

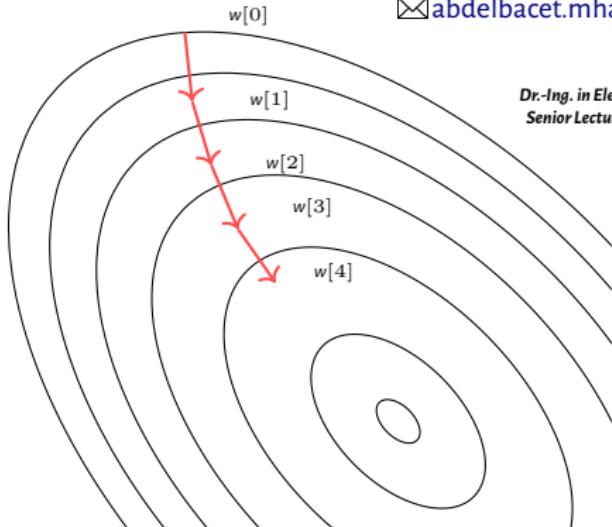
# An Introduction To Machine Learning Sorcery

(DEMYSTIFICATION PROCESS)<sup>1</sup>

Abdelbacet Mhamdi

✉ abdelbacet.mhamdi@bizerte.r-iset.tn

Dr.-Ing. in Electrical Engineering  
Senior Lecturer at ISETB Bizerte



“Computers are able to see, hear and learn.  
Welcome to the future.”

**Dave Waters**

“This is nothing. In a few years, that bot will move  
so fast you'll need a strobe light to see it.  
Sweet dreams...”

**Elon Musk**

“Machine intelligence is the last invention  
that humanity will ever need to make.”

**Nick Bostrom**

<sup>1</sup>Available @ <https://github.com/a-mhamdi/isetbz/>

## Disclaimer

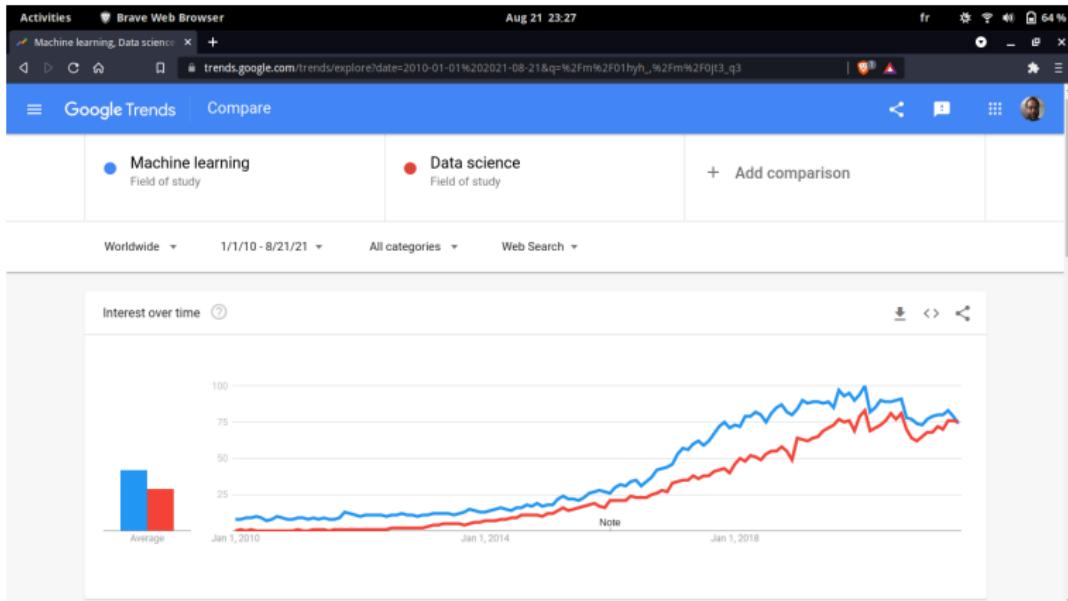
This document features some materials gathered from multiple online sources. Please note no copyright infringement is intended, and I do not own nor claim to own any of the original materials. They are used for educational purposes only. I have included links solely as a convenience to the reader. Some links within these slides may lead to other websites, including those operated and maintained by third parties. The presence of such a link does not imply a responsibility for the linked site or an endorsement of the linked site, its operator, or its contents.

- 1 An overview
- 2 Supervised Learning
- 3 Unsupervised Learning
- 4 Deep Learning
- 5 ML Landscape through Quizzes

**Next...**

- 1 An overview
- 2 Supervised Learning
- 3 Unsupervised Learning
- 4 Deep Learning
- 5 ML Landscape through Quizzes

# Trends



"Numbers represent search interest relative to the highest point on the chart for the given region and time."

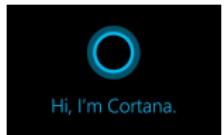
- A value of 100 is the peak popularity for the term;
- A value of 50 means that the term is half as popular;
- A score of 0 means there was not enough data for this term."

# Global Data Traffic



Update on the internet in real time is available [here](#).

# Top Uses



## Literature Review (1/3)

[Mit97]

"The field of machine learning is concerned with the question of how to construct computer programs that automatically improve with experience."

Mitchell, T. (1997) *Machine Learning*. McGraw-Hill International Editions. McGraw-Hill.

## Literature Review (2/3)

[Woj12]

“Machine learning (ML) is a scientific discipline that concerns developing learning capabilities in computer systems. Machine learning is one of central areas of Artificial Intelligence (AI). It is an interdisciplinary area that combines results from statistics, logic, robotics, computer science, computational intelligence, pattern recognition, data mining, cognitive science, and more.”

Wojtusiak, J. (2012) [Machine learning](#). In *Encyclopedia of the Sciences of Learning*, pages 2082–2083. Springer US.

## Literature Review (3/3)

[ENM15]

“Machine learning is an evolving branch of computational algorithms that are designed to emulate human intelligence by learning from the surrounding environment. They are considered the working horse in the new era of the so-called big data. Techniques based on machine learning have been applied successfully in diverse fields ranging from pattern recognition, computer vision, spacecraft engineering, finance, entertainment, and computational biology to biomedical and medical applications. [...] The ability of machine learning algorithms to learn from current context and generalize into unseen tasks would allow improvements in both the safety and efficacy of radiotherapy practice leading to better outcomes.”

El Naqa, I. and Murphy, M. J. (2015) *What Is Machine Learning?*, pages 3–11. Springer International Publishing.

# Debrief

Arthur Samuel (1959)

Machine Learning: Field of study that gives computers the ability to learn without being explicitly programmed.

Tom Mitchell (1998)

Well-posed Learning Problem: A computer is said to learn from experience  $\mathcal{E}$  with respect to some task  $\mathcal{T}$  and some performance measure  $\mathcal{P}$ , if its performance on  $\mathcal{T}$ , as measured by  $\mathcal{P}$ , improves with experience  $\mathcal{E}$  [Mit97].

# Debrief

Arthur Samuel (1959)

Machine Learning: Field of study that gives computers the ability to learn without being explicitly programmed.

Tom Mitchell (1998)

Well-posed Learning Problem: A computer is said to learn from experience  $\mathcal{E}$  with respect to some task  $\mathcal{T}$  and some performance measure  $\mathcal{P}$ , if its performance on  $\mathcal{T}$ , as measured by  $\mathcal{P}$ , improves with experience  $\mathcal{E}$  [Mit97].

## Task #1

Suppose your email program watches which emails you do or do not mark as spam, and based on that learns how to better filter spam. What is the task  $\mathcal{T}$  in this setting?

- ① Classifying emails as spam or not spam;
- ② Watching you label emails as spam or not spam;
- ③ The number (or fraction) of emails correctly classified as spam/not spam;
- ④ None of the above-this not a machine learning problem.

# Debrief

Arthur Samuel (1959)

Machine Learning: Field of study that gives computers the ability to learn without being explicitly programmed.

Tom Mitchell (1998)

Well-posed Learning Problem: A computer is said to learn from experience  $\mathcal{E}$  with respect to some task  $\mathcal{T}$  and some performance measure  $\mathcal{P}$ , if its performance on  $\mathcal{T}$ , as measured by  $\mathcal{P}$ , improves with experience  $\mathcal{E}$  [Mit97].

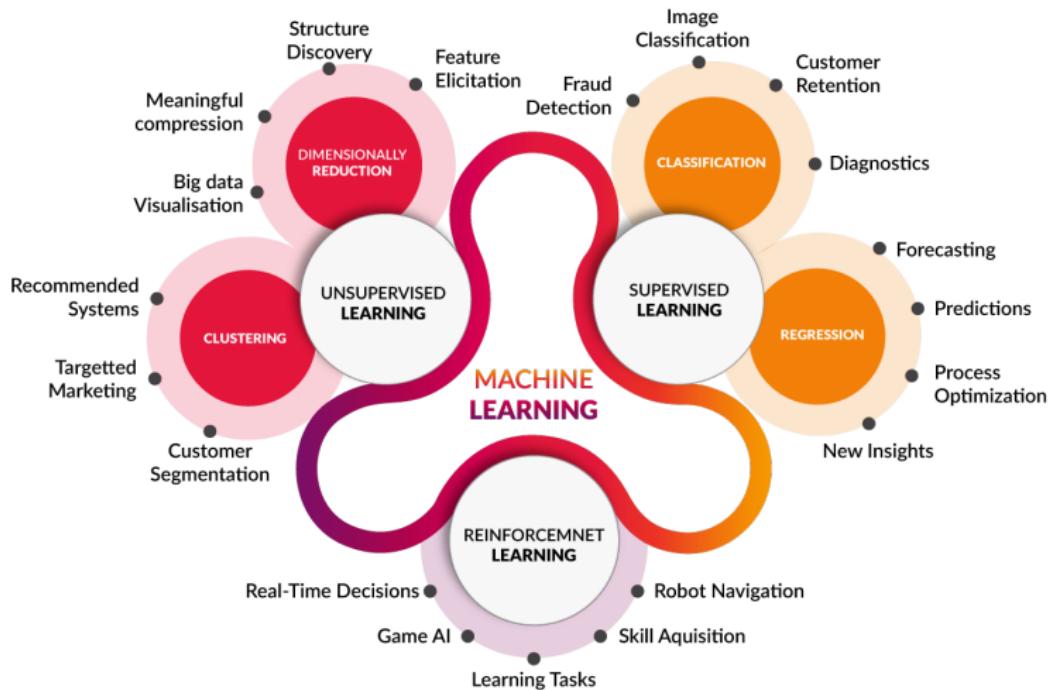
## Task #1

Suppose your email program watches which emails you do or do not mark as spam, and based on that learns how to better filter spam. What is the task  $\mathcal{T}$  in this setting?

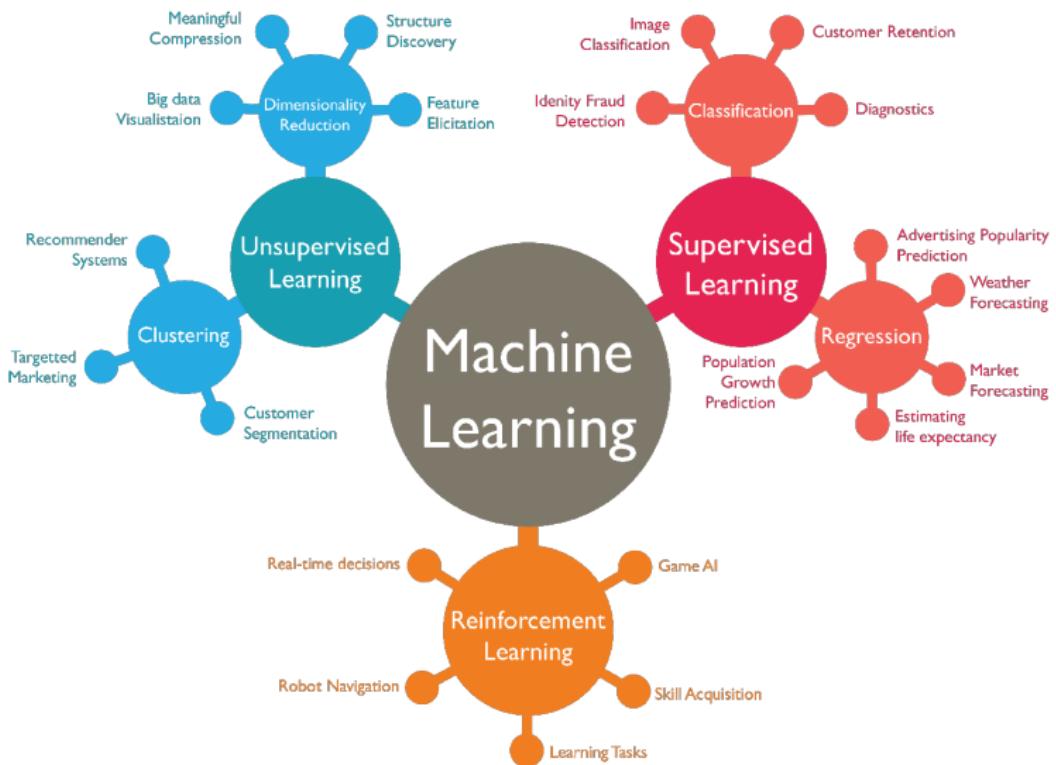
- ① Classifying emails as spam or not spam;
- ② Watching you label emails as spam or not spam;
- ③ The number (or fraction) of emails correctly classified as spam/not spam;
- ④ None of the above-this not a machine learning problem.

# Overall Methodology

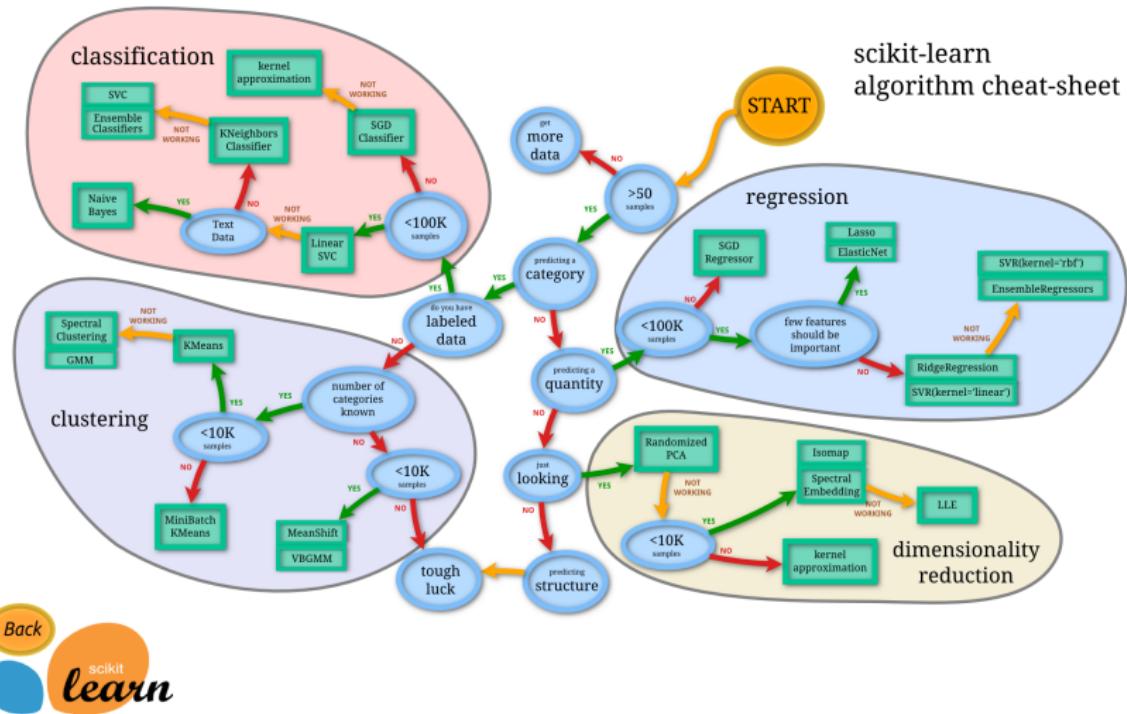
- ① Define the problem;
- ② Gather dataset;
- ③ Choose measure of success;
- ④ Decide evaluation protocol;
- ⑤ Prepare the data;
- ⑥ Develop a model;
- ⑦ Iterate models.



<https://www.cognub.com/index.php/cognitive-platform/>



<https://vitalflux.com/great-mind-maps-for-learning-machine-learning/>



[https://scikit-learn.org/stable/tutorial/machine\\_learning\\_map/index.html](https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html)

# Regression | Classification | Clustering

<https://github.com/MathWorks-Teaching-Resources/Machine-Learning-for-Regression>



# Programming Languages

A screenshot of a terminal window titled "python3". The window shows the Python interpreter running on a Linux system. The output of the command "print('Hello, World!')" is displayed, showing "Hello, World!" followed by a prompt ">>> |".

```
+ ls python3
Python 3.10.6 (main, Aug 10 2022, 11:40:04) [GCC 11.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> print('Hello, World!')
Hello, World!
>>> |
```

A screenshot of a terminal window titled "-/julia/julia". The window shows the Julia interpreter running on a Linux system. The output of the command "println("Hello, World!")" is displayed, showing "Hello, World!" followed by a prompt "julia>".

```
+ ls julia
Documentation: https://docs.julialang.org
Type "?" for help, "]?" for Pkg help.
Version 1.6.3 (2021-09-23)

julia> println("Hello, World!")
Hello, World!
julia>
```

# Development Environments



▲ \$ docker-compose up  
▼ \$ docker-compose down



# Required Packages

Valid only for...



- A full list is available @ <https://pypi.org/>

Numpy



Matplotlib



Pandas



Scikit – learn



Keras



```
$ pip install virtualenv
$ virtualenv -version
$ virtualenv «virtualenv_name»
$ source «virtualenv_name»/bin/activate # ACTIVATE
$ deactivate # DEACTIVATE
```

```
> pip install virtualenv
> virtualenv -version
> virtualenv «virtualenv_name»
> «virtualenv_name»\Scripts\activate :: ACTIVATE
> deactivate :: DEACTIVATE
```





# Source Control Management (SCM)

The screenshot shows a GitHub repository page for 'cosnip/Python/ml' at the 'main' branch. The repository has 2 pull requests, 1 issue, and 1 star. The commit history for the 'main' branch shows the following updates:

File	Commit Message	Time Ago
... (ellipsis)		
.vscode	update codes	7 months ago
exported-mdl	update codes	7 months ago
to-test	update codes	7 months ago
clf-ann.ipynb	update codes	7 months ago
clf-knn.ipynb	update codes	7 months ago
clf-svm.ipynb	update codes	7 months ago
cnn-build.py	di and gui-md files	6 months ago
gui-md.py	di and gui-md files	6 months ago
linear-regression.ipynb	update codes	7 months ago
mnist-cnn.ipynb	update codes	7 months ago
tuto-md.ipynb	update codes	12 months ago

<https://github.com/a-mhamdi/cosnip/tree/main/Python/ml>





# Continuous Integration (CI)

The screenshot shows a Docker Hub repository page for the user `abmhamdi` named `pyml`. The page includes a blue cube icon, the repository name, the owner's name, the last updated time, a description, and a 'Manage Repository' button. Below the main header, there are two tabs: 'Overview' (which is selected) and 'Tags'. The 'Overview' section contains a large heading 'PYML', a brief description of the repository's purpose, and a 'Docker Pull Command' section with a terminal command and a copy icon. A note at the bottom states that the latest image is built upon every push to this repo.

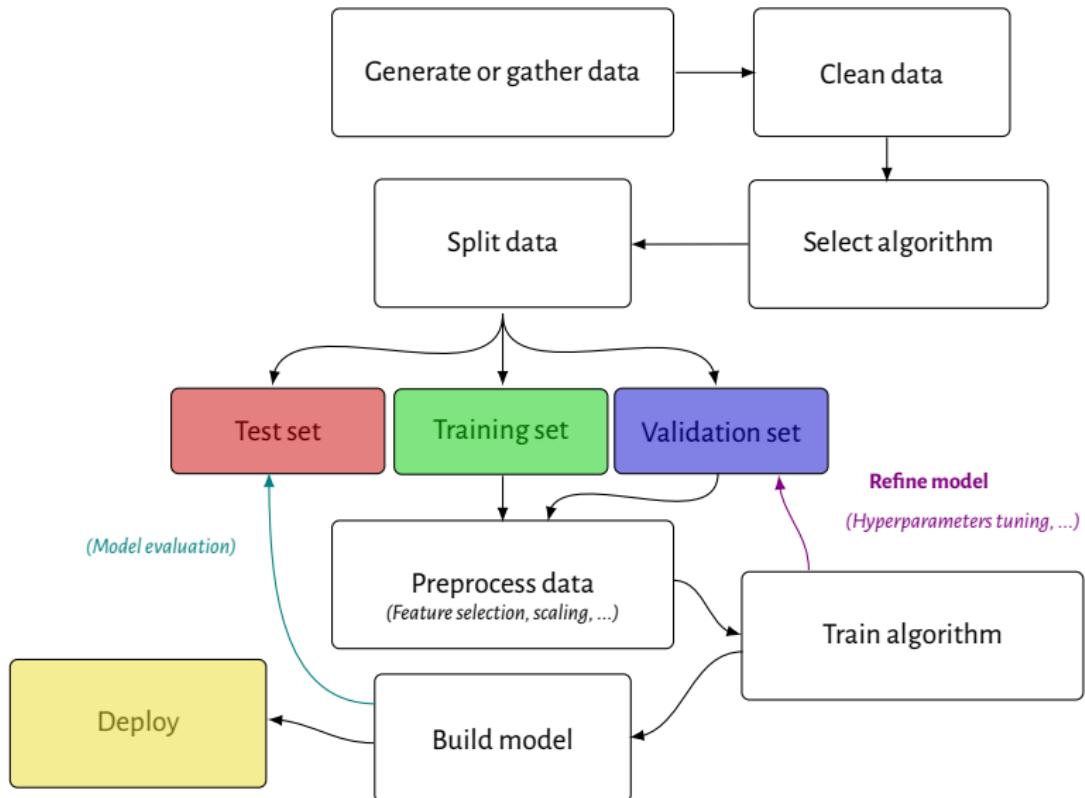
<https://hub.docker.com/repository/docker/abmhamdi/pyml>

Next...



- 1 An overview
- 2 **Supervised Learning**
- 3 Unsupervised Learning
- 4 Deep Learning
- 5 ML Landscape through Quizzes

# Workflow in Machine Learning



# Feature Scaling

## Normalisation

$$X = \frac{X - \min(X)}{X.\max() - X.\min()}$$

- ▲ No assumption on data distribution

## Standardisation

$$X = \frac{X - X.\text{mean}()}{X.\text{std}()}$$

- ▲ More recommended when following normal distribution

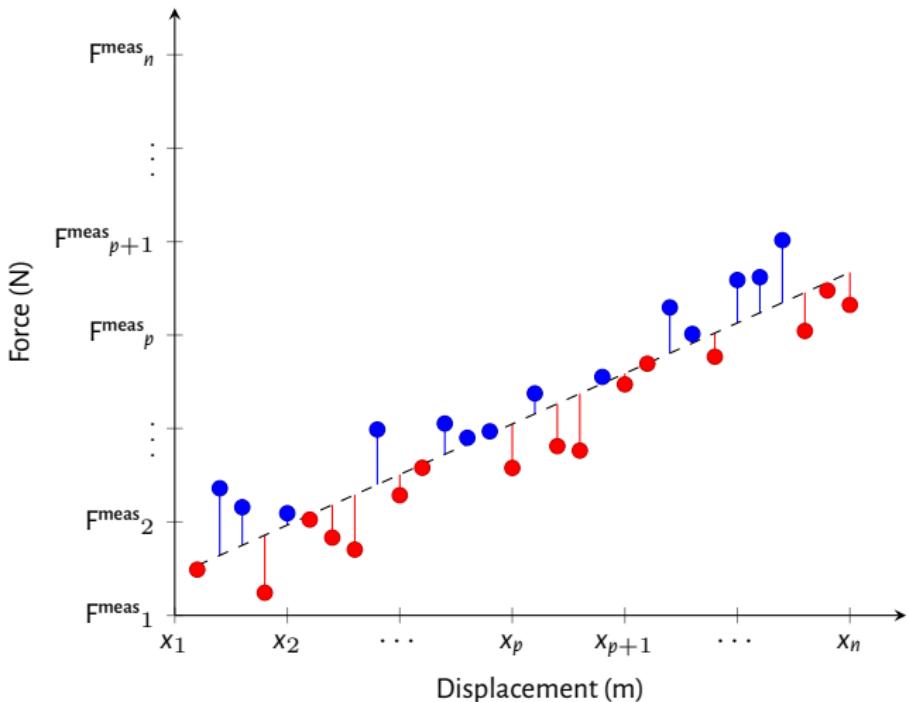
# Data Preprocessing Template

```
[]: from sklearn.preprocessing import StandardScaler
```

```
[]: sc = StandardScaler()  
X_train = sc.fit_transform(X_train)  
X_test = sc.transform(X_test)
```



The notebook is available at <https://github.com/a-mhamdi/isetbz/>  
→ Machine Learning → Codes → data-preprocessing-template.ipynb



Consider the example of a spring. Our main goal is to determine the stiffness  $k$  of this spring, given some experimental data. The mathematical model (*Hooke's law*):

$$F = kx \quad (1)$$

Restoring force is proportional to displacement.

**Table:** Measurements of couple  $(x_i, F^{\text{meas}}_i)$

$x_i$	$x_1$	$\dots$	$x_p$	$\dots$	$x_n$
$F^{\text{meas}}_i$	$F^{\text{meas}}_1$	$\dots$	$F^{\text{meas}}_p$	$\dots$	$F^{\text{meas}}_n$

$$\begin{aligned} F^{\text{meas}}_i &= F_i + \varepsilon_i \\ &= kx_i + \varepsilon_i, \end{aligned} \quad (2)$$

where  $F_i$  denotes the unknown real value of the force applied to the spring. In order to estimate the stiffness value  $k$ , we can consider the quadratic criterion:

$$\begin{aligned} \mathcal{J} &= \sum_{i=1}^n \varepsilon_i^2 \\ &= \sum_{i=1}^n (F^{\text{meas}}_i - kx_i)^2 \end{aligned}$$

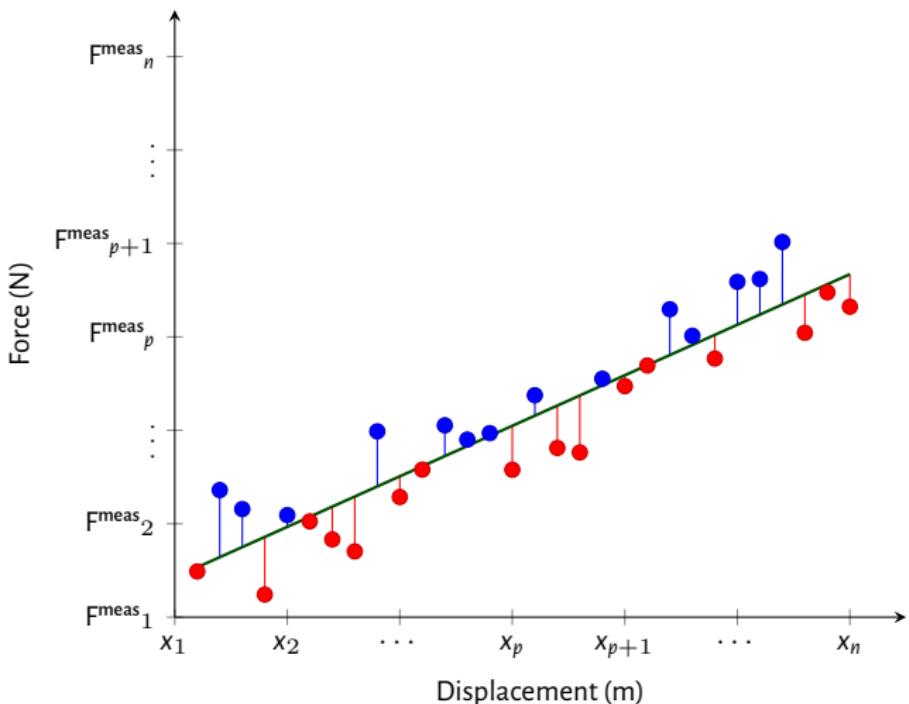
## Linear Regression

$$\frac{\partial \mathcal{J}}{\partial k} = 0 \quad (3)$$

$$2 \sum_{i=1}^n (\mathbf{F}^{\text{meas}}_i - kx_i) \sum_{i=1}^n \frac{\partial (\mathbf{F}^{\text{meas}}_i - kx_i)}{\partial k} = 0$$

$$\sum_{i=1}^n (\mathbf{F}^{\text{meas}}_i - kx_i) \sum_{i=1}^n x_i = 0$$

$$\sum_{i=1}^n \mathbf{F}^{\text{meas}}_i x_i = k \sum_{i=1}^n x_i^2 \iff \hat{k} = \frac{\sum_{i=1}^n \mathbf{F}^{\text{meas}}_i x_i}{\sum_{i=1}^n x_i^2}$$



# Simple Linear Regression

CODE SNIPPET

## Training the Simple Linear Regression model on the Training set

```
[1]: from sklearn.linear_model import LinearRegression  
  
[2]: regressor = LinearRegression()  
      regressor.fit(X_train, y_train)  
  
[3]: LinearRegression()
```

## Predicting the Test set results

```
[4]: y_pred = regressor.predict(X_test)
```



The notebook is available at <https://github.com/a-mhamdi/isetbz/>  
→ Machine Learning → Codes → simple-linear-regression.ipynb



This example consists on determining the unknown couple  $(y_0, v_0)$  of a mobile solid. We assume that the trajectory is linear. The mathematical model that relates the position  $y$  to time  $t$  is given by this equation:

$$y = y_0 + v_0 t \quad (4)$$

**Table:** Measurements of position  $y$

$t_i$	$t_1$	$\dots$	$t_p$	$\dots$	$t_n$
$y^{\text{meas}}_i$	$y^{\text{meas}}_1$	$\dots$	$y^{\text{meas}}_p$	$\dots$	$y^{\text{meas}}_n$

$$\begin{aligned} y^{\text{meas}}_i &= y_i + \varepsilon_i \\ &= y_0 + v_0 t_i + \varepsilon_i, \end{aligned} \quad (5)$$

where  $y_i$  denotes the unknown real value of the position  $y$  at time point  $t_i$ .

In order to estimate the values taken by the couple  $[y_0, v_0]^T$ , we consider the quadratic criterion again, as follows:

$$\begin{aligned} \mathcal{J} &= \sum_{i=1}^n \varepsilon_i^2 \\ &= \varepsilon^T \times \varepsilon \end{aligned}$$

The vector  $\varepsilon$  is set by  $\varepsilon_i, \forall i \geq 1$ :

$$\varepsilon = [ \varepsilon_1 \quad \cdots \quad \varepsilon_n ]^T$$

$$\frac{\partial \mathcal{J}}{\partial \begin{bmatrix} y_0 \\ v_0 \end{bmatrix}} = 0 \quad (6)$$

# Multiple Linear Regression

CODE SNIPPET

## Training the multiple linear regression model on the training set

```
[]: from sklearn.linear_model import LinearRegression
```

```
[]: lr = LinearRegression()
lr.fit(X_train, y_train)
```

```
[]: LinearRegression()
```

## Making predictions using the X test set and comparison

```
[]: y_pred = lr.predict(X_test)
```



The notebook is available at <https://github.com/a-mhamdi/isetbz/>  
→ Machine Learning → Codes → multiple-linear-regression.ipynb



Consider the following multivariate equation:

$$y = \theta_1 x_{(1)} + \theta_2 x_{(2)} + \cdots + \theta_m x_{(m)} \quad (7)$$

For a particular single measurement, eq. (7) can be updated as

$$y_k = \theta_1 x_{(1, k)} + \theta_2 x_{(2, k)} + \cdots + \theta_m x_{(m, k)} + \varepsilon_k \quad (8)$$

We denote hereafter by  $\theta$  the vector  $\begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{bmatrix}$ . The function  $y_k$  becomes:

$$y_k = \underbrace{[x_{(1, k)}, x_{(2, k)}, \dots, x_{(m, k)}]}_{x_k^T} \theta + \varepsilon_k$$

We assume that we have  $n$  measurements for  $y$ . Then we can transform the previous equation into

$$y = H\theta + \varepsilon,$$

where  $y^T = [y_1, y_2, \dots, y_n]$ ,  $X = \begin{bmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_n^T \end{bmatrix}$ , and  $\varepsilon^T = [\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n]$ .

We can consider the mean squared error or quadratic criterion in order to compute the approximated value of  $\theta$ :

$$\begin{aligned}\mathcal{J} &= \sum_{k=1}^n \varepsilon_k^2 \\ &= \varepsilon^T \varepsilon\end{aligned}$$

The best well estimated value of  $\hat{\theta}$  corresponds to the absolute minimum of  $\mathcal{J}$ . This leads to calculate the gradient of  $\mathcal{J}$  with respect to  $\theta$ :

$$\frac{\partial \mathcal{J}}{\partial \theta} = \frac{\partial (\varepsilon^T \varepsilon)}{\partial \theta} \quad (9)$$

$$\frac{\partial (\varepsilon^T \varepsilon)}{\partial \theta} = 2 \left( \frac{\partial \varepsilon}{\partial \theta} \right)^T \varepsilon \quad (10)$$

Recall that  $\varepsilon = y - X\theta$ , the term  $\frac{\partial \varepsilon}{\partial \theta}$  hence becomes:

$$\frac{\partial \varepsilon}{\partial \theta} = -X \quad (11)$$

$$\begin{aligned}\frac{\partial J}{\partial \theta} &= 2(-X)^T(y - X\theta) \\ &= 0\end{aligned}$$

The regressor is given by

$$\hat{\theta} = (X^T X)^{-1} X^T y$$



$X^T X$  is not invertible (singular/degenerate)

#### ▼ Redundant Features

Some features are linearly dependant, i.e.,  $\exists$  some  $x_p \propto$  some  $x_l$  for instance  $x_p$  in feet and  $x_l$  in m.

#### ▼ Too many features

Fewer observations compared to the number of features, i.e.,  $m \geq n$ .

- ▲ Delete some features
- ▲ Add extra observations
- ▲ Use regularization

# Gradient Descent

$$\theta_i = \theta_i - \underbrace{\alpha}_{\text{LEARNING RATE}} \frac{\partial \mathcal{J}}{\partial \theta_i}$$

Recall that  $\mathcal{J} = 1/2n \sum_{k=1}^n (y_k - h_\theta(x_k))^2 \implies \frac{\partial \mathcal{J}}{\partial \theta_i} = -1/n \sum_{k=1}^n (y_k - h_\theta(x_k)) x_{(i, k)}$

$$\theta_i \leftarrow \theta_i + \alpha \frac{1}{n} \sum_{k=1}^n (y_k - h_\theta(x_k)) x_{(i, k)}$$

$$\theta_0 \leftarrow \theta_0 + \alpha \frac{1}{n} \sum_{k=1}^n (y_k - h_\theta(x_k)) x_{(0, k)}$$

$$\theta_1 \leftarrow \theta_1 + \alpha \frac{1}{n} \sum_{k=1}^n (y_k - h_\theta(x_k)) x_{(1, k)}$$

⋮

$$\theta_m \leftarrow \theta_m + \alpha \frac{1}{n} \sum_{k=1}^n (y_k - h_\theta(x_k)) x_{(m, k)}$$

# Polynomial Regression

CODE SNIPPET

```
[]: from sklearn.preprocessing import PolynomialFeatures
```

```
[]: poly_reg = PolynomialFeatures(degree=4)
X_poly = poly_reg.fit_transform(X)
print(X_poly[:5])
lr_2 = LinearRegression()
lr_2.fit(X_poly, y)
```



The notebook is available at <https://github.com/a-mhamdi/isetbz/>  
→ Machine Learning → Codes → polynomial-regression.ipynb

## Task #2

The yield  $y$  of a chemical process is a random variable whose value is considered to be a linear function of the temperature  $x$ . The following data of corresponding values of  $x$  and  $y$  is found:

Temperature in °C (x)	0	25	50	75	100
Yield in grams (y)	14	38	54	76	95

The linear regression model  $y = \theta_0 + \theta_1 x$  is used. Determine the values of  $\theta_0$ ,  $\theta_1$ .

- ① Using normal equation,
- ② Using gradient descent for 5 iterations.

$$y = \begin{bmatrix} 14 \\ 38 \\ 54 \\ 76 \\ 95 \end{bmatrix} \quad \text{and} \quad X = \begin{bmatrix} 1 & 0 \\ 1 & 25 \\ 1 & 50 \\ 1 & 75 \\ 1 & 100 \end{bmatrix} \implies X^T X = \begin{bmatrix} 5 & 250 \\ 250 & 18750 \end{bmatrix}$$

$$\hat{\theta} = \begin{bmatrix} \hat{\theta}_0 \\ \hat{\theta}_1 \end{bmatrix} = \begin{bmatrix} 15.4 \\ 0.8 \end{bmatrix}$$

```

import matplotlib.pyplot as plt
import numpy as np

X = np.array([[1,0], [1,25], [1,50], [1,75], [1,100]], dtype=np.float32)
y = np.array([[14], [38], [54], [76], [95]])

# NORMAL EQUATION
XtX = X.T.dot(X)
invXtX = np.linalg.inv(XtX)
t_ne = invXtX.dot(np.matmul(X.T, y))
print(t_ne)
X[:,1] = (X[:,1]-X[:,1].min()) / X[:,1].max()
y = (y-y.min()) / y.max()

# GRADIENT DESCENT
t_gd = np.array([[1], [1]])
alpha = .1
vect = np.zeros(shape=(2, 1001))
vect[:,0] = t_gd[[0,1],[0]]
lost = []
for k in range(1000):
    eps = y-np.matmul(X, t_gd)
    lost.append(1/(2*len(y))*eps.T.dot(eps)[[0],[0]][0])
    t_gd = t_gd+alpha*1/len(y)*np.matmul((eps).T, X).T
    vect[:,k+1] = t_gd[[0,1],[0]]

print(vect[:, -1])
plt.plot(vect[0, :], label=r'$\hat{\theta}_0$')
plt.plot(vect[1, :], label=r'$\hat{\theta}_1$')
plt.legend(); plt.grid(); plt.show()

plt.plot(lost); plt.grid(); plt.show()

```

**F1-Score, Accuracy, Recall** and **Precision** are calculated as follow:

$$\text{f1-score} = \frac{2}{\frac{1}{\text{Recall}} + \frac{1}{\text{Precision}}}$$

f1-score denotes the *Harmonic Mean of Recall & Precision*

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}$$

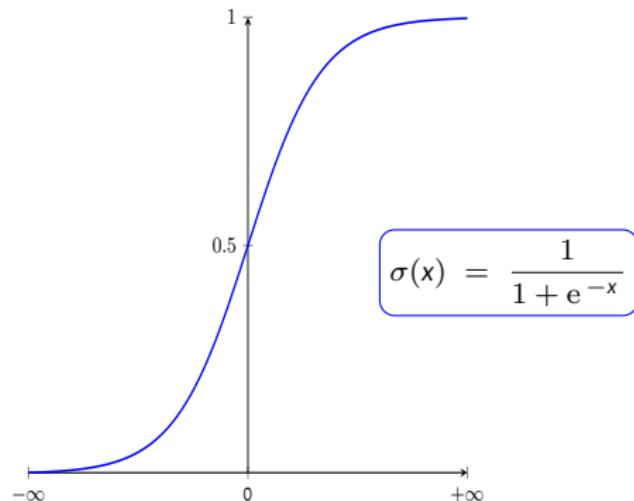
It denotes the ratio of how much we got right over all cases. Recall, on the other hand, designates the ratio of how much positives do we got right over all actual positive cases.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

Precision, at last, is how much positives we got right over all positive predictions. It is given by:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

## Logistic or S-shaped function $\sigma$



- ▲  $\sigma$  squashes range of distance from  $]-\infty, +\infty[$  to  $[0, 1]$
- ▲  $\sigma$  is differentiable and easy to compute:  $\dot{\sigma} = \sigma \times (1 - \sigma)$

# Decision boundary

$$y = \sigma(\theta_1 x_{(1)} + \theta_2 x_{(2)} + \cdots + \theta_m x_{(m)})$$

$$y = \frac{1}{1 + e^{-\theta^T x}}$$

Hypothesis:

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}} \quad h_\theta(x_k) = \frac{1}{1 + e^{-\theta^T x_k}}$$

Cost function:

$$\mathcal{J} = \begin{cases} -\ln(h_\theta(x)) & \text{if } y = 1 \\ -\ln(1 - h_\theta(x)) & \text{if } y = 0 \end{cases}$$

$$\boxed{\mathcal{J} = -y \ln(h_\theta(x)) - (1 - y) \ln(1 - h_\theta(x))}$$

# Gradient Descent

$$\theta_i = \theta_i - \underbrace{\alpha}_{\text{LEARNING RATE}} \frac{\partial \mathcal{J}}{\partial \theta_i}$$

Generalizing  $\mathcal{J}$  yields:  $\mathcal{J} = -\frac{1}{n} \sum_{k=1}^n (y_k \ln(h_\theta(x_k)) + (1-y_k) \ln(1-h_\theta(x_k)))$

$$\Rightarrow \frac{\partial \mathcal{J}}{\partial \theta_i} = -\frac{1}{n} \sum_{k=1}^n (y_k - h_\theta(x_k)) x_{(i, k)}$$

$$\theta_i \leftarrow \theta_i + \alpha \frac{1}{n} \sum_{k=1}^n (y_k - h_\theta(x_k)) x_{(i, k)}$$

$$\theta_0 \leftarrow \theta_0 + \alpha \frac{1}{n} \sum_{k=1}^n (y_k - h_\theta(x_k)) x_{(0, k)}$$

$$\theta_1 \leftarrow \theta_1 + \alpha \frac{1}{n} \sum_{k=1}^n (y_k - h_\theta(x_k)) x_{(1, k)}$$

⋮

$$\theta_m \leftarrow \theta_m + \alpha \frac{1}{n} \sum_{k=1}^n (y_k - h_\theta(x_k)) x_{(m, k)}$$

# Logistic Regression

CODE SNIPPET

## Training the logistic regressor

```
[]: from sklearn.linear_model import LogisticRegression
```

```
[]: classifier = LogisticRegression(random_state=0)
classifier.fit(X_train, y_train)
```

```
[]: LogisticRegression(random_state=0)
```

## Predicting a new result

```
[]: print(classifier.predict(sc.transform([[30,87000]])))
```



The notebook is available at <https://github.com/a-mhamdi/isetbz/>  
→ Machine Learning → Codes → logistic-regression.ipynb



# K Nearest Neighbors (1/4)

CODE SNIPPET

▶ Evelyn Fix and Joseph Hodges, 1951

▶ Thomas Cover, 1966

## Algorithm 1 Summary Construction

1: **procedure** How DOES KNN WORK? (Finding Nearest Neighbors)

**Input:** A query point;

**Output:** Assign a class label to that point.

2: Define how many neighbors will be checked to classify the specific query point;

3: Compute the distance  $d(x, y)$  of the query point to other data points;

4: Count the number of the data points in each category;

5: Assign the query point to the class with most frequent neighbors.

6: **end procedure**

Minkowski distance

$$d(x, y) = \left( \sum_{i=1}^n (y_i - x_i)^p \right)^{1/p}$$

Manhattan distance (p=1)

$$d(x, y) = \sum_{i=1}^n |y_i - x_i|$$

Euclidean distance (p=2)

$$d(x, y) = \sqrt{\sum_{i=1}^n (y_i - x_i)^2}$$

# K Nearest Neighbors (2/4)

CODE SNIPPET

## Task #3

Let be the following coordinate points:

$A(1; 6), B(2; 6), C(3; 1), D(4; 2), E(6; 0), F(7; 5), G(7; 3), H(10; 3)$

Using the Euclidean distance, what are the two closest neighbors of point  $P(5; 5)$ ?

# K Nearest Neighbors (3/4)

CODE SNIPPET

## Task #4<sup>a</sup>

<sup>a</sup>From Prof. Winston's book

We try to predict the color of a fruit according to its width ( $w$ ) and height ( $h$ ). The following training data is available:

$w$	2	5	2	6	1	4	2	6
$h$	6	6	5	5	2	2	1	1
Color	Red	Yellow	Orange	Purple	Red	Blue	Violet	Green

The goal here is to study the influence of neighbors on the color property of a fruit. Let  $\mathbf{U}$  be the new fruit of width  $w = 1$  and height  $h = 4$

- ① What is its color if we consider 1 neighbor?
- ② What is its color if we consider 3 neighbors?
- ③ Rather than majority voting, we would like to consider the vote of neighbors weighted by the distance. Each neighbor votes according to a weight  $w$  inversely proportional to the square of its distance:  $\mathcal{W} = \frac{1}{d^2}$ . We take 3 neighbors, what is the color of  $\mathbf{U}$ ? Compare your results to those in question 2.

# K Nearest Neighbors (4/4)

CODE SNIPPET

## Importing the classifier

```
[]: from sklearn.neighbors import KNeighborsClassifier
```

## Training the K-NN model on the training set

```
[]: classifier = KNeighborsClassifier(n_neighbors, metric, p)
classifier.fit(X_train, y_train)
```

## Predicting a new result

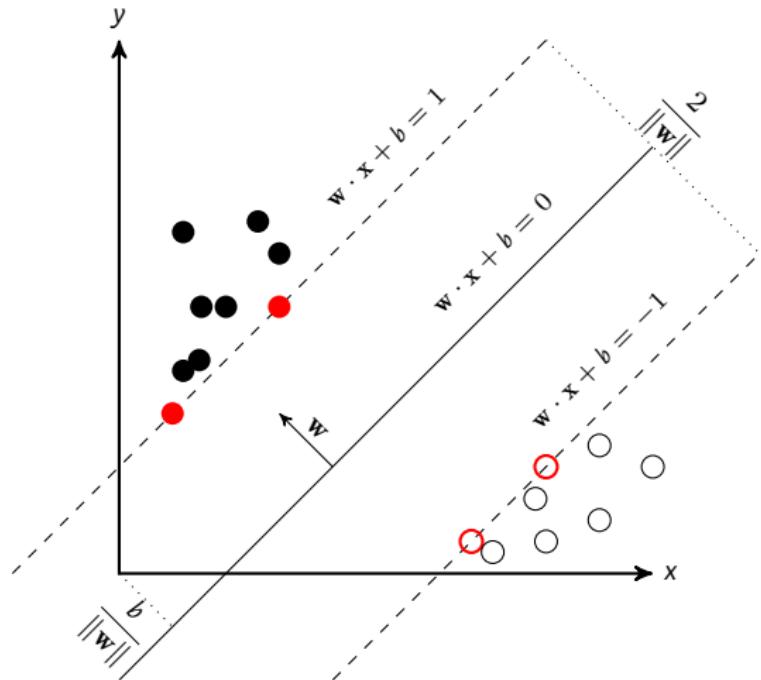
```
[]: print(classifier.predict(sc.transform([[X, y]])))
```



The notebook is available at <https://github.com/a-mhamdi/isetbz/>  
→ Machine Learning → Codes → k-nearest-neighbors.ipynb



# Support Vector Machine (SVM)



# Outroduction

Method	Pros	Cons
Logistic Regression	<ul style="list-style-type: none"><li>▲ Probabilistic</li><li>▲ Simple</li></ul>	<ul style="list-style-type: none"><li>▼ Almost linearly separable data</li></ul>
K-NN	<ul style="list-style-type: none"><li>▲ Fast</li><li>▲ Efficient</li></ul>	<ul style="list-style-type: none"><li>▼ Number of neighbors <math>k</math></li></ul>

Next...



- 1 An overview
- 2 Supervised Learning
- 3 Unsupervised Learning
- 4 Deep Learning
- 5 ML Landscape through Quizzes

## Task #5<sup>a</sup>

"From 'Machine Learning' course on 'Coursera'

Of the following examples, which would you address using an unsupervised learning algorithms? (*Check all that apply.*)

- ① Given email labeled as spam/not spam, learn a spam filter
- ② Given a set of news articles found on the web, group them into set of articles about the same story
- ③ Given a database of customer data, automatically discover market segments and group customers into different market segments
- ④ Given a dataset of patients diagnosed as either having diabetes or not, learn to classify new patients as having diabetes or not.

## Task #5<sup>a</sup>

"From 'Machine Learning' course on 'Coursera'

Of the following examples, which would you address using an unsupervised learning algorithms? (*Check all that apply.*)

- ① Given email labeled as spam/not spam, learn a spam filter
- ② Given a set of news articles found on the web, group them into set of articles about the same story
- ③ Given a database of customer data, automatically discover market segments and group customers into different market segments
- ④ Given a dataset of patients diagnosed as either having diabetes or not, learn to classify new patients as having diabetes or not.

# KMeans

CODE SNIPPET

**Import KMeans class**[]: 

```
from sklearn.cluster import KMeans
```

**Training the K-Means model on the dataset**[]: 

```
kmeans = KMeans(n_clusters, init, random_state)
y_kmeans = kmeans.fit_predict(X)
```



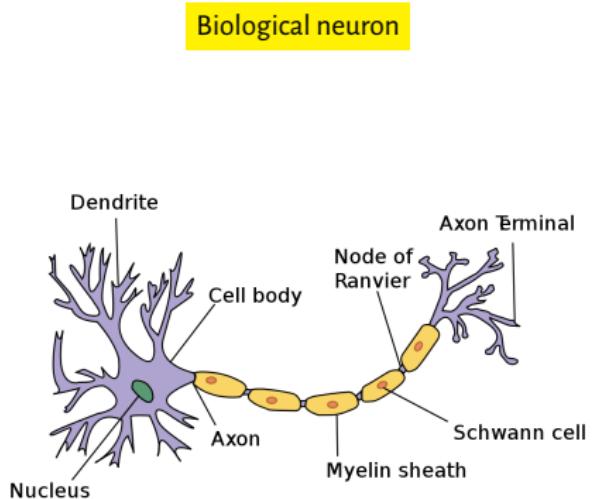
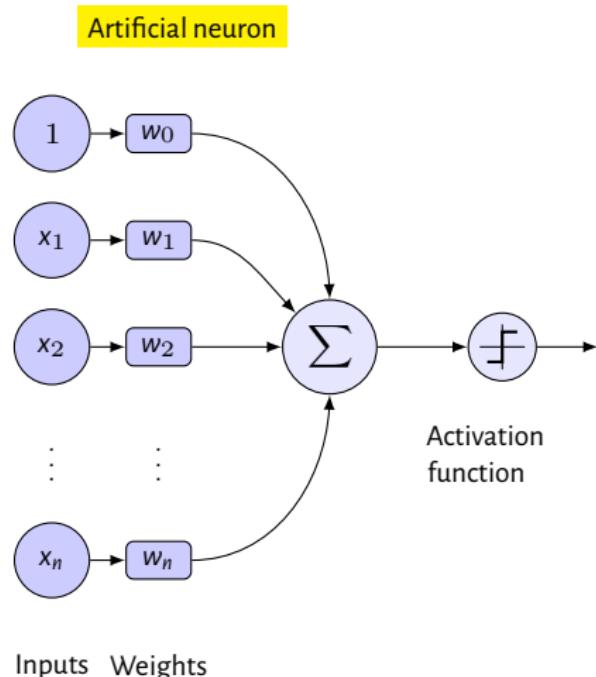
The notebook is available at <https://github.com/a-mhamdi/isetbz/>  
→ Machine Learning → Codes → k-means-clustering.ipynb

Next...



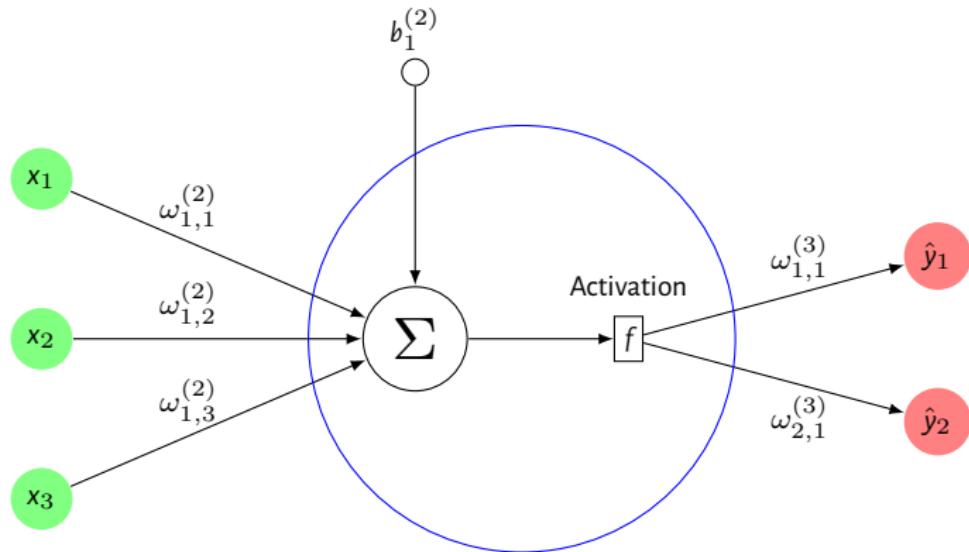
- 1 An overview
- 2 Supervised Learning
- 3 Unsupervised Learning
- 4 Deep Learning
- 5 ML Landscape through Quizzes

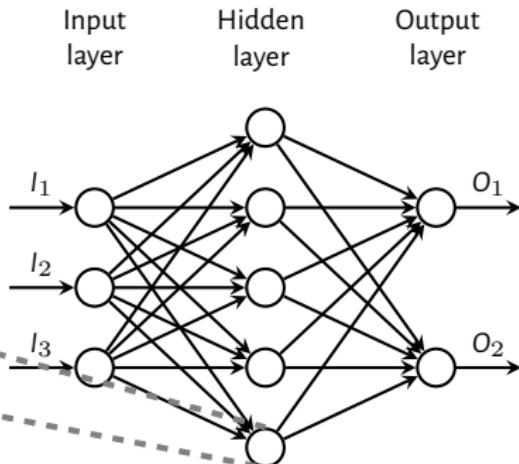
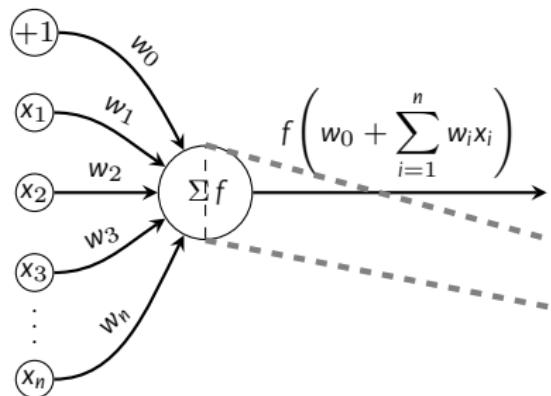
# Fundamental unit of a neural network (1/2)



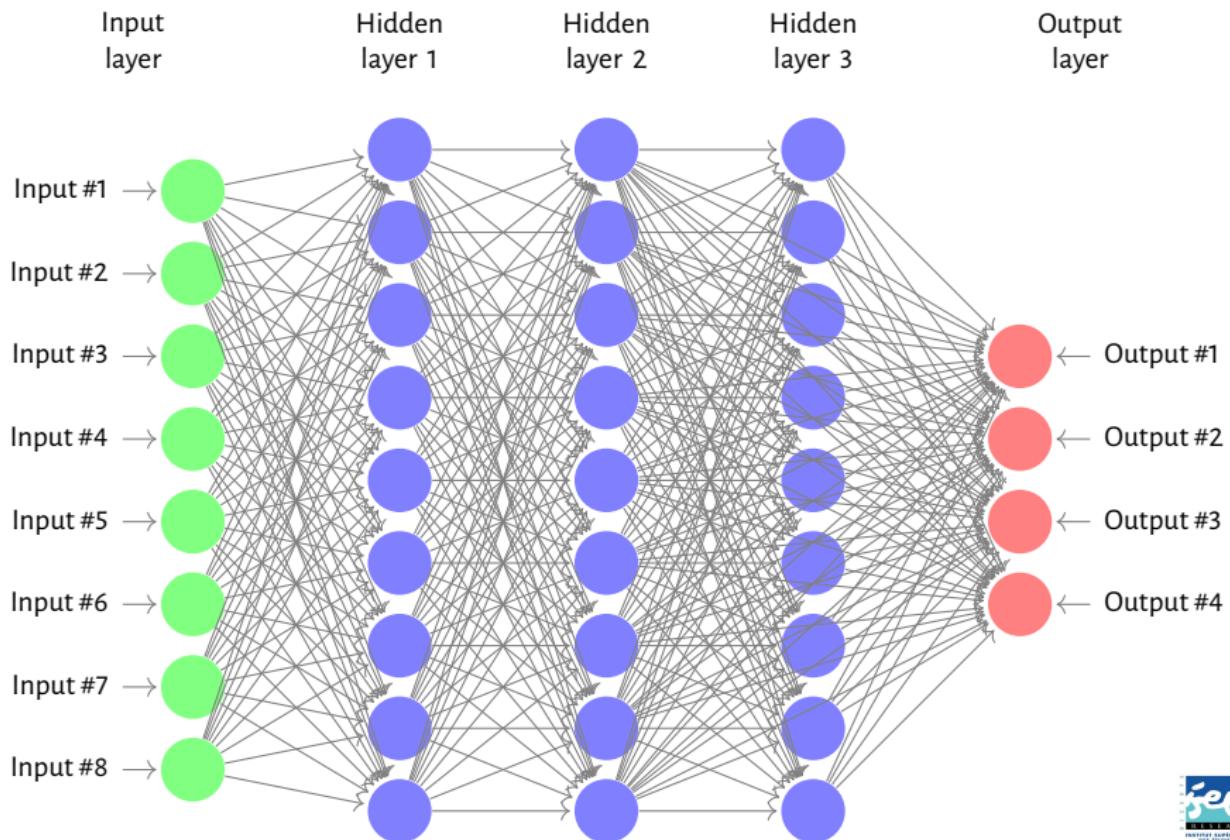
[https://id.wikipedia.org/wiki/Sel\\_saraf](https://id.wikipedia.org/wiki/Sel_saraf)

# Fundamental unit of a neural network (2/2)

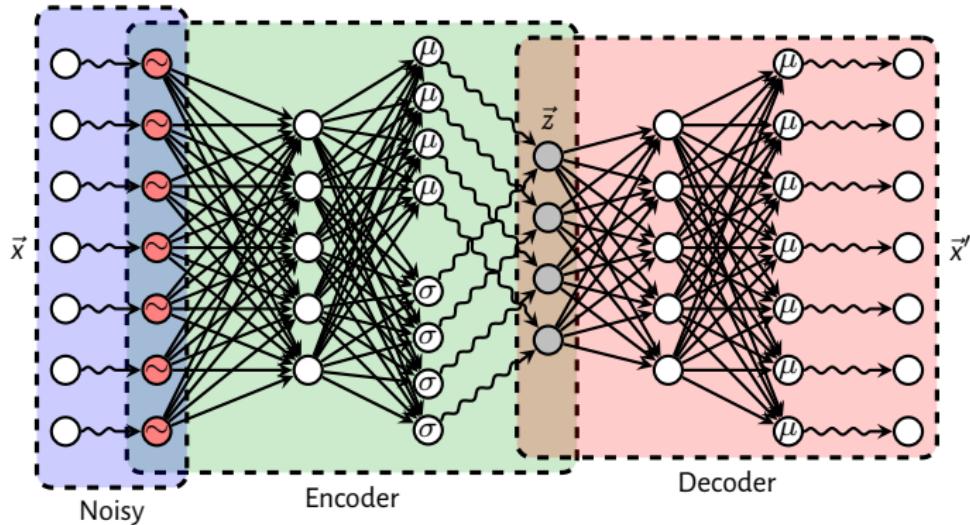




# Multilayer Perceptron (MLP)



# Variational Auto-Encoder



Next...



- 1 An overview
- 2 Supervised Learning
- 3 Unsupervised Learning
- 4 Deep Learning
- 5 ML Landscape through Quizzes

## MCQ (1/14)

Features of Machine Learning are ...

- ① Automation
- ② Improved customer experience
- ③ Business intelligence
- ④ All of the above

The term machine learning was coined by ...

- ① James Gosling
- ② Arthur Samuel
- ③ Guido van Rossum
- ④ None of the above

Which among the following algorithms are used in Machine learning?

- ① Naive Bayes
- ② Support Vector Machines
- ③ K-Nearest Neighbors
- ④ All of the above

## MCQ (2/14)

... is the machine learning algorithm that can be used with labeled data.

- ① Regression algorithm
- ② Clustering algorithm
- ③ Association algorithm
- ④ All of the above

Replace missing values with mean/median/mode helps to handle missing or corrupted data in a dataset. True/False?

- ① True
- ② False

The Real-world machine learning use cases are

- ① Digital assistants
- ② Chatbots
- ③ Fraud detection
- ④ All of the above

## MCQ (3/14)

... is a part of machine learning that works with neural networks.

- ① Deep learning
- ② Artificial intelligence
- ③ All of the above
- ④ None of the above

The supervised learning problems can be grouped as ...

- ① Regression problems
- ② Classification problems
- ③ All of the above
- ④ None of the above

The unsupervised learning problems can be grouped as ...

- ① Clustering
- ② Association
- ③ All of the above
- ④ None of the above

## MCQ (4/14)

Overfitting is a type of modelling error which results in the failure to predict future observations effectively or fit additional data in the existing model. Yes/No?

- ① Yes
- ② No
- ③ Can not say
- ④ Probably

... is the scenario when the model fails to decipher the underlying trend in the input data.

- ① Underfitting
- ② Overfitting
- ③ All of the above
- ④ None of the above

## MCQ (5/14)

Machine learning approaches can be traditionally categorized into ... categories.

- ① 3
- ② 4
- ③ 7
- ④ 9

The categories in which Machine learning approaches can be traditionally categorized are

...

- ① Supervised learning
- ② Unsupervised learning
- ③ Reinforcement learning
- ④ All of the above

## MCQ (6/14)

In general, to have a well-defined learning problem, we must identify which of the following

- ① The class of tasks
- ② The measure of performance to be improved
- ③ The source of experience
- ④ All of the above

The average positive difference between computed and desired outcome values

- ① Root Mean Squared Error
- ② Mean Squared Error
- ③ Mean Absolute Error
- ④ Mean Positive Error

... is used as an input to the machine learning model for training and prediction purposes.

- ① Target variable
- ② Feature vector
- ③ All of the above
- ④ None of the above

## MCQ (7/14)

Simple regression assumes a ... relationship between the input attribute and output attribute.

- ① linear
- ② quadratic
- ③ reciprocal
- ④ inverse

What is Machine Learning (ML)?

- ① The selective acquisition of knowledge through the use of computer programs
- ② The selective acquisition of knowledge through the use of manual programs
- ③ **The autonomous acquisition of knowledge through the use of computer programs**
- ④ The autonomous acquisition of knowledge through the use of manual programs

## MCQ (8/14)

The correlation between the number of years an employee has worked for a company and the salary of the employee is 0.75. What can be said about employee salary and years worked?

- ① There is no relationship between salary and years worked.
- ② Individuals that have worked for the company the longest have higher salaries.
- ③ Individuals that have worked for the company the longest have lower salaries.
- ④ The majority of employees have been with the company a long time.

## Successful applications of ML

- ① Learning to recognize spoken words
- ② Learning to drive an autonomous vehicle
- ③ Learning to classify new astronomical structures
- ④ Learning to play world-class backgammon
- ⑤ All of the above

## MCQ (9/14)

Which machine learning models are trained to make a series of decisions based on the rewards and feedback they receive for their actions?

- ① Supervised learning
- ② Unsupervised learning
- ③ Reinforcement learning
- ④ All of the above

Which of the following is not a type of supervised learning?

- ① Classification
- ② Regression
- ③ Clustering
- ④ None of the above

## MCQ (10/14)

As the amount of training data increases

- ① Training error usually increases and generalization error usually increases
- ② Training error usually increases and generalization error usually decreases
- ③ Training error usually decreases and generalization error usually decreases
- ④ Training error usually decreases and generalization error usually increases

Which of the following are not classification tasks?

- ① Find the gender of a person by analyzing his writing style
- ② Detect Pneumonia from Chest X-ray images
- ③ Predict the price of a house based on floor area, number of rooms, etc.
- ④ Predict whether there will be abnormally heavy rainfall next year

Which of the following is a categorical feature?

- ① Height of a person
- ② Price of petroleum
- ③ Amount of rainfall in a day
- ④ Mother tongue of a person

## MCQ (11/14)

What is the use of validation dataset in Machine Learning?

- ① To train the machine learning model.
- ② To tune the hyperparameters of the machine learning model
- ③ To evaluate the performance of the machine learning model
- ④ None of the above

Which of the following criteria is typically used for optimizing in linear regression.

- ① Maximize the number of points it touches.
- ② Minimize the number of points it touches.
- ③ Minimize the squared distance from the points.
- ④ Minimize the maximum distance of a point from a line.

For two runs of K-Mean clustering, is it expected to get same clustering results?

- ① Yes
- ② No

## MCQ (12/14)

Which of the following can act as possible termination conditions in K-Means?

- a) For a fixed number of iterations
  - b) Assignment of observations to clusters does not change between iterations. Except for cases with a bad local minimum.
  - c) Centroids do not change between successive iterations
  - d) Terminate when RSS falls below a threshold
- ① a, c & d  
② a, b & c  
③ a, b & d  
④ All of the above

In training a neural network, we notice that the loss does not increase in the first few starting epochs: What is the reason for this?

- ① The learning rate is low  
② Regularization parameter is high  
③ Stuck at the local minima  
④ All of the above

## MCQ (13/14)

Which of the following is true about model capacity (*where model capacity means the ability of neural network to approximate complex functions*)?

- ① As number of hidden layers increases, model capacity increases
- ② As dropout ratio increases, model capacity increases
- ③ As learning rate increases, model capacity increases
- ④ None of these

When there is noise in data, which of the following options would improve the performance of the KNN algorithm?

- ① Increase the value of k
- ② Decrease the value of k
- ③ Changing value of k will not change the effect of the noise
- ④ None of these

## MCQ (14/14)

Logistic Regression is used for ...

- ① regression purposes
- ② classification purposes
- ③ all of the above
- ④ none of the above

Which of the following methods do we use to best fit the data in Logistic Regression?

- ① Least Squared Error
- ② Maximum Likelihood
- ③ Jaccard distance

## Some Useful Links

- ① <https://www.ibm.com/downloads/cas/GB8ZMQZ3>
- ② <https://www.mathworks.com/company/mathworks-stories/deep-learning-uses-ai-to-translate-fcas-into-working-code.html>
- ③ <https://karpathy.github.io/2022/03/14/lecun1989/>
- ④ <https://github.com/Harislqbal88/PlotNeuralNet>
- ⑤ <https://towardsdatascience.com/how-to-easily-draw-neural-network-architecture-diagrams-a6b6138ed875>
- ⑥ <https://explore.mathworks.com/machine-learning-knowledge-quiz#>

## Further Reading (1/2)

- [Bur19] A. Burkov. *The Hundred-Page Machine Learning Book*. Andriy Burkov, Jan. 1, 2019. 160 pp.
- [Bur20] A. Burkov. *Machine Learning Engineering*. True Positive Inc., Sept. 8, 2020. 310 pp.
- [DFO20] M. P. Deisenroth, A. A. Faisal, and C. S. Ong. *Mathematics for Machine Learning*. Cambridge University Pr., Apr. 1, 2020. 398 pp.
- [ENM15] I. El Naqa and M. J. Murphy. "What Is Machine Learning?" In: *Machine Learning in Radiation Oncology: Theory and Applications*. Ed. by I. El Naqa, R. Li, and M. J. Murphy. Cham: Springer International Publishing, 2015, pp. 3–11. DOI: [10.1007/978-3-319-18305-3\\_1](https://doi.org/10.1007/978-3-319-18305-3_1) (cit. on p. 10).
- [Fla12] P. Flach. "References". In: *Machine Learning: The Art and Science of Algorithms that Make Sense of Data*. Cambridge University Press, Sept. 2012, pp. 367–382. DOI: [10.1017/CBO9780511973000.017](https://doi.org/10.1017/CBO9780511973000.017).
- [GBC16] I. Goodfellow, J. Bengio, and A. Courville. *Deep Learning*. MIT Press Ltd, Nov. 18, 2016. 800 pp.
- [Gé19] A. Géron. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, Oct. 15, 2019. 819 pp.
- [HYU21] T. J. Hui (York University). *Machine Learning Fundamentals*. Cambridge University Press, Nov. 25, 2021. 420 pp.
- [Jia22] H. Jiang. *Machine Learning Fundamentals*. Cambridge University Pr., Jan. 31, 2022.

## Further Reading (2/2)

- [JPM21] L. M. John Paul Mueller. *Machine Learning For Dummies*. Wiley John + Sons, Apr. 8, 2021.  
464 pp.
- [Mit97] T. Mitchell. *Machine Learning*. McGraw-Hill International Editions. McGraw-Hill, 1997  
(cit. on pp. 8, 11–13).
- [Pra18] M. L. de Prado. *Advances in Financial Machine Learning*. John Wiley & Sons Inc, May 4, 2018.  
400 pp.
- [SG16] A. C. M. Sarah Guido. *Introduction to Machine Learning with Python*. O'Reilly Media, July 31, 2016.
- [Woj12] J. Wojtusiak. “Machine Learning”. In: *Encyclopedia of the Sciences of Learning*. Springer US, 2012, pp. 2082–2083. DOI: 10.1007/978-1-4419-1428-6\_1927 (cit. on p. 9).