



UNIVERSIDAD DE CHILE

Deep Learning

Deeper, Better, _____, Stronger than Machine Learning

Valentin Barriere

Universidad de Chile – DCC

CC6204, Primavera 2025

Introducción

Deep Learning

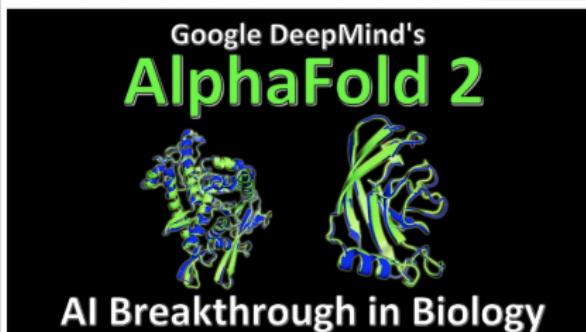
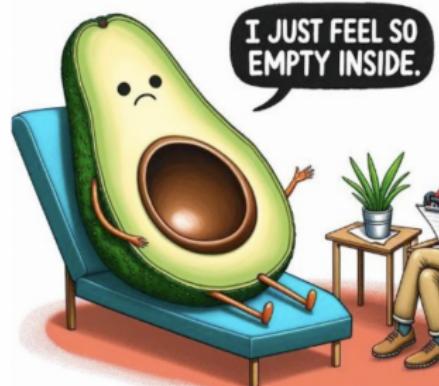
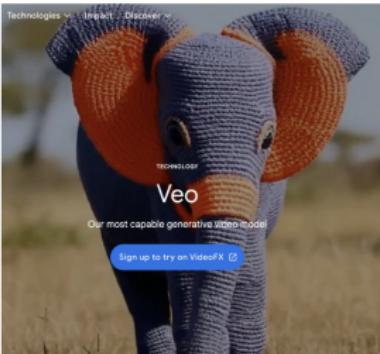


Figure 1: Conversational Agent, Image/Video generation, Protein structure

Outline : Significacion de las termas

Significacion de las termas

History

Principle

Applications

Ressources

The course

AI vs. Data Science vs. Machine Learning

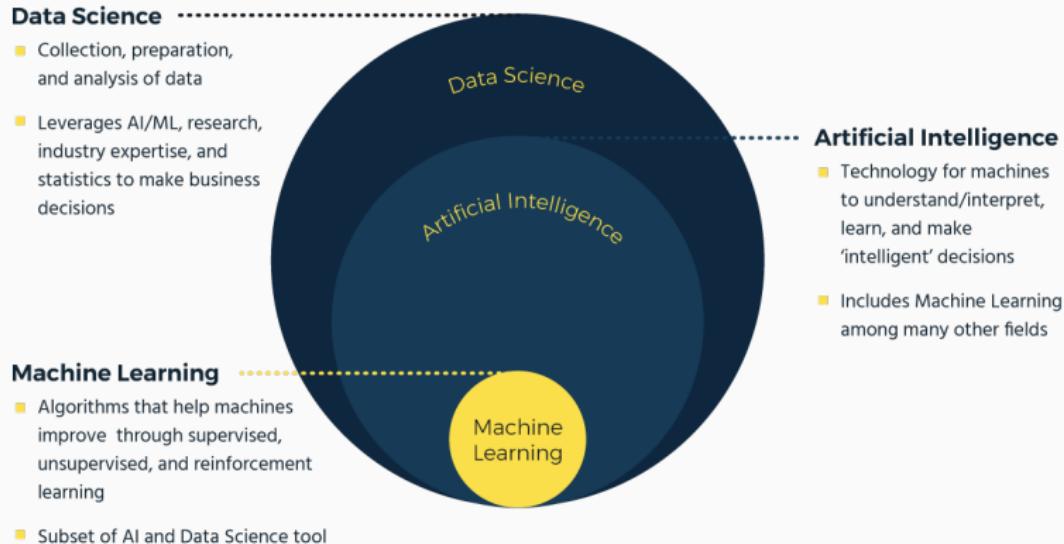


Figure 2: Diferencias entre campos

In summary

Data Science focuses on analyzing data to extract knowledge, Machine Learning uses algorithms to make predictions and decisions based on data, and Artificial Intelligence refers to the development of systems that can perform intelligent tasks autonomously.

Overly simplistic definition:

- Data mining generates understanding.
- Machine learning generates predictions.
- Artificial intelligence generates actions.

Example on a Music Platform

Data Scientist

Collects and analyzes user data from music platforms to identify patterns and musical preferences.

Machine Learner

Develops and optimizes a music recommendation model using machine learning algorithms to predict user preferences.

Artificial Intelligence

Implements a social agent that can interact with the user, to enhance the personalization of music recommendations and provide a more accurate and contextualized experience.

Data

Different types of data:

- Structured data:
 - Social data: Age, Salary, Skin color, Place of residence
 - Metric data: *Likes* on a post, Time spent on a page, Number of mutual friends
- Unstructured data:
 - Text: Sentence, Paragraph, Document
 - Sound: Song, Speech
 - Image: Photo, Video

Different types of Mining:

- Data exploration: Detect simple values, biases
- Classification/Regression task: Use data to characterize new data **by class or with a value** in a supervised manner
- Clustering task: Group data into classes in an unsupervised manner
- Dimensionality reduction: Develop common structures for compressed data representations

Different types of Mining:

- Data exploration: Detect simple values, biases
- Classification/Regression task: Use data to characterize new data **by class or with a value** in a supervised manner
- Clustering task: Group data into classes in an unsupervised manner
- Dimensionality reduction: Develop common structures for compressed data representations
- **Generation: generate data wrt a specific distribution particular**
(more complex)

Outline : History

Significacion de las termas

History

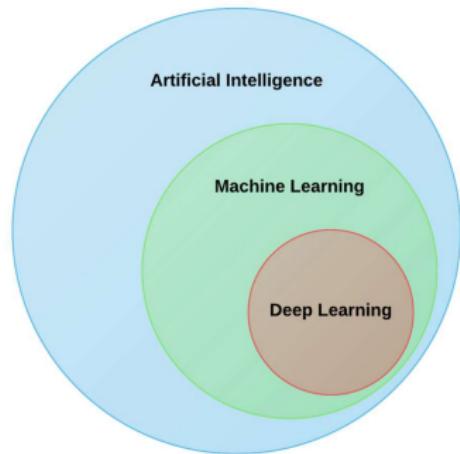
Principle

Applications

Ressources

The course

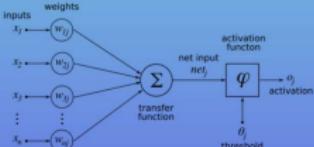
Qué es Deep learning?



Redes neuronales no siempre fueron profundas

1943 El primer modelo matemático de una neurona (McCulloch-Pitts)

1957 El perceptrón



1959 Hubel y Wiesel descubrieron las células simplex y complex en sistemas de visión biológica.

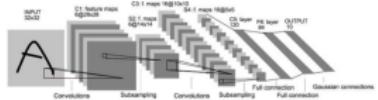
1965 Primera red profunda (8 capas) – Ivakhnenko and Lapa

1979 Red neuronal para reconocer patrones visuales (Neocognitron) – Fukushima

1982 Primera red recurrente – Hopfield

1986 Algoritmo Backpropagation

1989 Redes neuronales convolucionales – Handwritten recognition



1989 Reinforcement learning – Q-learning

Redes neuronales no siempre fueron profundas (no tanto)



2015 ~ Explosión de la industria DL

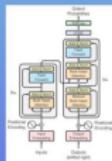


Historia

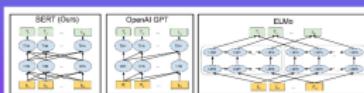
Redes neuronales profundas

2014 Mecanismo de atención – Bahdanau et al.

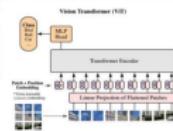
2017 Arquitectura Transformer – Machine translation.



2018-2020 Modelos de NLP: BERT, GPTs, XLM



2020 Vision Transformer

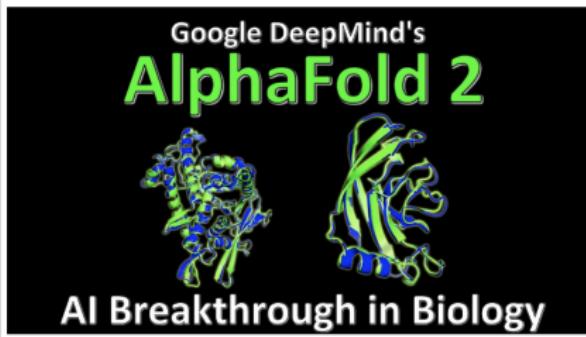
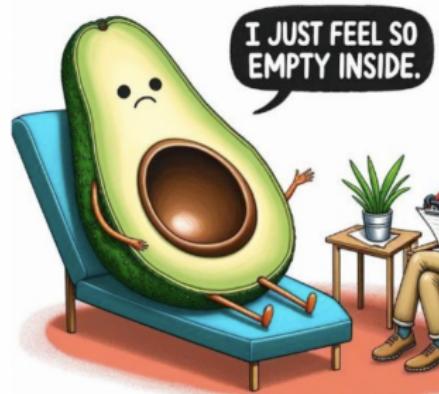
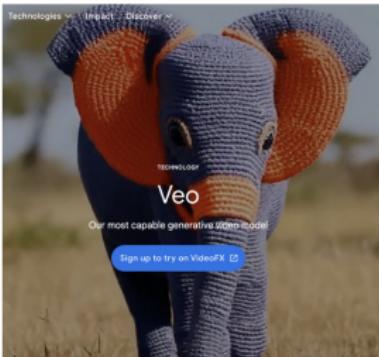


2021 MLP Mixer



2022 Dall-e, Imagen, LaMDA

And now this



Outline : Principle

Significacion de las termas

History

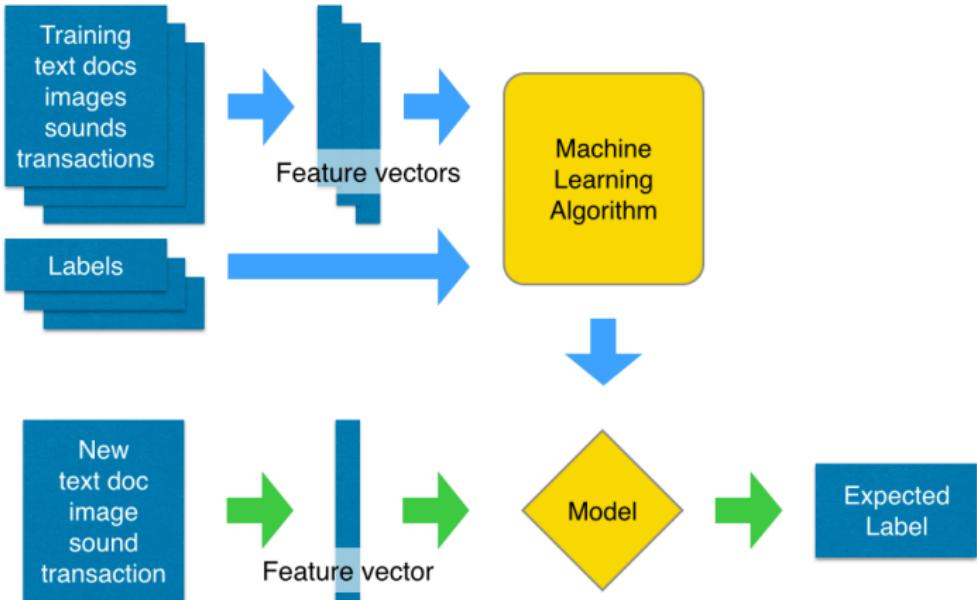
Principle

Applications

Ressources

The course

Supervised Machine Learning

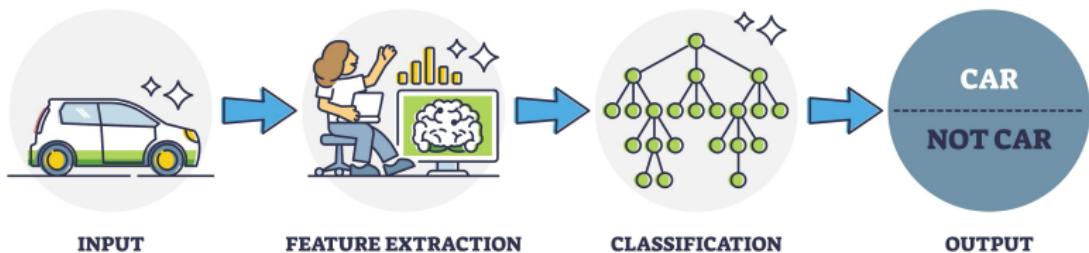


Predictive Modeling Data Flow

ML: Need to **create the feature vectors** to use them as input.

Differences between ML y DL: Feature Extraction

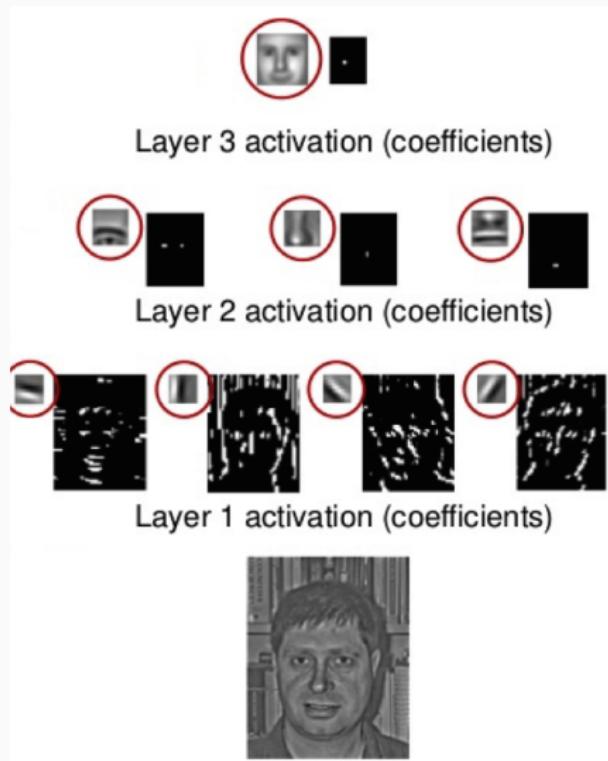
MACHINE LEARNING



DEEP LEARNING

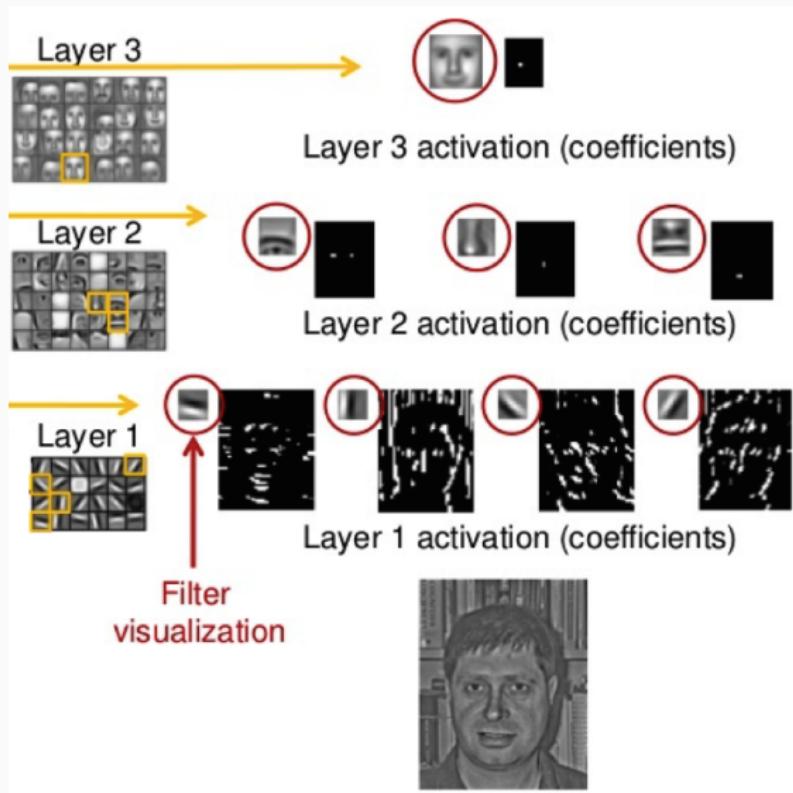


Representation Complexity: Example



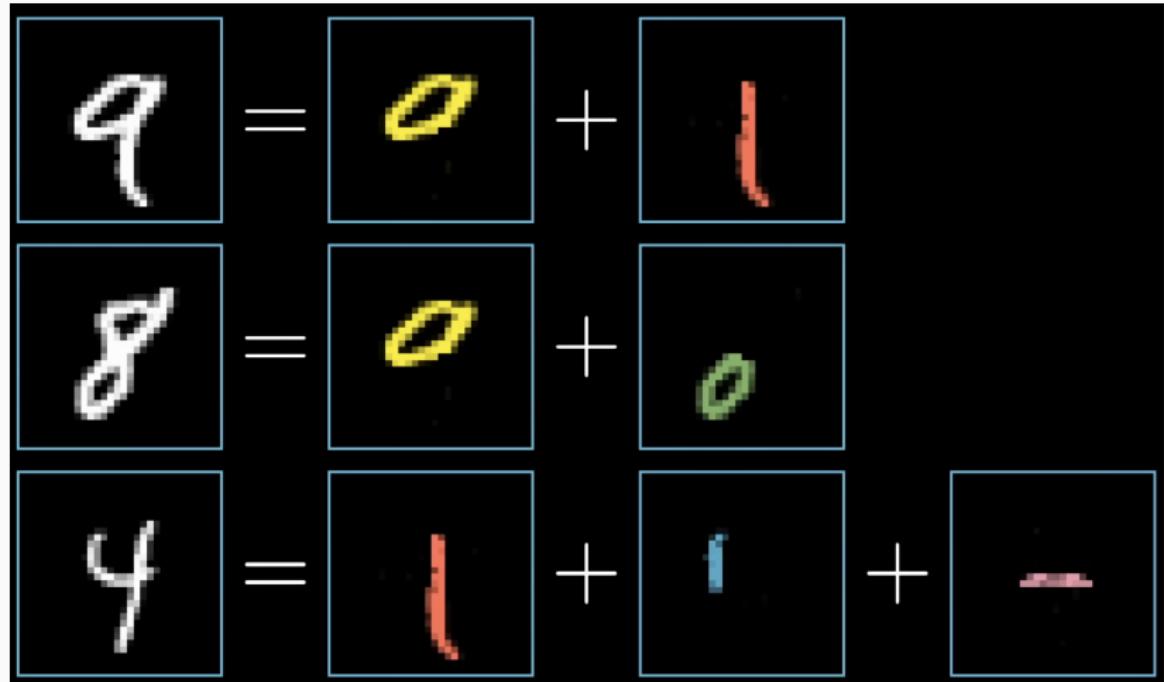
DL **extract feature vectors from raw data**

Representation Complexity: Compositionality



DL extract feature vectors from raw data and **combine them**

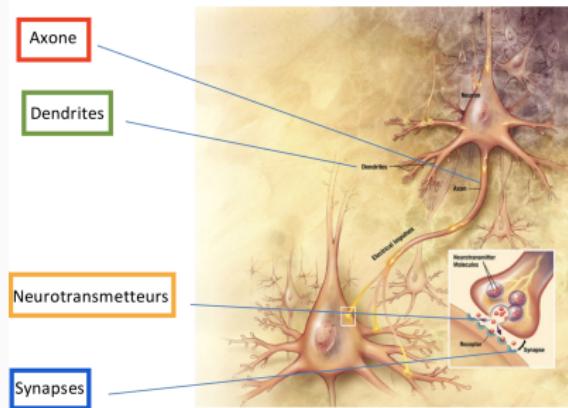
Representation Complexity: Compositionality



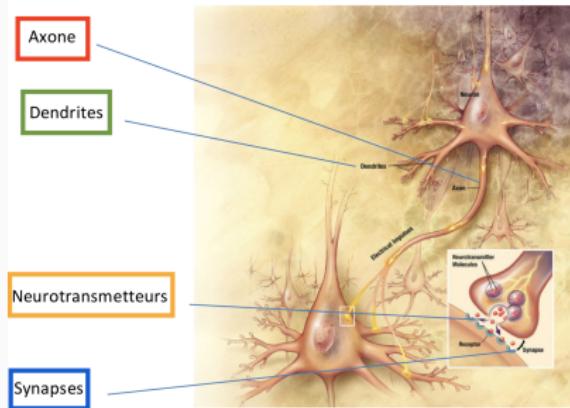
More information in [this video](#).

First, why this name?

A neuron receives multiple signals from other neurons at its **dendrites**. These signals are **neurotransmitters**, which are released at the other neurons' **synapses**. When the total input exceeds a certain **threshold**, the neuron "fires," sending an electrical impulse along its **axon**, which in turn allows it to release neurotransmitters through its own **synapses**.



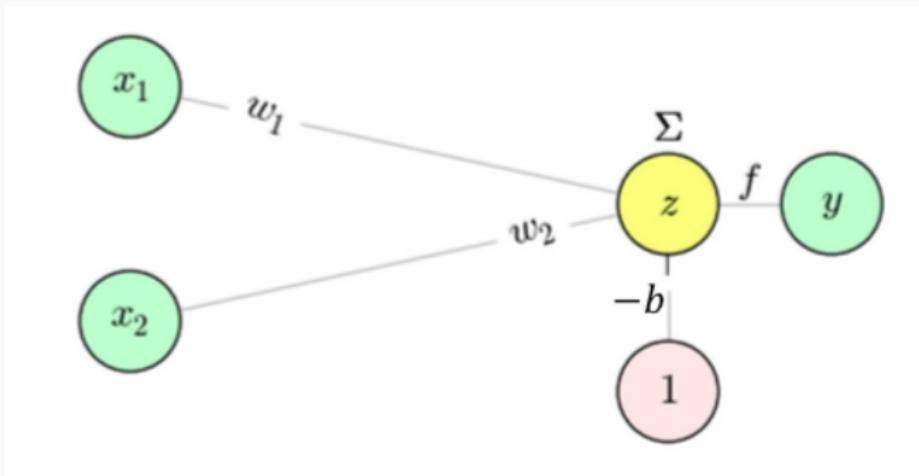
First, why this name?



To summarize:

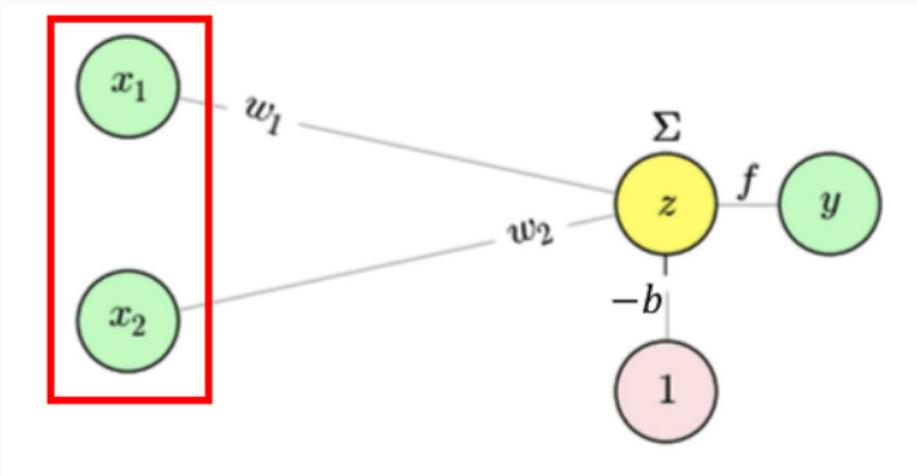
- The outputs of one neuron become the inputs of another.
- A neuron fires when it receives input exceeding a threshold.
- The strength of the signal sent to the next neuron is governed by synapses, which activate once their input crosses a threshold.

Perceptron – Overview



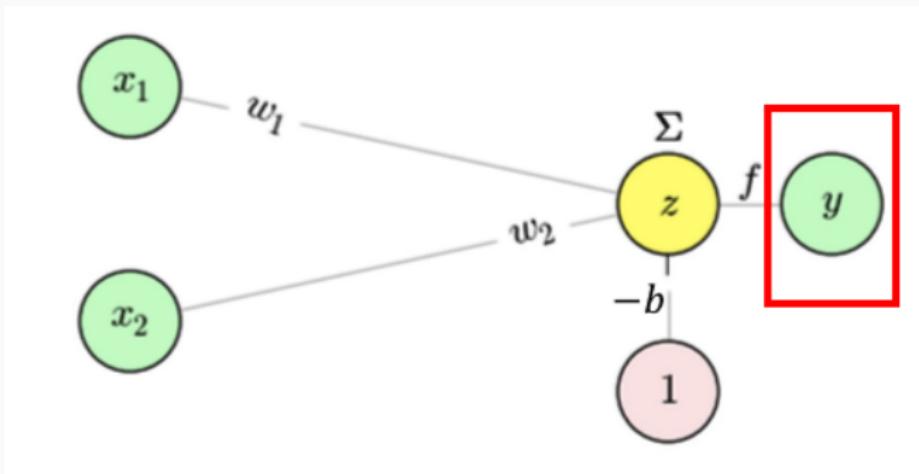
$$y = f(x_1 w_1 + x_2 w_2 - b)$$

Perceptron – Overview



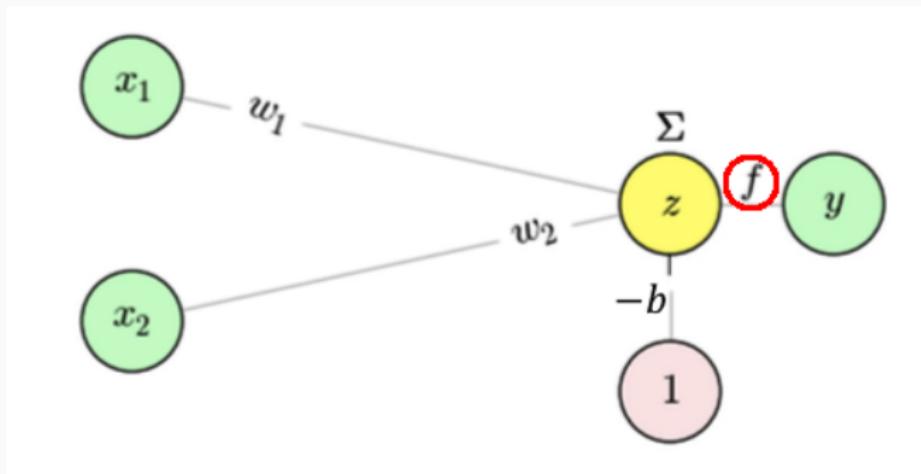
$$y = f(x_1 w_1 + x_2 w_2 - b)$$

Perceptron – Overview



$$y = f(x_1 w_1 + x_2 w_2 - b)$$

Perceptron – Overview



$$y = f(x_1 w_1 + x_2 w_2 - b)$$

In theory, f is the Heaviside step function:

$$f(z) = \mathbb{1}_{\mathbb{R}_+}(z) = \begin{cases} 1 & \text{if } z \geq 0, \\ 0 & \text{if } z < 0. \end{cases}$$

Outline : Applications

Significacion de las termas

History

Principle

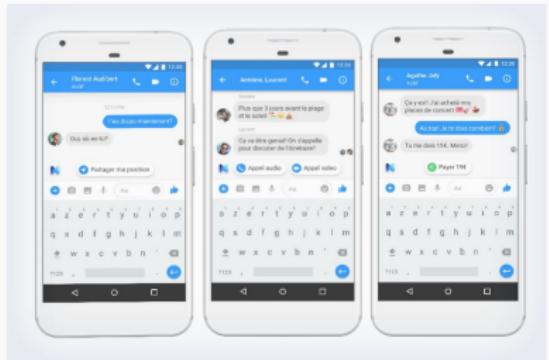
Applications

Ressources

The course

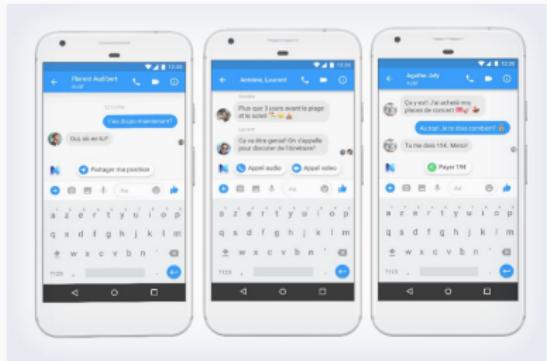
Applications (I/II)

- Event detection in text



Applications (I/II)

- Event detection in text

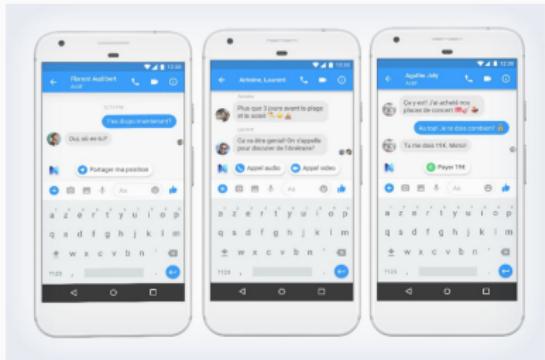


- Automatic processing of user opinions



Applications (I/II)

- Event detection in text



- Automatic processing of user opinions



Les connaissez-vous ?



Marina Dunion
Digital Marketing @Air France
& Co-Founder @FlexiFly
● Teddy Viraye-Chevalier et 3 autres relations



Salvatore Anzalone
Post-Doc at ISIR, University Pierre et Marie Curie, Paris
● Thomas Janssone et 2 autres relations

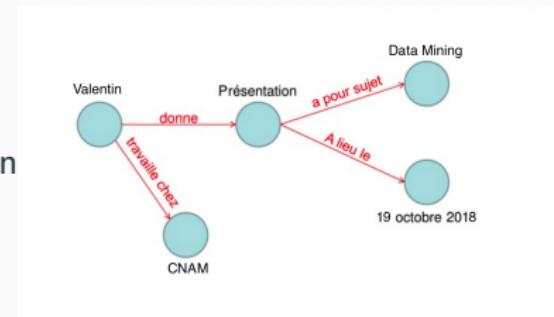


Halla Olafsdottir
Medical Solutions Project Manager | Chef de Projet
● Télécom ParisTech

- Recommending items to a user

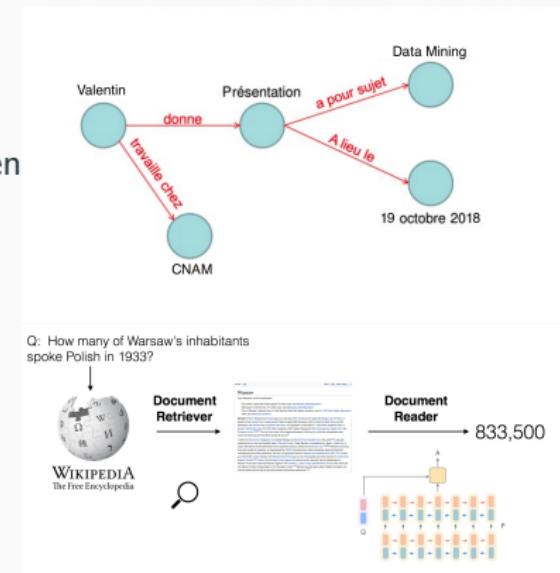
Applications (II/II)

- Detection of relationships between entities in a text



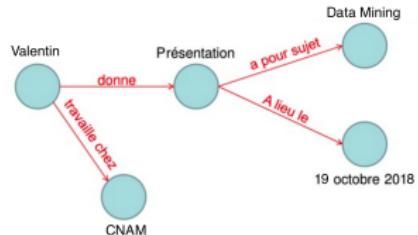
Applications (II/II)

- Detection of relationships between entities in a text
- Question answering

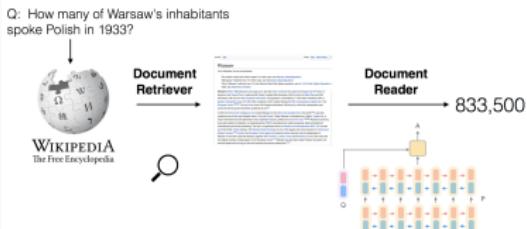


Applications (II/II)

- Detection of relationships between entities in a text



- Question answering

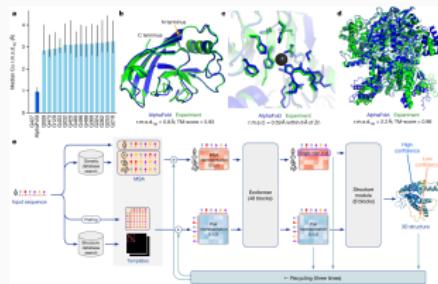


- IE module for a conversational agent



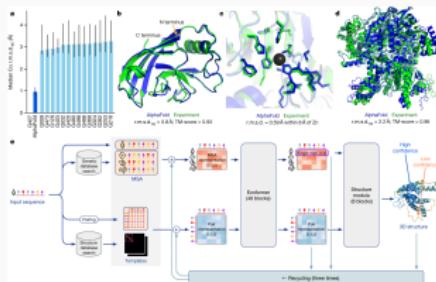
Meaning of the Job: Why Doing It?

- Scientific advancement



Meaning of the Job: Why Doing It?

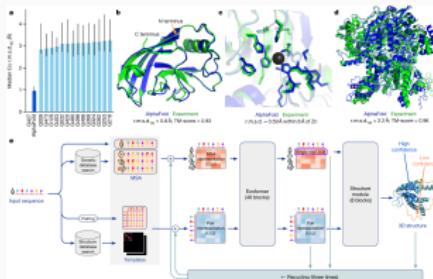
- Scientific advancement



- Prevention and management of natural disasters

Meaning of the Job: Why Doing It?

- Scientific advancement



- Prevention and management of natural disasters

- Impact on public health



Open Chronic

Améliorer la prise en charge des malades chroniques

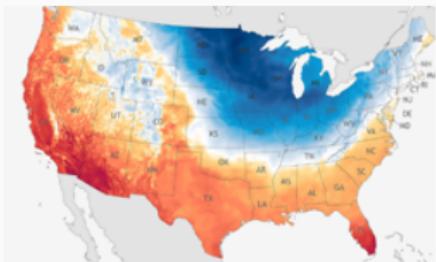
Response = 0

Ministère de la santé, Direction de la recherche, des études, de
et des statistiques



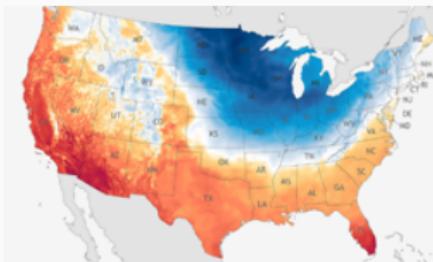
Meaning of the Job: Why Doing It?

- Environmental sustainability

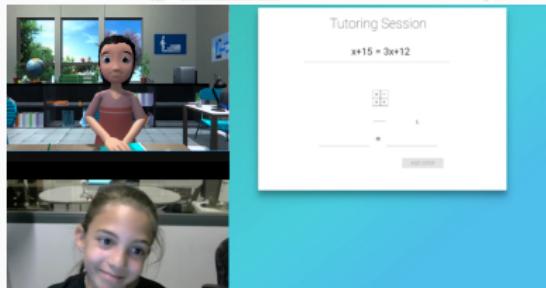


Meaning of the Job: Why Doing It?

- Environmental sustainability

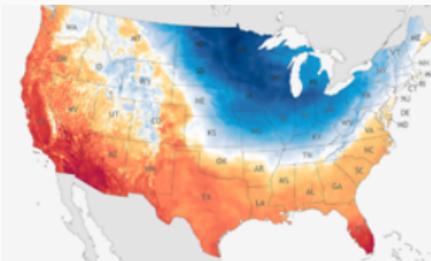


- Boost to education and research

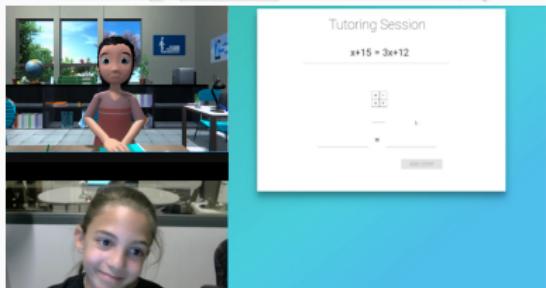


Meaning of the Job: Why Doing It?

- Environmental sustainability



- Boost to education and research



- Participatory democracy



Outline : Ressources

Significacion de las termas

History

Principle

Applications

Ressources

The course

Material

- Computer
- Jupyter Notebook and Anaconda:
<https://www.anaconda.com/download/>
- The notebooks and cheatsheets available online:

Python For Data Science Cheat Sheet

NumPy Basics

Learn Python for Data Science interactively at www.DataCamp.com

NumPy

The NumPy library is the core library for scientific computing in Python. It provides a high-performance multidimensional array object, and tools for working with these arrays.

Use the following import convention:

```
>>> import numpy as np
```

NumPy Arrays



Creating Arrays

More... [View on GitHub](#)

Python For Data Science Cheat Sheet

Pandas Basics

Learn Python for Data Science interactively at www.DataCamp.com

Pandas

The Pandas library is built on NumPy and provides easy-to-use data structures and data analysis tools for the Python programming language.

pandas

Use the following import convention:

```
>>> import pandas as pd
```

Pandas Data Structures

Series

A one-dimensional labeled array capable of holding any data type



Python For Data Science Cheat Sheet

Matplotlib

Learn Python interactively at www.DataCamp.com

Matplotlib

Matplotlib is a Python 2D plotting library which produces publication-quality figures in a variety of hardcopy formats and interactive environments across platforms.

1 Prepare The Data

Also see Lists & NumPy

```
>>> # Import numpy as np
>>> import numpy as np
>>> x = np.linspace(0, 10, 10)
>>> y = np.sin(x)
>>> x
>>> y
```

2D Data & Images

```
>>> data = 2 * np.random.random((10, 10))
>>> data2 = 3 * np.random.random((10, 10))
>>> data3 = -4 * np.random.random((10, 10))
>>> U = -1 + X**2 + Y
>>> V = 1 + Y - 4*X
```

Python For Data Science Cheat Sheet

Jupyter Notebook

Learn More Python for Data Science interactively at www.DataCamp.com

Saving/Loading Notebooks

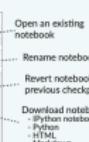
Create new notebook

Make a copy of the current notebook

Save current notebook and record checkpoint

Preview of the printed notebook

Close notebook & stop running any scripts



Writing Code And Text

Python For Data Science Cheat Sheet

Scikit-Learn

Learn Python for data science interactively at www.DataCamp.com

Scikit-learn

Scikit-learn is an open source Python library that implements a range of machine learning, preprocessing, cross-validation and visualization algorithms using a unified interface.

A Basic Example

```
>>> from sklearn.datasets import load_iris
>>> from sklearn.model_selection import train_test_split
>>> from sklearn.metrics import accuracy_score
>>> from sklearn.neighbors import KNeighborsClassifier
>>> X, y = iris.data[:, :-1], iris.target
>>> X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=42)
>>> knn = KNeighborsClassifier(n_neighbors=1)
>>> knn.fit(X_train, y_train)
>>> y_pred = knn.predict(X_test)
>>> accuracy_score(y_test, y_pred)
```

Python For Data Science Cheat Sheet

Keras

Learn Python for data science interactively at www.DataCamp.com

Keras

Keras is a powerful and easy-to-use deep learning library for Theano and TensorFlow that provides a high-level neural networks API to develop and evaluate deep learning models.

A Basic Example

```
>>> import numpy as np
>>> from keras.models import Sequential
>>> from keras.layers import Dense
>>> data = np.random.random((1000, 10))
>>> labels = np.random.randint(0, 9, size=(1000, 1))
>>> model = Sequential()
>>> model.add(Dense(32, activation='relu',
>>>                 input_dim=10))
>>> model.compile(optimizer='rmsprop',
>>>                 loss='categorical_crossentropy',
>>>                 metrics=['accuracy'])
>>> model.fit(data, labels, epochs=10, batch_size=32)
```

Creat

Sup

Lines

>>> f

>>> l

Supp

>>> t

>>> N

>>> E

>>> g

KNN

>>> E

>>> L

Uns

>>> P

>>> R

>>> F

>>> k

Mod

Sequ

>>> En

>>> ms

>>> mo

>>> Mult

>>> Briny G

>>> frc

>>> msc

Multi-Cl

>>> msc

>>> msu

>>> msu

>>> msu

>>> Regress

>>> msu

>>> msu

>>> Conv

Ressources

- El Calendario y las slides en el github
- Books:
 - Deep learning, Goodfellow, Bengio, Courville (
<https://www.deeplearningbook.org/>)
 - Neural networks and Deep learning, Nielsen (
<http://neuralnetworksanddeeplearning.com/>)
 - Dive into Deep Learning. Zhang, Lipton, Li, Smola (
<https://d2l.ai/>)
- Otros
 - CS231n - Stanford: Deep Learning for Computer Vision (
<http://cs231n.stanford.edu/>)
 - CS224N: Natural Language Processing with Deep Learning (
<https://web.stanford.edu/class/cs224n/>)
 - Deep learning – New York University (
<https://atcold.github.io/pytorch-Deep-Learning/>)
 - CS224W: Machine Learning with Graphs (
<http://web.stanford.edu/class/cs224w/>)
- Canal Discord: <https://discord.gg/hUJqa4AHCd>

A high-level Python library for Deep Learning, running on top of TensorFlow and other backends.



- Vision: [Keras tutorial for fine-tuning a pre-trained VGG16](#), which can be used with: [pre-trained CNNs available in Keras](#)
- Text and audio:
[Tutorial RNN-LSTM Seq2seq for machine translation](#)
- Text: [Use of pre-trained word embeddings](#)

A Python library for Deep Learning, a competitor to TensorFlow.



- Vision: [Tutorial for fine-tuning a pre-trained ResNet18](#)
- Audio: [Speech Recognition with Wav2Vec2](#)
- Text: [RNN-GRU Seq2seq tutorial for machine translation](#)

HuggingFace Transformers

A Python library for **Transformer** models (a class of deep neural networks).



Hugging Face

- Many pre-trained models for images, audio, multimodal data, text,
...
- From classical classifiers to generative models and embeddings, up
to large models (e.g. Llama3-70B).
- Related libraries: Diffusers, Datasets, Accelerate, PEFT,
bitsandbytes, TRL, ...

Outline : The course

Significacion de las termas

History

Principle

Applications

Ressources

The course

Evaluations

This class will be evaluated with 4 or 5 tasks to execute at home.
Careful: ponderations of the tasks vary wrt its difficulty!

Questions?

References i