Introduction to the basics of AI

F. TAIA-ALAOUI

Outline for today’s course

- Roundtable: Self-Introduction + Expectations of this course
- General Presentation of AI Methods
- Environment Setup
- Small program execution

Vocabulary

Learning:

A computer program is said to learn from **experience E** with respect to some class of **tasks T** and **performance measure P**, if its performance at tasks in T, as measured by P, improves with experience E. [Tom Mitchell] (link)

Vocabulary

Experience

Experience involves information integration, and information requires data.

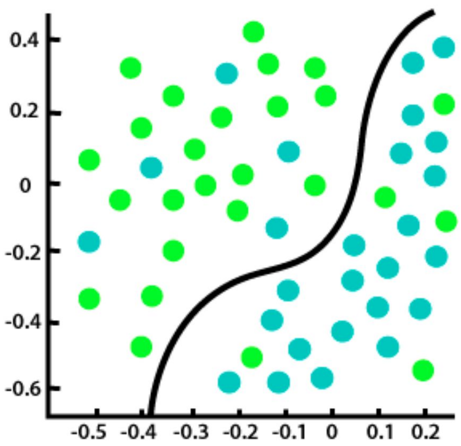
The more data a program is given, the more experience it has.

Tasks

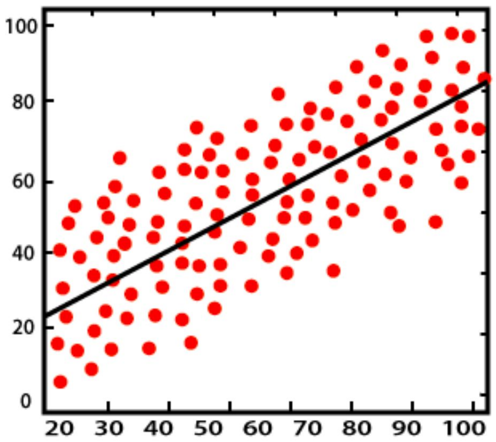
Tasks in AI are similar to those required from humans. Common tasks may involve face recognition, voice recognition, text generation, or simply the prediction of information B based on information A (for instance, tell whether a person is man of woman based on photos)

Vocabulary

Tasks involve two main processes: Classification for categorical variables, Regression for continuous variables.



Classification



Regression

Small Test - Tasks

Up to you… what kind of technique (classification or regression type) is required to perform the following tasks:

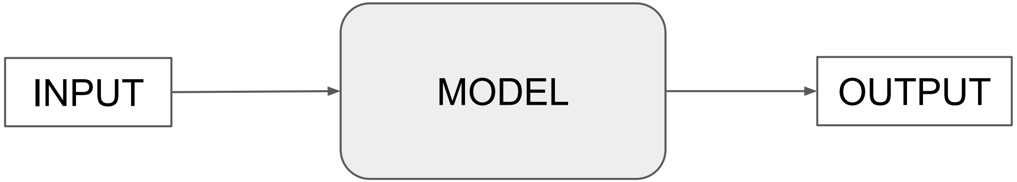
- Object recognition (for example, a car on an image)
- The temperature tomorrow based on the weather today
- The battery lifetime based on its features
- The team who will win at soccer
- Fraudulent email
- Patient with disease
- Patient’s heart rate in the next hour based on their current activity

Small test - Data

Up to you… what are the data needed to train a model that performs the following tasks :

- Object recognition (for example, a car on an image)
- The temperature tomorrow based on the weather today
- The battery lifetime based on its features
- The team who will win at soccer
- Fraudulent email
- Patient with disease

Models



INPUT

MODEL

OUTPUT

Image	Object label (face, cat, car, etc.)
Historical weather data	Future weather data
Historical patient data	Current health state
Historical patient data	Future health state
Voice records	Speech transcription
Email content	Fraud State: Yes or No
Text in English	Text in German

Data Types and Formats

Image	3D Array (row, column, channel)
Historical weather data	2D Array (time, weather features)
Historical patient data	2D Array (time, physiological features)
Voice records	2D Array (time, magnitude)
Email content	Vector (sequence of tokens)

AI pipelines and frameworks

Tensorflow

- Tensors (https://www.tensorflow.org/guide/tensor)
- Built-in Methods (https://www.tensorflow.org/guide/keras/training_with_built_in_methods)
- Tokenizers (https://www.tensorflow.org/text/guide/tokenizers)

Pytorch

- Tensors (https://pytorch.org/tutorials/beginner/nlp/pytorch_tutorial.html)
- Built-in Methods (https://pytorch.org/tutorials/beginner/introyt/modelsyt_tutorial.html)
- Tokenizers (https://pytorch.org/tutorials/beginner/torchtext_custom_dataset_tutorial.html)

Machine Learning VS Deep Learning

In ML, features are said to be “handcrafted” or “expert-made”

In DL, features are computed by the model itself

Link to video

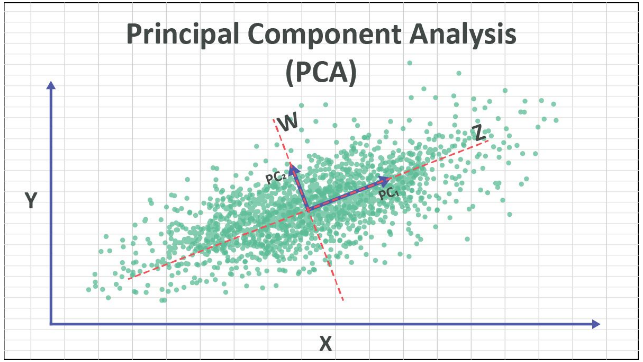
Feature Engineering Methods

Sometimes inputs need to be transformed in order to communicate the right information. For example, to recognize a face it has to be round, ro recognize a tree it has to be green, etc.

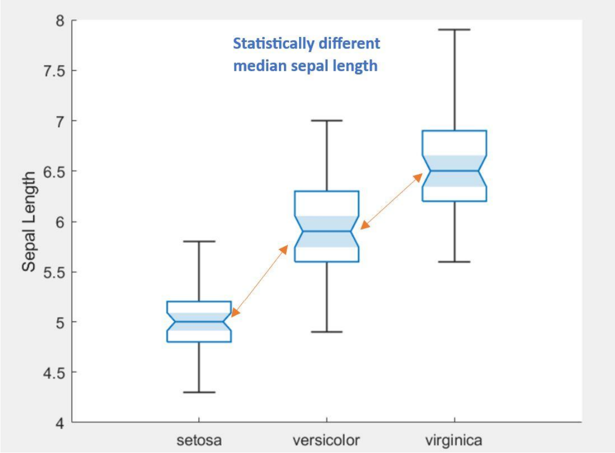
Some features are known thanks to the expertise of scientists, therefore the features are handcrafted

Sometimes handcrafted features do not lead to the desired results, hence, models are relied on to compute “deep features” that lose physical meaning but result into satisfying separation in the feature space

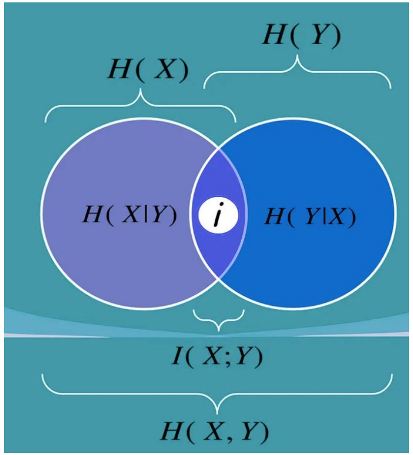
Feature Engineering Methods



MANOVA



Mutual Information

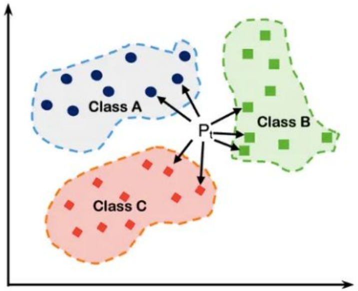


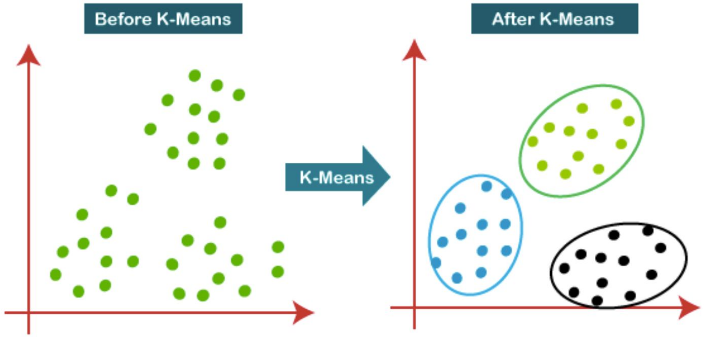
ML Methods

Supervised = known output	**Unsupervised = unknown outputs**
K-Nearest Neighbors (KNN)	K-MEANS
Support Vector Machines (SVM)	Decision Trees
Decision Trees	Expectation Minimization
Random Forest	Linear Regression
Naive Bayes	Principal Component Analysis
Neural Networks	AutoEncoders

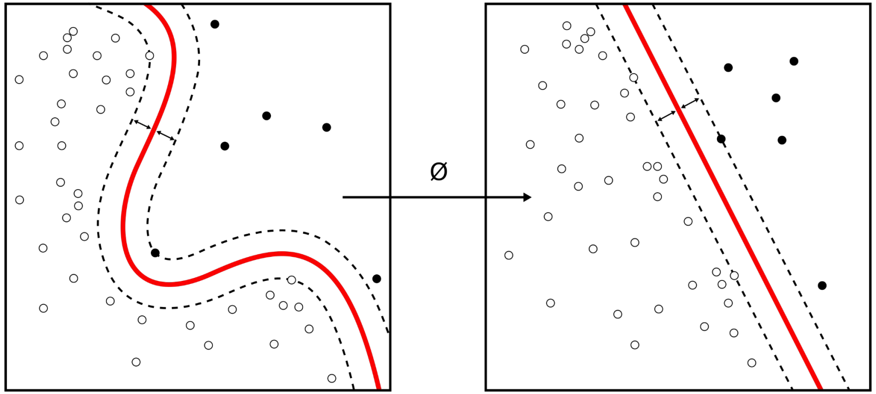
ML Methods

K Nearest Neighbors

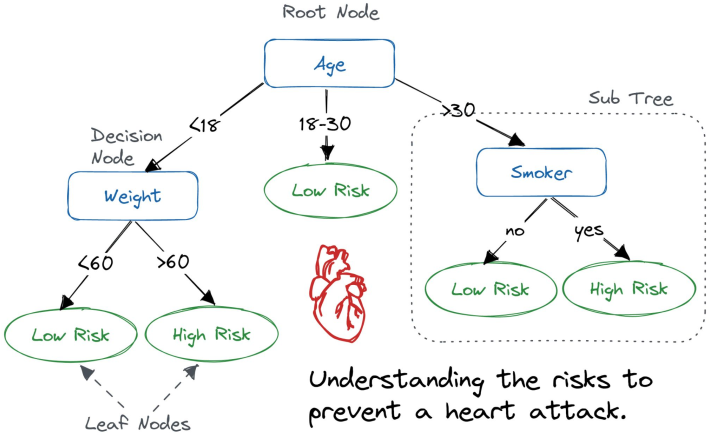




SVM

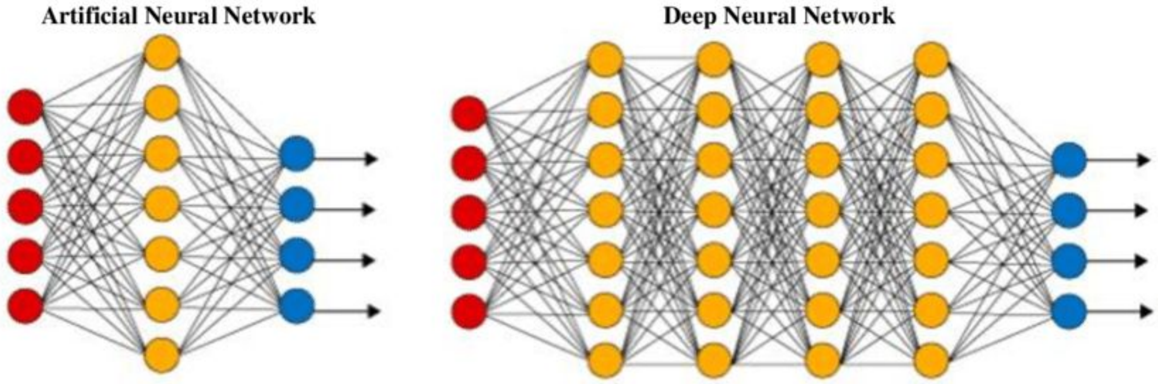


Decision Tree



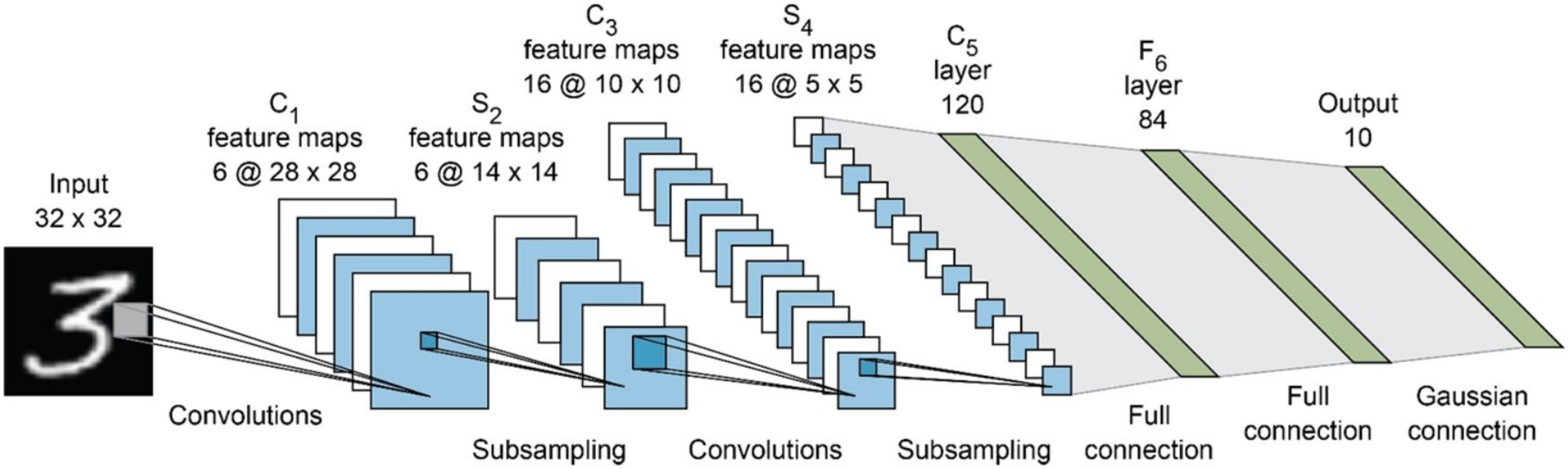
DL Methods

- Vanilla Neural Networks (shallow) & Deep neural networks (Deep networks)



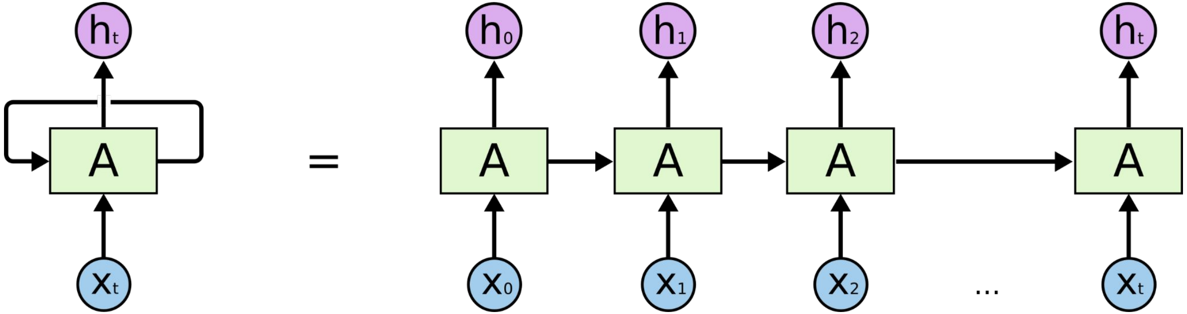
DL Methods

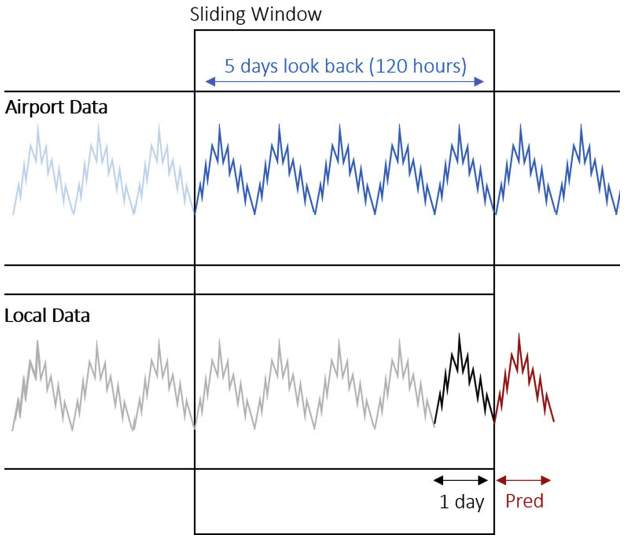
- Convolutional Neural Networks (CNN)



DL Methods

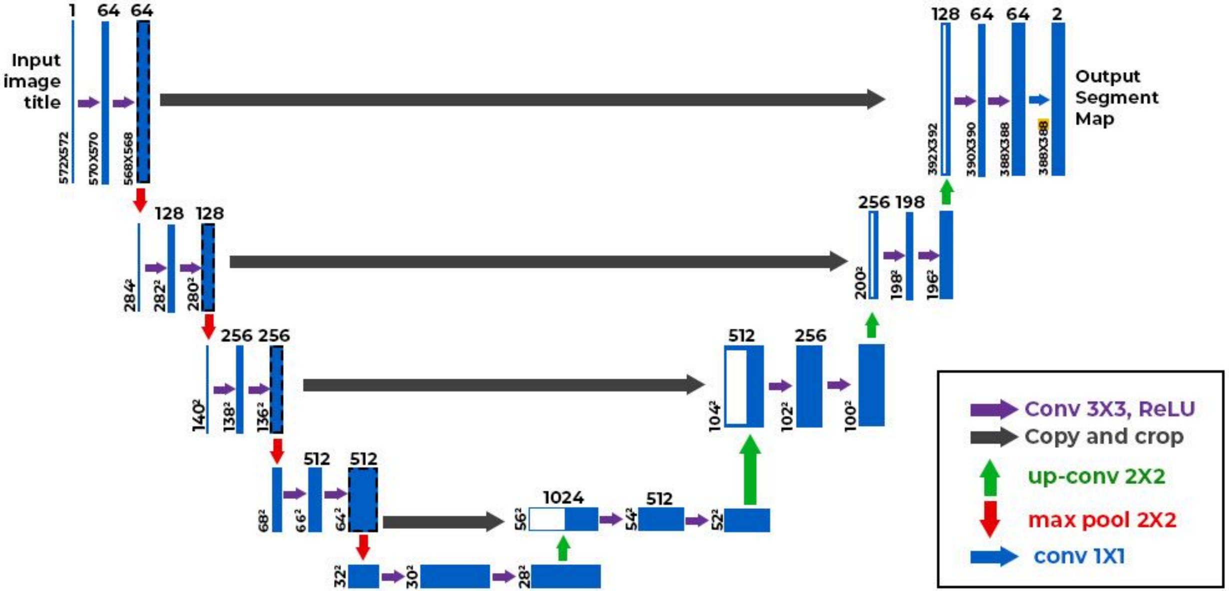
- Long Short Term Memory Neural Networks (LSTM)

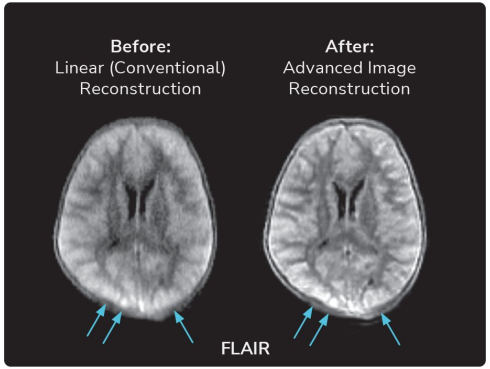




DL Methods

- Encoder-Decoders

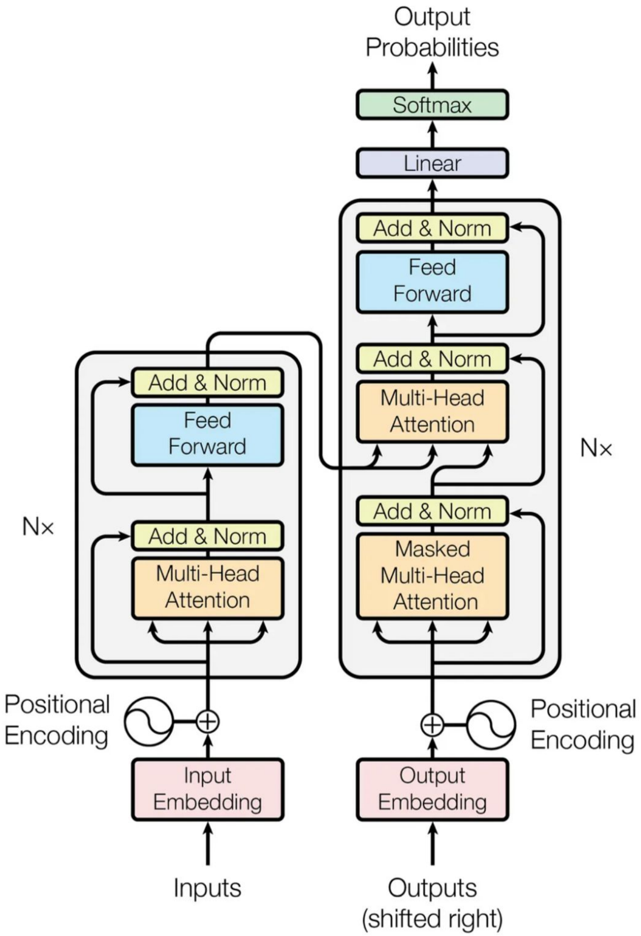


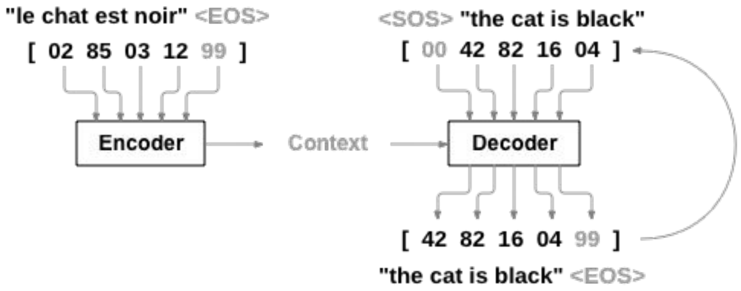


Before:
Linear (Conventional) Reconstruction
After:
Advanced Image Reconstruction
FLAIR

DL Methods

- Transformers





Environment Setup

- Install Python (https://www.python.org/downloads/)
- Install pip (https://pip.pypa.io/en/stable/installation/)
- Install numpy (https://numpy.org/install/)
- Install Anaconda (https://docs.anaconda.com/free/anaconda/install/)
- Install Jupyterlab , either using pip or conda (https://jupyterlab.readthedocs.io/en/stable/getting_started/installation.html)
- Pandas (https://pandas.pydata.org/docs/getting_started/install.html)
- Matplotlib (https://matplotlib.org/stable/install/index.html)
- Sklearn (https://pypi.org/project/scikit-learn/)

PCA program

https://github.com/TALAOUI/Introduction-to-the-basics-of-AI/tree/main