

ISEN

ALL IS DIGITAL!

NANTES

**PROJET BIG DATA : CONCEVOIR ET DEVELOPPER UNE
APPLICATION D'ETUDE DES DONNEES AIS**

Trinôme 13 : Ewan Ménahéze, Clément Bolomey, Matthieu Lebeau

Nombre de mots : 5777

Date rendu : 20/06/2025

I. Table des matières

II.	Introduction	3
III.	Besoin client 1 : Visualisation sur carte	4
A.	Première tentative : un clustering naïf basé sur des agrégats statistiques	4
B.	Tentative de clustering basé sur la forme des trajectoires (DTW + CAH)	5
C.	Tentatives de clustering avec DBSCAN et par CAH	6
D.	Clustering avancé : caractéristiques dérivées des trajectoires	7
1)	Extraction de descripteurs trajectoriels :	7
2)	Normalisation et sélection du nombre optimal de clusters :	7
3)	Résultats visuels : cartes des clusters.....	8
4)	Histogramme des Clusters	10
E.	Déploiement d'un outil de prédiction basé sur le modèle de clustering	10
IV.	Besoin client 2 : Modèle de prédiction du type de navire	12
A.	Analyse exploratoire des variables numériques	12
B.	Premier modèle erroné.....	13
C.	Création de nouvelles variables explicatives	14
D.	Évaluation des modèles	15
1)	Modèle Random Forest sans la variable Cargo	15
2)	Modèle Random Forest avec la variable Cargo	16
3)	Comparaison avec d'autres classifieurs.....	17
4)	Analyse des erreurs de prédiction	17
E.	Script de prédiction du type de navire	19
V.	Besoin client 3 : Prédiction de la trajectoire des navires.....	21
A.	Préparation du modèle et des données.....	21
1)	Contexte et justification du RandomForest	21
2)	Préparation du modèle et enrichissement des données.....	21
B.	Implémentation du script et interprétation des résultats	23
VI.	Bonus : Utilisation d'une vrai DB AIS (25-31/05/2024)	27
A.	Préparation des données	27
B.	Analyse par clustering.....	27
C.	Prédiction du type de navire (VesselType).....	29
D.	Résultats du classificateur	29
VII.	ANNEXE	31

II. Introduction

Ce rapport s'inscrit dans une démarche de valorisation des données AIS (Automatic Identification System) collectées dans le Golfe du Mexique. Ces données, issues de la navigation réelle des navires, constituent une ressource précieuse pour modéliser, analyser et anticiper les comportements maritimes. Notre étude se structure autour de trois problématiques principales, formulées sous forme de besoins clients.

Le premier besoin concerne l'exploration des trajectoires et la mise en évidence de comportements typiques via des méthodes de clustering non supervisées.

Le deuxième besoin vise à prédire le type d'un navire (ex. tanker, cargo, ferry) à partir de ses caractéristiques physiques et de ses données AIS statiques.

Enfin, le troisième besoin porte sur la prédiction de la position future d'un navire à court terme, pour des applications liées à la sécurité maritime et à l'optimisation portuaire.

Notre approche mêle analyse exploratoire, modélisation supervisée et validation croisée, avec un accent particulier sur la robustesse des méthodes et l'interprétabilité des résultats. L'utilisation d'une base AIS réelle et récente, en complément de la base initialement fournie, nous a permis de tester la généralisation des modèles et de renforcer la validité opérationnelle des conclusions.

III. Besoin client 1 : Visualisation sur carte

A. Première tentative : un clustering naïf basé sur des agrégats statistiques

Dans un premier temps, nous avons exploré une approche simple visant à regrouper les navires en clusters sur la base d'agrégats statistiques calculés à partir de leur trajectoire. Pour cela, nous avons sélectionné les variables directement liées à la navigation : latitude (LAT), longitude (LON), cap réel sur le sol (COG), direction (Heading), et vitesse (SOG). Pour chaque navire, nous avons alors extrait la moyenne et l'écart-type de chacune de ces variables, ce qui nous a permis d'obtenir un vecteur descriptif de 10 dimensions par navire.

L'idée était que ces statistiques pouvaient suffire à capturer le comportement global des navires, en termes de position moyenne, de direction générale et de régularité de mouvement. Nous avons ensuite appliqué un algorithme K-means pour partitionner l'ensemble des navires en 4 groupes, puis réattribué les labels de cluster à chaque point de la trajectoire, afin de visualiser l'ensemble des itinéraires sur une carte avec un code couleur représentant les différents clusters.

Le résultat de cette première visualisation est présenté ci-dessous :

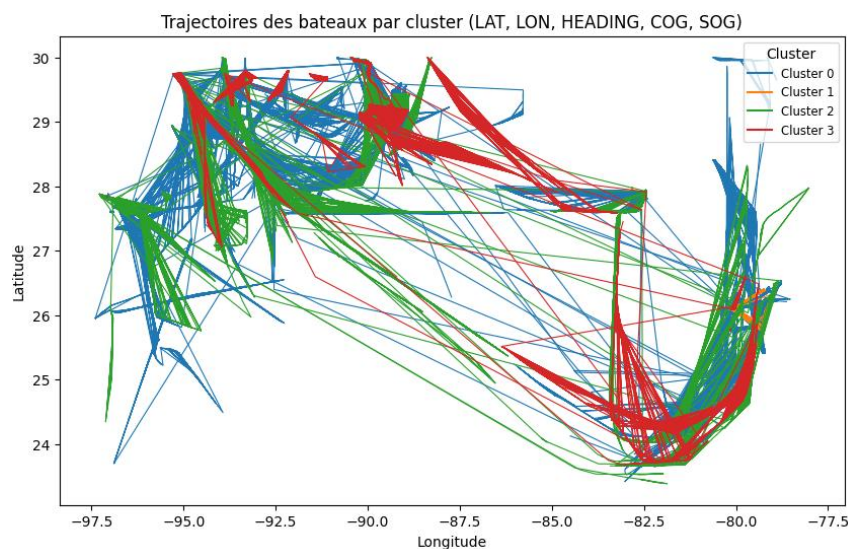


Figure 1 : Trajectoires des bateaux par cluster – clustering naïf

Bien que cette approche ait permis une première séparation grossière des comportements, nous l'avons rapidement jugée insuffisante. Les clusters ainsi obtenus ne correspondaient pas à des logiques géographiques ou de navigation réellement distincte : certains groupes étaient trop dispersés spatialement, et des navires aux trajectoires très différentes se retrouvaient dans le même cluster. Cela s'explique notamment par la perte d'information temporelle et dynamique dans l'agrégation

statistique : deux navires peuvent avoir une position moyenne similaire tout en suivant des itinéraires radicalement opposés.

Ainsi, cette première méthode a été considérée comme exploratoire, mais non concluante pour répondre au besoin d'identification de schémas de navigation typiques. Elle nous a toutefois permis de poser les bases de la démarche de clustering, et a motivé la recherche d'une méthode plus fine, intégrant davantage de caractéristiques de la trajectoire complète.

B. Tentative de clustering basé sur la forme des trajectoires (DTW + CAH)

Afin d'explorer d'autres pistes de regroupement des trajectoires, nous avons testé une méthode de classification non supervisée basée sur la similarité de forme entre les trajectoires complètes.

Cette approche repose sur l'algorithme DTW (Dynamic Time Warping) pour mesurer les distances entre séries temporelles multivariées, et sur une classification ascendante hiérarchique (CAH) appliquée sur la matrice des distances obtenues.

Le DTW est particulièrement adapté aux données de trajectoires, car il permet de comparer des séquences de points ayant potentiellement des vitesses ou durées différentes, en autorisant des déformations temporelles. Contrairement à notre première méthode, qui ne considère que des indicateurs globaux (aire, centre, etc.), le DTW tient compte de la dynamique spatiale complète du déplacement.

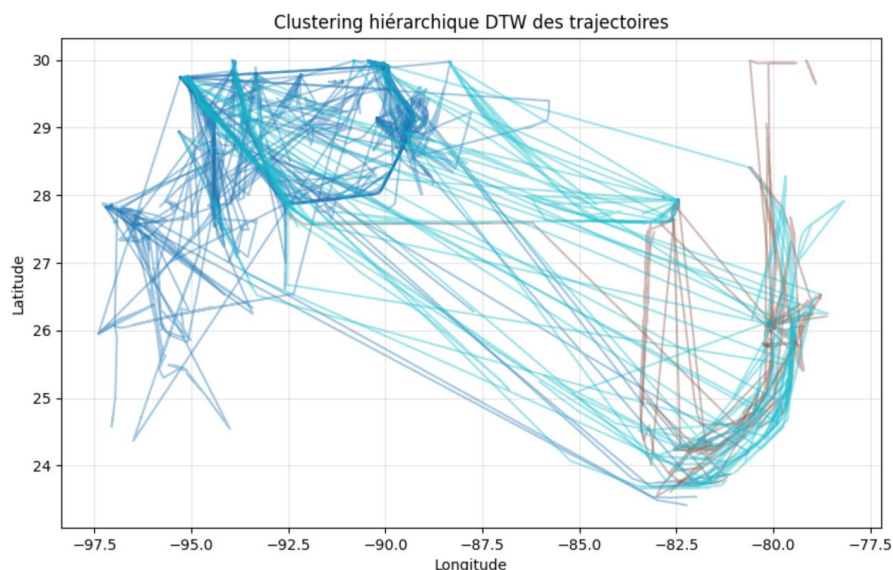


Figure 2 : Trajectoires des bateaux avec 3 clusters – clustering par forme de trajectoire

Le résultat, bien qu'encore perfectible, montre une meilleure cohérence spatiale dans les regroupements que notre première tentative avec KMeans sur les features géométriques brutes. Les trajets semblent ici rassemblés en fonction de leur forme

générale et non plus seulement de leurs points extrêmes. Toutefois, la méthode reste coûteuse en temps de calcul, et sensible au prétraitement des trajectoires (normalisation, interpolation, etc.).

C. Tentatives de clustering avec DBSCAN et par CAH

Par souci de rigueur méthodologique, nous avons également exploré d'autres approches de clustering, afin de comparer objectivement les performances. Nous avons ainsi testé DBSCAN, une méthode basée sur la densité, mais les résultats se sont avérés peu satisfaisants (voir carte ci-contre) : les clusters sont peu cohérents, loin du niveau des autres méthodes testées.

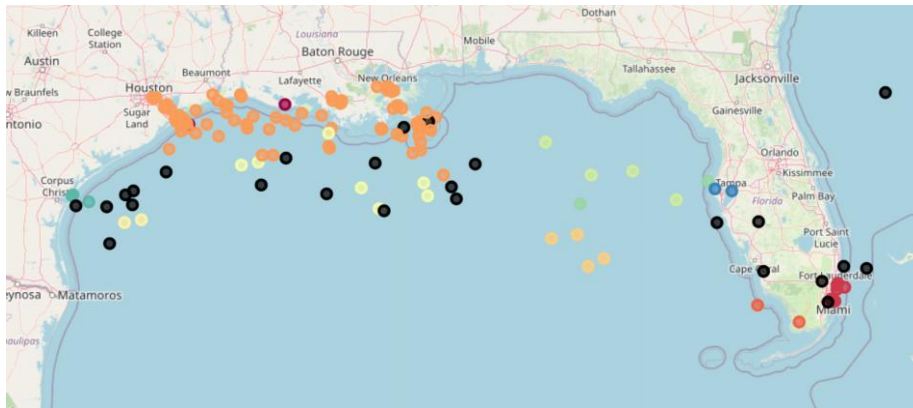


Figure 3 : Résultats Clustering DBSCAN

Nous avons ensuite appliqué une Classification Ascendante Hiérarchique (CAH), en forçant la formation de trois clusters. Les résultats étaient cette fois plus convaincants : la carte montre une segmentation relativement cohérente du trafic maritime.

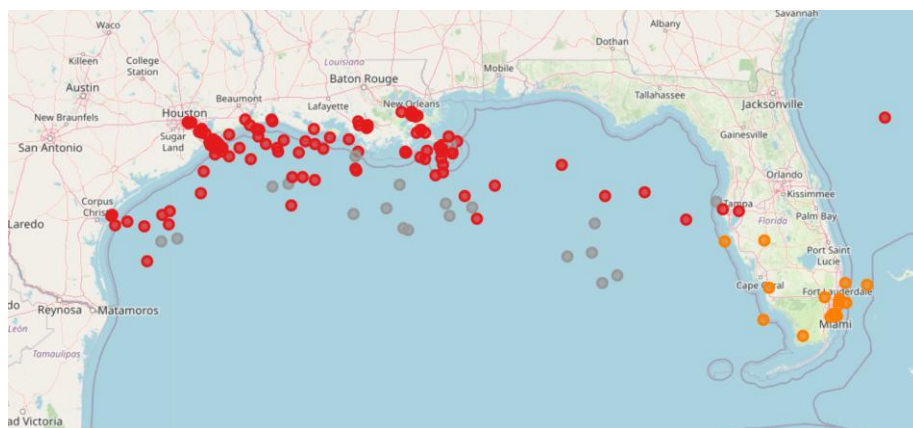


Figure 4 : Résultat Clustering CAH

Toutefois, ces méthodes n'ont pas encore atteint la qualité des résultats obtenus avec notre méthode principale, détaillée dans la section suivante.

D. Clustering avancé : caractéristiques dérivées des trajectoires

Pour donner suite aux limites identifiées dans notre première approche, nous avons développé une méthode plus robuste visant à mieux caractériser la dynamique des navires dans le Golfe du Mexique. Plutôt que de se limiter à des moyennes simples, cette méthode repose sur l'extraction de descripteurs géométriques issus des trajectoires individuelles, sans perdre totalement l'information spatiale.

1) Extraction de descripteurs trajectoriels :

Pour chaque navire, nous avons extrait une série de dix variables numériques à partir des coordonnées de sa trajectoire :

- ❖ lon_min, lon_max, lat_min, lat_max : les bornes géographiques,
- ❖ lon_ctr, lat_ctr : les coordonnées moyennes,
- ❖ lon_rng, lat_rng : les étendues de la trajectoire,
- ❖ area : une approximation de la surface couverte ($\text{lon_rng} \times \text{lat_rng}$),
- ❖ total_d : la distance totale parcourue (somme des distances euclidiennes entre points consécutifs).

Ces variables visent à résumer à la fois l'emplacement, l'ampleur spatiale et l'intensité du mouvement de chaque navire.

2) Normalisation et sélection du nombre optimal de clusters :

Les variables ainsi calculées ont été standardisées, puis un algorithme K-means a été appliqué pour identifier des regroupements. Afin de déterminer le nombre optimal de clusters, nous avons analysé plusieurs indices de qualité de partition pour des valeurs de K allant de 2 à 10 :

- ❖ **Silhouette score** : mesure la cohésion intra-cluster et la séparation inter-cluster. Il varie entre -1 et 1 et doit être maximisé.
- ❖ **Indice de Calinski-Harabasz** : ratio entre la dispersion inter-cluster et intra-cluster. Plus il est élevé, meilleure est la séparation ; il doit donc être maximisé.
- ❖ **Indice de Davies-Bouldin** : évalue la similarité entre les clusters. Un score faible indique une bonne séparation ; cet indice doit être minimisé.
- ❖ **Méthode du coude** : basée sur l'analyse de l'inertie intra-cluster. Le nombre optimal de clusters est déterminé au point où l'ajout d'un cluster supplémentaire n'apporte qu'un gain marginal (forme visuellement un « coude »).

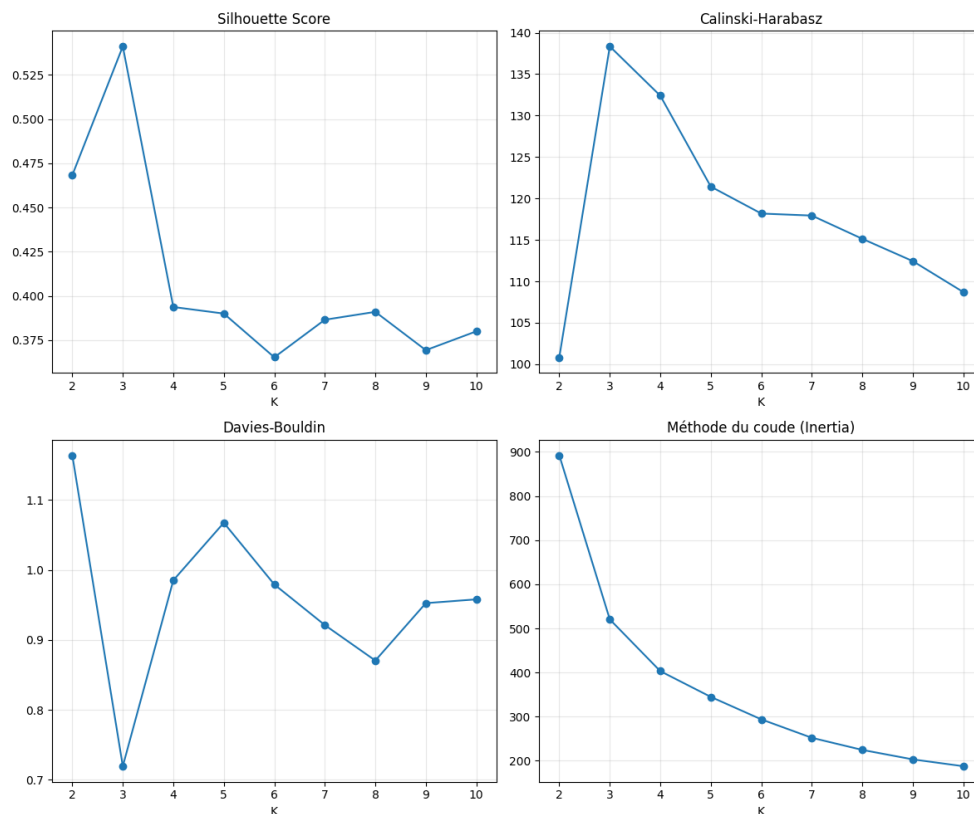


Figure 55 : Évaluation du nombre optimal de clusters selon différents indices

L'ensemble de ces critères a convergé autour de $K = 3$, qui a donc été retenu comme nombre final de clusters. Ce choix s'est appuyé sur un vote majoritaire entre les différents indices, chaque indicateur proposant sa valeur optimale.

3) Résultats visuels : cartes des clusters

Une fois les clusters assignés à chaque navire, nous avons construit deux visualisations cartographiques interactives afin de valider visuellement la pertinence des regroupements :

La première carte affiche les trajectoires individuelles, colorées selon leur cluster. La seconde carte représente les centres géographiques des trajectoires (milieu du rectangle englobant chaque itinéraire), toujours avec le même code couleur, elle permet d'observer plus distinctement les regroupements réalisés.

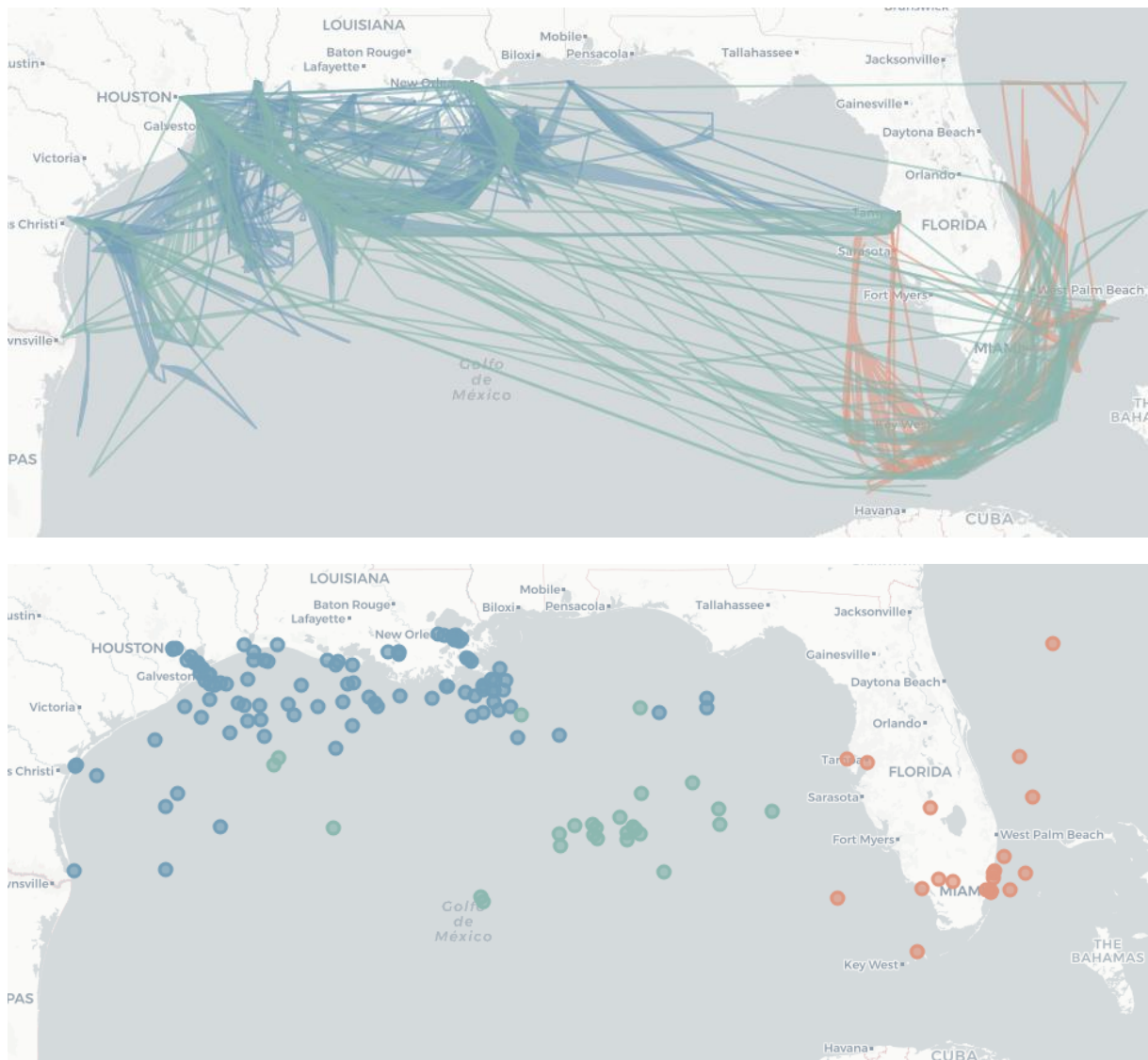


Figure 6 6 : Trajectoires de navires par cluster et Centres de trajectoire par cluster (Cluster 0 : bleu, 1 : orange, 2 : vert)

Ces représentations ont permis d'identifier des groupes de navires ayant des comportements et zones d'activité nettement différenciés. Contrairement à la première méthode, les clusters obtenus ici apparaissent beaucoup plus cohérents spatialement : certains sont concentrés au large des côtes texanes, d'autres longent la Floride ou dessinent des arcs plus larges.

4) Histogramme des Clusters

Afin d'approfondir notre analyse des clusters, nous avons également représenté la distribution des types de navires au sein de chaque cluster à l'aide d'histogrammes :

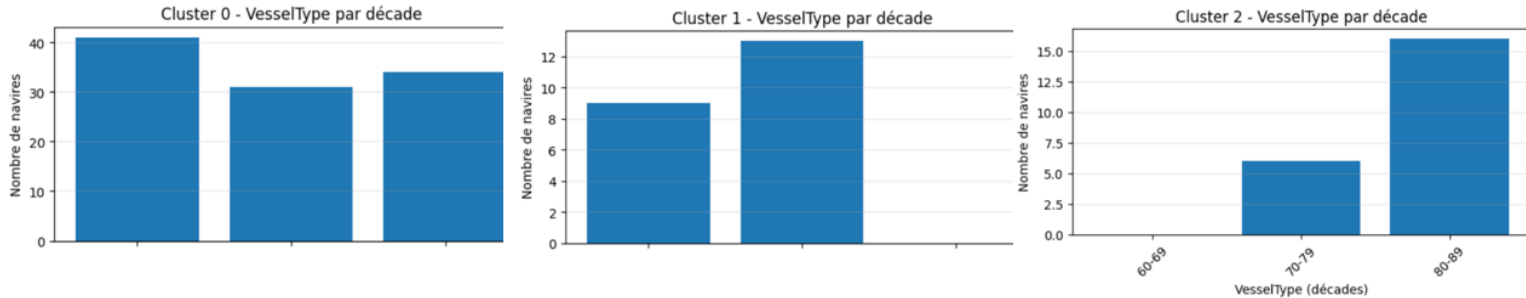


Figure 7 : Histogrammes de la composition des 3 clusters

Ces graphiques révèlent des tendances marquées :

- ❖ Le cluster 0 présente une répartition relativement équilibrée entre les trois types de navires (cargo, tanker, passager).
- ❖ Le cluster 1, quant à lui, exclut complètement les tankers et est largement dominé par les cargos.
- ❖ Enfin, le cluster 2 ne contient aucun navire de type passager et est principalement constitué de tankers.

Ces résultats suggèrent que le clustering basé sur les trajectoires capture également des dynamiques propres aux types de navires, ce qui renforce la pertinence de cette approche.

E. Déploiement d'un outil de prédiction basé sur le modèle de clustering

Afin de prolonger l'analyse effectuée avec notre modèle de clustering, nous avons développé un petit script Python permettant de prédire à quel cluster appartiendrait une trajectoire hypothétique, définie uniquement par un point de départ et un point d'arrivée.

L'objectif de ce script est double : vérifier que le modèle peut être utilisé de manière interactive sur de nouveaux itinéraires, et illustrer la réutilisabilité des objets exportés (le StandardScaler et le KMeans) via les fichiers .pkl.

Les features utilisées (min/max/centre des latitudes et longitudes, aire couverte, distance euclidienne) sont calculées à partir des deux points d'entrée. Après mise à l'échelle avec le StandardScaler, les données sont passées au modèle KMeans entraîné précédemment pour déterminer le cluster de rattachement.

Pour rendre l'usage plus intuitif, nous avons également intégré une interface console permettant de choisir parmi une liste de ports connus du Golfe du Mexique, simulant ainsi un scénario d'utilisation utilisateur :

```
PRÉDICTION DE CLUSTER POUR TRAJETS MARITIMES

=====
PORTS DISPONIBLES:
=====
1. Miami (Lat: 25.7617, Lon: -80.1918)
2. New Orleans (Lat: 29.9511, Lon: -90.0715)
3. Houston (Lat: 29.7604, Lon: -95.3698)
4. Beaumont (Lat: 30.0860, Lon: -94.1018)
5. Corpus Christi (Lat: 27.7322, Lon: -97.0734)
6. Tampa (Lat: 27.9506, Lon: -82.4572)
7. Key West (Lat: 24.5551, Lon: -81.7821)
=====

Choisissez le port de DÉPART:
Votre choix: Miami

Choisissez le port d'ARRIVÉE:
Votre choix: 

TRAJET : Miami -> Houston
Distance approximative : 15.6959°

Prédiction en cours...

RÉSULTAT : CLUSTER 2

Fin du script. Merci d'avoir utilisé le prédicteur !
```

Figure 8 : Interface console de l'outil de prédiction basé sur le modèle de clustering

Ce type d'outil pourrait être intégré à une application web ou une API REST dans un contexte plus opérationnel (par exemple, prédire automatiquement la typologie d'une trajectoire nouvelle selon le modèle historique).

IV. Besoin client 2 : Modèle de prédiction du type de navire

Dans le cadre du deuxième besoin exprimé, nous avons cherché à prédire le type de navire à partir de données AIS, en exploitant les caractéristiques géométriques et dynamiques issues des transmissions. Cette tâche constitue un problème de classification supervisée, où la variable cible est le VesselType, et les variables explicatives sont extraites ou dérivées des attributs bruts comme la longueur, la largeur, le tirant d'eau, ou encore le type de cargaison.

A. Analyse exploratoire des variables numériques

L'analyse exploratoire a débuté par l'étude des corrélations entre les variables numériques du jeu de données. Deux matrices de corrélation ont été générées : l'une intégrant l'ensemble des variables numériques disponibles, et l'autre restreinte aux variables liées aux caractéristiques physiques et dynamiques des navires (LAT, LON, SOG, COG, Heading, Length, Width, Draft), en y incluant la variable cible VesselType.

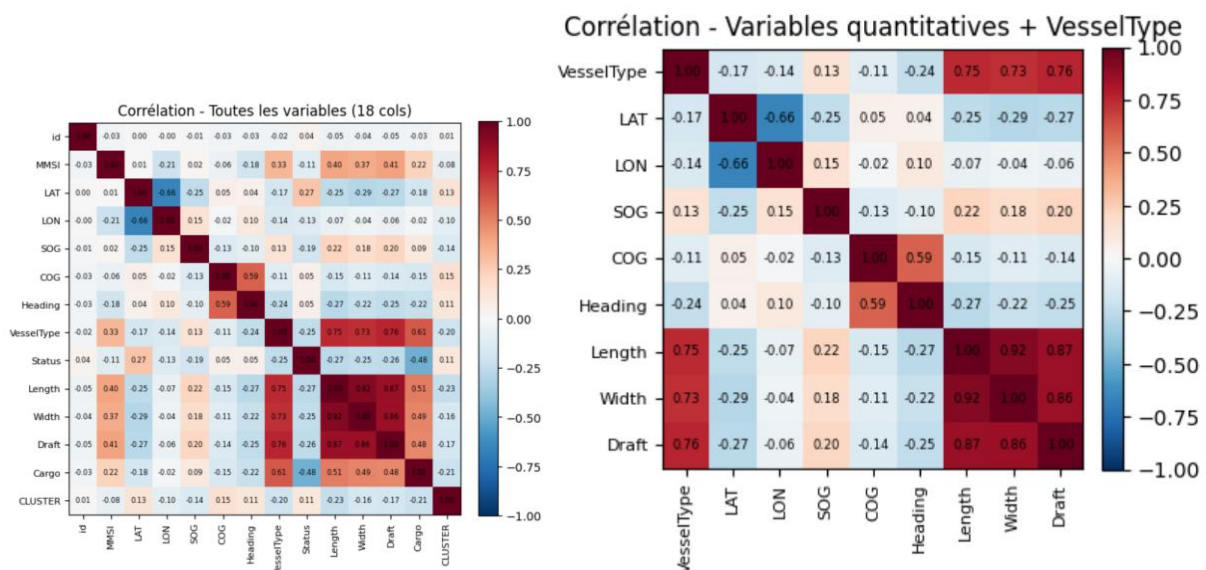


Figure 9 : Matrices de corrélation des variables brutes

L'interprétation des cartes de chaleur met en évidence plusieurs corrélations notables. VesselType est fortement corrélée aux dimensions des navires, avec des coefficients de 0,75 pour la longueur, 0,73 pour la largeur et 0,76 pour le tirant d'eau. Une corrélation significative est également observée avec la variable Cargo (0,61).

Ces résultats suggèrent que les dimensions et la capacité de transport sont des indicateurs discriminants pour prédire le type de navire, tandis que les données liées au mouvement sont moins informatives à l'état brut. Cette observation orientera les choix de modélisation et de sélection de variables dans les étapes suivantes.

B. Premier modèle erroné

Une première version du modèle présentait une fuite de données : la variable VesselType avait été conservée dans les colonnes explicatives via l'agrégation.

Cette erreur a été corrigée en excluant cette colonne lors de la création du jeu de données agrégé par navire. Les observations ont été regroupées par VesselName, en prenant les premières valeurs connues de variables telles que Length, Width, Draft, et Cargo. Nous avons également filtré les classes trop rares pour garantir une classification équilibrée, ne conservant que les types de navires avec au moins deux occurrences.

Deux approches ont été comparées comprenant cette erreur :

- ❖ Modèle 1 : Il s'appuie sur un jeu de variables étendu combinant des caractéristiques dynamiques (SOG, COG, Heading), géométriques (Length, Width, Draft) ainsi que les coordonnées géographiques (LAT, LON).
- ❖ Modèle 2 : Il repose uniquement sur les caractéristiques statiques des navires — en particulier leurs dimensions (Length, Width, Draft).

Ces deux modèles ont été initialement testés à l'aide d'un Random Forest Classifier, qui sera conservé comme modèle principal pour la suite. Par souci de comparaison, des essais ont également été réalisés avec une régression logistique et un Gradient Boosting Classifier sur les mêmes variables, révélant également des performances artificiellement élevées en raison de la fuite de données.

=== Modèle 1 : features étendues ===										Accuracy moyenne (5-fold CV) : 1.000									
Accuracy (test) : 1.000																			
	Prédit 60	Prédit 61	Prédit 69	Prédit 70	Prédit 71	Prédit 74	\	Vrai 60	Vrai 61	Prédit 60	Prédit 61	Prédit 69	Prédit 70	Prédit 71	Prédit 74	\			
Vrai 60	6346	0	0	0	0	0	0	Vrai 60	14805	0	0	0	0	0	0	0			
Vrai 61	0	66	0	0	0	0	0	Vrai 61	0	155	0	0	0	0	0	0			
Vrai 69	0	0	194	0	0	0	0	Vrai 69	0	0	452	0	0	0	0	0			
Vrai 70	0	0	0	3678	0	0	0	Vrai 70	0	0	0	8582	0	0	0	0			
Vrai 71	0	0	0	0	248	0	0	Vrai 71	0	0	0	0	578	0	0	0			
Vrai 74	0	0	0	0	0	58	0	Vrai 74	0	0	0	0	0	0	135	0			
Vrai 79	0	0	0	0	0	0	0	Vrai 79	0	0	0	0	0	0	0	0			
Vrai 80	0	0	0	0	0	0	0	Vrai 80	0	0	0	0	0	0	0	0			
Vrai 82	0	0	0	0	0	0	0	Vrai 82	0	0	0	0	0	0	0	0			
Vrai 84	0	0	0	0	0	0	0	Vrai 84	0	0	0	0	0	0	0	0			
Vrai 89	0	0	0	0	0	0	0	Vrai 89	0	0	0	0	0	0	0	0			
	Prédit 79	Prédit 80	Prédit 82	Prédit 84	Prédit 89					Prédit 79	Prédit 80	Prédit 82	Prédit 84	Prédit 89					
Vrai 60	0	0	0	0	0			Vrai 60	0	0	0	0	0	0					
Vrai 61	0	0	0	0	0			Vrai 61	0	0	0	0	0	0					
Vrai 69	0	0	0	0	0			Vrai 69	0	0	0	0	0	0					
Vrai 70	0	0	0	0	0			Vrai 70	0	0	0	0	0	0					
Vrai 71	0	0	0	0	0			Vrai 71	0	0	0	0	0	0					
Vrai 74	0	0	0	0	0			Vrai 74	0	0	0	0	0	0					
Vrai 79	195	0	0	0	0			Vrai 79	456	0	0	0	0	0					
Vrai 80	0	3290	0	0	0			Vrai 80	0	7677	0	0	0	0					
Vrai 82	0	0	57	0	0			Vrai 82	0	0	132	0	0	0					
Vrai 84	0	0	0	50	0			Vrai 84	0	0	0	118	0	0					
Vrai 89	0	0	0	0	248			Vrai 89	0	0	0	0	579	0					

Figure 10 : Matrice de confusion du premier modèle avec fuite de données — 100% de précision artificielle

C. Création de nouvelles variables explicatives

Afin d'améliorer les performances du modèle, nous avons exploré des techniques de feature engineering. Inspirés par des recherches sur l'architecture navale, nous avons introduit le DWT estimé (Deadweight Tonnage), une variable construite selon la formule :

$$DWT_estime = CB \times Length \times Width \times Draft \times \rho,$$

où CB est le coefficient de block, une valeur comprise entre 0.6 et 0.9 selon le type de navire, et $\rho = 1.025$ la densité moyenne de l'eau de mer. Une cartographie du coefficient de block a été associée à des intervalles de VesselType (ex. : 60–69 → 0.6), permettant d'estimer le DWT en l'absence de données réelles.

Remarque : Il est à noter que l'estimation initiale du DWT reposait sur une correspondance entre le coefficient de block (CB) et les classes de VesselType. Cette approche introduit une forme de circularité, dans la mesure où la variable cible est utilisée indirectement pour construire une variable explicative. Afin d'éviter toute fuite d'information lors de l'entraînement des modèles prédictifs, cette version du DWT n'a été conservée que pour l'exploration préliminaire. Pour la modélisation finale, un coefficient CB fixe (0.7) a été adopté de manière uniforme.

Nous avons également créé deux ratios géométriques : $L_W = Length / Width$ et $L_D = Length / Draft$, susceptibles de capturer la finesse ou la profondeur relative du navire.

Une analyse exploratoire basée sur des boxplots et une matrice de corrélation annotée a confirmé que DWT_estime était la variable dérivée la plus discriminante, montrant des variations marquées selon les catégories de navires :

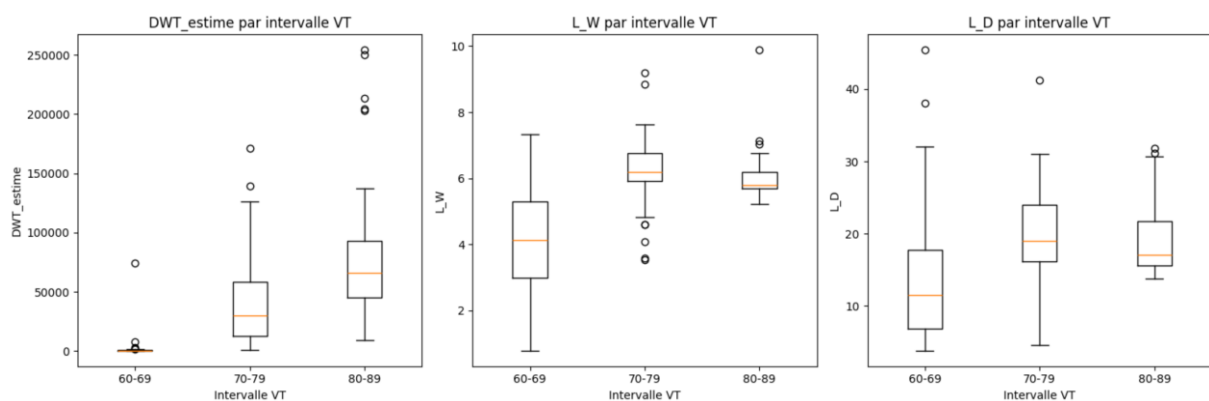


Figure 11 : boxplots des variables artificielles avec le VesselType

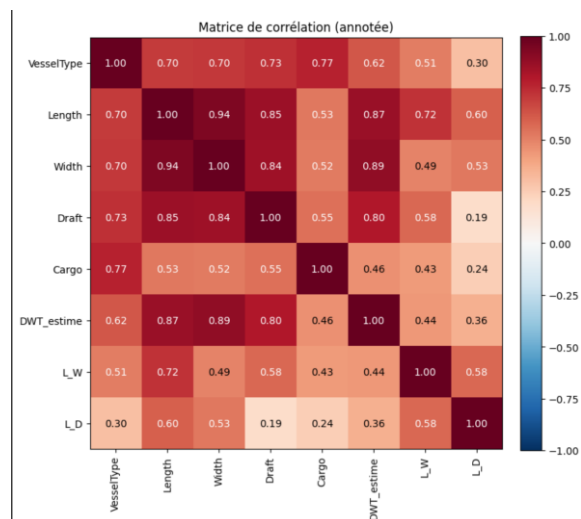


Figure 12 : Matrice de corrélation enrichie avec les variables artificielles — DWT estimé corrélé à plusieurs variables clés

Les visualisations produites mettent en évidence une certaine capacité des variables dérivées, notamment le DWT estimé, à distinguer les types de navires.

Les deux ratios géométriques montrent également des tendances intéressantes, bien qu'ils soient construits à partir de variables déjà existantes. Leur ajout peut néanmoins contribuer à simplifier l'apprentissage en rendant certaines caractéristiques plus directement exploitables.

Ces variables ont donc été intégrées à la modélisation finale, car elles enrichissent la représentation des navires sans introduire de redondance préjudiciable sur nos résultats.

D. Évaluation des modèles

1) Modèle Random Forest sans la variable Cargo

Dans un premier temps, un modèle a été entraîné à l'aide d'un RandomForestClassifier, en utilisant exclusivement des variables continues : les dimensions géométriques (longueur, largeur, tirant d'eau) ainsi que les variables dérivées issues du feature engineering (DWT estimé, ratios géométriques).

Un pipeline de traitement a été mis en place pour garantir une préparation homogène des données : les variables numériques ont été standardisées (centrées réduites), ce qui permet d'en uniformiser l'échelle, et ainsi d'éviter que certaines dominent l'apprentissage.

Ce modèle, bien que ne s'appuyant que sur des variables structurelles, a donné de bons résultats en termes de précision, confirmant la forte valeur prédictive des caractéristiques physiques des navires.

```

=== Test set performance ===
Accuracy      : 0.818
F1-score macro : 0.366

Matrice de confusion (test set):
      Prédit 60 Prédit 69 Prédit 70 Prédit 71 Prédit 79 Prédit 80 \
Vrai 60      13      0      0      0      0      1
Vrai 69       1      0      0      0      0      0
Vrai 70       0      0     13      0      0      0
Vrai 71       0      0      0      0      0      1
Vrai 79       0      0      1      0      0      0
Vrai 80       0      0      3      0      0     10
Vrai 89       0      0      0      0      0      1

      Prédit 89
Vrai 60       0
Vrai 69       0
Vrai 70       0
Vrai 71       0
Vrai 79       0
Vrai 80       0
Vrai 89       0

=== 5-fold CV performance ===
Accuracy      : 0.726 ± 0.080
F1-score macro : 0.493 ± 0.099

```

Figure 13 : Précision du modèle sans la variable Cargo

2) Modèle Random Forest avec la variable Cargo

Une seconde variante du modèle a été testée en ajoutant la variable catégorielle Cargo, qui apporte une indication sur la fonction ou la spécialisation du navire.

Le pipeline de traitement a ici intégré un encodage one-hot de cette variable, transformant ses modalités en variables binaires exploitables par les algorithmes de classification.

Cette version du modèle a permis de légèrement améliorer les performances globales, suggérant que Cargo capte une information utile complémentaire à la géométrie.

Une réflexion a été menée concernant la pertinence de la variable Cargo, certains la jugeant potentiellement redondante avec la variable cible VesselType. Toutefois, l'analyse du jeu de données a montré que ces deux variables, bien que liées, ne se recouvrent pas totalement, et présentent des différences significatives dans leur distribution. Cette distinction est d'autant plus marquée lorsqu'on examine d'autres jeux de données AIS similaires. Il a donc été décidé de conserver la variable Cargo dans le modèle, en considérant qu'elle apportait une information complémentaire utile à la classification.

```

=== Test set performance ===
Accuracy      : 0.909
F1-score macro : 0.640

Matrice de confusion (test set):
      Prédit 60 Prédit 69 Prédit 70 Prédit 71 Prédit 79 Prédit 80 \
Vrai 60      14      0      0      0      0      0
Vrai 69       1      0      0      0      0      0
Vrai 70       0      0     12      0      1      0
Vrai 71       0      0      0      1      0      0
Vrai 79       0      0      0      0      1      0
Vrai 80       0      0      1      0      0     12
Vrai 89       0      0      0      0      0      1

      Prédit 89
Vrai 60       0
Vrai 69       0
Vrai 70       0
Vrai 71       0
Vrai 79       0
Vrai 80       0
Vrai 89       0

=== 5-fold CV performance ===
Accuracy      : 0.863 ± 0.063
F1-score macro : 0.630 ± 0.069

```

Figure 14 : Précision du modèle avec la variable Cargo

3) Comparaison avec d'autres classifieurs

Pour compléter l'analyse, deux autres algorithmes de classification ont été testés sur le même jeu de données intégrant Cargo : une régression logistique et un classifieur par Gradient Boosting.

Ces modèles, bien qu'efficaces, se sont révélés légèrement moins performants que la Random Forest :

```

LogisticRegression    Accuracy: 0.886
GradientBoosting      Accuracy: 0.864

```

Figure 15 : Performances des modèles alternatifs avec Cargo

Leur inclusion dans cette étude montre cependant une démarche comparative, validant la robustesse du modèle retenu.

4) Analyse des erreurs de prédiction

Afin de mieux comprendre les limites de notre modèle, une analyse ciblée des erreurs a été menée sur les prédictions du meilleur classificateur : le RandomForestClassifier entraîné avec les variables enrichies (dont Cargo). Le modèle a été évalué à la fois sur un jeu de test séparé et par validation croisée à 5 plis, avec des performances satisfaisantes (accuracy et F1-score macro élevés), mais certaines erreurs de classification subsistaient.

Pour les analyser, nous avons identifié les instances du jeu de test pour lesquelles la prédiction était incorrecte, puis utilisé un interpréteur (DALEX) pour comprendre localement les décisions du modèle.

```
A new explainer has been created!

Instance 3 | Vrai : 70 → Prédit : 79
<dalex.predict_explanations._break_down.object.BreakDown object at 0x7be0d2584f50>

Instance 9 | Vrai : 69 → Prédit : 60
<dalex.predict_explanations._break_down.object.BreakDown object at 0x7be0d39f6050>

Instance 20 | Vrai : 89 → Prédit : 80
<dalex.predict_explanations._break_down.object.BreakDown object at 0x7be0f5daa390>

Instance 42 | Vrai : 80 → Prédit : 70
<dalex.predict_explanations._break_down.object.BreakDown object at 0x7be0f5e04790>
Modèle et LabelEncoder enregistrés dans /content/drive/MyDrive
```

Figure16 : Initialisation de l'interpréteur à partir des erreurs

Cet outil permet de visualiser l'influence individuelle de chaque variable sur une prédiction, via des diagrammes de décomposition (*breakdown plots*), illustrant comment le modèle est arrivé à sa décision :

(Remarque : ces deux breakdown plot sont disponible en plus grand en Annexe)

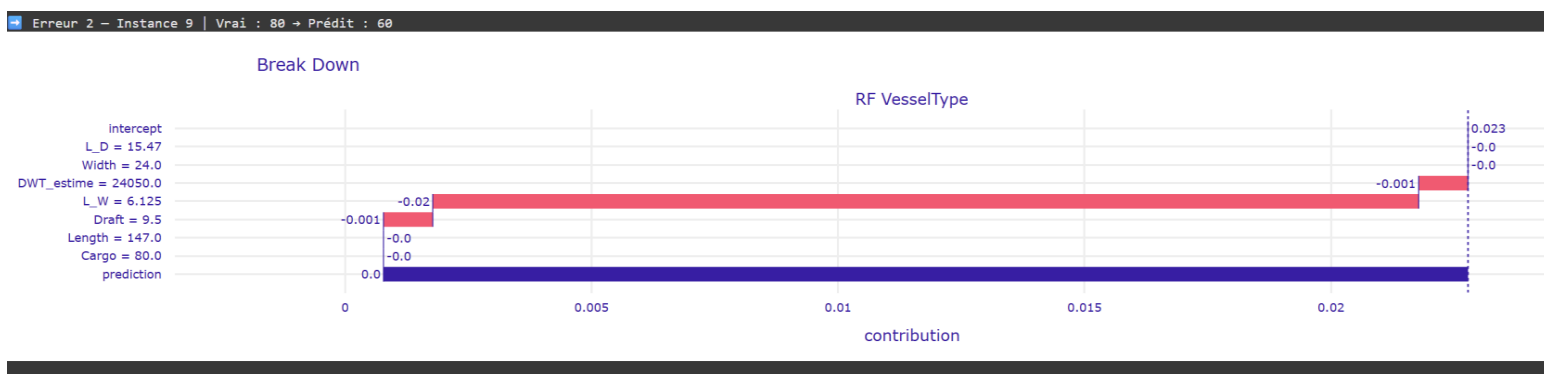


Figure 17 : Breakdown plot de l'erreur n°2

Par exemple, ce graphique montre comment le modèle, initialement orienté vers la bonne classe (80) avec une probabilité de 23 % (qui est calculé via la formule de l'intercept) , a progressivement revu sa prédiction à la baisse. La variable DWT estimé, notamment, a fortement contribué à éloigner la prédiction de la classe correcte, jusqu'à atteindre une probabilité finale nulle.

$$\text{Intercept} = \frac{1}{N} \sum_{i=1}^N P(Y = k | X_i)$$

À l'inverse, ce second graphique illustre un cas de classification correcte. Le modèle

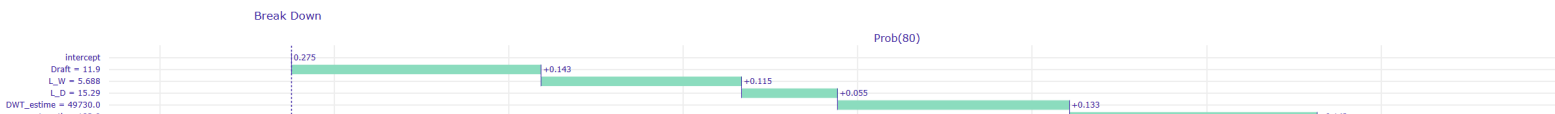


Figure 18 : Breakdown plot d'une réussite

démontre avec une probabilité de 27,5 % pour la classe 80, puis l'analyse des variables renforce progressivement cette hypothèse, jusqu'à atteindre un niveau de confiance de 97 %.

E. Script de prédiction du type de navire

Un script interactif a été conçu afin de rendre le modèle de prédiction exploitable de manière simple et autonome. Ce script permet à un utilisateur de saisir manuellement les caractéristiques principales d'un navire (longueur, largeur, tirant d'eau, type de cargo) et d'obtenir en retour une prédiction du type de navire, accompagnée d'un indicateur de confiance.

À partir des valeurs renseignées, le script calcule automatiquement les variables dérivées nécessaires au modèle, notamment le DWT estimé, le ratio longueur/largeur (L_W) et le ratio longueur/tirant d'eau (L_D).

L'ensemble de ces variables est ensuite traité par le pipeline complet (prétraitement + Random Forest), précédemment entraîné et sauvegardé au format .pkl.

Une gestion d'erreur est prévue en cas de fichiers manquants ou de saisies incorrectes, et l'utilisateur peut enchaîner plusieurs prédictions sans relancer le programme. Cette interface console, bien que simple, permet d'expérimenter rapidement le modèle sur des cas concrets.

```

Pipeline et LabelEncoder chargés depuis ~/content/drive/MyDrive

Entrez les caractéristiques du navire à prédire :
→ Largeur (Width) : 20
→ Tirant d'eau (Draft) : 3.3
→ Longueur (Length) : 60
→ Type de cargo (numérique, ex: 70, 80, 90, etc.) : 60

Résultat de la prédiction :
→ Type de navire prédit : 60
→ Confiance du modèle : 75.00%

Faire une autre prédiction ? (o/n): 
```

Figure 19 : Interface console de l'outil de prédiction du VesselType

V. Besoin client 3 : Prédiction de la trajectoire des navires

A. Préparation du modèle et des données

1) Contexte et justification du RandomForest

Dans le Golfe du Mexique, la densité de trafic de cargos pétroliers, de ferries et de navires de soutien offshore impose une prévision à court terme, de cinq à quinze minutes, pour garantir la sécurité maritime, fluidifier les approches portuaires et anticiper les manœuvres de chargement ou de ravitaillement.

Les données AIS que nous exploitons combinent des variables continues (vitesse, cap, orientation, dimensions) et catégorielles (type de navire) relevées à intervalles irréguliers. L'objectif de prédire simultanément deux cibles continues (latitude et longitude) exclut les méthodes de classification et favorise un MultiOutputRegressor. On va chercher 2 cibles avec un modèle. Le RandomForestRegressor répond à plusieurs besoins : il supporte naturellement le multioutput, il modélise la non-linéarité inhérente aux virages et accélérations, et il reste robuste face aux outliers et au bruit, sans supposer de distribution gaussienne des features.

D'autres approches, telles que le K-Nearest Neighbors ou le SVR, présentent un coût de calcul prohibitif et une sensibilité excessive à la distance spatiale, tandis que les réseaux de neurones récurrents, s'ils sont prometteurs pour la dynamique temporelle, nécessitent une infrastructure et un réglage d'hyperparamètres trop lourds pour ce prototype.

2) Préparation du modèle et enrichissement des données

Dans cette section, nous présentons les étapes successives qui nous ont permis de construire et d'améliorer notre jeu de variables dans le but de réduire simultanément le MAE, le RMSE et la distance moyenne de prédiction. À partir des relevés AIS bruts, nous avons d'abord trié chronologiquement les observations par MMSI et par horodatage, puis associé à chaque instant la position cinq, dix et quinze minutes plus tard au moyen d'un `merge_asof` avec une tolérance pour pallier l'irrégularité des relevés.

	MAE_lat	MAE_lon	RMSE_lat	RMSE_lon	Dist_mean_km
tolerance					
30s	0.049270	0.095587	0.194374	0.508255	12.528254
45s	0.035207	0.055706	0.151667	0.306607	8.014220
1min	0.020467	0.042943	0.098445	0.269807	5.704652
2min	0.015108	0.041026	0.090610	0.277474	4.999919
3min	0.015108	0.036221	0.081960	0.225794	4.576238
5min	0.015195	0.039404	0.087968	0.257217	4.865440

Figure 20 : évaluation pour définir la tolérance optimale sur le modèle

La tolérance dans `merge_asof` sert à limiter l'écart temporel maximal accepté entre un relevé initial et le relevé futur qu'on lui associe. Concrètement, lorsque l'on veut construire la position « + 5 min » pour chaque point, les AIS n'étant pas relevés exactement toutes les 5 minutes, on tolère un décalage (par ex. ± 3 min) pour trouver le point le plus proche en temps tout en évitant de faire correspondre des instants trop éloignés. Sans cette tolérance, on risquerait soit de ne pas associer de cible (si le pas est irrégulier), soit d'appliquer des appariements totalement hors délai, entraînant des paires input-cible incohérentes et faussant l'entraînement du modèle.

Nous avons ensuite enrichi chaque point de trois moyennes glissantes sur cinq instants pour la vitesse (SOG), le cap (COG) et l'orientation (Heading) afin de lisser le bruit des mesures instantanées. Pour mieux capter la dynamique des manœuvres, nous avons calculé la dérivée première de la vitesse et du cap pour en extraire l'accélération moyenne et le taux de virage moyen, eux aussi moyennés sur cinq points. Enfin, pour tenir compte des variations de comportement entre jour et nuit—manœuvres portuaires, rotations de ferries—nous avons ajouté un encodage cyclique de l'heure du jour à l'aide de fonctions sinus et cosinus.

L'image ci-dessous présente les résultats chiffrés obtenus pour chaque jeu de features (dynamiques, statiques, dyn+stat, dyn+stat+oh, enrichi_roll5 et full_features) pour les indicateurs MAE_lat, MAE_lon, RMSE_lat, RMSE_lon, Q1, distance moyenne en kilomètres et Q3 :

	MAE_lat	MAE_lon	RMSE_lat	RMSE_lon	Dist_mean_km
features					
dynamiques	0.4936	1.9143	0.8870	3.3448	199.7484
statiques	0.2949	0.7366	0.6305	1.7556	84.0805
dyn+stat	0.0345	0.0810	0.1455	0.4006	9.8189
dyn+stat+oh	0.0334	0.0772	0.1425	0.4055	9.4337
enrichi_roll5	0.0309	0.0710	0.1301	0.3839	8.7871
full_features	0.0205	0.0429	0.0984	0.2698	5.7047

Figure 21 : évaluation des combinaisons de features pour optimiser le modèle

La construction de notre jeu de features s'est faite entièrement sur la base de résultats empiriques.

Avec les seules variables dynamiques (SOG, COG, Heading), l'erreur moyenne de distance dépassait 200 km, inacceptable pour des prévisions opérationnelles dans le Golfe du Mexique.

L'ajout des dimensions statiques (Length, Width, Draft) a fait chuter cette erreur à 84 km.

En combinant dynamiques et statiques, le MAE sur latitude/longitude est alors tombé à 0,034°/0,081° et la distance moyenne à 9,8 km.

L'encodage One-Hot de VesselType a permis de gagner encore un peu (MAE 0,0334°/0,0772°, 9,4 km) en capturant les spécificités de comportement par type de navire.

L'introduction des moyennes glissantes sur cinq instants pour SOG, COG et Heading a réduit l'écart moyen à 8,8 km (MAE 0,0309°/0,0710°) en atténuant les fluctuations à court terme.

Enfin, en ajoutant les variables cinématiques (accélération moyenne et taux de virage moyen) et l'encodage cyclique de l'heure du jour, la version full_features atteint un MAE de 0,0205° en latitude et 0,0429° en longitude, un RMSE de 0,0984°/0,2698° et une distance moyenne d'erreur de seulement 5,7 km.

Ces résultats confirment l'intérêt d'enrichir progressivement les données par des moyennes glissantes, des dérivées et un encodage cyclique pour atteindre une meilleure précision adaptée à la navigation dans le Golfe du Mexique.

B. Implémentation du script et interprétation des résultats

Le pipeline s'appuie sur un ColumnTransformer appliquant un StandardScaler aux variables numériques, y compris moyennes et dérivées, et un OneHotEncoder à la colonne VesselType. La division train/test s'effectue à 80/20 avec un random state fixe pour garantir la reproductibilité.

Le script interactif invite l'utilisateur à choisir parmi une liste d'exemples de MMSI représentatifs du golfe, puis à sélectionner l'horizon de prédiction, soit 5, 10 ou 15 minutes. Après sous-échantillonnage d'un point sur cinq pour alléger l'affichage, le code filtre les relevés manquants en entrée ou en cible, affiche le nom du navire (lorsqu'il est disponible) et calcule les statistiques d'erreur, notamment les MAE sur latitude et longitude, la médiane, la moyenne et le troisième quartile de la distance en kilomètres.

On se propose de tester le script pour le bateau MMSI : 538003366, le STOLT LOTUS.

```

Exemples de MMSI : 538003366, 215583000, 311000808, 636021151, 311070400, 636022111, 244561000, 309587000, 369390000, 241770000
Entrez le MMSI : 538003366

Navire : ALGOMA VALUE (MMSI 538003366)

Horizons disponibles : [5, 10, 15]
Choisissez un horizon (min) : 5
Modèle chargé.

Stats pour +5 min :
MAE LAT : 0.0147°
MAE LON : 0.0058°
Méd dist : 0.180 km
Q3 dist : 0.431 km
Moy dist : 1.999 km

```

Figure 22 : Script 3

Le quartile d'ordre 75 % (aussi appelé troisième quartile ou Q3) est la valeur en dessous de laquelle se trouvent 75 % des données d'un échantillon trié. Dans notre cas, 75% de nos prédictions atterriront à moins de 431 mètres et 50% à moins de 180 mètres. On remarquera tout de même que la valeur moyenne des distances entre les points réels et prédits semble élevée par rapport au quartile d'ordre 75% ce qui reflète la présence de points prédits aberrants.

Enfin, la carte Folium superpose la trajectoire réelle en bleu et les positions prédites en drapeaux rouges avec une ligne orange qui relie les deux, offrant une visualisation claire et immédiatement exploitable.

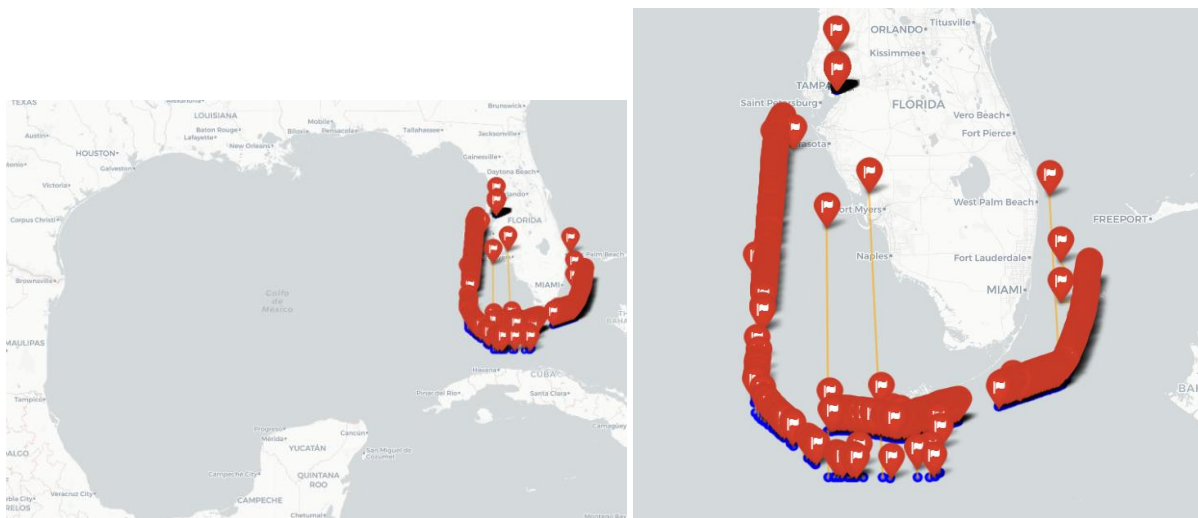


Figure 23 : Affichage des prédictions de Le STOLT LOTUS pour 5 minutes

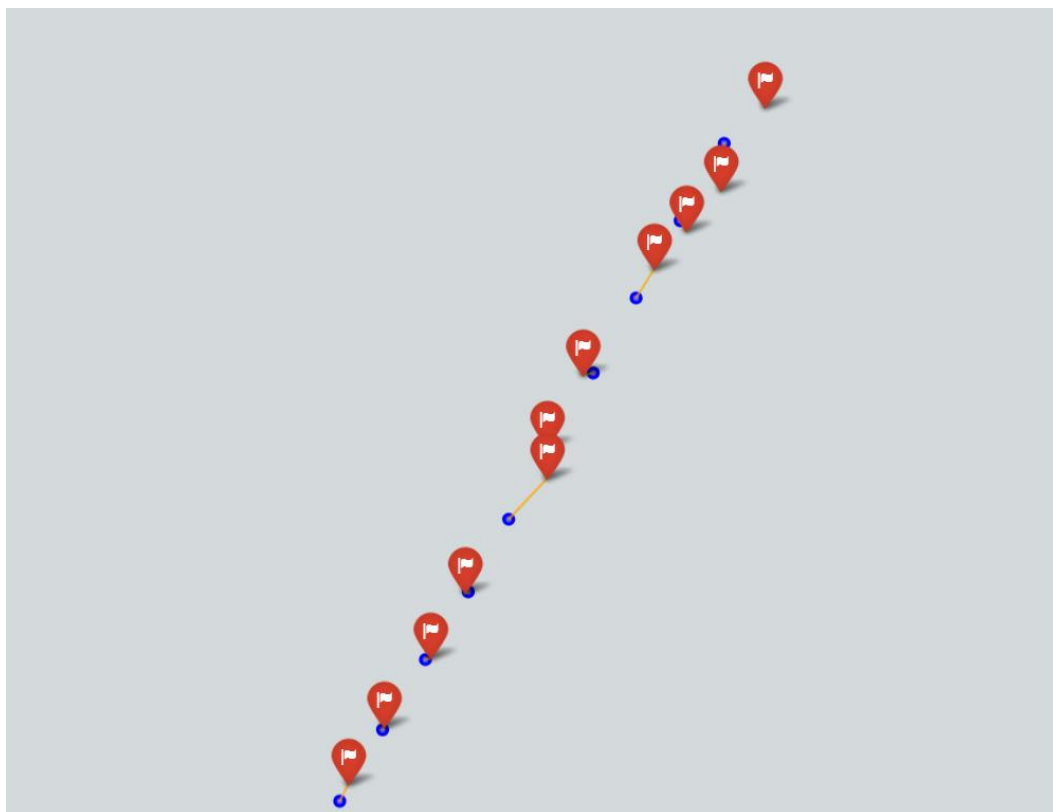


Figure 2411 : Zoom figure 12

Stats pour +5 min :	Stats pour +10 min :	Stats pour +15 min :
MAE LAT : 0.0341°	MAE LAT : 0.0345°	MAE LAT : 0.0523°
MAE LON : 0.0061°	MAE LON : 0.0110°	MAE LON : 0.0191°
Méd dist : 0.162 km	Méd dist : 1.764 km	Méd dist : 3.485 km
Q3 dist : 0.445 km	Q3 dist : 2.374 km	Q3 dist : 4.399 km
Moy dist : 4.125 km	Moy dist : 4.471 km	Moy dist : 6.899 km

Figure 2512 : métriques pour 5, 10 et 15 pour le STOLT LOTUS

On observe que plus l'horizon de prédiction augmente (passant de +5 à +10 puis +15 minutes), plus les erreurs moyennes (MAE), le troisième quartile (Q3) et la distance moyenne entre les points réels et prédits augmentent également. Cela s'explique naturellement par l'accumulation d'incertitudes au fil du temps : plus on cherche à prédire loin dans le futur, plus il est difficile d'anticiper avec précision les changements de cap, de vitesse ou de trajectoire du navire. Cette dégradation progressive des performances est donc attendue et met en évidence la difficulté croissante du problème à mesure que l'horizon s'éloigne.

On rappelle qu'une quarantaine de bateaux sont à l'arrêt dans nos données, sachant que nous disposons de 150 bateaux en tout pour entraîner notre modèle. Cela signifie qu'environ un tiers de l'ensemble des navires ne présentent que peu ou pas de mouvement, ce qui peut biaiser l'apprentissage du modèle. En effet, pour ces navires à

l'arrêt, la prédiction de la position future devient triviale (elle ne change pas ou très peu), ce qui peut faussement améliorer certaines métriques globales, notamment la MAE.

Par ailleurs, certaines valeurs apparaissent comme aberrantes : on observe par exemple des vitesses nulles combinées à des variations de cap, ou des coordonnées GPS qui changent brutalement malgré une vitesse très faible. Cela peut s'expliquer par du bruit dans les capteurs, des erreurs de transmission AIS, ou encore des erreurs de synchronisation temporelle entre deux points.

Enfin, certaines trajectoires peuvent être mal échantillonnées, avec des intervalles irréguliers entre les points, ce qui dégrade la qualité des variables dérivées comme l'accélération ou le taux de giration. Ces artefacts introduisent de la variabilité non représentative du comportement réel du navire, et peuvent dégrader la performance du modèle dans les cas plus dynamiques.

Malgré tout, la tendance des prédictions semble suivre celle des valeurs réelles. Cela montre que le modèle parvient globalement à capturer la dynamique des navires en déplacement, malgré la présence de bruit ou de données atypiques. Les trajectoires prédites restent souvent proches des trajectoires observées, même si des écarts peuvent apparaître ponctuellement, notamment lors de virages brusques ou de changements de vitesse mal anticipés. Cela confirme que, bien que perfectible, le modèle est capable de restituer les grandes lignes du comportement spatio-temporel des navires.

VI. Bonus : Utilisation d'une vrai DB AIS (25-31/05/2024)

Afin d'évaluer la robustesse de notre pipeline sur des données réelles issues du système AIS, nous avons décidé d'exploiter une base externe.

Cette démarche n'a pas été simple : la majorité des bases AIS complètes et récentes sont réservées aux autorités maritimes, aux assureurs ou aux chercheurs institutionnels. Il a donc été particulièrement difficile de trouver une base librement accessible, complète et suffisamment documentée.

Après plusieurs recherches, nous nous sommes tournés vers le site officiel gouvernemental de la marine des Etats-Unis, qui propose en libre accès des archives quotidiennes de données AIS complète depuis 2009. Nous avons choisi de concentrer notre étude sur les mêmes jours que la base initialement fournie, mais pour l'année suivante (2024), soit du 25 au 31 mai.

A. Préparation des données

Après avoir téléchargé les 7 fichiers .zip correspondant aux 7 jours, nous avons réalisé un nettoyage et une concaténation en langage R. Les données brutes étant très volumineuses (environ 1 Go par jour), il était indispensable d'en réduire la taille :

- ❖ Filtrage géographique : nous avons restreint les observations à la zone du golfe du Mexique, comme pour la base précédente.
- ❖ Filtrage par type de navire : nous avons conservé uniquement les navires dont le VesselType est compris entre 60 et 89, correspondant aux types commerciaux d'intérêt (tankers, cargos, etc.).
- ❖ Réduction des colonnes : nous avons supprimé les variables non pertinentes pour l'analyse afin d'alléger les fichiers finaux.

B. Analyse par clustering

Une fois le prétraitement terminé, nous avons reconduit la même analyse non supervisée que précédemment. Cela a consisté en un clustering des trajectoires de navires selon leurs caractéristiques de navigation (position, vitesse, cap, etc.).

Nous avons testé plusieurs valeurs de k (entre 2 et 10) et évalué leur pertinence grâce à différents indices de validation (Silhouette score, Davies-Bouldin, Calinski-Harabasz...).

Le nombre optimal de clusters obtenu est $k=4$:

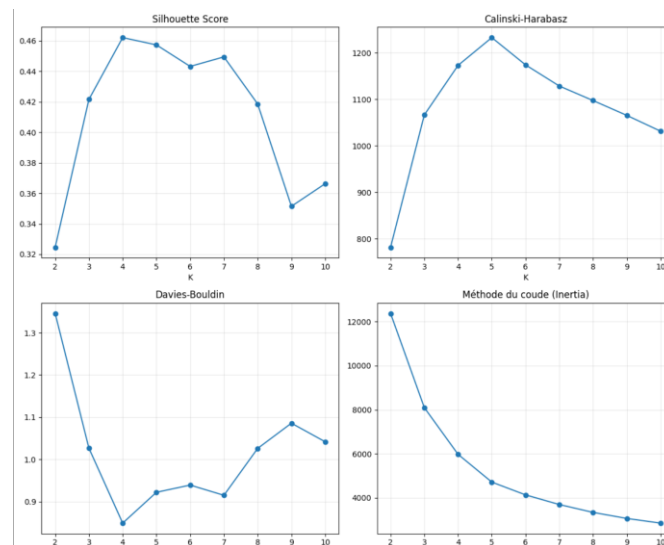


Figure 26 : graphique des 4 scores pour déduire k optimal

Une fois le nombre de clusters connu on passe à l'affichage des clusters. L'affichage de toutes les trajectoires individuelles étant trop dense pour être lisible, nous avons représenté uniquement les centres de trajectoires (moyennes de coordonnées), ce qui explique quelques points apparaissant légèrement en dehors de leur cluster théorique.

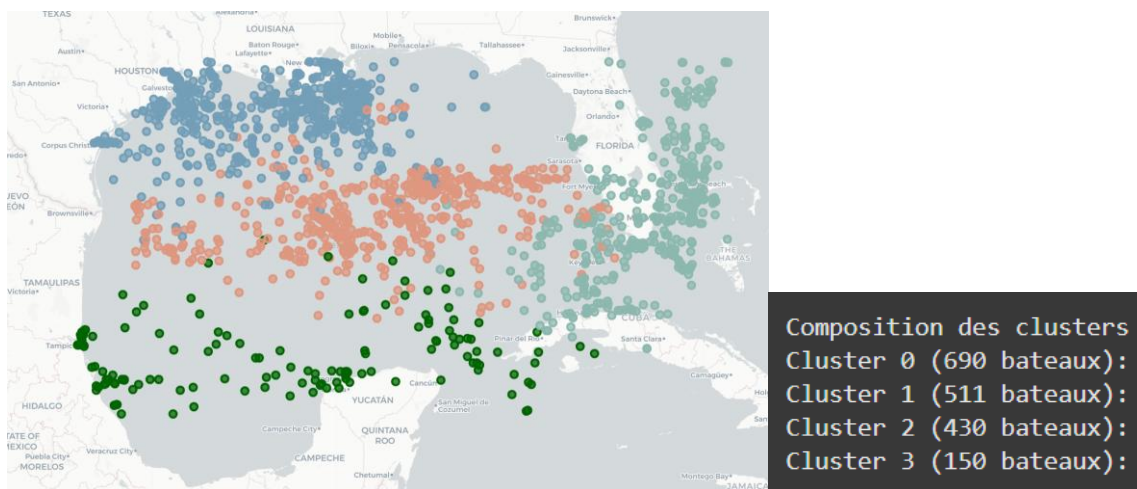


Figure 27 : répartition des 4 clusters

On observe que les 4 clusters sont bien distincts, avec trois d'entre eux très similaires à ceux obtenus avec la base initiale. Un changement important cependant : l'apparition de points dans la zone sud du golfe du Mexique, probablement due à une couverture plus complète dans les données réelles.

C. Prédiction du type de navire (VesselType)

Nous avons ensuite appliqué un modèle de classification supervisée, comme précédemment. Le classificateur utilisé est à nouveau le RandomForestClassifier, qui s'avère performant dans la détection de corrélations complexes ou non linéaires.

Nous avons commencé par calculer la matrice de corrélation entre les variables explicatives et la variable cible (VesselType) :

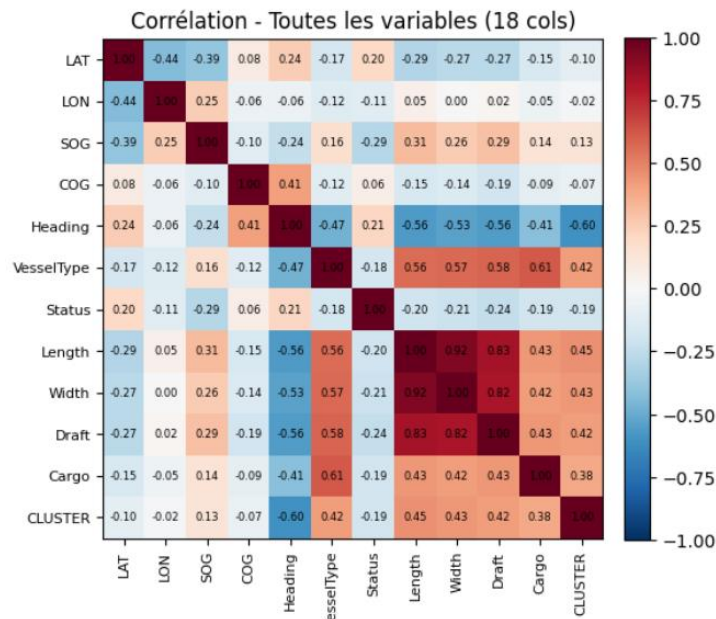


Figure 28 : Matrice de corrélation

Bien que la structure générale soit proche de celle obtenue avec la première base, on note que certaines corrélations significatives se sont atténuées. Par exemple, la corrélation entre VesselType et Cargo est passée de 0.77 à 0.61, ce qui peut nuire à la performance du modèle.

D. Résultats du classificateur

Comme précédemment, nous avons utilisé GridSearchCV pour optimiser les hyperparamètres de la forêt aléatoire.

Nous avons ensuite testé trois variantes de modèles :

- ❖ Résultat du modèle avec cargo :

```
=== Test set performance ===  
Accuracy      : 0.908  
F1-score macro : 0.818
```

- ❖ Résultat du modèle sans cargo :

```
=== Test set performance ===  
Accuracy      : 0.611  
F1-score macro : 0.170
```

- ❖ Sans Cargo mais en prenant en compte les variances de vitesse (SOG), et cap(COG) :

```
Accuracy globale : 0.637
```

Ces résultats confirment l'importance de la variable Cargo dans la prédiction, bien que sa corrélation ait diminué. La baisse de performance du modèle sans cette variable illustre la fragilité du lien entre géométrie du navire et usage commercial, surtout dans un jeu de données non nettoyé manuellement.

VII. ANNEXE

Site où on a trouvé de nouvelles Data Base :
<https://hub.marinecadastre.gov/pages/vesseltraffic>

