

Faire deviner à une machine le genre d'une chanson

Mise en situation. Quand vous écoutez de la musique en streaming, l'application vous propose souvent des morceaux "qui pourraient vous plaire". Derrière cette suggestion se cache un algorithme d'**intelligence artificielle (IA)** qui a appris, à partir de milliers de chansons déjà classées, à reconnaître des points communs entre elles.

Une des idées les plus simples pour construire une telle IA s'appelle la méthode des **k plus proches voisins** (en anglais, *k-Nearest Neighbors*, ou **k-NN**). Le principe : pour deviner la catégorie d'un nouvel élément, on regarde simplement à quelle catégorie appartiennent les éléments déjà connus qui lui ressemblent le plus.

Dans cette activité, vous allez utiliser cette méthode pour apprendre à une machine à reconnaître automatiquement le **genre musical** (Pop ou Rap) d'un morceau, à partir de deux critères simples : son **tempo** (en battements par minute, bpm) et sa **durée** (en secondes).

1. Comment une machine peut-elle "reconnaître" quelque chose ?

Définition

Une intelligence artificielle ne "comprend" jamais vraiment ce qu'est une chanson : elle ne fait que comparer des **nombres**. On appelle **caractéristiques** (ou *features* en anglais) les nombres qui décrivent chaque élément à classer. Ici, chaque chanson sera décrite par 2 caractéristiques : son tempo et sa durée.

Remarque

On appelle **données d'apprentissage** un ensemble d'exemples déjà classés (ici, des chansons dont on connaît déjà le genre), qui vont servir de référence à la machine pour classer de nouveaux éléments inconnus.

2. La méthode du plus proche voisin

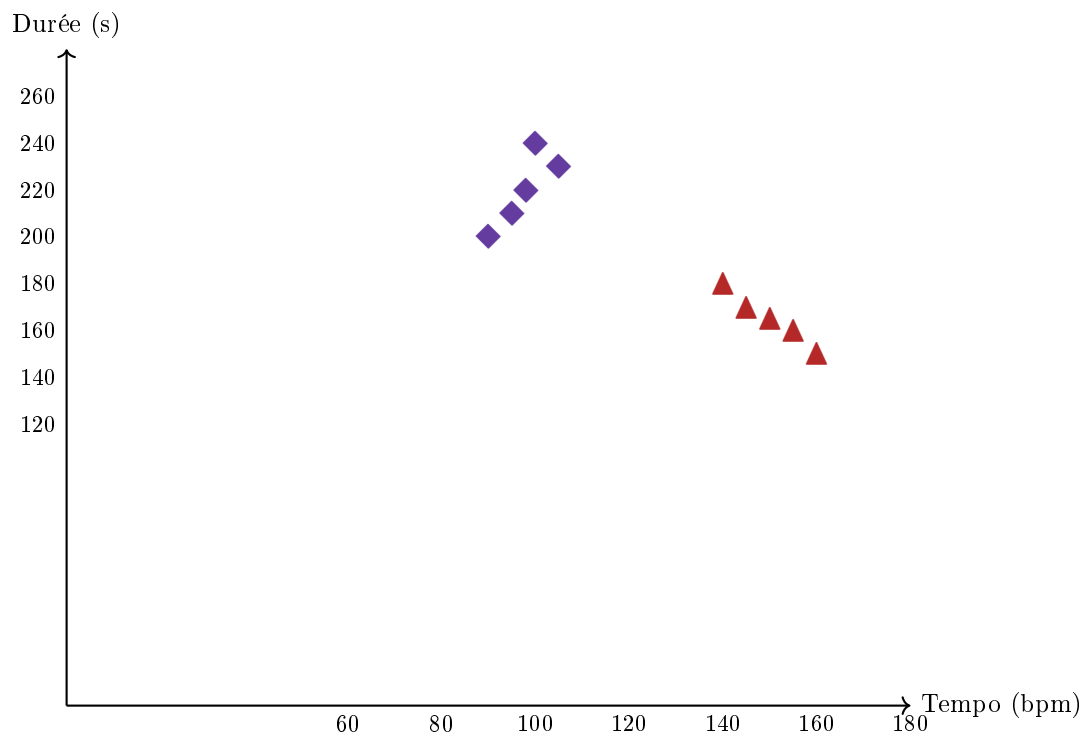
Données d'apprentissage : 10 chansons déjà classées

Chanson	Tempo (bpm)	Durée (s)	Genre
C1	95	210	Pop
C2	140	180	Rap
C3	100	240	Pop
C4	150	165	Rap
C5	90	200	Pop
C6	160	150	Rap
C7	105	230	Pop
C8	145	170	Rap
C9	98	220	Pop
C10	155	160	Rap

Définition

Méthode du plus proche voisin (1-NN). Pour classer un nouvel élément, on calcule sa **distance** avec chacune des données d'apprentissage, puis on lui attribue la **catégorie de la donnée la plus proche** (son "plus proche voisin").

● **Nouvelle chanson (148 bpm, 175 s)**



Les carrés violets représentent les chansons Pop, les triangles rouges les chansons Rap. On représentera la nouvelle chanson à classer par un point vert.

Exemple

On rappelle la formule de la distance entre deux points $A(x_A; y_A)$ et $B(x_B; y_B)$ dans un repère orthonormé :

$$AB = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}$$

► À vous de jouer !

On souhaite classer une nouvelle chanson de tempo 148 bpm et de durée 175 s (point vert sur le graphique).

1. Sur le graphique, repérez à l'œil le point qui semble le plus proche de la nouvelle chanson. À quel genre appartient-il ?
2. Calculez la distance entre la nouvelle chanson (148 ; 175) et la chanson C8 (145 ; 170).
3. Calculez la distance entre la nouvelle chanson et la chanson C2 (140 ; 180).
4. D'après vos calculs, quelle est, parmi C2 et C8, la plus proche voisine de la nouvelle chanson ? En déduire le genre attribué à la nouvelle chanson par la méthode du plus proche voisin.

Coup de pouce

Pour la question 2 : $AB = \sqrt{(148 - 145)^2 + (175 - 170)^2} = \sqrt{3^2 + 5^2} = \sqrt{34} \approx 5,8$.

3. Généraliser : la méthode des k plus proches voisins**Remarque**

La méthode du **seul** plus proche voisin a un défaut : si cette unique donnée est un cas "bizarre" ou mal classé, la prédiction sera fausse. Pour être plus robuste, on préfère souvent regarder **plusieurs** voisins à la fois, et faire un **vote majoritaire**.

Définition

Méthode des k plus proches voisins (k-NN). On choisit un entier k (par exemple $k = 3$). Pour classer un nouvel élément :

1. on calcule sa distance à **toutes** les données d'apprentissage ;
2. on retient les k données dont la distance est la plus petite (les " k plus proches voisins") ;
3. on attribue au nouvel élément la catégorie **majoritaire** parmi ces k voisins.

► À vous de jouer !

On reprend la nouvelle chanson (148 bpm ; 175 s) et on choisit $k = 3$.

1. En complétant le tableau ci-dessous, calculez la distance entre la nouvelle chanson et chacune des 5 chansons de genre Rap (C2, C4, C6, C8, C10).

Chanson	Tempo	Durée	Distance à (148 ; 175)
C2	140	180	
C4	150	165	
C6	160	150	
C8	145	170	
C10	155	160	

2. Sans même calculer les distances aux chansons Pop, pouvez-vous justifier qu'elles seront toutes nettement plus grandes que celles du tableau ? (Comparez les tempos.)
3. D'après vos résultats, quels sont les 3 plus proches voisins de la nouvelle chanson ? Quelle catégorie est majoritaire parmi eux ?
4. La méthode des k plus proches voisins donne-t-elle, ici, le même résultat que la méthode du seul plus proche voisin (partie 2) ?

Coup de pouce

Les tempos des chansons Pop sont tous inférieurs à 105 bpm, alors que la nouvelle chanson a un tempo de 148 bpm : l'écart de tempo seul (au moins $148 - 105 = 43$) dépasse déjà toutes les distances calculées avec les chansons Rap.

4. Programmer la distance en Python

Exemple

La fonction Python suivante calcule la distance entre deux points $A(x_A; y_A)$ et $B(x_B; y_B)$, grâce au module `math` qui fournit la fonction racine carrée `sqrt` :

```
from math import sqrt

def distance(xA, yA, xB, yB):
    d = sqrt((xB - xA)**2 + (yB - yA)**2)
    return d

print(distance(148, 175, 145, 170))  # distance a la chanson C8
```

► À vous de jouer !

1. Sans exécuter le code, donner la valeur (arrondie au dixième) renvoyée par `distance(148, 175, 140, 180)`.
2. Écrire une fonction `test1(x, y)` qui renvoie `True` si le point $M(x; y)$ est à l'intérieur du cercle de centre $A(150; 170)$ et de rayon 10, et `False` sinon.

Coup de pouce

On pourra réutiliser la fonction `distance` déjà écrite : le point M est à l'intérieur du cercle si et seulement si `distance(x, y, 150, 170) ≤ 10`.

```
def test1(x, y):
    if distance(x, y, 150, 170) <= 10:
        return True
    else:
        return False
```

★ Question étoilée

Écrire une fonction `test2(x, y, xA, yA, R)` qui renvoie `True` si le point $M(x; y)$ est à l'intérieur du cercle de centre $A(x_A; y_A)$ et de rayon R , et `False` sinon (la fonction doit fonctionner pour n'importe quel cercle, pas seulement celui de la question précédente).

5. Trouver le plus proche voisin en Python**Exemple**

On stocke les données d'apprentissage sous forme de **listes** : une liste des tempos, une liste des durées, et une liste des genres associés (dans le même ordre).

```
tempos = [95, 140, 100, 150, 90, 160, 105, 145, 98, 155]
durees = [210, 180, 240, 165, 200, 150, 230, 170, 220, 160]
genres = ["Pop", "Rap", "Pop", "Rap", "Pop", "Rap", "Pop", "Rap", "Pop", "Rap"]
```

► À vous de jouer !

On souhaite écrire une fonction `plus_proche_voisin(x, y)` qui parcourt toutes les chansons connues et renvoie le **genre** de la plus proche.

1. Compléter le code ci-dessous (les zones à compléter sont indiquées par des points de suspension ...).

```
def plus_proche_voisin(x, y):
    meilleure_distance = distance(x, y, tempos[0], durees[0])
    meilleur_indice = 0
    for i in range(1, len(tempos)):
        d = distance(x, y, ..., ...)
        if d < meilleure_distance:
            meilleure_distance = ...
            meilleur_indice = ...
    return genres[meilleur_indice]

print(plus_proche_voisin(148, 175))
```

2. Une fois le code complété, quel résultat doit afficher `print(plus_proche_voisin(148, 175))` ? Comparez avec votre réponse de la partie 2.

Remarque

Écrire le code des **k** plus proches voisins (avec $k > 1$ et un vote majoritaire) demande des outils un peu plus avancés (tri de listes, comptage d'occurrences) qui seront étudiés plus tard dans l'année. Retenez surtout l'**idée** : plus k est grand, plus la décision est "lissée" et robuste aux cas particuliers, mais un k trop grand peut aussi noyer l'information utile dans trop de voisins non pertinents.

6. Pour aller plus loin

Définition

Le choix de k . Le nombre de voisins k est un paramètre que l'on doit choisir avant d'utiliser l'algorithme. Un k trop petit (comme $k = 1$) rend la méthode très sensible aux erreurs ou aux cas particuliers des données d'apprentissage. Un k trop grand risque au contraire de "diluer" la décision parmi des voisins qui ne sont plus vraiment proches.

► À vous de jouer !

1. Dans notre exemple, que se passerait-il si l'on choisissait $k = 10$ (c'est-à-dire tous les points d'apprentissage) ? La méthode serait-elle encore utile pour classer une nouvelle chanson ?
2. Citez au moins une autre application concrète où une IA pourrait utiliser la méthode des k plus proches voisins pour classer un nouvel élément (par exemple dans le sport, les jeux vidéo, la santé, le commerce en ligne...).

Coup de pouce

Avec $k = 10$, on prendrait toujours la catégorie majoritaire parmi **toutes** les chansons connues (ici 5 Pop et 5 Rap : il y aurait même une égalité!), quelle que soit la nouvelle chanson à classer. La méthode perdrait alors tout son intérêt, puisque le résultat ne dépendrait plus des caractéristiques de la nouvelle chanson.

Remarque

La méthode des k plus proches voisins est utilisée dans de nombreux systèmes de recommandation réels (musique, films, produits en ligne), mais aussi en reconnaissance d'images ou en diagnostic médical assisté. Sa force : elle est simple à comprendre et ne nécessite pas de longue phase d'apprentissage. Sa limite : elle devient lente lorsque le nombre de données d'apprentissage est très grand, car il faut calculer une distance avec **chacune** d'entre elles pour classer un seul nouvel élément.