Analyse des Séries Temporelles avec Dynamic Time Warping (DTW)

Travaux Pratiques — Analyse et Alignement de Séries Temporelles

Matricule	Noms & Prénoms	Email
24F2456	ESSUTHI MBANGUE ANGE ARMEL	ange.essuthi@facsciences-uy1.cm
22W2188	BOKOU-BOUNA-ANGE-LARISSA	ange.bokou@facsciences-uy1.cm
21T2437	DJAMPA MBIANGANG PLATINY CABREL	platiny.djampa@facsciences-uy1.cm

Université de Yaoundé 1

Département Informatique

Année académique 2025-2026

Objectifs du TP

Objectif général : Comprendre comment comparer deux séries temporelles désynchronisées à l'aide du Dynamic Time Warping (DTW).

Objectifs spécifiques :

- Définir ce qu'est une série temporelle : univariée vs multivariée.
- Comprendre pourquoi la distance euclidienne échoue en cas de décalage temporel.
- Expliquer le principe du DTW : alignement optimal, étirement et compression.
- Réaliser un calcul DTW pas-à-pas (cas simple et cas avec étirements).
- Étendre la méthode au cas multivarié (exemple : vecteurs MFCC).
- Comparer DTW et distance euclidienne, et discuter leurs forces et limites.

Analogie intuitive : pourquoi le DTW ?

Problème : Deux séries peuvent représenter le **même phénomène** mais être observées **à des vitesses différentes**.

Analogie des coureurs :

- Deux coureurs suivent exactement **la même piste**.
- Le premier démarre vite, puis ralentit.
- Le second démarre lentement, puis accélère.
- Si on compare leur position au **même instant**, ils semblent différents.

Mais : si on compare leurs positions en laissant le temps se **dilater ou se comprimer**, on voit qu'ils suivent **la même trajectoire**.

Idée clé du DTW : Aligner intelligemment les points de deux séries, même si elles ne sont pas synchronisées dans le temps.

Définition

- Une série temporelle est une suite de valeurs ordonnées dans le temps.
- Exemple univarié : température $T(t)$ jour après jour.
- Exemple multivarié : signal audio représenté par plusieurs coefficients (ex. MFCC).

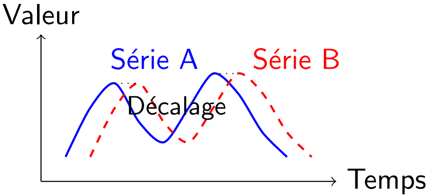
Problématique

- Deux séries peuvent représenter le **même phénomène** mais être **désynchronisées**.
- Une distance naïve (euclidienne) les considère alors "différentes".

$\rightarrow$ **Motivation : comment comparer deux séries qui ne sont pas alignées dans le temps ?**

1. **Sensibilité au décalage temporel :**
   - La distance euclidienne compare strictement point-à-point : $d_E(A, B) = \sum_i (A_i - B_i)^2$.
   - Si un événement (pic) est à $t = 3$ dans A et à $t = 5$ dans B, la comparaison directe oppose un pic à un creux $\rightarrow$ grande distance.

Illustration : deux séries identiques mais décalées



2. **Incapable de gérer :**
   - **Étirements** (une partie de la série répétée),
   - **Compressions** (partie abrégée),
   - **Décalages horizontaux** (même forme, temps différent).

Principe du DTW (1/2)

Idée intuitive : on autorise l'alignement d'un point d'une série à plusieurs points de l'autre (étirer/compresser le temps) pour minimiser le coût total d'alignement.

Soient $A = (a_1, \ldots, a_m)$ et $B = (b_1, \ldots, b_n)$.
Définissons la distance locale :

$$c(i, j) = d(a_i, b_j) \quad (\text{ex. } (a_i - b_j)^2 \text{ ou } \|a_i - b_j\|^2 \text{ pour vecteurs}).$$

Principe du DTW (2/2)

La matrice cumulative $D(i,j)$ est construite par récurrence :

$$\begin{cases} D(1,1) = c(1,1), \\ D(i,1) = c(i,1) + D(i-1,1), \quad i = 2 \ldots m, \\ D(1,j) = c(1,j) + D(1,j-1), \quad j = 2 \ldots n, \\ D(i,j) = c(i,j) + \min\{D(i-1,j),\, D(i,j-1),\, D(i-1,j-1)\}, \end{cases}$$

La distance DTW finale = $D(m,n)$.
Le *warp path* se récupère par backtracking depuis (m,n) en choisissant à chaque étape le voisin (haut/gauche/diag) qui a fourni le minimum.

Exemple univarié : calcul pas-à-pas (1/2)

Énoncé : Comparons deux séries temporelles :

$$A = [1,\ 3,\ 4] \qquad B = [1,\ 2,\ 4]$$

Étape 1 : Matrice des distances locales

$$c(i,j) = |A_i - B_j|$$

$c(i,j)$	$b_1 = 1$	$b_2 = 2$	$b_3 = 4$
$a_1 = 1$	0	1	3
$a_2 = 3$	2	1	1
$a_3 = 4$	3	2	0

Exemple :

- $c(1,1) = |1 - 1| = 0$
- $c(1,2) = |1 - 2| = 1$
- $c(1,3) = |1 - 4| = 3$

Exemple univarié : calcul pas-à-pas (2/2)

Rappel : $A = [1, 3, 4]$, $B = [1, 2, 4]$

Étape 2 : Matrice cumulative $D(i, j)$

$$D(i, j) = c(i, j) + \min\{D(i-1, j), D(i, j-1), D(i-1, j-1)\}$$

$D(i, j)$	$b_1 = 1$	$b_2 = 2$	$b_3 = 4$
$a_1 = 1$	0	1	4
$a_2 = 3$	2	1	2
$a_3 = 4$	5	3	1

Calculs :

- $D(1, 1) = c(1, 1) = 0$
- $D(1, 2) = c(1, 2) + D(1, 1) = 1 + 0 = 1$
- $D(2, 1) = c(2, 1) + D(1, 1) = 2 + 0 = 2$
- $D(2, 2) = c(2, 2) + \min\{1, 2, 0\} = 1 + 0 = 1$

Exemple univarié : résultat (1/2)

Résultat final :

$$\text{DTW}(A, B) = D(3, 3) = 1$$

Matrice cumulative final e :

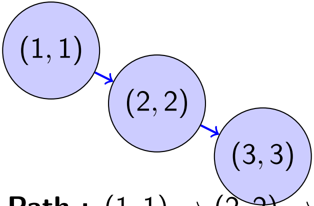
$D(i,j)$	$b_1 = 1$	$b_2 = 2$	$b_3 = 4$
$a_1 = 1$	0	1	4
$a_2 = 3$	2	1	2
$a_3 = 4$	5	3	**1**

Méthode de backtracking :

- Partir de $(m, n) = (3, 3)$
- Remonter vers $(1, 1)$ en suivant le chemin minimal
- À chaque étape, choisir le voisin avec la plus petite valeur D

Exemple univarié : résultat (2/2)

Backtracking (Warp Path) :



Warp Path : $(1,1) \to (2,2) \to (3,3)$

Correspondances :

$\boxed{a_1 \leftrightarrow b_1}$ $\boxed{a_2 \leftrightarrow b_2}$ $\boxed{a_3 \leftrightarrow b_3}$

Interprétation :

- Chemin diagonal $\to$ pas d'étirement temporel
- Alignement parfait entre les séries
- Distance DTW minimale (1)

Énoncé : Montrer un étirement temporel où une même valeur de A s'aligne sur plusieurs valeurs de B.
Choix :

$$A = [1,\ 2,\ 3] \quad \text{(plus court)}$$

$$B = [1,\ 1,\ 2,\ 2,\ 3] \quad \text{(plus long, répétitions = étirements)}$$

Étape 1 — Matrice des distances locales $c(i,j) = |A_i - B_j|$

$c(i,j)$	$b_1 = 1$	$b_2 = 1$	$b_3 = 2$	$b_4 = 2$	$b_5 = 3$
$a_1 = 1$	0	0	1	1	2
$a_2 = 2$	1	1	0	0	1
$a_3 = 3$	2	2	1	1	0

Exemple univarié : cas avec étirements (1/3)

Problème : Montrer comment DTW gère l'étirement temporel.

Séries à comparer :

$$A = [1,\ 2,\ 3] \quad \text{(courte, 3 points)}$$

$$B = [1,\ 1,\ 2,\ 2,\ 3] \quad \text{(longue, 5 points avec répétitions)}$$

Interprétation :

- B est une version "ralentie" de A
- Les valeurs 1 et 2 sont répétées dans B
- La série A doit s'étirer pour s'aligner sur B

Matrice des distances locales :

$$c(i,j) = |A_i - B_j|$$

Exemple univarié : cas avec étirements (2/3)

Étape 1 : Matrice des distances locales $c(i,j) = |A_i - B_j|$

$c(i,j)$	$b_1 = 1$	$b_2 = 1$	$b_3 = 2$	$b_4 = 2$	$b_5 = 3$
$a_1 = 1$	**0**	**0**	1	1	2
$a_2 = 2$	1	1	**0**	**0**	1
$a_3 = 3$	2	2	1	1	**0**

Observations :

- a_1 a distance 0 avec b_1 et b_2 (deux fois la valeur 1)
- a_2 a distance 0 avec b_3 et b_4 (deux fois la valeur 2)
- a_3 a distance 0 avec b_5 (valeur 3)
- Les zéros montrent où l'alignement est parfait

Exemple univarié : cas avec étirements (3/3)

Étape 2 : Matrice cumulative $D(i,j)$

Formule de récurrence :

$$D(i,j) = c(i,j) + \min\{D(i-1,j), D(i,j-1), D(i-1,j-1)\}$$

$D(i,j)$	$b_1 = 1$	$b_2 = 1$	$b_3 = 2$	$b_4 = 2$	$b_5 = 3$
$a_1 = 1$	0	0	1	2	4
$a_2 = 2$	1	1	0	0	1
$a_3 = 3$	3	3	1	1	**0**

Calculs clés :

- $D(1,1) = 0, \ D(1,2) = 0 + 0 = 0$
- $D(2,3) = 0 + \min\{1,1,0\} = 0$
- $D(3,5) = 0 + \min\{1,1,0\} = 0$

Résultat : $DTW(A,B) = D(3,5) = 0$

Exemple avec étirements : backtracking (1/2)

Étape 3 : Backtracking depuis $(3, 5)$

Chemin Départ Arrivée

Matrice $D(i, j)$ **:**

0	0	1	2	4
1	1	0	0	1
3	3	1	1	0

Étapes du backtracking :

$$(3, 5) \rightarrow (2, 4) \rightarrow (2, 3) \rightarrow (1, 2) \rightarrow (1, 1)$$

Exemple avec étirements : backtracking (2/2)

Warp Path final (ordre chronologique) :

$$(1,1) \to (1,2) \to (2,3) \to (2,4) \to (3,5)$$

Représentation graphique du warp path :

$$(1,1) \to (1,2) \to (2,3) \to (2,4) \to (3,5)$$

$$a_1 \leftrightarrow b_1 \quad a_1 \leftrightarrow b_2 \quad a_2 \leftrightarrow b_3 \quad a_2 \leftrightarrow b_4 \quad a_3 \leftrightarrow b_5$$

Alignement obtenu :

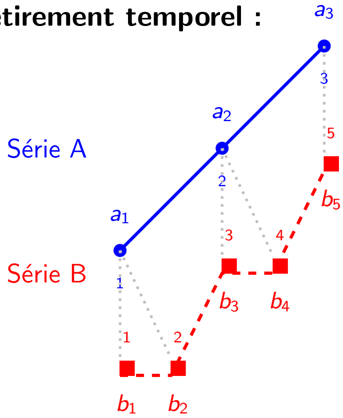
- $a_1 \leftrightarrow b_1$ et $a_1 \leftrightarrow b_2$ (étirement)
- $a_2 \leftrightarrow b_3$ et $a_2 \leftrightarrow b_4$ (étirement)
- $a_3 \leftrightarrow b_5$ (direct)

Exemple avec étirements : visualisation

Résumé de l'exemple :

- **Série A :** $[1, 2, 3]$ (courte, 3 points)
- **Série B :** $[1, 1, 2, 2, 3]$ (longue, 5 points avec répétitions)
- **DTW final :** $D(3, 5) = 0$ (alignement parfait)

Représentation de l'étirement temporel :



Exemple avec étirements : alignements

Alignements DTW identifiés :

Point Série A	Point(s) Série B	Type
a_1 (valeur 1)	b_1 et b_2 (valeur 1)	**Étirement**
a_2 (valeur 2)	b_3 et b_4 (valeur 2)	**Étirement**
a_3 (valeur 3)	b_5 (valeur 3)	**Direct**

Interprétation

- La série B est une version **ralentie** de la série A
- DTW **répète** certains points pour compenser la différence de vitesse
- L'étirement permet d'aligner des séries de **longueurs différentes**

Conclusion

DTW réussit à **étirer** la série courte (A) pour l'aligner avec la série longue (B)
en faisant correspondre chaque point de A avec **un ou plusieurs** points de B ayant la même valeur.

DTW : gestion des étirements et compressions

Pourquoi la distance euclidienne échoue ici ?

- Si une portion de la série A est plus longue que la même portion dans B, l'alignement point-à-point devient impossible.
- La comparaison brute crée des écarts artificiels.

Ce que DTW autorise :

- Un point de A peut être aligné avec **plusieurs** points de B (étirement).
- Plusieurs points de A peuvent être alignés avec **un seul** point de B (compression).

Intuition : DTW adapte le temps pour trouver la correspondance la plus naturelle.

DTW multivarié : principe général (1/3)

Motivation : Beaucoup de signaux réels sont décrits par un **vecteur** à chaque instant :

$$\mathbf{x}(t) = (x_1(t), x_2(t), \ldots, x_d(t))$$

Exemples concrets :

Signal audio (MFCC)

Chaque fenêtre = vecteur de 13 coefficients

Capteur d'accélération

Chaque instant = vecteur 3D (a_x, a_y, a_z)

Données financières

Chaque jour = vecteur (prix, volume, volatilité)

Exemple : $(150.5, 1000000, 0.15)$

DTW multivarié : principe général (2/3)

Idée fondamentale :

- Le DTW ne change pas structurellement
- Seule la **distance locale** devient une distance entre vecteurs
- L'algorithme reste identique

Distance euclidienne multivariée :

$$c(i,j) = \|\mathbf{x}_i - \mathbf{y}_j\|_2 = \sqrt{\sum_{k=1}^{d} (x_{ik} - y_{jk})^2}$$

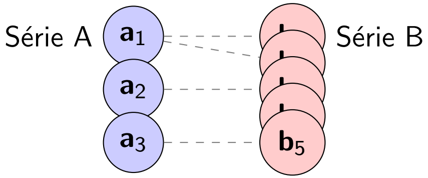
Formulation matricielle (identique au cas univarié) :

Récurrence DTW

$$D(i,j) = c(i,j) + \min\{D(i-1,j), D(i,j-1), D(i-1,j-1)\}$$

DTW multivarié : principe général (3/3)

Visualisation du processus :



Exemple multivarié : MFCC 2D simplifié (1/3)

Scénario : Comparaison de deux signaux audio représentés par des MFCC simplifiés.

Séries multivariées (2 dimensions) :

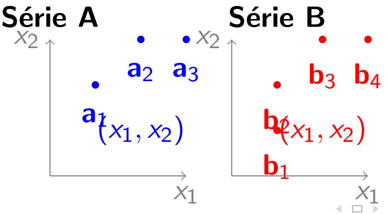
Série A (3 points):

$$\mathbf{a}_1 = (1, 2)$$
$$\mathbf{a}_2 = (2, 3)$$
$$\mathbf{a}_3 = (3, 3)$$

Série B (4 points):

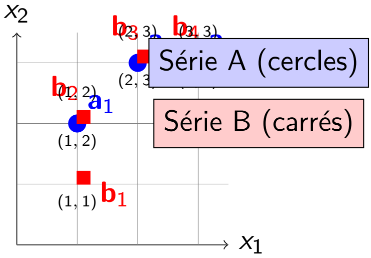
$$\mathbf{b}_1 = (1, 1)$$
$$\mathbf{b}_2 = (1, 2)$$
$$\mathbf{b}_3 = (2, 3)$$
$$\mathbf{b}_4 = (3, 3)$$

Représentation des séries :



Exemple multivarié : MFCC 2D simplifié (2/3)

Représentation dans l'espace 2D :



Observation :

- $\mathbf{a}_1$ et $\mathbf{b}_2$ ont mêmes coordonnées $(1, 2)$
- $\mathbf{a}_2$ et $\mathbf{b}_3$ ont mêmes coordonnées $(2, 3)$
- $\mathbf{a}_3$ et $\mathbf{b}_4$ ont mêmes coordonnées $(3, 3)$
- $\mathbf{b}_1 = (1, 1)$ n'a pas d'équivalent direct dans A

Exemple multivarié : MFCC 2D simplifié (3/3)

Étape 1 : Calcul des distances locales

Distance euclidienne 2D :

$$c(i,j) = \sqrt{(a_{i1} - b_{j1})^2 + (a_{i2} - b_{j2})^2}$$

$c(i,j)$	$\mathbf{b}_1 = (1,1)$	$\mathbf{b}_2 = (1,2)$	$\mathbf{b}_3 = (2,3)$	$\mathbf{b}_4 = (3,3)$
$\mathbf{a}_1 = (1,2)$	1.00	**0.00**	1.41	2.24
$\mathbf{a}_2 = (2,3)$	2.24	1.41	**0.00**	1.00
$\mathbf{a}_3 = (3,3)$	2.83	2.24	1.00	**0.00**

Étape 2 : Matrice cumulative $D(i,j)$

$D(i,j)$	$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$	$\mathbf{b}_4$
$\mathbf{a}_1$	1.00	0.00	1.41	3.65
$\mathbf{a}_2$	3.24	1.41	**0.00**	1.00
$\mathbf{a}_3$	6.07	3.65	1.00	**0.00**

Warp Path : $(1,2) \to (2,3) \to (3,4)$

Étape 3 : Backtracking depuis $D(3,4) = 0.00$

Matrice D **:**

	$\mathbf{b}_1$	$\mathbf{b}_2$	$\mathbf{b}_3$	$\mathbf{b}_4$
$\mathbf{a}_1$	1.00	0.00	1.41	3.65
$\mathbf{a}_2$	3.24	1.41	0.00	1.00
$\mathbf{a}_3$	6.07	3.65	1.00	0.00

Départ
Chemin
Flèche

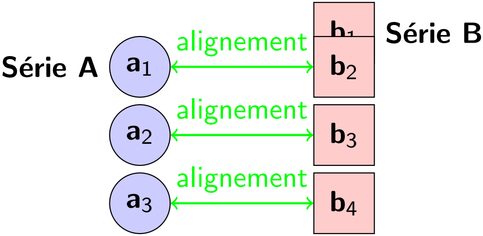
Processus :

- Départ : $(3,4)$ avec $D = 0.00$
- Voisins : $(2,4) = 1.00$, $(3,3) = 1.00$, $(2,3) = 0.00$
- Choix : $(2,3)$ (minimum)
- Voisins de $(2,3)$: $(1,3) = 1.41$, $(2,2) = 1.41$, $(1,2) = 0.00$
- Choix : $(1,2)$

Warp Path obtenu :

$$(1,2) \rightarrow (2,3) \rightarrow (3,4)$$

Représentation du chemin d'alignement :



$\boxed{\textbf{Warp Path :} \ (1,2) \rightarrow (2,3) \rightarrow (3,4)}$

Alignement final :

$\boxed{a_1 \leftrightarrow b_2}$ $\boxed{a_2 \leftrightarrow b_3}$ $\boxed{a_3 \leftrightarrow b_4}$

Résumé des résultats :

- **DTW multivarié :** $D(3,4) = 0.00$
- **Warp Path :** $(1,2) \to (2,3) \to (3,4)$
- **Alignement :**
  - $\mathbf{a}_1 = (1,2) \leftrightarrow \mathbf{b}_2 = (1,2)$
  - $\mathbf{a}_2 = (2,3) \leftrightarrow \mathbf{b}_3 = (2,3)$
  - $\mathbf{a}_3 = (3,3) \leftrightarrow \mathbf{b}_4 = (3,3)$

Interprétation :

Ce que DTW a fait :

- A sauté $\mathbf{b}_1 = (1,1)$ qui n'avait pas d'équivalent dans A
- A aligné chaque point de A avec son équivalent exact dans B
- A trouvé un chemin optimal de distance totale nulle

Exemple multivarié : interprétation (2/2)

Signification dans le contexte audio :

- Les deux signaux ont les mêmes caractéristiques MFCC
- Ils sont identiques mais de longueurs différentes
- $\mathbf{b}_1$ pourrait être un bruit ou un artefact
- DTW ignore les éléments non pertinents

Comparaison : DTW vs Distance Euclidienne (1/3)

Distance Euclidienne multivariée :

$$d_E(A, B) = \sum_{i=1}^{\min(m,n)} \|\mathbf{a}_i - \mathbf{b}_i\|_2$$

- Compare strictement point à point
- Ignore les décalages temporels
- Exige que les séries aient même longueur

DTW multivarié :

- Autorise l'étirement/compression temporel
- Compare points similaires, pas nécessairement aux mêmes positions
- Gère des séries de longueurs différentes

Application à notre exemple :

$$A = [(1,2),(2,3),(3,3)] \quad B = [(1,1),(1,2),(2,3),(3,3)]$$

Comparaison	**Euclidienne**	**DTW**
Alignement	Point à point	Optimisé
Longueurs	Doivent être égales	Peuvent différer
Résultat	3.41	0.00

Comparaison : DTW vs Distance Euclidienne (3/3)

Calcul détaillé des distances :

Distance Euclidienne :

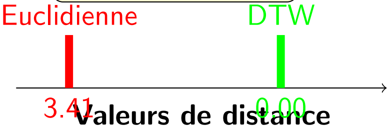
- $\|\mathbf{a}_1 - \mathbf{b}_1\| = \sqrt{(1-1)^2 + (2-1)^2} = 1.00$
- $\|\mathbf{a}_2 - \mathbf{b}_2\| = \sqrt{(2-1)^2 + (3-2)^2} = 1.41$
- $\|\mathbf{a}_3 - \mathbf{b}_3\| = \sqrt{(3-2)^2 + (3-3)^2} = 1.00$
- **Total :** 3.41

DTW multivarié :

- Aligne $\mathbf{a}_1$ avec $\mathbf{b}_2$: 0.00
- Aligne $\mathbf{a}_2$ avec $\mathbf{b}_3$: 0.00
- Aligne $\mathbf{a}_3$ avec $\mathbf{b}_4$: 0.00
- **Total :** 0.00

Représentation graphique :

Différence : 3.41



DTW - Points forts :

- ✓Flexible : séries de longueurs différentes
- ✓Robuste aux décalages temporels
- ✓Excellente reconnaissance de formes
- ✓Applicable aux données multivariées

DTW - Points faibles :

- 55 Complexité élevée : $O(n \times m)$
- 55 Calcul coûteux pour grandes séries
- 55 Paramètres à ajuster (fenêtre, contraintes)

Comparaison : DTW vs Euclidienne (2/2)

Distance Euclidienne - Points forts :

- ✓Simple et rapide : $O(n)$
- ✓Interprétation intuitive
- ✓Peu de paramètres
- ✓Efficace pour grandes séries

Distance Euclidienne - Points faibles :

- Séries doivent avoir même longueur
- Très sensible aux décalages
- Ne gère pas les variations de vitesse

Euclidienne est idéale pour : Données synchronisées, séries alignées temporellement, applications en temps réel

Tableau récapitulatif :

	DTW	**Euclidienne**
Complexité	Élevée	Faible
Flexibilité	Excellente	Limitée
Vitesse	Lente	Rapide
Applications	Séquences désynchronisées	Données alignées

Conclusion du TP

Récapitulatif des points clés :

- ✓**DTW** est une méthode puissante pour comparer des séries temporelles désynchronisées
- ✓Il permet l'étirement/compression temporel via un warp path optimal
- ✓Applicable aux séries univariées et multivariées
- ✓Surpasse la distance euclidienne dans les cas réels avec variations de timing

Applications pratiques vues :

- Reconnaissance vocale (MFCC + DTW)
- Analyse de séries financières
- Traitement de signaux biométriques
- Alignement de séquences biologiques

Message principal : Le DTW est un outil essentiel quand la similarité de forme est plus importante que la synchronisation temporelle exacte.

Références (1/2)

Ressources principales :

1. **Articles académiques :**
   - Sakoe, H., & Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition
   - Berndt, D. J., & Clifford, J. (1994). Using dynamic time warping to find patterns in time series
2. **Livres et cours :**
   - Keogh, E., & Ratanamahatana, C. A. (2005). Exact indexing of dynamic time warping
   - Université de Yaoundé 1 - Cours d'Analyse de Séries Temporelles
3. **Ressources en ligne :**
   - Müller, M. (2007). Information Retrieval for Music and Motion
   - Wikipedia : Dynamic Time Warping
   - Towards Data Science : Introduction to DTW

Collaborateurs et outils :

- **Assistance IA :** ChatGPT (OpenAI) Et Deepseek pour la génération de code LaTeX et explications
- **Rédacteur principal :** Étudiants Essuthi/Bokou/Djampa - Université de Yaoundé 1
- **Encadrement :** Pr. TSOPZE NORBERT
- **Outils :** Overleaf, TikZ pour les visualisations

Remerciements : Nous remercions les ressources en ligne ainsi que les outils d'assistance basés sur l'intelligence artificielle qui ont contribué à l'enrichissement de ce travail pratique.

Disponible sur GitHub

https://github.com/Time-series-group/Tp-groupe-dtw

- Notebook Jupyter complet
- Données et visualisations
- Documentation et cahier de suivi
- Code open-source (licence MIT)