

# Exercice TP : Prédiction du Marché Boursier à l'aide des Modèles Cachés de Markov (HMM)

---

## Contexte :

Le marché boursier est influencé par de nombreux facteurs économiques et peut être modélisé comme un système stochastique. Les modèles de Markov cachés (HMM) sont souvent utilisés pour détecter des tendances cachées dans des séries temporelles complexes, telles que les fluctuations des prix des actions. Dans ce TP, vous allez explorer l'utilisation des HMM pour prédire les mouvements de prix des actions en utilisant un jeu de données historique.

## Objectifs pédagogiques :

- Comprendre les bases des modèles cachés de Markov (HMM) et leur application dans la prédiction de séries temporelles.
- Apprendre à utiliser les bibliothèques Python pour modéliser et prédire les prix des actions.
- Explorer l'importance de la normalisation des données et de la réduction de dimensions.
- Mettre en œuvre un modèle prédictif et évaluer ses performances.

## Remarque :

Le dataset utilisé dans cet exercice est constitué des données historiques des actions. La dataset **^GSPC** est fourni sous forme d'un fichier csv, ou bien télécharger votre dataset depuis la source Yahoo Finance avec cette condition : **start 2000-01-01 end 2023-12-31**.

## Partie 1 : Préparation des Données

### 1. Importation des données :

- Importez un dataset de données historiques des actions (Yahoo Finance). Les colonnes importantes incluent la Date, Open, Close, High, Low, et Volume.

### 2. Visualisation des données :

- Affichez les données dans un graphique (par exemple, l'évolution des prix de clôture au cours du temps).
- Que remarquez-vous quant aux fluctuations de prix ?

### 3. Prétraitement des données :

- Remplissez les valeurs manquantes s'il y en a.
- Normalisez les données (min-max scaling ou z-score normalization).
- Divisez les données en un ensemble d'entraînement (train) et de test (test).

## Partie 2 : Implémentation du HMM

### 4. Construction du Modèle :

- Utilisez la bibliothèque `hmmlearn` ou `sklearn` pour définir un modèle HMM avec un nombre de  $n$  états cachés que vous jugez pertinent (essayez différentes valeurs et comparez les résultats).

### 5. Analyse des États Cachés :

- Après l'entraînement, interprétez les états cachés en observant leur distribution.
- À quoi correspondent ces états en termes de comportement de marché (hausse, baisse, stagnation) ?

## Partie 3 : Prédiction

### 6. Prévion des tendances futures :

- Utilisez le modèle HMM pour prédire les futurs mouvements du marché sur l'ensemble de test. Comparez les prédictions avec les valeurs réelles des prix de clôture.

### 7. Évaluation des Performances :

- Évaluez la performance de votre modèle en utilisant des métriques comme l'erreur quadratique moyenne (RMSE) ou l'erreur absolue moyenne (MAE).
- Tracez un graphique comparant les prédictions aux données réelles.

## Partie 4 : Discussion

### 8. Analyse des Résultats :

- Quelle est la précision des prédictions de votre modèle ? Dans quelle mesure les prédictions sont-elles proches de la réalité ?
- Comment amélioreriez-vous le modèle ? Essayez différentes configurations (nombre d'états, méthode de normalisation, etc.).

## Ressources Utiles :

- [Documentation hmmlearn](#):
- [Introduction aux modèles cachés de Markov](#):
- [Séries temporelles avec Python](#) :