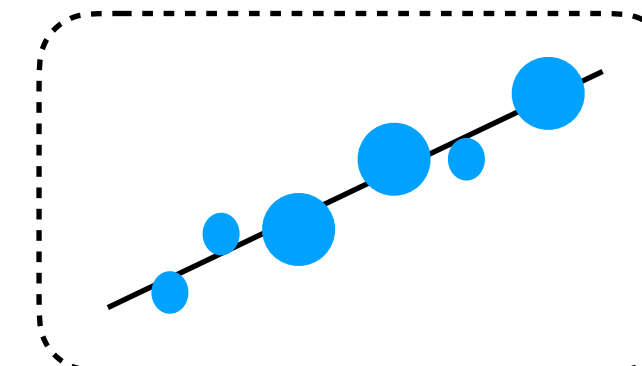


Tipos de
Problemas

Clasificación



Regresión

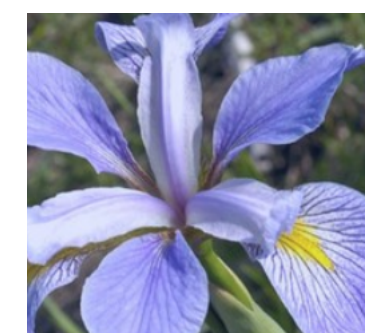


Un Problema de Clasificación

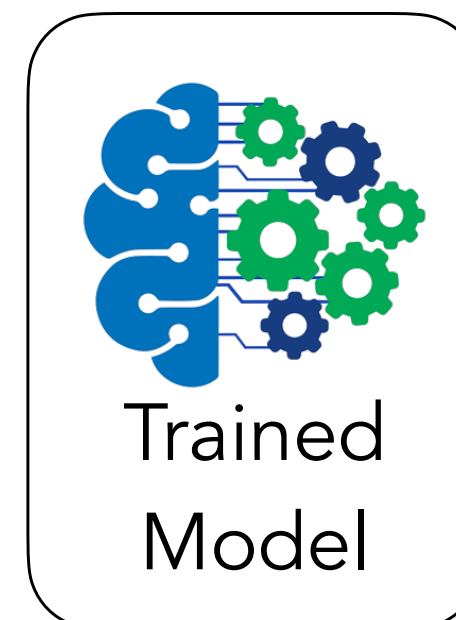
Iris
flowers



Problema: crear un modelo que pueda clasificar distintas especies de la flor Iris.



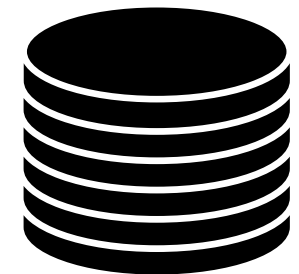
Iris flower



Flower type

Pipeline

Preparar los Datos



Iris data

Get the data

Cleaning

Encoding

Tagging

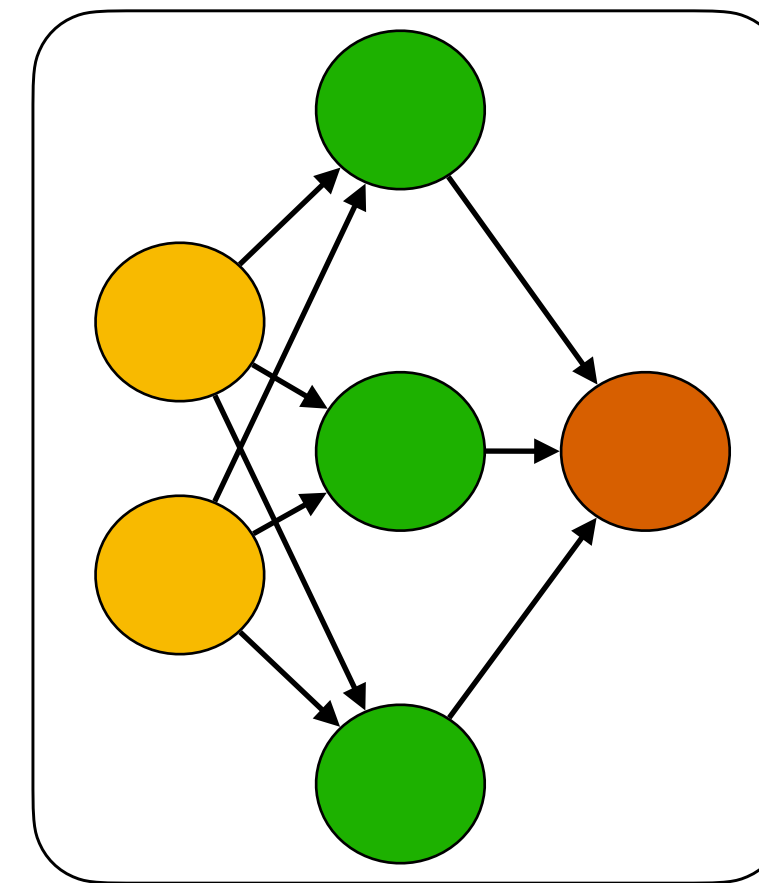
Construir & Entrenar el Modelo

Split the data

Model selection

Model training

Model validation



Red Neuronal

Deploy

Deployment

Monitoring

Stack: Pandas & Scikit-Learn

Preparación de los Datos



Librería open-source de análisis
y manipulación de datos

<https://pandas.pydata.org/>

Entrenamiento y Validación del Modelo

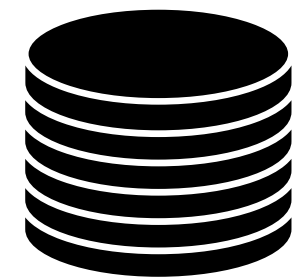


Librería open-source para
Machine Learning en Python

<https://scikit-learn.org/>

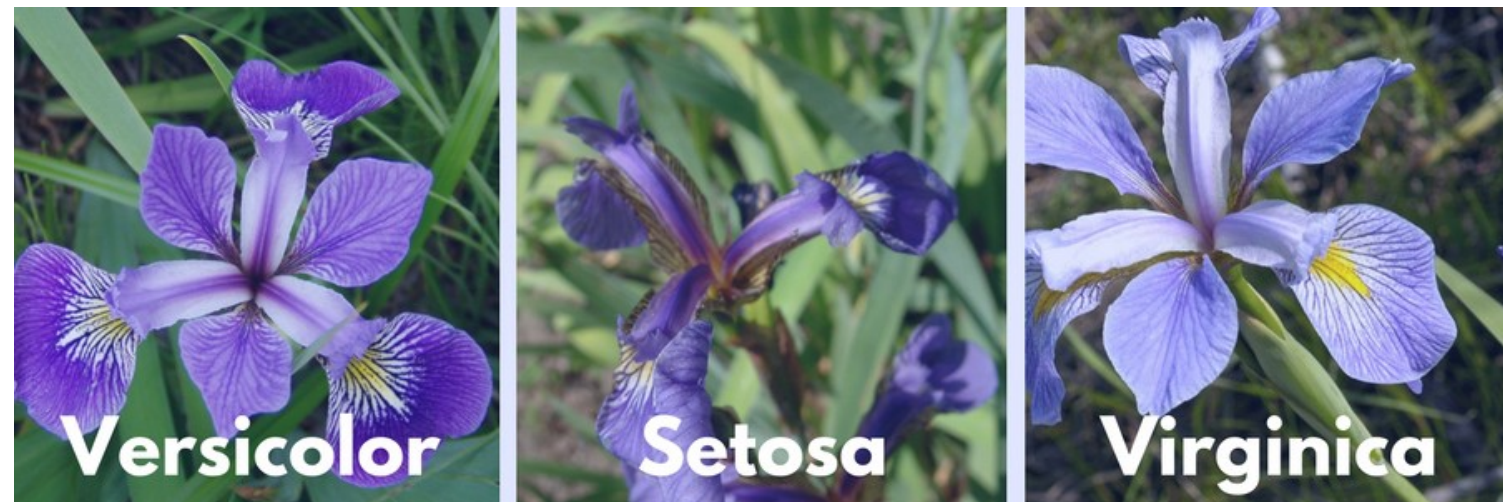
Preparación de los Datos

Obteniendo los Datos

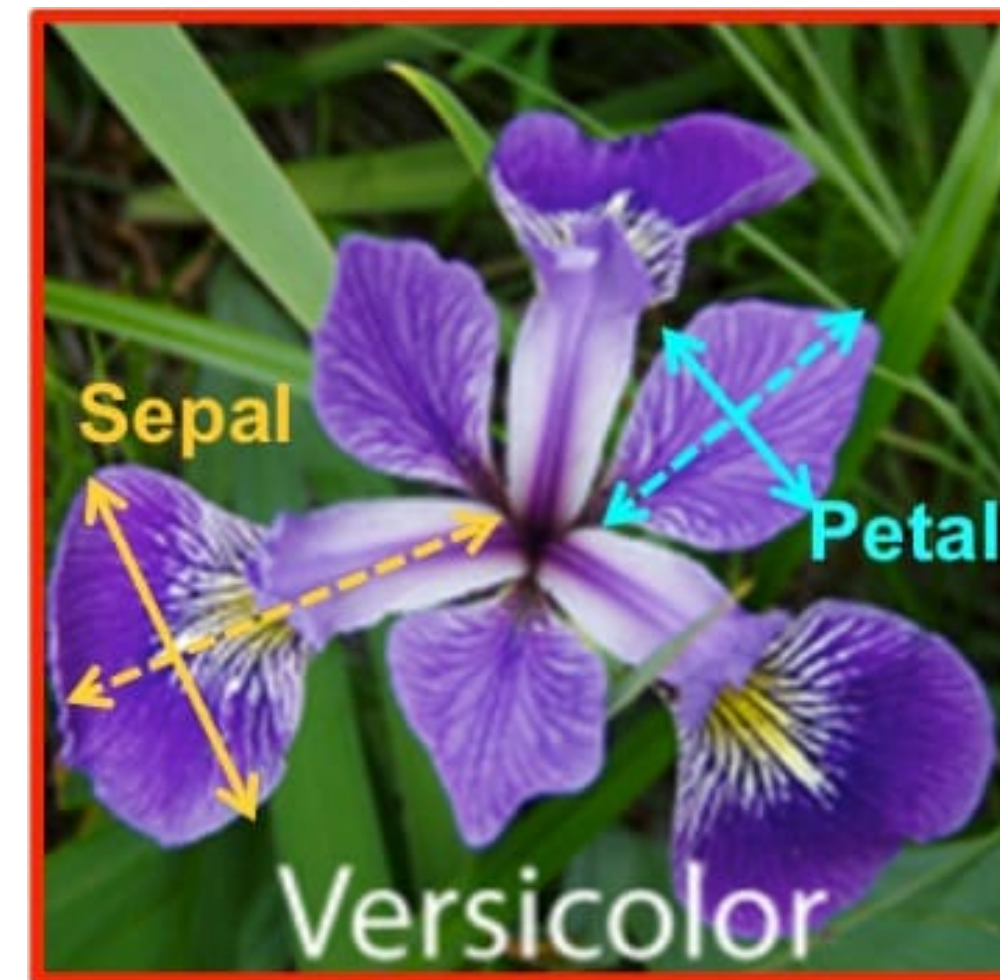


= Flowers description + class

Iris data



Codificando los Datos



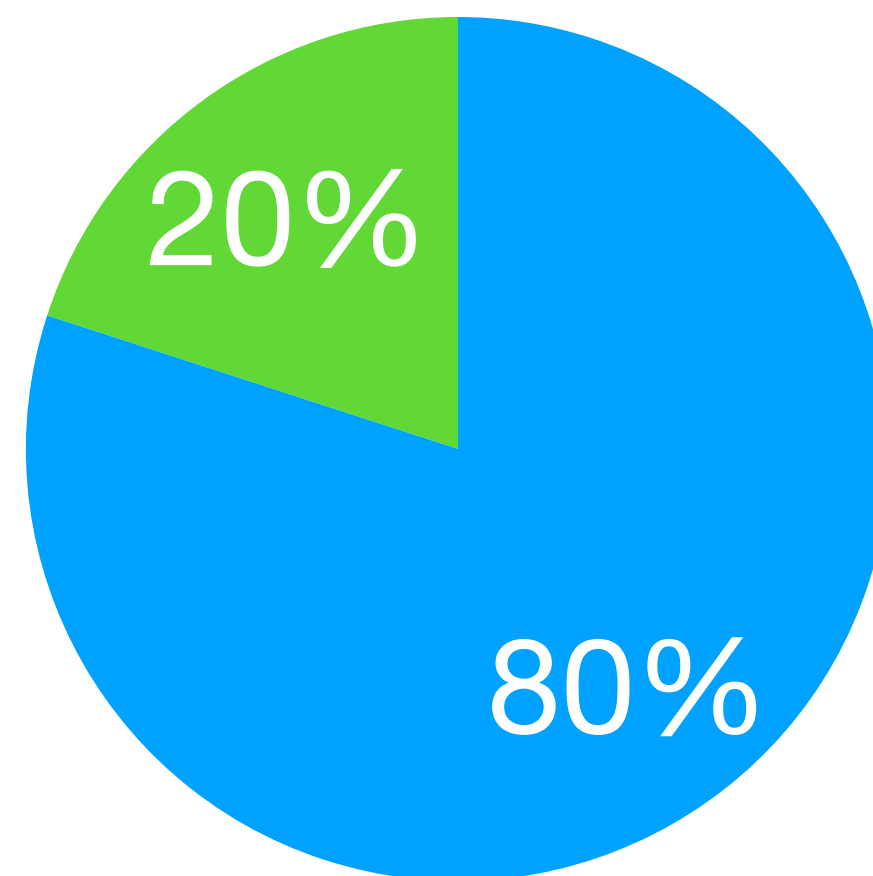
<https://archive.ics.uci.edu/ml/datasets/iris>

[sepal-length, sepal-width, petal-length, petal-width, class]

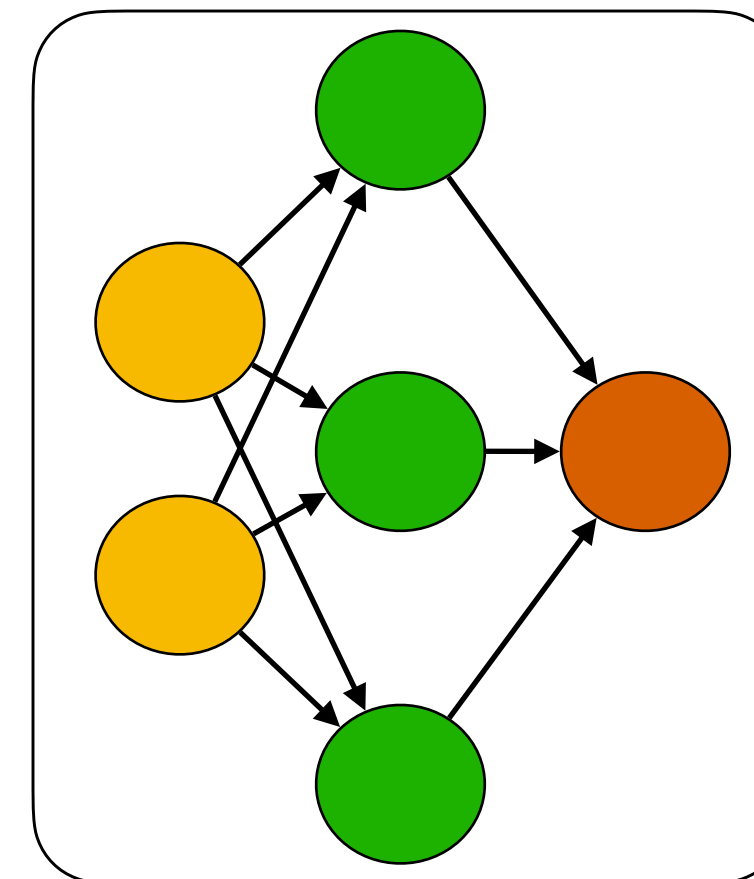
Entrenando una Red Neuronal

Split de los Datos

● Training ● Testing



Construcción & Entrenamiento del modelo



Red Neuronal

Cuántas **neuronas** en el **input layer**?

Cuántos **layers** intermedios?

Cuántas **neuronas** en cada **layer**?

Cuántas **iteraciones** de training?

Entrenando una Red Neuronal

Validación del modelo

Matriz de confusión - Precision/Recall

| | | True Class | |
|-----------------|----------|------------|----------|
| | | Positive | Negative |
| Predicted Class | Positive | TP | FP |
| | Negative | FN | TN |

Positive class

Precision: $tp / (tp + fp)$

Porcentaje de instancias que el modelo predijo como positivas y realmente lo eran

Recall: $tp / (tp + fn)$

Porcentaje de instancias realmente positivas que el modelo predijo como tales

Conclusiones

Las Redes Neuronales pueden funcionar muy bien en problemas de Clasificación y de Regresión

Pandas & Scikit-learn son dos librerías básicas para la resolución de problemas con Machine Learning

<https://github.com/facumolina/demo-neural-nets>

Algunas Recomendaciones

Libros

- *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*, Aurélien Géron, 2017.
- *Deep Learning*, Ian Goodfellow and Yoshua Bengio and Aaron Courville, 2016

Datasets (conjuntos de datos listos para distintas tareas de ML)

- UCI Machine Learning Repository - <https://archive.ics.uci.edu/ml/datasets.php>
- Machine Learning Mastery Repository - <https://github.com/jbrownlee/Datasets>

Tools (otras herramientas además de Pandas y Scikit-Learn para resolver problemas de ML en Python)

- <https://keras.io/>
- <https://www.tensorflow.org/>
- <https://pytorch.org/>

Websites

- <https://machinelearningmastery.com/> (guías y tutoriales sobre problemas y estrategias en el mundo ML principalmente en Python. En esta sección hay un resumen de todas las cosas que pueden ver: <https://machinelearningmastery.com/start-here/#getstarted>)
- <https://www.kaggle.com/> (sitio muy famoso por las competencias de problemas de ML que hay, entre otras cosas)