



UNIVERSITÉ MOHAMMED-V DE RABAT  
ÉCOLE NATIONALE SUPÉRIEURE  
D'INFORMATIQUE ET D'ANALYSE DES  
SYSTÈMES  
ENSIAS



DÉPARTEMENT GÉNIE LOGICIEL

DATA DRIVING MAKING DECISION

---

# Prévision de la demande des produits du terroir Marocain sur le marché national et international.

---

Élèves ingénieurs :  
KENDI MOHAMMED AMINE  
KARMOUCHI ASMAE

Professeur Encadrant :  
Pr. Y. Tabii

Année Académique 2023/2024

# Résumé

Ce projet porte sur la prévision de la demande des produits du terroir marocain sur les marchés national et international. L'objectif principal est d'élaborer des stratégies de développement et de commercialisation basées sur une analyse approfondie des données collectées à partir de sources officielles. La démarche suivie inclut la collecte, le nettoyage, et l'analyse des données, la sélection et la justification des algorithmes de prévision, ainsi que la visualisation des résultats pour faciliter la prise de décisions informées.

Ce projet vise à fournir des recommandations stratégiques basées sur une analyse rigoureuse des données pour promouvoir efficacement les produits du terroir marocain et répondre aux attentes des marchés ciblés. Les livrables incluent un rapport détaillé, une présentation PowerPoint, le code source commenté, et un fichier ReadMe pour l'exécution du code.

En résumé, ce projet démontre comment l'utilisation de techniques avancées d'analyse de données et d'intelligence artificielle peut améliorer la précision des prévisions de demande et informer des stratégies de commercialisation efficaces pour les produits du terroir marocain.

**Mots-clés :** Prévision de la demande, Produits marocain, Marchés national et international, Analyse des données, Algorithmes de prévision, Visualisation des résultats, Recommandations stratégiques, Techniques avancées, Intelligence artificielle,

# Table des acronymes

**AI** :Artificial Intelligence

**ML** :Machine Learning

**LSTM** :Long Short-Term Memory

**ARIMA** :AutoRegressive Integrated Moving Average

**SARIMA** : Seasonal AutoRegressive Integrated Moving Average

**EDA** :Exploratory Data Analysis

**MAE** :Mean Absolute Error

**RMSE** :Root Mean Square Error

**AIC** :Akaike Information Criterion

**RNN** :Recurrent Neural Network

# Table des matières

<b>Résumé</b>	<b>1</b>
<b>Table des acronymes</b>	<b>2</b>
<b>1 Introduction</b>	<b>5</b>
1.1 Contexte et importance du projet . . . . .	5
1.2 Objectifs du projet . . . . .	5
1.3 Approche et Méthodologie . . . . .	6
<b>2 Dataset</b>	<b>7</b>
2.1 Source des Données : . . . . .	7
2.1.1 Description des Données . . . . .	7
2.2 Visualition des données recueillies : . . . . .	11
2.2.1 barplot : . . . . .	11
2.2.2 contribution produit : . . . . .	13
2.2.3 Heat map . . . . .	14
<b>3 Nettoyage et Préparation des Données</b>	<b>15</b>
3.1 Techniques de nettoyage des données . . . . .	15
3.1.1 Réduction du nombre de colonnes : . . . . .	15
3.1.2 Suppression des produits avec des valeurs d'exportation nulles : . . .	16
3.2 Transformation des données . . . . .	17
3.2.1 Objectifs de la transformation des données . . . . .	17
3.2.2 Étapes de transformation . . . . .	18
3.2.3 Résultats de la transformation des données . . . . .	19
<b>4 Sélection et Justification des Algorithmes de Prédiction</b>	<b>23</b>
4.1 Algorithmes de Prédiction Sélectionnés . . . . .	23
4.1.1 ARIMA[1] . . . . .	23
4.1.2 LSTM Networks [4] . . . . .	24
4.1.3 Prophet (développé par Facebook) [3] . . . . .	24
4.2 Justification des choix des modèles . . . . .	24
4.3 Application aux Données d'Exportation du Maroc . . . . .	25
4.3.1 Préparation des Données . . . . .	25
4.3.2 Résultats des Modèles . . . . .	26
<b>Conclusion</b>	<b>31</b>

# Table des figures

2.1	The shape of dataset . . . . .	8
2.2	The 6 first rows of the dataset . . . . .	8
2.3	méthode <b>info()</b> : détail sur les colonnes . . . . .	9
2.4	résultat méthode <b>isnull().sum()</b> . . . . .	10
2.5	Nombre de case contenant des zéros . . . . .	10
2.6	Visualisation barplot des 5 premiers produits . . . . .	11
2.7	Visualisation des 15 produits suivants . . . . .	12
2.8	Contribution des premiers produits . . . . .	13
2.9	Heat map . . . . .	14
3.1	Diminution de nombre de colonnes . . . . .	16
3.2	Visualisation de la nouvelle <i>Dataframe</i> . . . . .	20
3.3	Série temporelle pour des produits au fil des années . . . . .	21
3.4	Heatmap des valeurs par année par produit . . . . .	22
4.1	Série temporelle pour 10 produits aléatoirement choisis . . . . .	24
4.2	Arima visualisation . . . . .	27
4.3	Lstm visualisation . . . . .	28
4.4	Prophet visualisation . . . . .	28
4.5	Arima visualisation . . . . .	29
4.6	Lstm visualisation . . . . .	29
4.7	Prophet visualisation . . . . .	30

# Chapitre 1

## Introduction

### 1.1 Contexte et importance du projet

La promotion des produits du terroir marocain, tels que l'huile d'argan, le safran et les dattes, est essentielle pour le développement économique du pays. Ces produits, reconnus pour leur qualité unique et leur valeur culturelle, ont un potentiel significatif sur les marchés national et international. Toutefois, la demande fluctuante et la concurrence accrue nécessitent une stratégie bien informée pour optimiser la production et la commercialisation. Nous visons à fournir une analyse approfondie sur la prévision de la demande des produits du terroir marocain sur les marchés national et international. L'objectif principal est de développer des stratégies de développement et de commercialisation basées sur une analyse rigoureuse des données collectées à partir de sources officielles. La méthodologie suivie inclut la collecte, le nettoyage et l'analyse des données, ainsi que la sélection et la justification des algorithmes de prévision, en utilisant des techniques avancées d'analyse de données et d'intelligence artificielle.

### 1.2 Objectifs du projet

L'objectif principal de ce projet est de prévoir la demande des produits du terroir marocain sur les marchés national et international et de développer des stratégies de développement et de commercialisation basées sur les données. Les objectifs spécifiques incluent :

- **Collecter et analyser les données de vente et de marché :** Rassembler des données pertinentes sur les ventes et le marché des produits marocain à partir de sources fiables telles que les instituts statistiques, les organismes gouvernementaux et les études de marché.
- **Nettoyer et préparer les données pour une analyse précise :** Traiter les données collectées pour éliminer les erreurs, les doublons et les valeurs manquantes, et les formater de manière à les rendre aptes à une analyse précise.
- **Sélectionner et justifier les algorithmes de prévision :** Choisir les méthodes et les modèles de prévision appropriés en fonction des caractéristiques des données et des objectifs du projet, en justifiant les choix effectués.
- **Visualiser les résultats pour faciliter la prise de décisions informées :** Utiliser des techniques de visualisation de données telles que les graphiques, les tableaux.
- **Formuler des recommandations stratégiques basées sur les prévisions .**

## 1.3 Approche et Méthodologie

Dans ce projet, nous avons adopté une approche systématique pour analyser l'ensemble de données et découvrir des idées sur la demande des produits du terroir marocain. La méthodologie suivie comprend plusieurs étapes clés :

1. Collecte de Données

2. Nettoyage et Préparation des Données

3. Analyse Exploratoire des Données (EDA)

4. Sélection et Justification des Algorithmes de Prévision [2]

- **Modèles ARIMA** : ARIMA est un modèle statistique classique utilisé pour modéliser et prévoir les séries temporelles. Il est efficace pour capturer les tendances et les motifs de variation dans les données.
- **Modèles SARIMA (Seasonal ARIMA)** : SARIMA étend le modèle ARIMA pour prendre en compte les variations saisonnières dans les séries temporelles.
- **Modèles LSTM** : Les réseaux LSTM sont une forme de réseaux de neurones récurrents (RNN) conçus pour modéliser les dépendances à long terme dans les séquences de données. Ils sont efficaces pour capturer les modèles complexes et non linéaires dans les séries temporelles.
- **Modèle Prophet** : Prophet est un modèle développé par Facebook, conçu pour prévoir les séries temporelles avec des points de changement fréquents et des tendances non linéaires. Il est particulièrement utile pour les données comportant des saisons multiples ou des jours spéciaux.

5. Mise en Œuvre et Évaluation des Modèles

# Chapitre 2

## Dataset

### 2.1 Source des Données :

Les données utilisées dans cette étude ont été extraites de Trade Map, une plateforme fournissant des statistiques commerciales pour le développement des affaires internationales. Trade Map offre des données mensuelles, trimestrielles et annuelles sur le commerce international, comprenant les valeurs d'importation et d'exportation, les volumes, les taux de croissance, les parts de marché, et bien d'autres indicateurs pertinents pour l'analyse du commerce mondial.

#### 2.1.1 Description des Données

Les données d'exportation par produit sont fournies sous forme de valeurs annuelles de 2003 à 2022. Le jeu de données contient les colonnes suivantes :

- **Code** : Le code unique identifiant chaque produit.
- **Product label** : La description du produit.
- **Exported value in 2003** : La valeur exportée du produit en 2003.
- **Exported value in 2004** : La valeur exportée du produit en 2004.
- **Exported value in 2005** : La valeur exportée du produit en 2005.
- **Exported value in 2006** : La valeur exportée du produit en 2006.
- **Exported value in 2007** : La valeur exportée du produit en 2007.
- **Exported value in 2008** : La valeur exportée du produit en 2008.
- **Exported value in 2009** : La valeur exportée du produit en 2009.
- **Exported value in 2010** : La valeur exportée du produit en 2010.
- **Exported value in 2011** : La valeur exportée du produit en 2011.
- **Exported value in 2012** : La valeur exportée du produit en 2012.
- **Exported value in 2013** : La valeur exportée du produit en 2013.
- **Exported value in 2014** : La valeur exportée du produit en 2014.
- **Exported value in 2015** : La valeur exportée du produit en 2015.
- **Exported value in 2016** : La valeur exportée du produit en 2016.
- **Exported value in 2017** : La valeur exportée du produit en 2017.
- **Exported value in 2018** : La valeur exportée du produit en 2018.
- **Exported value in 2019** : La valeur exportée du produit en 2019.
- **Exported value in 2020** : La valeur exportée du produit en 2020.
- **Exported value in 2021** : La valeur exportée du produit en 2021.



— **Exported value in 2022** : La valeur exportée du produit en 2022.

Nous sommes d'abord curieux de connaître les dimensions de notre ensemble de données. Nous le découvrons en utilisant l'attribut `shape` du dataframe. La figure 2.1 représente la forme des données que nous allons utiliser, tandis que la figure 2.2 représente les première 6 lignes des données itilisées.

```
df.shape
✓ 0.0s
(5326, 22)
```

FIGURE 2.1 – The shape of dataset

	Code	Product label	Exported value in 2003	Exported value in 2004	Exported value in 2005	Exported value in 2006	Exported value in 2007	Exported value in 2008	Exported value in 2009	Exported value in 2010
0	'TOTAL	All products	8738341	9870179	11197413	12731303	15321255	20094617	14027369	17764791
1	'310530	Diammonium hydrogenorthophosphate "diammonium ...	187480	169600	201128	250901	379138	556772	426977	858530
2	'870322	Motor cars and other motor vehicles principall...	154	38	0	68	15011	8739	103	216
3	'280920	Phosphoric acid; polyphosphoric acids, whether...	507748	713155	861974	897193	1106980	2945756	999254	1633333
4	'310540	Ammonium dihydrogenorthophosphate "monoammoniu...	120350	155382	138635	168734	236700	359355	140933	399515
5	'854430	Ignition wiring sets and other wiring sets for...	1496	1419	67305	99879	239595	296663	482438	1075276
6	'251010	Natural calcium phosphates and natural alumini...	362620	420784	518104	577642	742872	2281656	552711	1066957
...			Exported value in 2013	Exported value in 2014	Exported value in 2015	Exported value in 2016	Exported value in 2017	Exported value in 2018	Exported value in 2019	Exported value in 2020
...	21967951	23815816	22197567	22850021	25619900	29317739	29592492	27704922	36578743	42331463
...	729453	776074	603765	607097	796871	1271422	939489	1391742	1983471	3458926
...	329837	1079008	1054759	1395613	1682384	2107521	2054644	1775361	2586182	3389603
...	1433671	1523131	1649823	1138528	1096644	1477068	1413013	1200014	2204279	2254490
...	691868	654685	608446	715663	811963	1026339	1042180	1117906	2138153	2056357
...	1149521	1251319	1237521	1360902	1616389	1915199	2156375	1698805	1677065	1654677
...	1082093	977560	1023670	756017	862979	884160	760461	773549	994759	1322481

FIGURE 2.2 – The 6 first rows of the dataset

En utilisant la méthode `info()` comme illustré dans la figure 2.3, nous pouvons exposer des détails plus typiques de chaque colonne comme suit :

```
df.info()
✓ 0.0s

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5326 entries, 0 to 5325
Data columns (total 22 columns):
#   Column                                     Non-Null Count  Dtype
---  -
0   Code                                     5326 non-null   object
1   Product label                           5326 non-null   object
2   Exported value in 2003                   5326 non-null   int64
3   Exported value in 2004                   5326 non-null   int64
4   Exported value in 2005                   5326 non-null   int64
5   Exported value in 2006                   5326 non-null   int64
6   Exported value in 2007                   5326 non-null   int64
7   Exported value in 2008                   5326 non-null   int64
8   Exported value in 2009                   5326 non-null   int64
9   Exported value in 2010                   5326 non-null   int64
10  Exported value in 2011                   5326 non-null   int64
11  Exported value in 2012                   5326 non-null   int64
12  Exported value in 2013                   5326 non-null   int64
13  Exported value in 2014                   5326 non-null   int64
14  Exported value in 2015                   5326 non-null   int64
15  Exported value in 2016                   5326 non-null   int64
16  Exported value in 2017                   5326 non-null   int64
17  Exported value in 2018                   5326 non-null   int64
18  Exported value in 2019                   5326 non-null   int64
19  Exported value in 2020                   5326 non-null   int64
20  Exported value in 2021                   5326 non-null   int64
21  Exported value in 2022                   5326 non-null   int64
dtypes: int64(20), object(2)
memory usage: 915.5+ KB
```

FIGURE 2.3 – méthode `info()` : détail sur les colonnes

Passant au traitement des valeurs manquantes, notre dataset ne contient aucune valeur manquante. Pour le montrer plus clairement, nous avons utilisé la méthode `isnull().sum()` qui permet de compter le nombre de valeurs nulles dans chaque colonne. Le fait que le résultat de cette opération soit nul (figure 2.4) indique qu'il n'y a pas de valeurs manquantes dans notre ensemble de données. Cela peut être attribué à une collecte de données rigoureuse effectuée par Trade Map.

```

Code                                0
Product label                       0
Exported value in 2003              0
Exported value in 2004              0
Exported value in 2005              0
Exported value in 2006              0
Exported value in 2007              0
Exported value in 2008              0
Exported value in 2009              0
Exported value in 2010              0
Exported value in 2011              0
Exported value in 2012              0
Exported value in 2013              0
Exported value in 2014              0
Exported value in 2015              0
Exported value in 2016              0
Exported value in 2017              0
Exported value in 2018              0
Exported value in 2019              0
Exported value in 2020              0
Exported value in 2021              0
Exported value in 2022              0
dtype: int64

```

FIGURE 2.4 – résultat méthode `isnull().sum()`

Dans notre analyse du dataset, nous avons remarqué la présence de nombreuses valeurs nulles. Cependant, il est difficile de déterminer si ces valeurs nulles sont significatives ou non pour notre étude. Pour mieux comprendre l'impact de ces valeurs nulles sur nos données, nous avons effectué une évaluation quantitative.

Nous avons utilisé le code Python suivant pour calculer le nombre total de valeurs nulles dans notre dataset :

```

1
2 total_cases = df.shape[0] * df.shape[1]
3 total_zero_values = df.eq(0).sum().sum()
4
5 print("Nombre total de cases contenant des zéros / cases dans
  le Dataset :", total_zero_values, "/", total_cases)

```

Nombre total de cases contenant des zéros / cases dans le Dataset : 49572 / 117172

FIGURE 2.5 – Nombre de case contenant des zéros

D'après la figure 2.5, nous constatons qu'il y a un total de 49 572 cases contenant des zéros sur un total de 117 172 cases dans le dataset. Cela signifie qu'environ 42,3% des cases dans notre dataset contiennent des valeurs nulles.

Ce résultat met en lumière l'importance des valeurs nulles dans nos données. Il serait judicieux d'examiner de plus près ces valeurs nulles pour déterminer leur impact sur nos analyses et décider des meilleures stratégies de traitement à adopter.

## 2.2 Visualisation des données recueillies :

Dans cette section, nous allons visualiser nos données en utilisant différents types de graphiques pour mieux les connaître.

### 2.2.1 barplot :

Cette visualisation présente les valeurs exportées par année pour les cinq premiers produits de notre ensemble de données. Chaque sous-plot représente un produit spécifique, avec les années sur l'axe des x et les valeurs exportées sur l'axe des y. Les barres colorées indiquent les valeurs exportées pour chaque année, avec une couleur différente pour chaque produit. Cette visualisation permet de comparer les tendances d'exportation entre les produits et de comprendre les variations des valeurs exportées au fil du temps pour chaque produit individuel.

La figure 2.6 présente les valeurs exportées par année pour le total des produits et 4 premiers produits de notre ensemble de données.

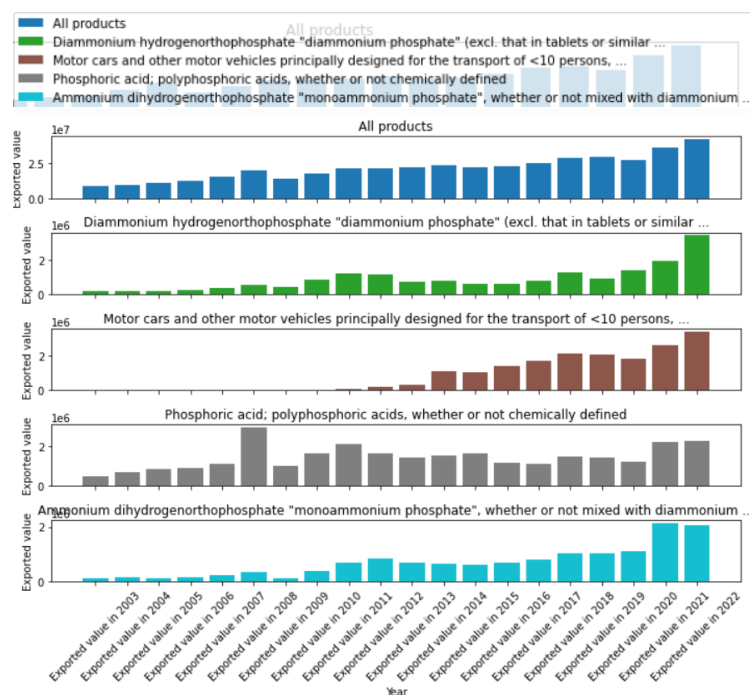


FIGURE 2.6 – Visualisation barplot des 5 premiers produits

L'analyse des barplots révèle des variations significatives dans les valeurs exportées au fil du temps pour chaque produit. Certains produits montrent une croissance constante des exportations au fil des années, indiquant une demande croissante sur le marché international. D'autres produits peuvent présenter des fluctuations dans les exportations d'une année à l'autre. En examinant de près les barplots, nous pouvons identifier les années où les valeurs exportées ont atteint leur pic ou leur creux pour chaque produit. Cette analyse temporelle peut fournir des informations précieuses sur les performances passées des produits sur le

marché mondial et aider à orienter les stratégies futures d'exportation et de développement de produits.  
Si on affiche les bar plot pour les 15 produits suivants (figure 2.7) :

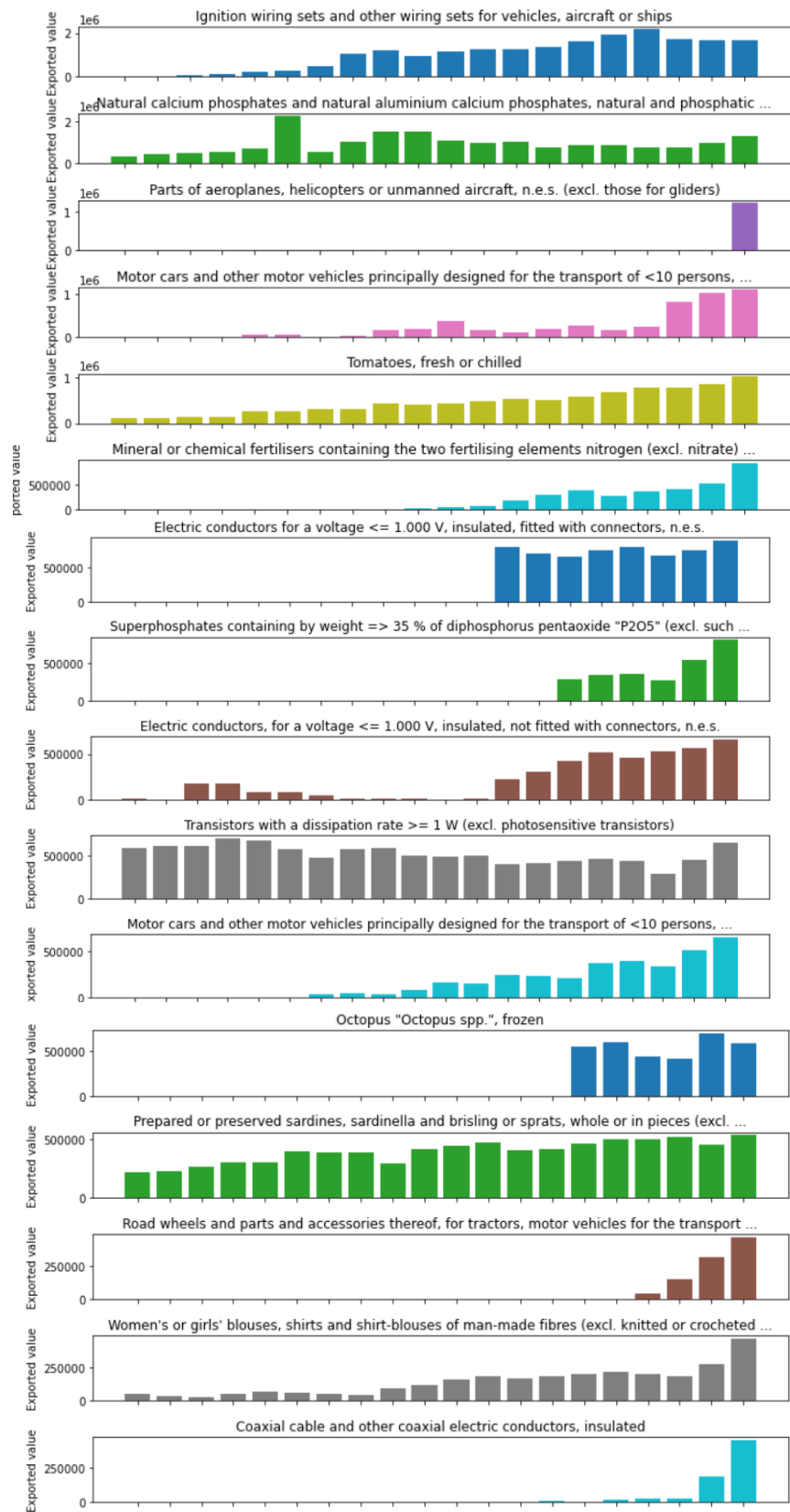


FIGURE 2.7 – Visualisation des 15 produits suivants

D'après ces graphes qui représentent juste 20 produits sur 5324, nous pouvons constater une grande variation entre les valeurs d'exportation au fil des années pour chaque produit.

Cette analyse montre une diversité de comportements parmi les produits exportés au cours de la période considérée.

- Certains produits ont maintenu une présence constante sur le marché mondial, avec des exportations enregistrées chaque année et une tendance à la hausse des valeurs exportées au fil du temps. Cette stabilité dans les exportations suggère une demande persistante pour ces produits sur le marché international, ainsi qu'une croissance continue de leur popularité ou de leur utilité.
- D'autres produits ont présenté des tendances plus volatiles, avec des fluctuations significatives dans les valeurs exportées d'une année à l'autre. Certains de ces produits ont initialement enregistré des exportations à grande échelle, suivies d'une diminution des valeurs exportées au cours des années intermédiaires, pour ensuite observer une reprise dans les années les plus récentes. Cette variabilité dans les tendances d'exportation peut être attribuée à des facteurs tels que la concurrence sur le marché, les changements de politique commerciale ou les fluctuations des conditions économiques mondiales.
- Certains produits ont émergé comme des exportations récentes, n'apparaissant dans les données qu'au cours des dernières années de la période considérée. Ces produits ont peut-être bénéficié de nouveaux développements technologiques, de l'ouverture de nouveaux marchés ou de changements dans les préférences des consommateurs, ce qui a conduit à leur introduction tardive sur la scène mondiale de l'exportation.

### 2.2.2 contribution produit :

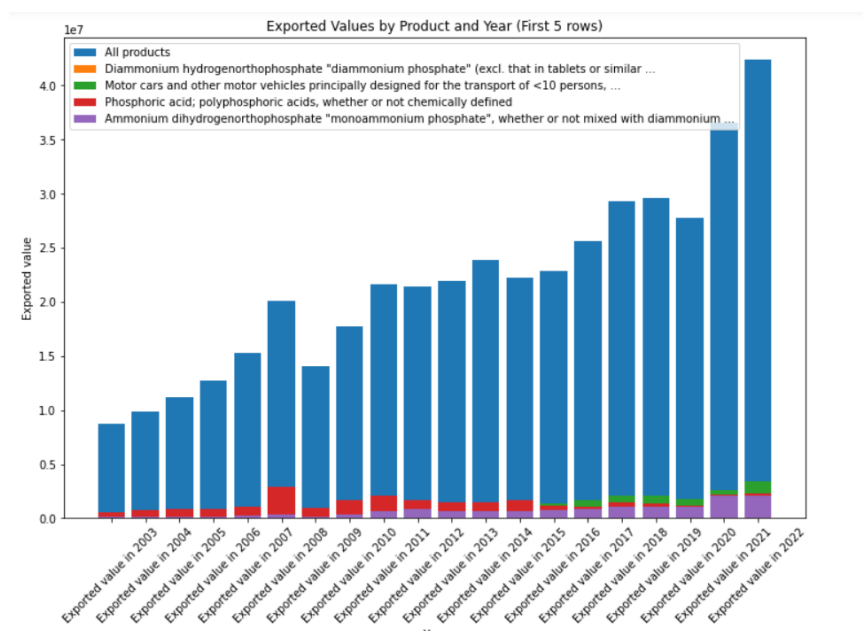


FIGURE 2.8 – Contribution des premiers produits

La visualisation représente la contribution de chaque produit aux valeurs exportées par année, permettant ainsi de comprendre la part relative de chaque produit dans le total des exportations annuelles.

Si on prend la figure 2.9 qui représente 4 produits.

Pour chaque année, on observe une répartition des exportations entre les différents produits avec des produits qui représentent une part significative des exportations totales et continue de dominer la contribution aux exportations totales.

Ainsi que d'autres produits commencent également à jouer un rôle significatif au fil des années

### 2.2.3 Heat map

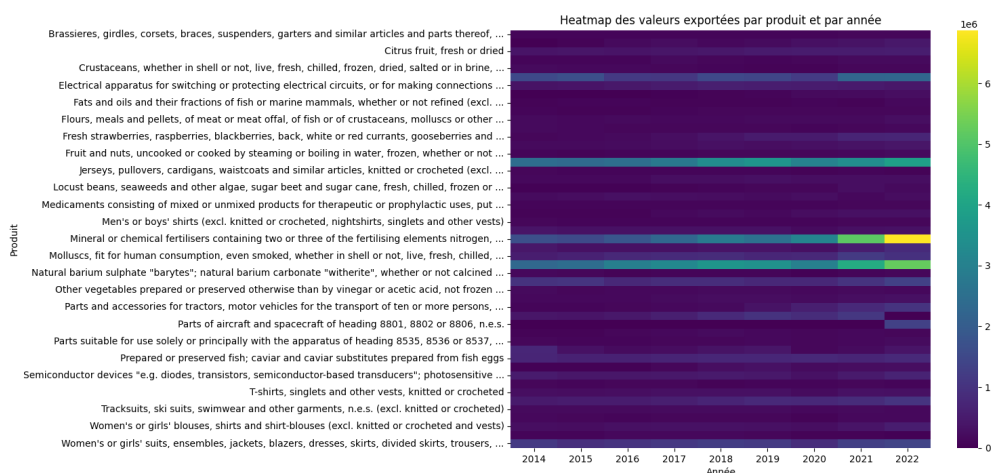


FIGURE 2.9 – Heat map

La heatmap montre l'évolution de la valeur les exportations par produit sur la période 2014-2022. Le produit dont la valeur d'exportation est la plus élevée en 2022 est "Les engrais minéraux ou chimiques contenant deux ou trois éléments fertilisants azotés". Ce produit a également connu la plus forte croissance des exportations depuis 2014. Un autre produit notable est «Fraises fraîches, framboises, mûres, dos, groseilles blanches ou rouges, groseilles et autres baies» qui a connu une augmentation importante de sa valeur d'exportation depuis 2017. Dans l'ensemble, la carte thermique suggère que les exportations ont été relativement stables sur la période 2014-2022, certains produits connaissant une forte croissance tandis que d'autres ont vu leur valeur diminuer.

# Chapitre 3

## Nettoyage et Préparation des Données

Dans ce chapitre, nous décrivons les différentes techniques de nettoyage et de préparation des données que nous avons utilisées pour assurer la qualité et la pertinence de notre ensemble de données. L'objectif est de transformer les données brutes en un format adapté à l'analyse et à la modélisation.

### 3.1 Techniques de nettoyage des données

Le nettoyage des données est une étape cruciale qui consiste à identifier et corriger les erreurs, incohérences et valeurs manquantes dans les données. Cela permet d'améliorer la qualité des données et de garantir des résultats plus fiables lors de l'analyse et de la modélisation.

#### 3.1.1 Réduction du nombre de colonnes :

Nous avons opté pour une réduction du nombre de colonnes, en sélectionnant uniquement celles correspondant aux années de 2012 à 2022 au lieu des années de 2003 jusqu'à 2022, et cela pour plusieurs raisons :

1. **Pertinence temporelle :** Nous avons choisi de nous concentrer sur les données les plus récentes, c'est-à-dire celles des années les plus récentes, car elles reflètent mieux les tendances actuelles du marché. Les données plus anciennes peuvent être moins pertinentes pour notre analyse, car les conditions économiques et les préférences des consommateurs peuvent avoir changé au fil du temps.
2. **Précision de la prévision :** En se limitant aux années les plus récentes, nous pouvons obtenir des prévisions plus précises en utilisant des données plus récentes et donc plus représentatives des conditions actuelles du marché. Cela peut améliorer la qualité de nos modèles de prévision et de prédiction, car ils seront basés sur des données plus actuelles et pertinentes.
3. **Simplification de l'analyse :** En réduisant le nombre de colonnes à celles des années de 2012 à 2022, nous simplifions notre ensemble de données et facilitons son analyse. Cela nous permet de nous concentrer sur les données les plus pertinentes et d'éviter la complexité inutile associée à l'inclusion de données plus anciennes potentiellement moins significatives.



4. **Optimisation des ressources** : En excluant les données plus anciennes, nous réduisons également la charge de calcul associée à l'analyse de données et à la formation de modèles. Cela peut être particulièrement important lorsque l'ensemble de données est volumineux, car cela permet d'optimiser l'utilisation des ressources informatiques disponibles.

Code	0		
Product label	0		
Exported value in 2003	0		
Exported value in 2004	0		
Exported value in 2005	0		
Exported value in 2006	0		
Exported value in 2007	0		
Exported value in 2008	0		
Exported value in 2009	0		
Exported value in 2010	0	Code	0
Exported value in 2011	0	Product label	0
Exported value in 2012	0	Exported value in 2012	0
Exported value in 2013	0	Exported value in 2013	0
Exported value in 2014	0	Exported value in 2014	0
Exported value in 2015	0	Exported value in 2015	0
Exported value in 2016	0	Exported value in 2016	0
Exported value in 2017	0	Exported value in 2017	0
Exported value in 2018	0	Exported value in 2018	0
Exported value in 2019	0	Exported value in 2019	0
Exported value in 2020	0	Exported value in 2020	0
Exported value in 2021	0	Exported value in 2021	0
Exported value in 2022	0	Exported value in 2022	0
dtype: int64		dtype: int64	

FIGURE 3.1 – Diminution de nombre de colonnes

La figure 3.1 représente le passage de **"2003 à 2022"** vers **"2012 à 2022"** la réduction du nombre de colonnes aux années de 2012 à 2022 est une décision stratégique qui nous permet de concentrer notre analyse sur les données les plus récentes et les plus pertinentes pour notre objectif de prévision de la demande des produits du terroir marocain.

### 3.1.2 Suppression des produits avec des valeurs d'exportation nulles :

Lors de l'analyse initiale de notre ensemble de données, nous avons constaté que certaines entrées présentaient des valeurs nulles pour toutes les années de 2012 à 2022. Ces valeurs nulles indiquent que ces produits n'ont pas été exportés du tout pendant cette période. Par conséquent, nous avons décidé de supprimer ces entrées pour plusieurs raisons :

1. **Contribution insignifiante à l'analyse** : Les produits qui n'ont aucune valeur d'exportation sur une longue période ne fournissent aucune information utile pour notre analyse de prévision de la demande. Leur inclusion dans l'ensemble de données pourrait diluer les résultats et rendre les modèles moins précis.
2. **Amélioration de la qualité des données** : Les données nulles peuvent introduire du bruit dans l'analyse et les modèles de prédiction. En les supprimant, nous

améliorons la qualité globale de notre ensemble de données, ce qui peut mener à des prévisions plus précises et plus fiables.

### 3. Optimisation des ressources :

- Travailler avec un ensemble de données épuré et pertinent permet d’optimiser l’utilisation des ressources informatiques. Les algorithmes d’apprentissage automatique peuvent s’entraîner plus rapidement et de manière plus efficace lorsque l’ensemble de données est débarrassé des valeurs nulles et des entrées non significatives.
- La visualisation des données devient également plus facile et plus intuitive, car nous nous concentrons uniquement sur les produits qui présentent des comportements intéressants et exploitables.

4. **Focus sur les produits actifs :** En nous concentrant sur les produits qui ont des valeurs d’exportation enregistrées, nous pouvons mieux comprendre les dynamiques actuelles du marché et élaborer des stratégies de développement et de commercialisation plus pertinentes. Cela nous permet également de détecter les tendances, les variations saisonnières, et les anomalies de manière plus précise.

## 3.2 Transformation des données

La transformation des données est une étape cruciale dans la préparation des données pour l’analyse et la modélisation. Elle nous permet de structurer les données de manière à faciliter l’analyse temporelle et la prévision de la demande des produits du terroir marocain.

### 3.2.1 Objectifs de la transformation des données

#### Organiser les données pour les adapter aux analyses de séries temporelles :

- Les séries temporelles sont essentielles pour analyser et prévoir les tendances au fil du temps. En structurant les données en séries temporelles, nous pouvons appliquer des méthodes statistiques et des algorithmes d’apprentissage automatique spécifiques à ce type de données.
- Les données initiales étaient sous forme tabulaire, avec chaque colonne représentant une année d’exportation pour un produit donné. Pour effectuer des analyses temporelles, il est nécessaire de convertir ces données en un format où chaque ligne représente une observation unique à un moment donné pour un produit spécifique.

#### Faciliter l’utilisation des données dans les algorithmes de prévision :

- Les algorithmes de prévision, tels que ARIMA, SARIMA, Prophet et les réseaux de neurones LSTM, nécessitent des données structurées en séries temporelles pour fonctionner efficacement. En structurant les données de cette manière, nous permettons une intégration plus fluide avec ces modèles.
- La conversion des données en séries temporelles simplifie également le processus d’entraînement des modèles et l’évaluation des performances, car les algorithmes peuvent exploiter la continuité et les tendances des données chronologiques.

## Optimiser l'accès et la manipulation des données :

Créer une structure de données qui permet une manipulation facile et efficace lors de l'analyse et de la modélisation :

- Une structure de données bien définie facilite les opérations de filtrage, d'agrégation et de transformation nécessaires pour les analyses avancées.
- En convertissant les données en un DataFrame de séries temporelles, nous avons simplifié l'accès et la manipulation des données pour les étapes ultérieures, telles que la visualisation et la modélisation.
- Cette organisation permet également d'exécuter des opérations de transformation en bloc, ce qui améliore l'efficacité du traitement des données et réduit les erreurs potentielles.

### 3.2.2 Étapes de transformation

#### Extraction des produits uniques

Nous avons commencé par extraire les noms des produits uniques présents dans notre ensemble de données. Cette étape est cruciale pour identifier chaque produit individuellement et suivre ses exportations au fil des années. Puisque l'extraction des produits uniques permet de créer une base de données structurée où chaque produit peut être analysé séparément.

```
1 # Sauter la première ligne si elle ne contient pas de
   donn es pertinentes
2 dt = df_filtered.iloc[1:]
3
4 # Extraire les noms de colonnes pour s'assurer qu'ils sont
   corrects
5 dt.columns = [col.strip() for col in df_filtered.columns]
6
7 # Obtenir la liste des produits
8 products = dt['Product label'].unique()
```

Cette étape assure que chaque produit a des enregistrements pour chaque année de la période d'analyse, même si certaines valeurs d'exportation sont manquantes. Cela garantit une continuité dans les séries temporelles et permet d'appliquer des méthodes de prévision de manière cohérente et fiable.

#### Création d'une liste d'années

Nous avons généré une liste des années allant de 2012 à 2023. Cette période correspond à la plage temporelle retenue pour notre analyse, car elle contient les données les plus récentes et pertinentes pour la prévision.

```
1 years = np.arange(2012, 2023)
```

#### Initialisation d'un DataFrame pour les séries temporelles

Nous avons initialisé un DataFrame vide pour stocker les données transformées sous forme de séries temporelles. Ce DataFrame sert de structure de base pour organiser les données de manière à faciliter les analyses temporelles.

```
1 time_series = pd.DataFrame()
```

## Remplissage du DataFrame de séries temporelles

Pour chaque produit et chaque année, nous avons extrait les valeurs d'exportation correspondantes et les avons organisées dans un format structuré. Cette étape implique plusieurs sous-étapes :

1. **Identification des valeurs d'exportation** : Pour chaque produit et chaque année, nous avons cherché la colonne correspondante dans le DataFrame initial pour obtenir les valeurs.
2. **Création des enregistrements** : Pour chaque combinaison produit-année, nous avons créé un enregistrement individuel dans le DataFrame de séries temporelles.
3. **Concaténation des enregistrements** : Nous avons ajouté chaque enregistrement au DataFrame de séries temporelles de manière itérative.

```
1
2 for product in products:
3     for year in years:
4         # Extraire les valeurs correspondant au produit et
         l'annee
5         value_column = f'Exported value in {year}'
6         if value_column in dt.columns:
7             value = dt.loc[dt['Product label'] == product,
            value_column].values
8             if len(value) > 0:
9                 value = value[0]
10            else:
11                value = np.nan # ou 0 selon votre besoin
12        else:
13            value = np.nan # ou 0 selon votre besoin
14
15        # Créer un enregistrement pour la serie temporelle
16        data = pd.DataFrame({'Product': [product], 'Year': [
            year], 'Value': [value]})
17        time_series = pd.concat([time_series, data],
            ignore_index=True)
18
19 # Reinitialiser l'index
20 time_series = time_series.reset_index(drop=True)
```

La transformation de nos données en séries temporelles nous facilite l'application des techniques d'analyse de séries temporelles, telles que les modèles ARIMA, SARIMA, Prophet et LSTM, et nous permet également d'effectuer des analyses de tendances, de saisonnalités et de cycles économiques de manière plus précise et efficace.

### 3.2.3 Résultats de la transformation des données

Après avoir transformé nos données en séries temporelles, nous avons obtenu un DataFrame structuré et prêt pour les analyses et les prévisions. Voici un aperçu des résultats obtenus et de leur pertinence pour notre étude.

Le DataFrame final, nommé `time_series`, contient les colonnes suivantes :

- **Product** : Le nom du produit exporté.
- **Year** : L'année de l'exportation.
- **Value** : La valeur des exportations pour ce produit et cette année.

Ce format permet de visualiser facilement les exportations de chaque produit au fil des années, facilitant ainsi les analyses temporelles et les prévisions.

## Nouvelle Data\_frame

Notre nouvelle Data\_frame `time_series` est composé de 49170 lignes et 3 colonnes comme il est représenté dans la figure 3.2

```
print(time_series.head())
```

		Product	Year	Value
0	Diammonium hydrogenorthophosphate	"diammonium ...	2012	1147443.0
1	Diammonium hydrogenorthophosphate	"diammonium ...	2013	729453.0
2	Diammonium hydrogenorthophosphate	"diammonium ...	2014	776074.0
3	Diammonium hydrogenorthophosphate	"diammonium ...	2015	603765.0
4	Diammonium hydrogenorthophosphate	"diammonium ...	2016	607097.0

```
print(time_series.tail())
```

		Product	Year	Value
49165	Construction sets and constructional toys (exc...		2018	0.0
49166	Construction sets and constructional toys (exc...		2019	0.0
49167	Construction sets and constructional toys (exc...		2020	0.0
49168	Construction sets and constructional toys (exc...		2021	0.0
49169	Construction sets and constructional toys (exc...		2022	0.0

```
time_series.shape
```

```
(49170, 3)
```

FIGURE 3.2 – Visualisation de la nouvelle Data\_frame

La figure 3.3 montre la valeur d'exportation de différents produits sur une période de dix ans, de 2012 à 2022 avec :

- **Axe vertical** : Représente la valeur d'exportation en USD, avec une échelle logarithmique indiquée par "1e6", signifiant que les chiffres doivent être multipliés par un million.
- **Axe horizontal** : Indique les années, de 2012 à 2022.
- **Lignes de couleur** : Chaque ligne représente une catégorie de produit différente, avec des fluctuations montrant les changements dans les valeurs d'exportation au fil des ans.

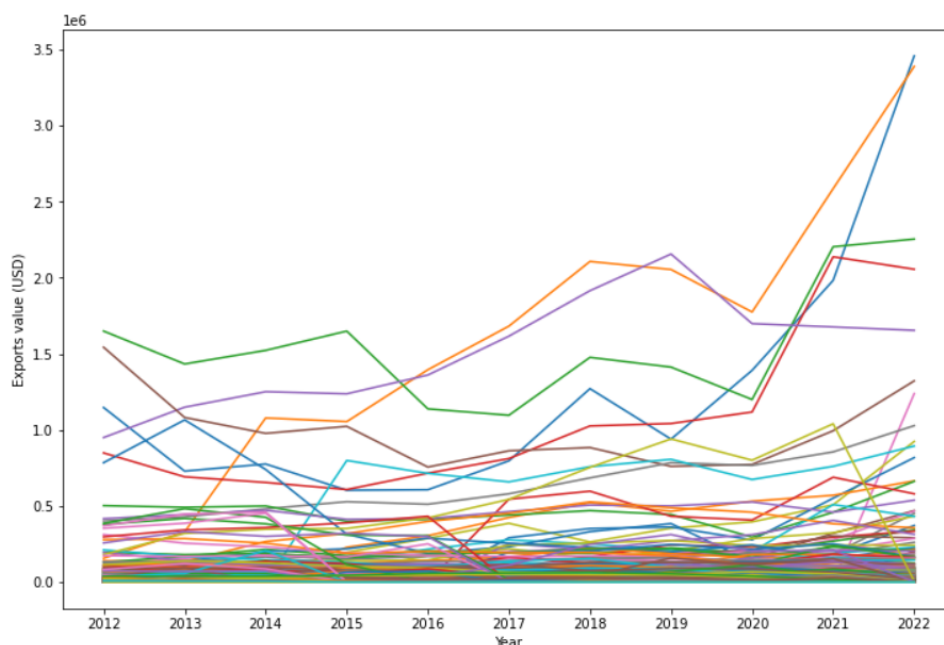


FIGURE 3.3 – Série temporelle pour des produits au fil des années

La figure 3.4 représente plusieurs cellules où chaque cellule indique la valeur d'exportation pour un produit donné durant une année spécifique, avec une échelle de couleur à droite montrant que les couleurs plus foncées représentent des valeurs plus élevées. Donc cette carte est utile pour identifier rapidement les tendances, comme les produits ayant constamment des valeurs d'exportation élevées ou les années marquant des augmentations ou des diminutions significatives des exportations.

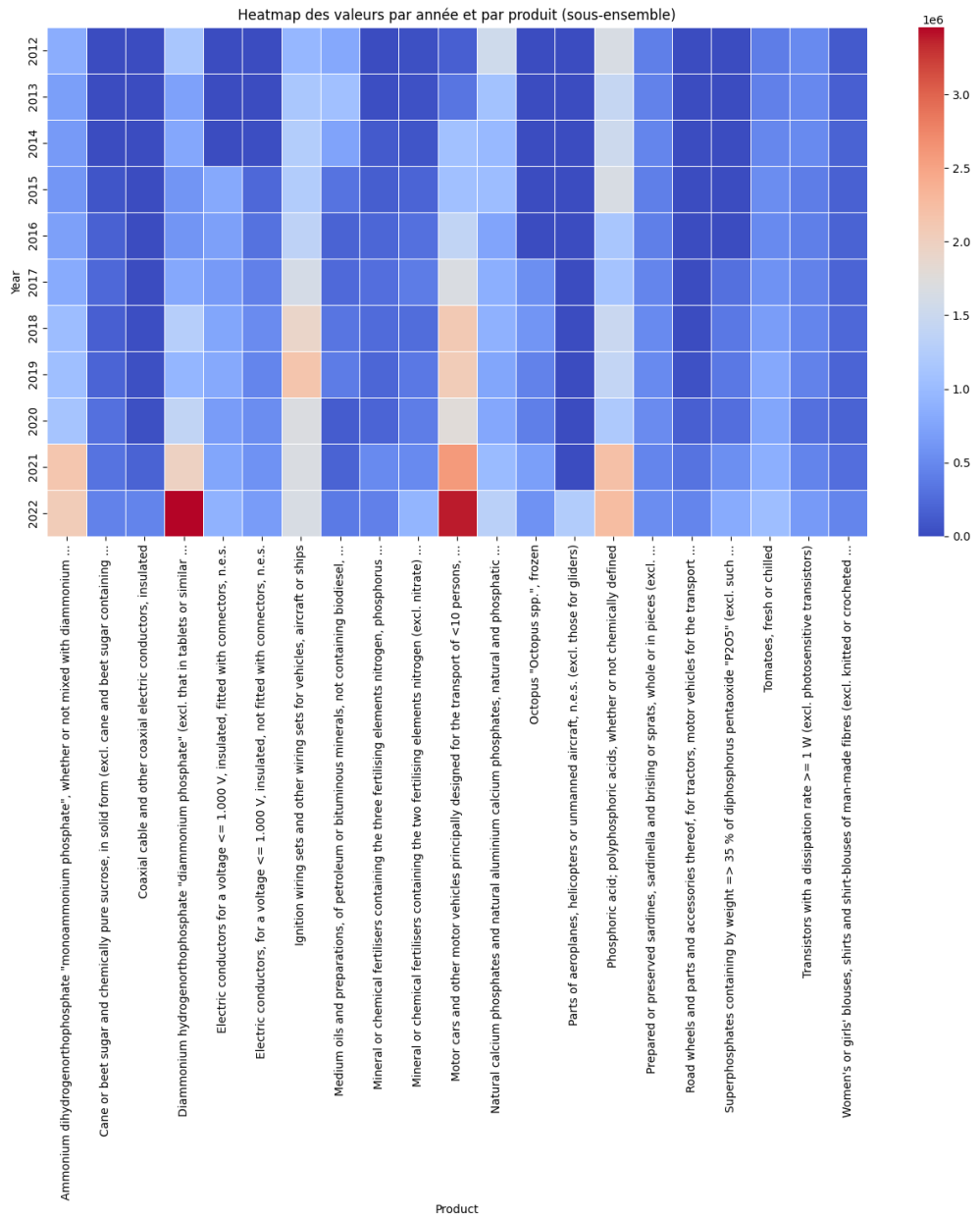


FIGURE 3.4 – Heatmap des valeurs par année par produit

# Chapitre 4

## Sélection et Justification des Algorithmes de Prévision

Après avoir nettoyé et préparé nos données, nous avons sélectionné les algorithmes de prévision les plus appropriés pour notre analyse. Ce chapitre décrit les algorithmes choisis et les justifications de leur sélection.

### 4.1 Algorithmes de Prévision Sélectionnés

Pour effectuer des prévisions précises et robustes, nous avons choisi d'utiliser les algorithmes suivants :

#### 4.1.1 ARIMA[1]

Nous avons utilisé un modèle ARIMA (AutoRegressive Integrated Moving Average) pour entraîner et faire des prédictions sur les séries temporelles de chaque produit. Il est particulièrement adapté pour les séries stationnaires après différenciation.

##### Paramètres du Modèle :

- **Auto-régression (AR)** : Cela fait référence à la composante autorégressive du modèle, qui capture les dépendances linéaires entre les valeurs passées et la valeur actuelle de la série temporelle
- **Intégration (I)** : Cela fait référence à la composante d'intégration du modèle, qui représente le nombre de différences nécessaires pour rendre la série temporelle stationnaire. Cela permet de traiter les séries temporelles non stationnaires en prenant des différences successives entre les valeurs.
- **Moyenne mobile (MA)** : Cela fait référence à la composante de moyenne mobile du modèle, qui capture les erreurs ou les résidus de la série temporelle après avoir pris en compte les valeurs précédentes.

Dans le code, nous avons utilisé la fonction ARIMA de statsmodels pour créer et entraîner un modèle ARIMA. Ensuite, nous avons utilisé ce modèle pour faire des prédictions sur les données de test.



### 4.1.2 LSTM Networks [4]

Les réseaux de neurones LSTM (Long Short-Term Memory) sont un type de réseau de neurones récurrents (RNN) capable de capturer les dépendances à long terme dans les séries temporelles.

#### Caractéristiques des LSTM :

- **Cellules de Mémoire** : Retiennent l'information sur des périodes prolongées.
- **Gates** : Mécanismes qui régulent le flux d'information à travers les cellules de mémoire.

### 4.1.3 Prophet (développé par Facebook) [3]

Facebook Prophet est une bibliothèque Python open-source offrant une approche intuitive et automatisée pour capturer les tendances, saisons et événements exceptionnels dans les séries temporelles.

L'une des grandes avancées technologiques récentes est la capacité à prédire les tendances futures à partir de données historiques.

## 4.2 Justification des choix des modèles

D'après la figure 4.7, on constate que les données ne sont pas linéaires et contiennent des points de changement significatifs. Cela nécessite l'utilisation d'algorithmes capables de capturer ces complexités et irrégularités dans les séries temporelles.

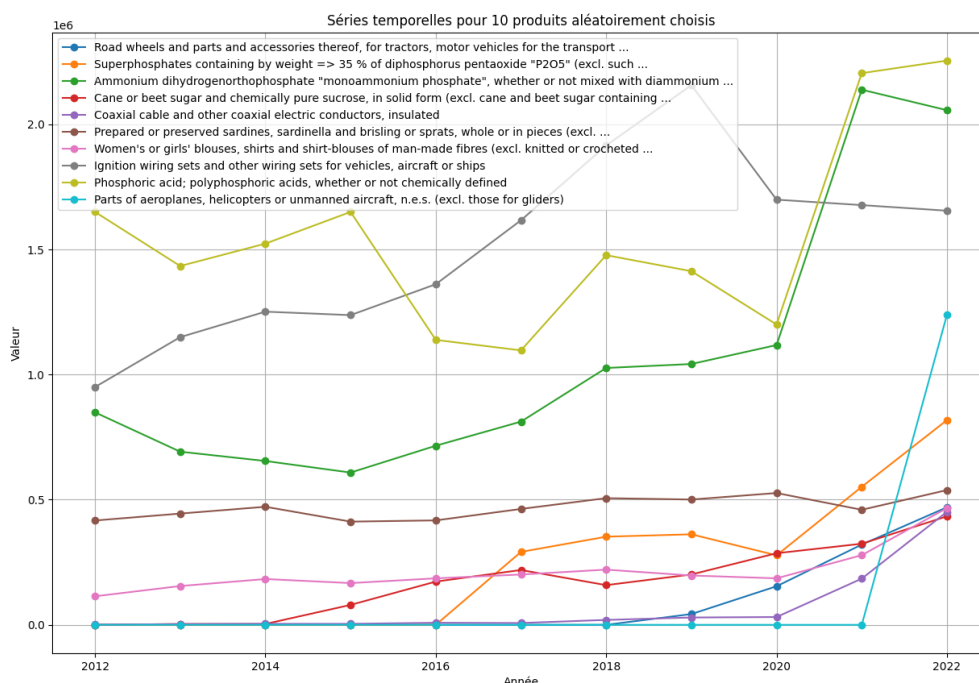


FIGURE 4.1 – Série temporelle pour 10 produits aléatoirement choisis

Alors nous avons sélectionné les modèles ARIMA, LSTM et Prophet pour les raisons suivantes :

## ARIMA :

Il est efficace pour les séries temporelles qui présentent des comportements linéaires et des tendances à court terme. Sa capacité à gérer les séries non stationnaires le rend adapté à notre ensemble de données, qui peut contenir des changements brusques dans les exportations d'une année à l'autre.

## LSTM :

Il est un modèle de réseau de neurones récurrents (RNN) spécialement conçu pour gérer les dépendances à long terme dans les données séquentielles. Il est capable de modéliser les relations complexes et non linéaires dans les séries temporelles, ce qui est crucial pour notre analyse des données d'exportation.

## Prophet :

Il est un modèle flexible développé par Facebook pour les séries temporelles présentant des tendances non linéaires avec des effets saisonniers. Il est particulièrement utile pour les données qui contiennent des cycles saisonniers et des points de changement, ce qui le rend idéal pour notre analyse des exportations, souvent influencées par des facteurs saisonniers et économiques.

## 4.3 Application aux Données d'Exportation du Maroc

Dans cette section, nous détaillons comment les techniques et les modèles de prévision ont été appliqués aux données d'exportation du Maroc. Cette application pratique nous permet d'évaluer l'efficacité de notre approche et de fournir des prévisions utiles pour la demande future des produits du terroir marocains.

### 4.3.1 Préparation des Données

La division des données a été effectuée comme suit :

- **Ensemble d'entraînement** : Comprend les données d'exportation des produits de 2012 à 2020. Cet ensemble est utilisé pour ajuster les modèles et apprendre les tendances historiques.
- **Ensemble de test** : Comprend les données d'exportation des produits de 2021 et 2022. Cet ensemble est utilisé pour évaluer la performance des modèles sur des données récentes et non vues pendant l'entraînement.

```
1 # S parer les donn es pour chaque produit
2 for product in time_series22['Product'].unique():
3     # Extraire les donn es pour le produit courant
4     product_data = time_series22[time_series22['Product'] ==
5     product]
6     # S parer les donn es en entra nement et test
7     product_train = product_data[product_data['Year'] <
    cutoff_year]
    product_test = product_data[product_data['Year'] >=
    cutoff_year]
```

### 4.3.2 Résultats des Modèles

Après avoir entraîné et testé les modèles ARIMA, LSTM et Prophet sur les données d'exportation du Maroc, nous avons comparé leurs performances en termes de RMSE (Root Mean Squared Error).

#### ARIMA :

##### Paramètres :

p (Auto-Regressive Order) : 5

d (Differencing Order) : 1

q (Moving Average Order) : 0

```
1 def forecast_product1(product, order=(5,1,0)):  
2     # Extract training and testing data  
3     df_train = product_store[product]['train']  
4     df_test = product_store[product]['test']  
5     # Fit the ARIMA model  
6     model = ARIMA(df_train['Value'], order=order)  
7     model_fit = model.fit()
```

Ces paramètres ont été choisis après une analyse initiale des données de séries temporelles et des critères d'information comme AIC (Akaike Information Criterion) pour optimiser la sélection des paramètres.

Pour chaque produit, nous avons ajusté le modèle ARIMA en utilisant les données d'entraînement et prédit les valeurs pour l'ensemble de test.

Les paramètres (5, 1, 0) ont été choisis après une série de tests pour obtenir le meilleur ajustement aux données.

#### Modèle LSTM :

##### Paramètres :

Unités LSTM : 32

Époques d'entraînement : 5

Optimizer : Adam

Loss Function : MSE (Mean Squared Error)

```
1 def train_lstm_model(train_data, test_data):  
2     # Extraire les valeurs des colonnes 'value' des DataFrames  
3     train_values = train_data['Value'].values.reshape(-1, 1,  
4     1)  
5     train_targets = train_data['Value'].values.reshape(-1, 1,  
6     1)  
7     test_values = test_data['Value'].values.reshape(-1, 1, 1)  
8  
9     model = Sequential([  
10         LSTM(32, input_shape=(1, 1)),  
11         Dense(1)  
12     ])  
13     model.compile(optimizer='adam', loss='mse')
```

11

```
model.fit(train_values, train_targets, epochs=5, verbose=0)
```

Les unités LSTM et le nombre d'époques ont été choisis en tenant compte de la complexité des données et du compromis entre performance et temps de calcul. L'optimizer Adam a été sélectionné pour sa capacité à adapter le taux d'apprentissage et à converger rapidement. Pour chaque produit, nous avons entraîné un modèle LSTM en utilisant les données d'entraînement et prédit les valeurs pour l'ensemble de test.

## Modèle Prophet :

### Paramètres :

Les paramètres par défaut de Prophet ont été utilisés, car ils sont généralement suffisants pour capturer les tendances et les saisonnalités des données économiques. Les ajustements spécifiques peuvent être faits selon les besoins, mais pour cette étude, les paramètres par défaut ont donné des résultats satisfaisants.

## Comparaison des Modèles

Pour évaluer la performance de nos modèles de prévision sur un produit spécifique de notre jeu de données, nous avons comparé les erreurs moyennes absolues (MAE) et les erreurs quadratiques moyennes (RMSE) pour trois modèles :

Modèle	RMSE	MAE
ARIMA	1283098.70	1004651.53
LSTM	2307722.82	2156090.34
Prophet	1554913.61	1296157.14

TABLE 4.1 – Comparaison des performances des modèles de prévision

### 1er exemple : ARIMA :

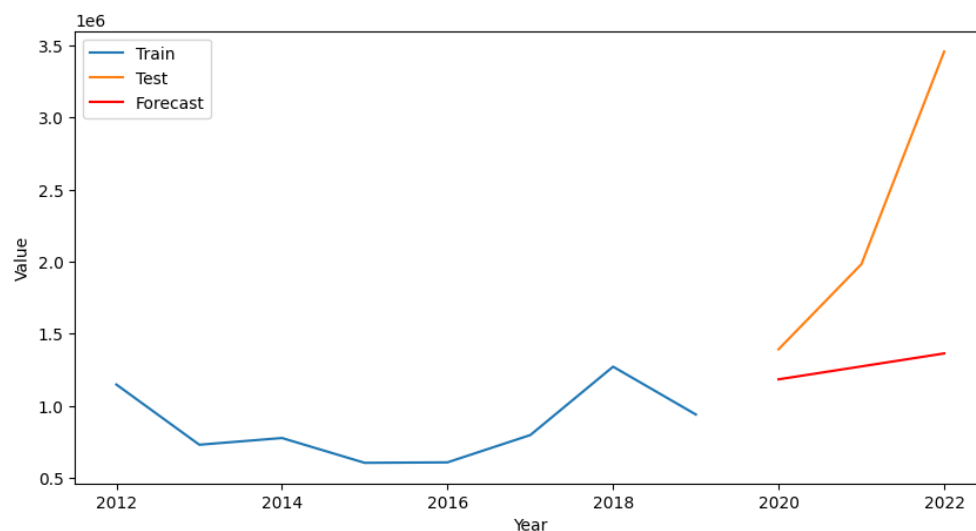


FIGURE 4.2 – Arima visualisation

**LSTM :**

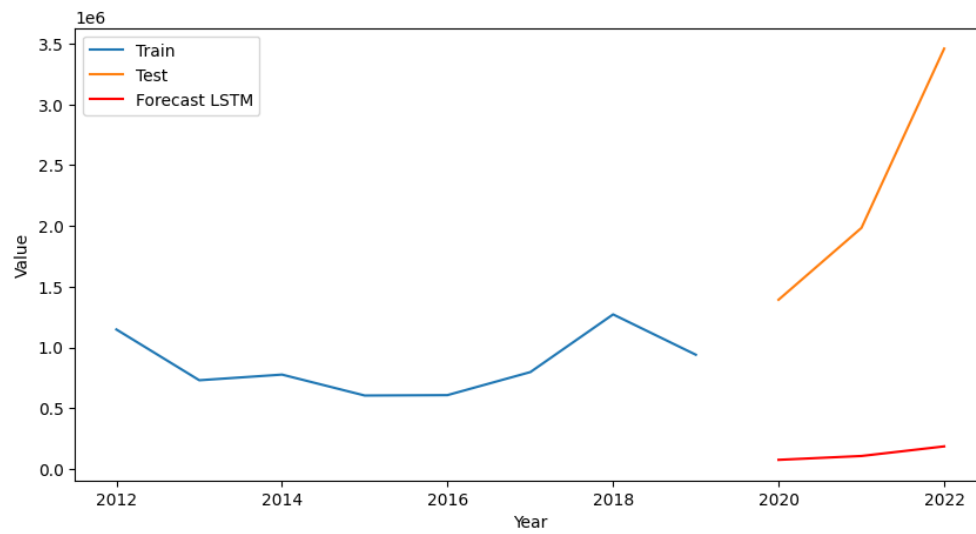


FIGURE 4.3 – Lstm visualisation

**PROPHET :**

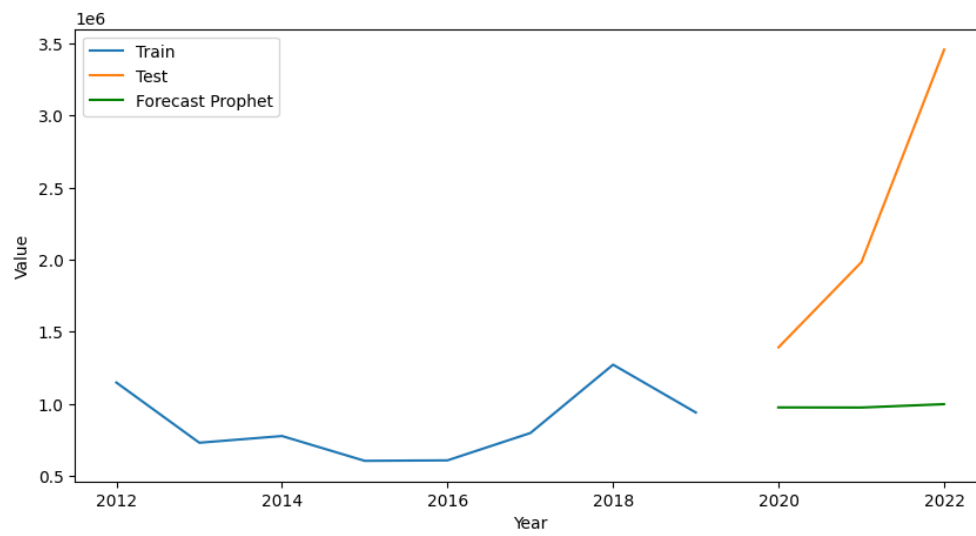


FIGURE 4.4 – Prophet visualisation

2eme exemple :  
ARIMA :

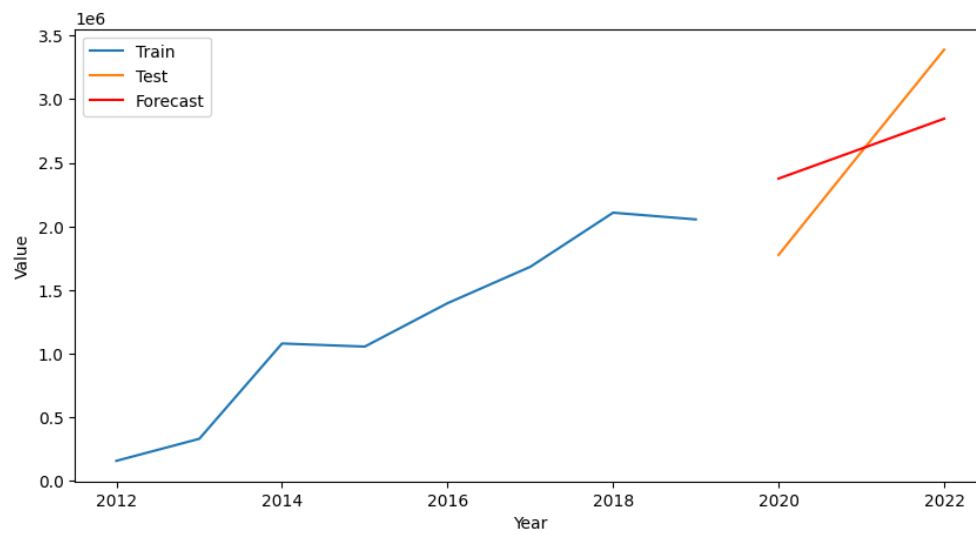


FIGURE 4.5 – Arima visualisation

LSTM :

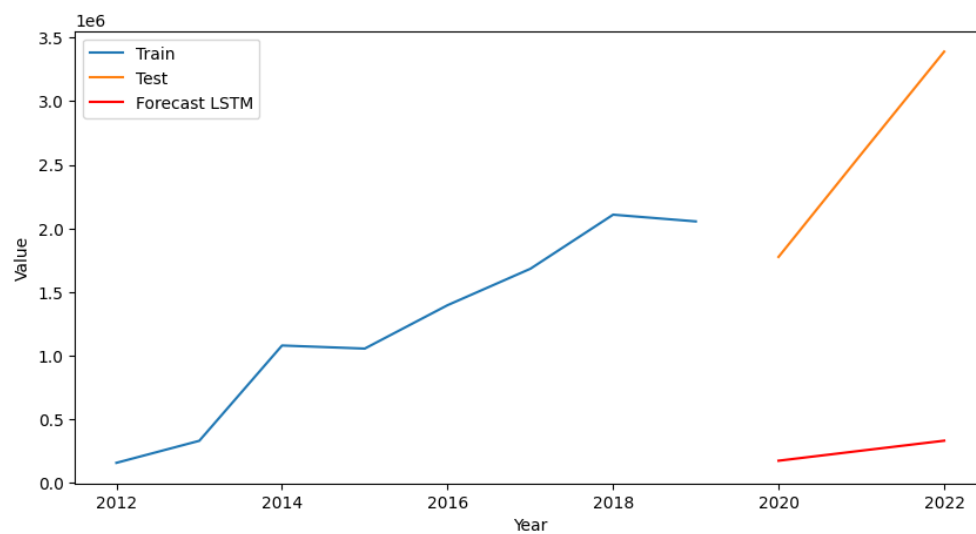


FIGURE 4.6 – Lstm visualisation

PROPHET :

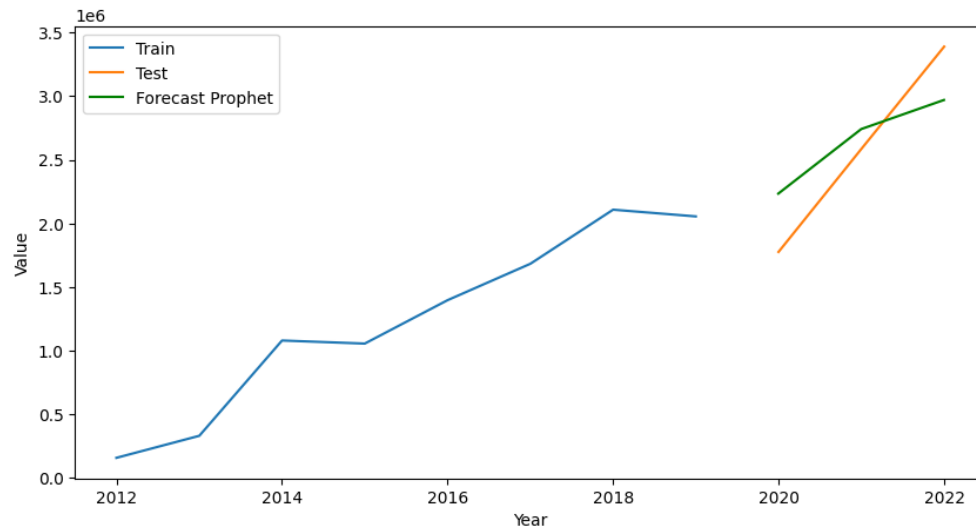


FIGURE 4.7 – Prophet visualisation

Le modèle ARIMA a la plus petite erreur moyenne absolue (MAE), mais aussi le plus grand erreur quadratique moyen (RMSE). Le modèle LSTM a des performances moins précises que les modèles ARIMA et Prophet. Le modèle Prophet se situe entre les deux autres en termes de performances. En conclusion, le modèle ARIMA semble être le meilleur choix en termes de précision de prévision.

# Conclusion

Ce rapport a exploré de manière approfondie la prévision de la demande des produits du terroir marocain, en se concentrant sur des produits emblématique.

La promotion de ces produits, reconnus pour leur qualité unique et leur valeur culturelle, est essentielle pour le développement économique du Maroc, tant sur les marchés national qu'international.

Toutefois, la demande fluctuante et la concurrence accrue exigent des stratégies bien informées pour optimiser leur production et commercialisation. Le projet a été méthodologiquement rigoureux, commençant par la collecte de données pertinentes provenant de sources fiables. Les données ont ensuite été nettoyées et transformées pour garantir leur qualité et pertinence pour l'analyse. Les visualisations initiales des données, comme les barplots et les heatmaps, ont permis de mettre en évidence des tendances et motifs cruciaux.

Pour la prévision de la demande, plusieurs modèles ont été sélectionnés et justifiés, notamment ARIMA, SARIMA, LSTM et Prophet, chacun offrant des avantages spécifiques pour capturer les tendances et les variations saisonnières dans les séries temporelles. Les résultats des modèles de prévision ont été visualisés, révélant des tendances significatives et des motifs saisonniers, fournissant des insights précieux pour l'élaboration de stratégies optimales de production et de commercialisation.

Ce projet a démontré l'importance cruciale de l'analyse de données et de la prévision pour la gestion des produits du terroir marocain. En utilisant des techniques avancées d'analyse de données et d'intelligence artificielle.



# Bibliographie

- [1] Abderrahmane BRAHIM and Mohamed BENSAADIA. *Modèle hybride pour la prévision de séries chronologiques*. PhD thesis, Université Ibn Khaldoun, 2023.
- [2] Alejandro González Mata. A comparison between lstm and facebook prophet models : a financial forecasting case study. 2020.
- [3] Mario Merveille Olufemi LIGAN. Developpement d'un systeme pour la prévision des prix des produits agricoles par l'analyse des données et l'apprentissage automatique : cas du maïs.
- [4] Bibhuti Bhusan Sahoo, Ramakar Jha, Anshuman Singh, and Deepak Kumar. Long short-term memory (lstm) recurrent neural network for low-flow hydrological time series forecasting. *Acta Geophysica*, 67(5) :1471–1481, 2019.