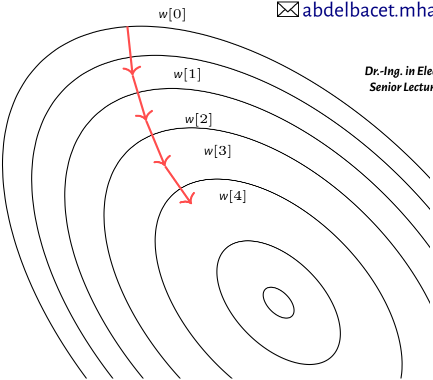
An Introduction To Machine Learning Sorcery

(Demystification Process)[1]

Abdelbacet Mhamdi
abdelbacet.mhamdi@bizerte.r-iset.tn

Dr.-Ing. in Electrical Engineering
Senior Lecturer at ISET 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

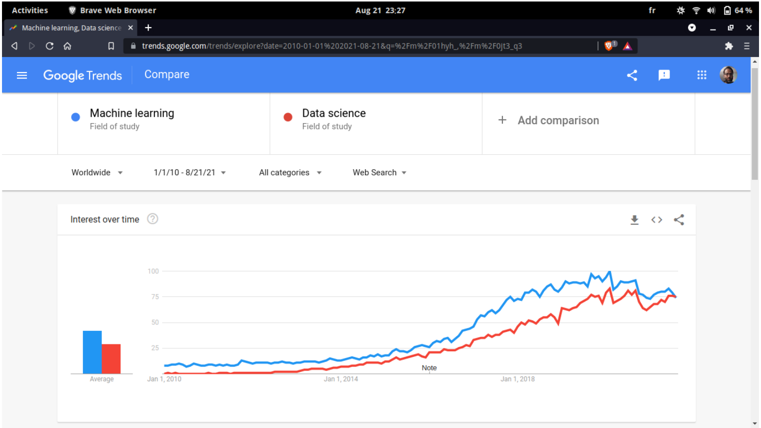
[1]Available @ https://github.com/a-mhamdi/isetbz/

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.

Roadmap

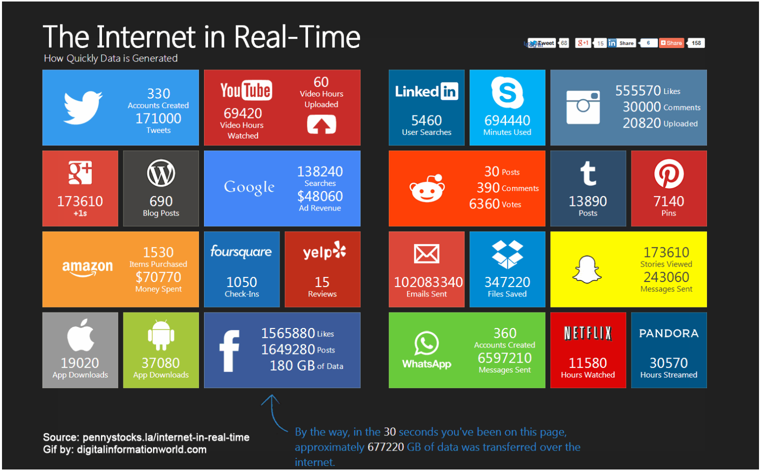
Next...



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

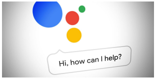
- A value of 100 is the peak popularity for the term;
- A value of 50 means that the term is half as popular;
- A score of 0 means there was not enough data for this term."

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].

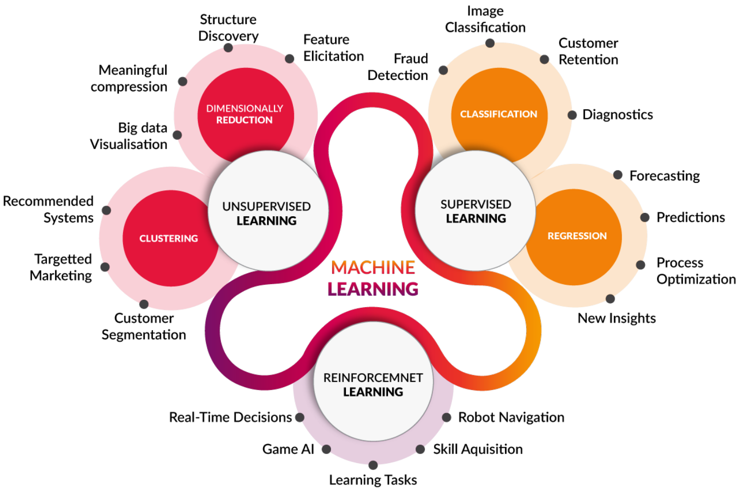
Task #1

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?

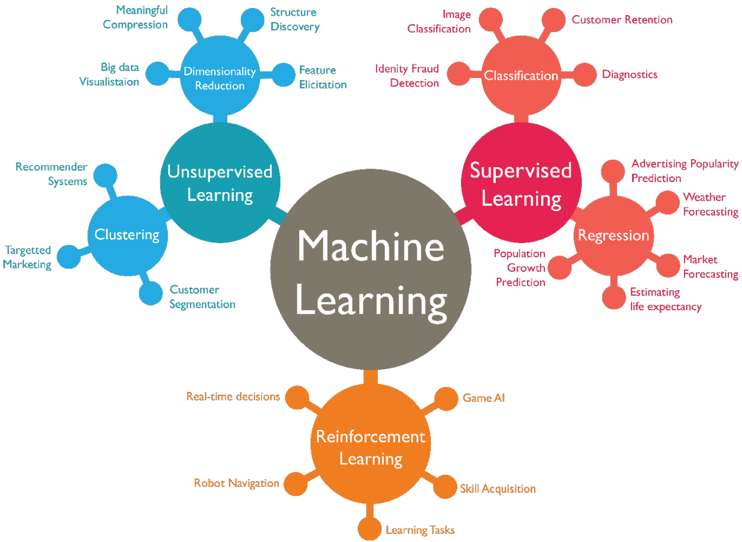
1. Classifying emails as spam or not spam;
2. Watching you label emails as spam or not spam;
3. The number (or fraction) of emails correctly classified as spam/not spam;
4. None of the above-this not a machine learning problem.

Overall Methodology

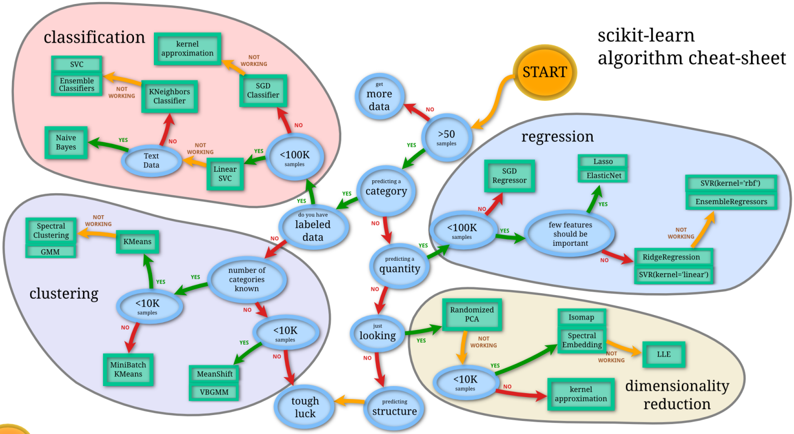
1. Define the problem;
2. Gather dataset;
3. Choose measure of success;
4. Decide evaluation protocol;
5. Prepare the data;
6. Develop a model;
7. Iterate models.



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



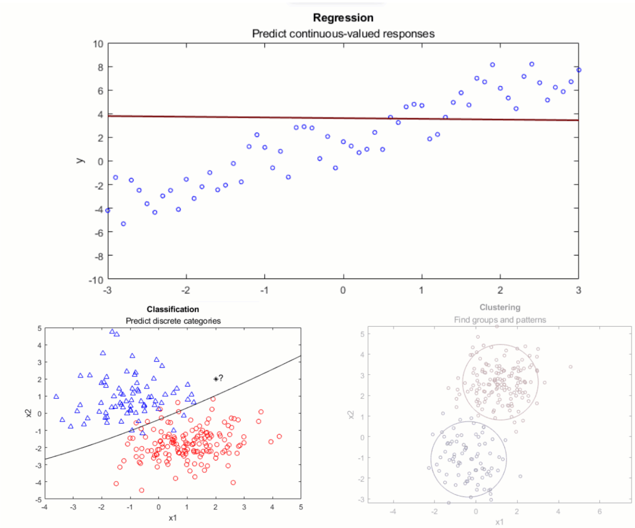
https://vitalflux.com/great-mind-maps-for-learning-machine-learning/



Back

https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

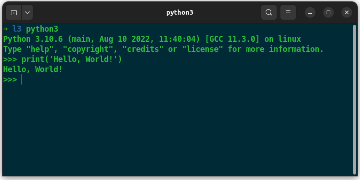
Regression | Classification | Clustering



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

Programming Languages

python.org/



julialang.org/

Development Environments

ANACONDA

Docker Compose

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

spyder

jupyter

Pluto.jl

Required Packages

Valid only for...

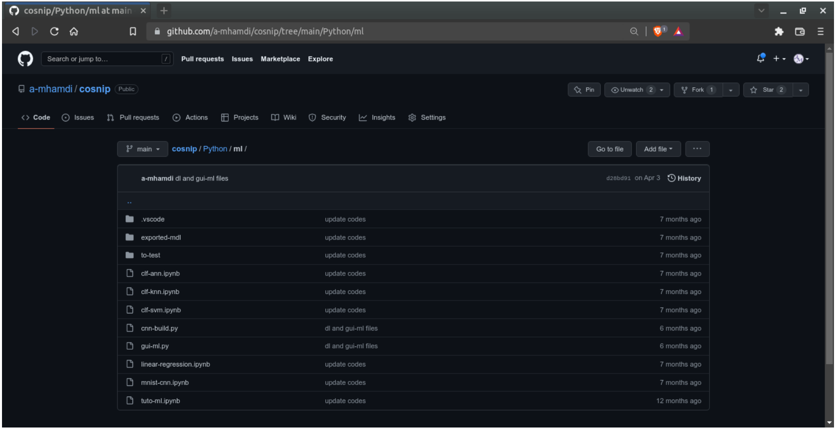
- A full list is available @ https://pypi.org/

Numpy | *Matplotlib* | *Pandas* | *Scikit — learn* | *Keras*

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

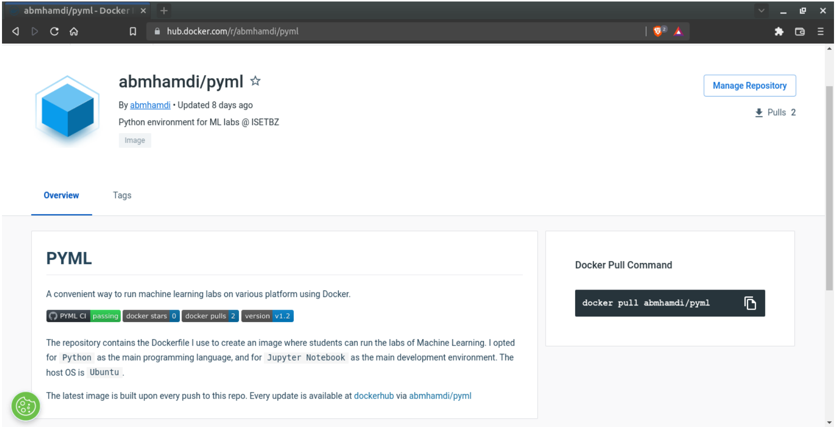
> pip install virtualenv
> virtualenv - -version
> virtualenv «virtualenv_name»
> «virtualenv_name»\Scripts\activate :: ACTIVATE
> deactivate :: DEACTIVATE

Source Control Management (SCM)



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

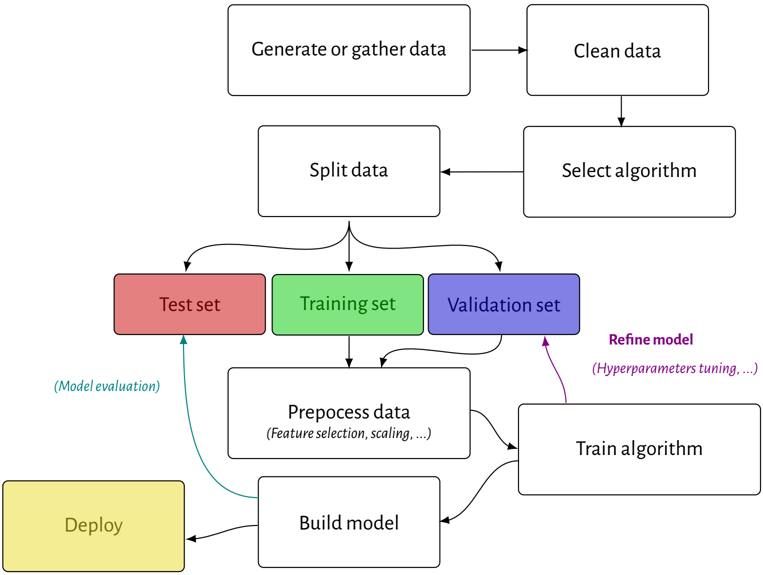
Continuous Integration (CI)



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

Next...

Workflow in Machine Learning



Feature Scaling

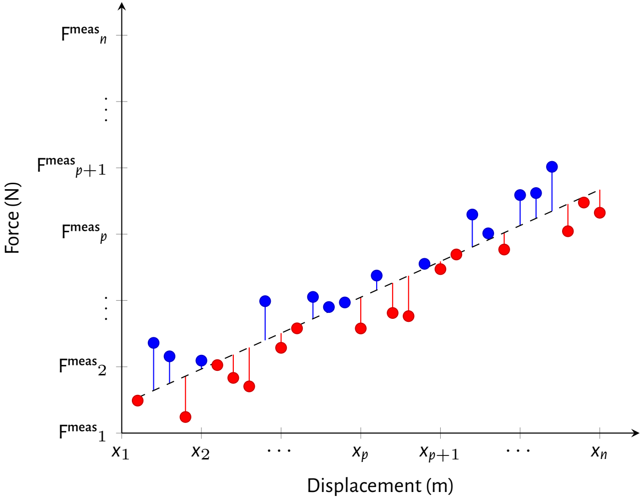
Normalisation	Standardisation
$X = \dfrac{X - \min(X)}{X.\max() - X.\min()}$	$X = \dfrac{X - X.\text{mean}()}{X.\text{std}()}$
▲ No assumption on data distribution	▲ More recommended when following normal distribution

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



Force (N)
$F^{meas}{}_n$
$\vdots$
$F^{meas}{}_{p+1}$
$F^{meas}{}_p$
$\vdots$
$F^{meas}{}_2$
$F^{meas}{}_1$
x_1
x_2
$\cdots$
x_p
x_{p+1}
$\cdots$
x_n
Displacement (m)

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)*:

$$F = kx \tag{1}$$

Restoring force is proportional to displacement.

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

x_i	x_1	$\cdots$	x_p	$\cdots$	x_n
$\mathrm{F^{meas}}_i$	$\mathrm{F^{meas}}_1$	$\cdots$	$\mathrm{F^{meas}}_p$	$\cdots$	$\mathrm{F^{meas}}_n$

$$\begin{aligned}\mathrm{F^{meas}}_i &= \mathrm{F}_i + \varepsilon_i \\ &= kx_i + \varepsilon_i,\end{aligned} \tag{2}$$

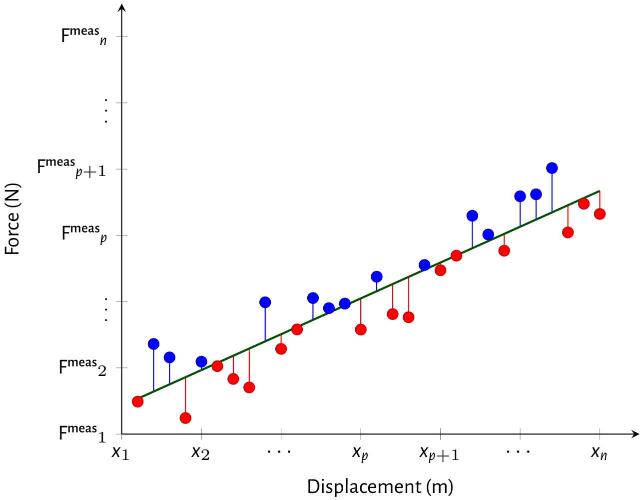
where F_i denotes the unknown real value of the force applied to the spring. In order to estimate the stiffness value k, we can consider the quadratic criterion:

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

$$\frac{\partial \mathcal{J}}{\partial k} = 0 \tag{3}$$

$$\begin{aligned}
2\sum_{i=1}^{n}\left(\mathrm{F^{meas}}_i - kx_i\right)\sum_{i=1}^{n}\frac{\partial\left(\mathrm{F^{meas}}_i - kx_i\right)}{\partial k} &= 0\\
\sum_{i=1}^{n}\left(\mathrm{F^{meas}}_i - kx_i\right)\sum_{i=1}^{n}x_i &= 0
\end{aligned}$$

$$\sum_{i=1}^{n}\mathrm{F^{meas}}_i\, x_i = k\sum_{i=1}^{n}x_i^2 \iff \boxed{\hat{k} = \frac{\sum_{i=1}^{n}\mathrm{F^{meas}}_i\, x_i}{\sum_{i=1}^{n}x_i^2}}$$



Force (N)
$F^{meas}{}_n$
$\vdots$
$F^{meas}{}_{p+1}$
$F^{meas}{}_p$
$\vdots$
$F^{meas}{}_2$
$F^{meas}{}_1$
x_1
x_2
$\cdots$
x_p
x_{p+1}
$\cdots$
x_n
Displacement (m)

Simple Linear Regression

CODE SNIPPET

Training the Simple Linear Regression model on the Training set

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

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

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

Predicting the Test set results

[ ]:
```
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:

$$y = y_0 + v_0 t \tag{4}$$

Table: Measurements of position y

t_i	t_1	$\cdots$	t_p	$\cdots$	t_n
$\mathrm{y^{meas}}_i$	$\mathrm{y^{meas}}_1$	$\cdots$	$\mathrm{y^{meas}}_p$	$\cdots$	$\mathrm{y^{meas}}_n$

$$\begin{aligned} \mathrm{y^{meas}}_i &= y_i + \varepsilon_i \\ &= y_0 + v_0 t_i + \varepsilon_i, \end{aligned} \tag{5}$$

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 $\begin{bmatrix} y_0, & v_0 \end{bmatrix}^T$, 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 ε is set by $\varepsilon_i, \ \forall i \geq 1$:

$$\varepsilon = \begin{bmatrix} \varepsilon_1 & \cdots & \varepsilon_n \end{bmatrix}^T$$

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

Multiple Linear Regression

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/
$\longrightarrow$ `Machine Learning` $\longrightarrow$ `Codes` $\longrightarrow$ `multiple-linear-regression.ipynb`

Consider the following multivariate equation:

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

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

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

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

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

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

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

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

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

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

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 θ:

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

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

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

$$\frac{\partial \varepsilon}{\partial \theta} = -X \tag{11}$$

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

The regressor is given by

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

$X^T X$ **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

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

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

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

$$\boxed{\theta_0 \;\leftarrow\; \theta_0 + \alpha \frac{1}{n} \sum_{k=1}^{n} (y_k - h_\theta\,(x_k))\, x_{(0,\,k)}} \qquad \boxed{\theta_1 \;\leftarrow\; \theta_1 + \alpha \frac{1}{n} \sum_{k=1}^{n} (y_k - h_\theta\,(x_k))\, x_{(1,\,k)}}$$

$$\vdots$$

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

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.

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

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

```python
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:

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

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

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

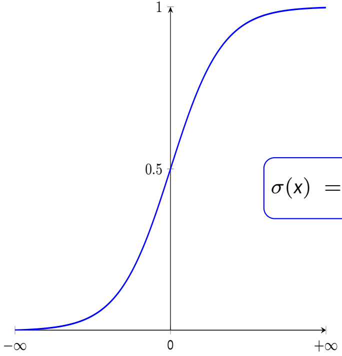
It denotes the ratio of how much we got right over all cases. Recall, on the other hand, designates the ratio of how much positives do we got right over all actual positive cases.

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

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 σ



$$\boxed{\sigma(x) = \frac{1}{1+e^{-x}}}$$

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

Decision boundary

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

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

Hypothesis:

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

Cost function:

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

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

Gradient Descent

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

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

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

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

$$\boxed{\theta_0 \;\leftarrow\; \theta_0 + \alpha\frac{1}{n}\sum_{k=1}^{n}\left(y_k - h_\theta\left(x_k\right)\right)x_{(0,\,k)}} \qquad \boxed{\theta_1 \;\leftarrow\; \theta_1 + \alpha\frac{1}{n}\sum_{k=1}^{n}\left(y_k - h_\theta\left(x_k\right)\right)x_{(1,\,k)}}$$

$$\vdots$$

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

Logistic Regression

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/
$\rightarrow$ `Machine Learning` $\rightarrow$ `Codes` $\rightarrow$ `logistic-regression.ipynb`

K Nearest Neighbors (1/4)

▸ Evelyn Fix and Joseph Hodges, 1951 ▸ Thomas Cover, 1966

Algorithm 1 Summary Construction

```
procedure HOW DOES KNN WORK? (Finding Nearest Neighbors)
Input: A query point;
Output: Assign a class label to that point.
    Define how many neighbors will be checked to classify the specific query point;
    Compute the distance d(x, y) of the query point to other data points;
    Count the number of the data points in each category;
    Assign the query point to the class with most frequent neighbors.
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)

Task #4[a]

[a] 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 **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 **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

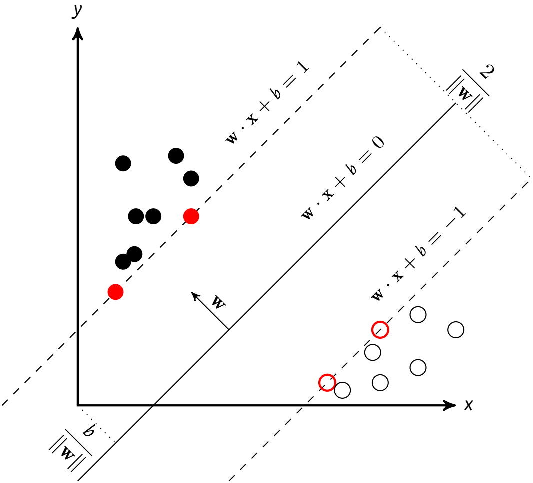
[ ]:
```
classifier = KNeighborsClassifier(n_neighbors, metric, p)
classifier.fit(X_train, y_train)
```

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	▲	Probabilistic	▼	Almost linearly separable data
K-NN	▲	Simple	▼	Number of neighbors k
	▲	Fast		
	▲	Efficient		

Next...

Task #5[a]

[a]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 stroy
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[a]

[a]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 stroy
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

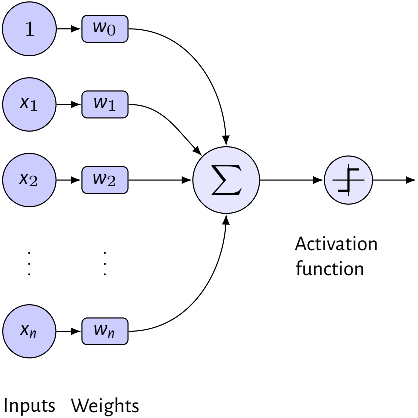
```
[ ]: kmeans = KMeans(n_clusters, init, random_state)
     y_kmeans = kmeans.fit_predict(X)
```

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

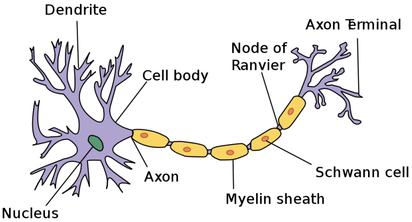
Next...

Fundamental unit of a neural network (1/2)

Artificial neuron

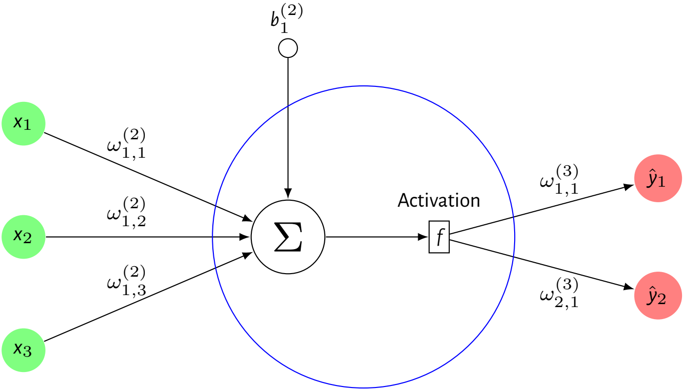


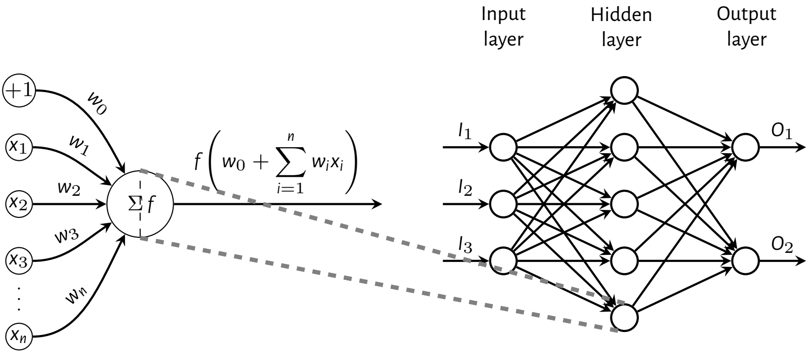
Biological neuron



https://id.wikipedia.org/wiki/Sel_saraf

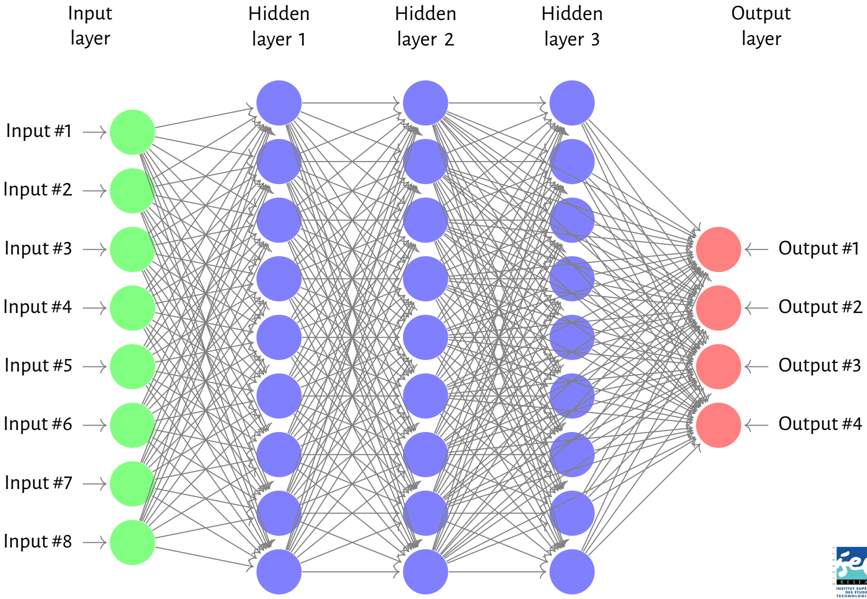
Fundamental unit of a neural network (2/2)



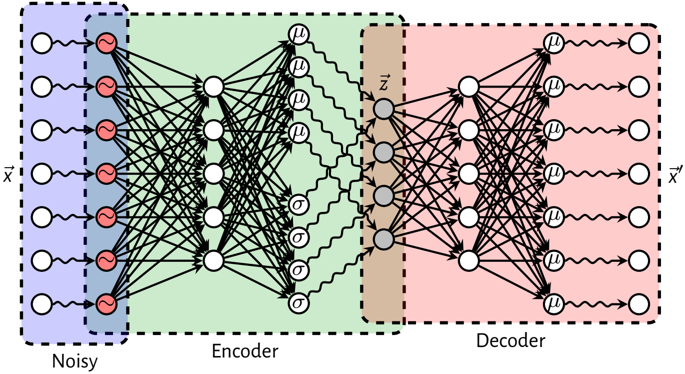


+1
x_1
x_2
x_3
x_n
w_0
w_1
w_2
w_3
w_n
Σf
$f\left(w_0 + \sum_{i=1}^{n} w_i x_i\right)$
Input layer
Hidden layer
Output layer
I_1
I_2
I_3
O_1
O_2

Multilayer Perceptron (MLP)



Variational Auto-Encoder



Next...

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 identity 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 emplyee has worked for a company and the salary of the employee is 0.75. What can be said about emplyee 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/HarisIqbal88/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 (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.

[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.

[Sch+19] J. Schmidt et al. "Recent advances and applications of machine learning in solid-state materials science". In: *npj Computational Materials* 5.1 (Aug. 2019). DOI: 10.1038/s41524-019-0221-0.

[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).