|  |  |  |
| --- | --- | --- |
| Numérique et Sciences Informatiques | | |
| 3h30 | **Algorithmique des k plus proches voisins** |  |
| **Objectif** : savoir écrire un algorithme qui prédit la classe d’un élément en fonction de la classe majoritaire de ses k plus proches voisins. | | |
| **Matériel**: Python | | |

**Contexte**

Vidéo de 8min23 sur l'IA :https://www.youtube.com/watch?v=yQLmgw3rClM

Vous pouvez regarder toute la chaine ...

L’intelligence artificielle est l’ensemble des théories et des techniques mises en œuvre afin de réaliser des machines capables de **simuler l’intelligence humaine**.

Les IA sont souvent amenées à classer des données.

Par exemple, dans le CAPTCHA ci-contre, en sélectionnant les carrés qui contiennent une partie du panneau STOP vous ferez un classement vers la catégorie « panneau de STOP », ce qui permettra à une IA de voiture autonome « d’apprendre » à identifier ce type de panneaux.

C’est ce qu’on appelle de l’apprentissage supervisé.

Pour le classement un des algorithmes les plus utilisés est la méthode des **k plus proches voisins (kppv)**,aussi appelée **knn** (k nearest neighbors).

Cette méthode consiste à classer un élément par rapport à ses voisins les plus proches ayant les mêmes caractéristiques afin de lui attribuer une classe (étiquette).

**Étiquette (classe)**

Lors d’un apprentissage supervisé on souhaite classer des éléments d’un **jeu de données (dataset)** à partir de leurs **caractéristiques** (**features**), pour cela il va falloir déterminer des **classes** pour ces éléments (ex : les pommes, les oranges, les kiwis …).

Supposons que sur un plan en deux dimensions nous disposions d’un ensemble de points que nous appellerons **échantillons** (**samples**) et qu’un expert a colorié certains de ces points en rouge et d’autres en bleu en fonction de leurs caractéristiques et à partir d’une logique que vous ne connaissez pas.

Bleu et rouge vont être deux **étiquettes** (**labels**) que nous pouvons attribuer à ces échantillons pour les classer. Un échantillon ne peut avoir qu’une seule étiquette.

Si on ajoute un nouveau point dans l’espace, nous souhaitons déterminer quelle étiquette lui attribuer.

**Les k plus proches voisins**

Le "k"correspond au nombre de voisins que nous allons prendre en compte.

Le but de l’algorithme des **k plus proches voisins** est de regarder les k plus proches voisins d’un **nouveau vecteur de caractéristiques** (point à classer) parmi les **échantillons connus** (points déjà classées) et de lui attribuer l'**étiquette** qui est majoritaire parmi ses voisins.

* Un expert a déjà classé des points avec des étiquettes bleus et des étiquettes rouges.

En appliquant les k plus proches voisins, déterminez l’étiquette des points p et m pour :

|  |  |
| --- | --- |
|  | k = 1 : P\_rouge – M\_bleu |
| k = 3 : P\_bleu – M\_rouge |
| k = 5 : P\_bleu – M\_rouge |
| k = 9 : P\_rouge – M\_rouge |
| k = 2 : ? - ? |

On constate que pour appliquer les **k plus proches voisins** il faut **k impair**.

**Les mal classés**

Sur le cas précédent on a pu voir qu’en fonction du k choisis, l’étiquette attribuée n’est pas toujours la même, pourtant chacun de ces points ne peut avoir qu’une seule étiquette !

Un expert sera en capacité de dire quelle étiquette correspond à tel point et on pourra comparer notre résultat avec celui de l’expert pour vérifier si notre algorithme fonctionne correctement.

Un exemple concret de classement est le **test PCR** (Covid notamment) car il permet de savoir si un patient est atteint d’une maladie virale, il permet donc de donner l'étiquette "malade" ou "pas malade" à un patient.

Si un test PCR est positif alors qu’un patient n’est pas malade → le résultat est un **faux-positif**.

Si un test est négatif alors que le patient est vraiment malade → le résultat est un **faux-négatif**.

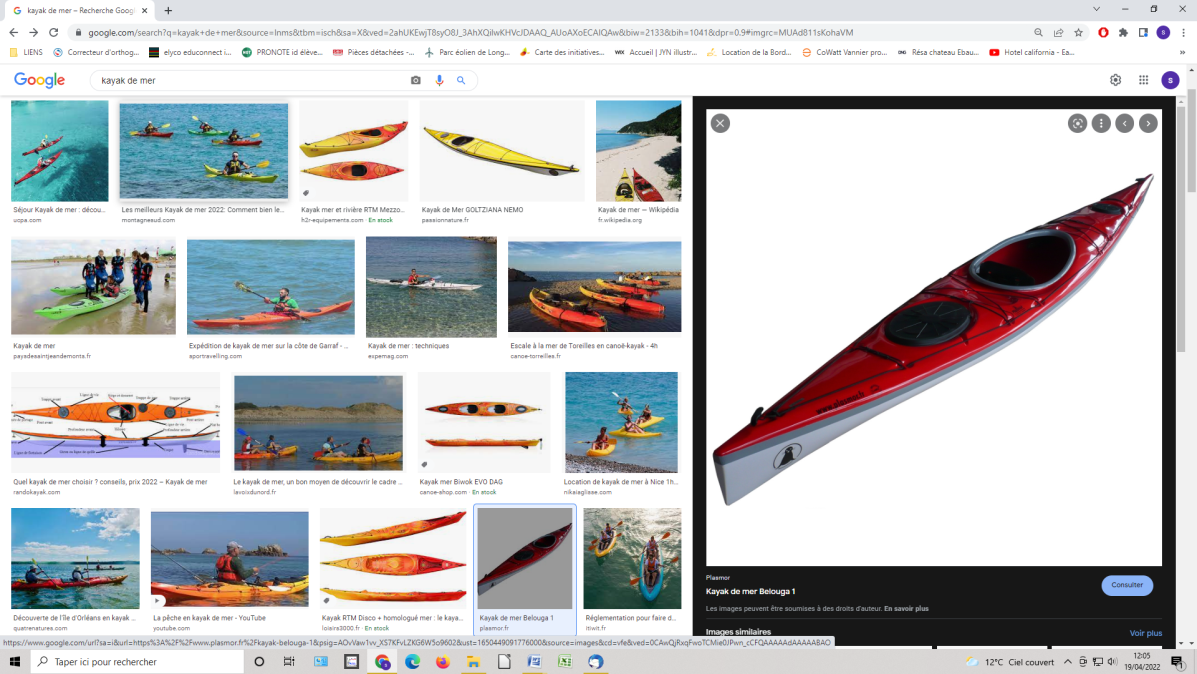
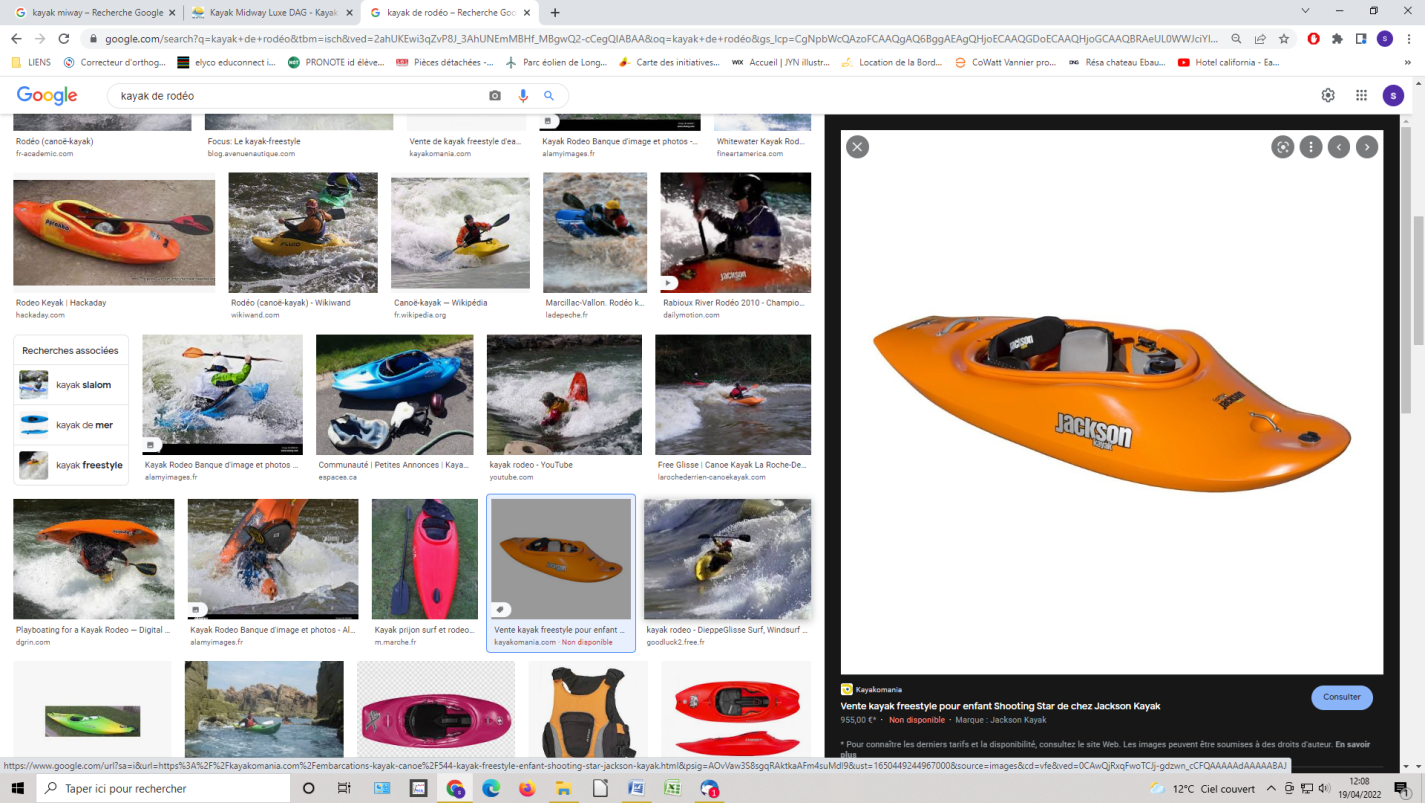
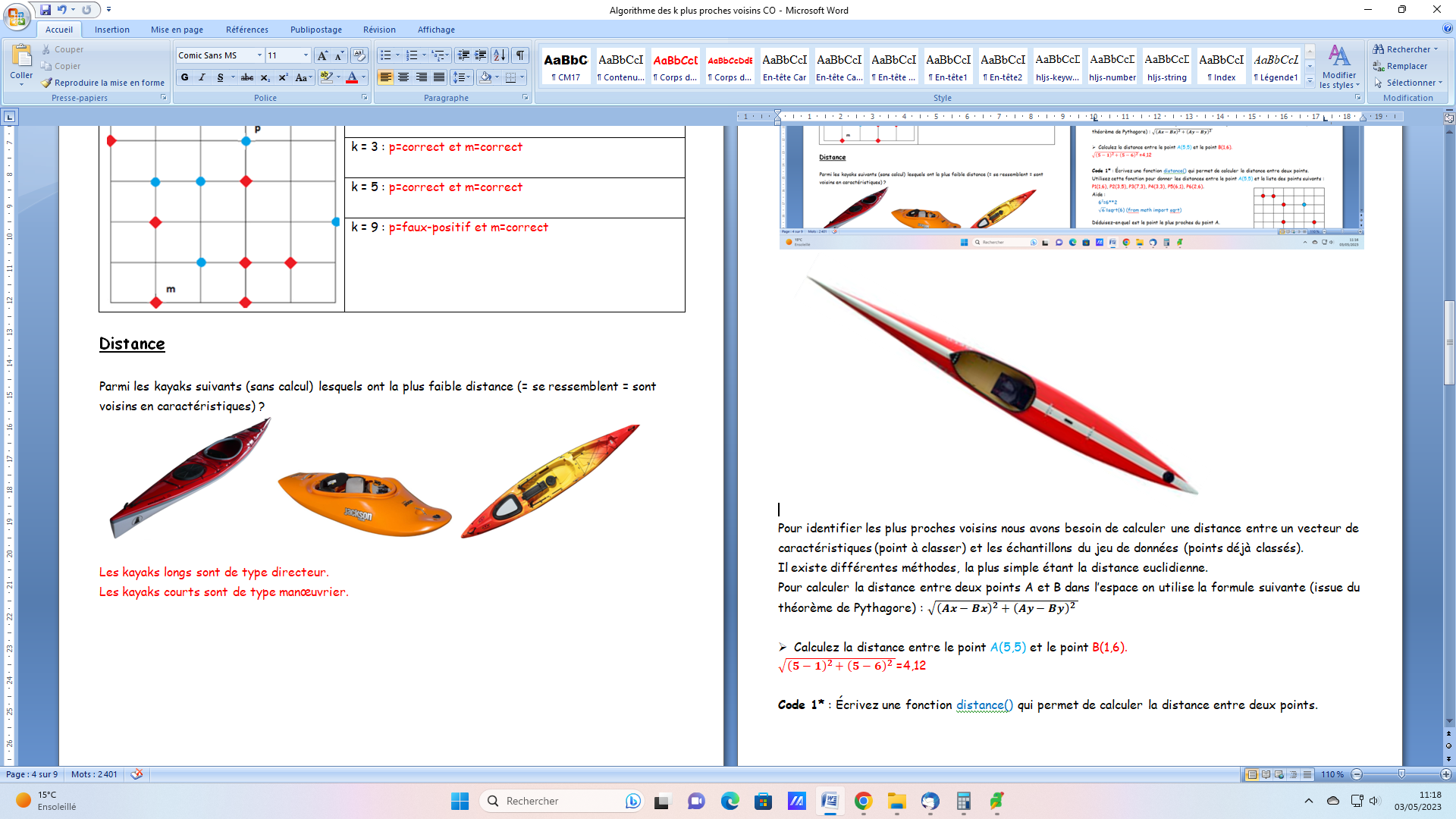
* Un expert a identifié les étiquettes suivantes pour p (patient sain) et m (patient malade).

À partir des étiquettes attribuées aux questions précédentes, indiquez si p et m sont **corrects**, **faux-positif** ou **faux-négatif** :

|  |  |
| --- | --- |
|  | k = 1 : P\_faux\_p – M\_faux\_n |
| k = 3 : P\_correct – M\_correct |
| k = 5 : P\_correct – M\_correct |
| k = 9 : P\_faux\_p – M\_correct |

**Distance**

Parmi les kayaks suivants (sans calcul) lesquels ont la plus faible distance (= se ressemblent = sont voisins en caractéristiques) ?

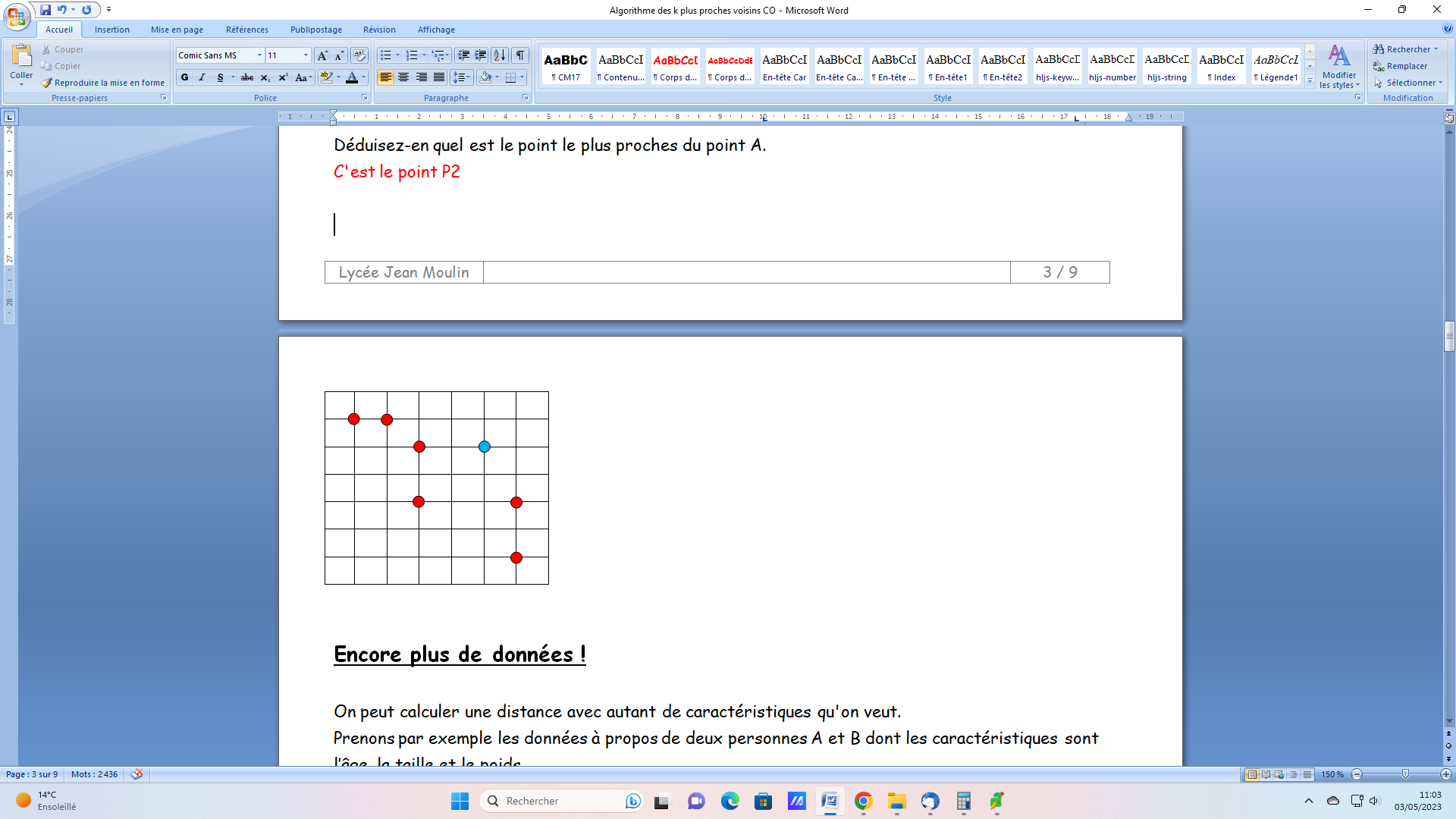
Pour identifier **les plus proches voisins** nous avons besoin de calculer une distance entre un **vecteur de caractéristiques** (point à classer) et les **échantillons connus** du jeu de données (points déjà classés).

Il existe différentes méthodes, la plus simple étant celle de la distance euclidienne.

Pour calculer la distance entre deux points A et B dans l’espace on utilise la formule suivante (issue du théorème de Pythagore) :

* Calculez la distance entre le point A(5,5) et le point B(1,6).

**Code 1\*** : Écrivez une fonction distance() qui permet de calculer la distance entre deux points.

Utilisez cette fonction pour donner les distances entre le point A(5,5) et la liste des points suivants : P1(1,6), P2(3,6), P3(3,3), P4(3,5), P5(6,1), P6(6,3).

Aide :

62=6\*\*2

=sqrt(6) (from math import sqrt)

Déduisez-en quel est le point le plus proches du point A.

**Encore plus de données !**

On peut calculer une distance avec autant de caractéristiques qu'on veut.

Prenons par exemple les **caractéristiques : âge, taille, poids**, pour distinguer deux personnes A et B.

Pour calculer la distance entre ces deux personnes nous utilisons la formule suivante :

Pour A : **Aa**=âge, **At**=taille, **Ap**=poids.

Pour B : **Ba**=âge, **Bt**=taille, **Bp**=poids.

**Code 2\*\*** : Modifiez la fonction distance() pour qu’elle fonctionne avec trois caractéristiques.

Quelle est la distance entre A et B :

A={"age":50.0,"taille":168.0,"poids":59.0}

B={"age":51.0,"taille":183.0,"poids":80.0}.

**L’encodage des données**

Nous avons donc la possibilité de calculer la distance entre deux vecteurs de caractéristiques lorsque les données sont numériques, mais les caractéristiques peuvent ne pas être des nombres.

Par exemple si on veut calculer une distance entre personnes ayant un niveau sportif différent :

* non-sportif (pas de sport),
* sportif occasionnel (quelque fois par mois),
* sportif régulier (une fois par semaine),
* sportif amateur (plusieurs fois par semaine),
* sportif professionnel (pratique du sport tous les jours au niveau professionnel).

Proposition pour représenter ces caractéristiques pour qu'elles puissent être utilisées dans notre calcul de distance :

non-sportif=0, sportif occasionnel=1, sportif régulier=2, sportif amateur=3 et sportif professionnel=4