Rapport : Prédiction des Hospitalisations COVID-19 en France

Nom: KANGNI-SOUKPE AKOKO NADEGE

Introduction

Dans ce rapport, je présente mon travail sur l'analyse et la prédiction des hospitalisations liées au COVID-19 en France. L'objectif est d'identifier les tendances des hospitalisations, de comprendre les classes d'âge les plus vulnérables et de proposer des modèles prédictifs pour anticiper l'évolution des hospitalisations.

1. Analyse exploratoire des données

Présentation des données

Les données utilisées proviennent d'un fichier CSV contenant les taux d'hospitalisation et de soins critiques par âge et par semaine. Elles incluent notamment les colonnes suivantes :

- `jour` : date des observations
- `clage_90` : classe d'âge des patients
- `tx_indic_7J_hosp` : taux d'hospitalisation sur 7 jours
- `tx_prev_hosp` : taux prédit des hospitalisations
- `tx_indic_7J_SC` : taux de soins critiques sur 7 jours
- `tx_prev_SC` : taux prédit des soins critiques

	fra	jour	clage_90	PourAvec	tx_indic_7J_DC	tx_indic_7J_hosp	tx_indic_7J_SC	tx_prev_hosp	tx_prev_SC
0	FR	2020-03-07	0	0	0.000000	0.000000	0.00000	1.169634	0.144528
1	FR	2020-03-07	0	1	NaN	0.000000	0.00000	0.000000	0.000000
2	FR	2020-03-07	0	2	NaN	0.000000	0.00000	0.000000	0.000000
3	FR	2020-03-08	0	0	0.000000	0.000000	0.00000	1.303732	0.175818
4	FR	2020-03-08	0	1	NaN	0.000000	0.00000	0.000000	0.000000
...	...	...	...	...	...	...	...	...	...
39826	FR	2023-06-25	90	1	NaN	2.750531	0.10579	97.114915	1.375266
39827	FR	2023-06-25	90	2	NaN	1.692635	0.00000	54.270100	0.211579
39828	FR	2023-06-26	90	0	0.423159	4.125797	0.10579	151.702384	1.481055
39829	FR	2023-06-26	90	1	NaN	2.644742	0.10579	97.432284	1.269476
39830	FR	2023-06-26	90	2	NaN	1.481055	0.00000	54.270100	0.211579

39831 rows × 9 columns

2. Prétraitement des données

2.1 Analyse et nettoyage des données

Avant d'effectuer les analyses, j'ai nettoyé les données en traitant les valeurs manquantes et en convertissant les colonnes au bon format (dates, variables numériques, etc.).

```
Valeurs manquantes par colonne :
fra                0
jour               0
clage_90           0
PourAvec           0
tx_indic_7J_DC     26554
tx_indic_7J_hosp   0
tx_indic_7J_SC     0
tx_prev_hosp       0
tx_prev_SC         0
dtype: int64
```

```
[ ]  ## Suppression des valeurs manquantes par la moyenne

     df.drop(columns=['tx_indic_7J_DC'], inplace=True)
```

```
[ ]  ## Vérifier s'il reste des valeurs manquantes

     df.isna().sum()
```

	0
fra	0
jour	0
clage_90	0
PourAvec	0
tx_indic_7J_hosp	0
tx_indic_7J_SC	0
tx_prev_hosp	0
tx_prev_SC	0

dtype: int64

2.2 Transformation de la colonne `jour` en format datetime

La colonne `jour` a été convertie au format datetime pour permettre des analyses temporelles précises.

```
[ ]  # Convertir la colonne 'Jour' en datetime

     df['jour'] = pd.to_datetime(df['jour'])
```

```
[ ]  ## Vérifier si la conversion a marcher

     df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 39831 entries, 0 to 39830
Data columns (total 8 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   fra               39831 non-null  object
 1   jour              39831 non-null  datetime64[ns]
 2   clage_90          39831 non-null  int64
 3   PourAvec          39831 non-null  int64
 4   tx_indic_7J_hosp  39831 non-null  float64
 5   tx_indic_7J_SC    39831 non-null  float64
 6   tx_prev_hosp      39831 non-null  float64
 7   tx_prev_SC        39831 non-null  float64
dtypes: datetime64[ns](1), float64(4), int64(2), object(1)
memory usage: 2.4+ MB
```

2.3 Création d'une colonne `semaine`

Une nouvelle colonne `semaine` a été générée pour regrouper les données par périodes hebdomadaires, facilitant ainsi l'analyse des tendances sur le long terme.

	fra	jour	clage_90	PourAvec	tx_indic_7J_hosp	tx_indic_7J_SC	tx_prev_hosp	tx_prev_SC	semaine
0	FR	2020-03-07	0	0	0.000000	0.00000	1.169634	0.144528	1
1	FR	2020-03-07	0	1	0.000000	0.00000	0.000000	0.000000	1
2	FR	2020-03-07	0	2	0.000000	0.00000	0.000000	0.000000	1
3	FR	2020-03-08	0	0	0.000000	0.00000	1.303732	0.175818	1
4	FR	2020-03-08	0	1	0.000000	0.00000	0.000000	0.000000	1
...	...	...	...	...	...	...	...	...	...
39826	FR	2023-06-25	90	1	2.750531	0.10579	97.114915	1.375266	173
39827	FR	2023-06-25	90	2	1.692635	0.00000	54.270100	0.211579	173
39828	FR	2023-06-26	90	0	4.125797	0.10579	151.702384	1.481055	173
39829	FR	2023-06-26	90	1	2.644742	0.10579	97.432284	1.269476	173
39830	FR	2023-06-26	90	2	1.481055	0.00000	54.270100	0.211579	173

39831 rows × 9 columns

2.4 Agrégation des données par semaine et par classe d'âge

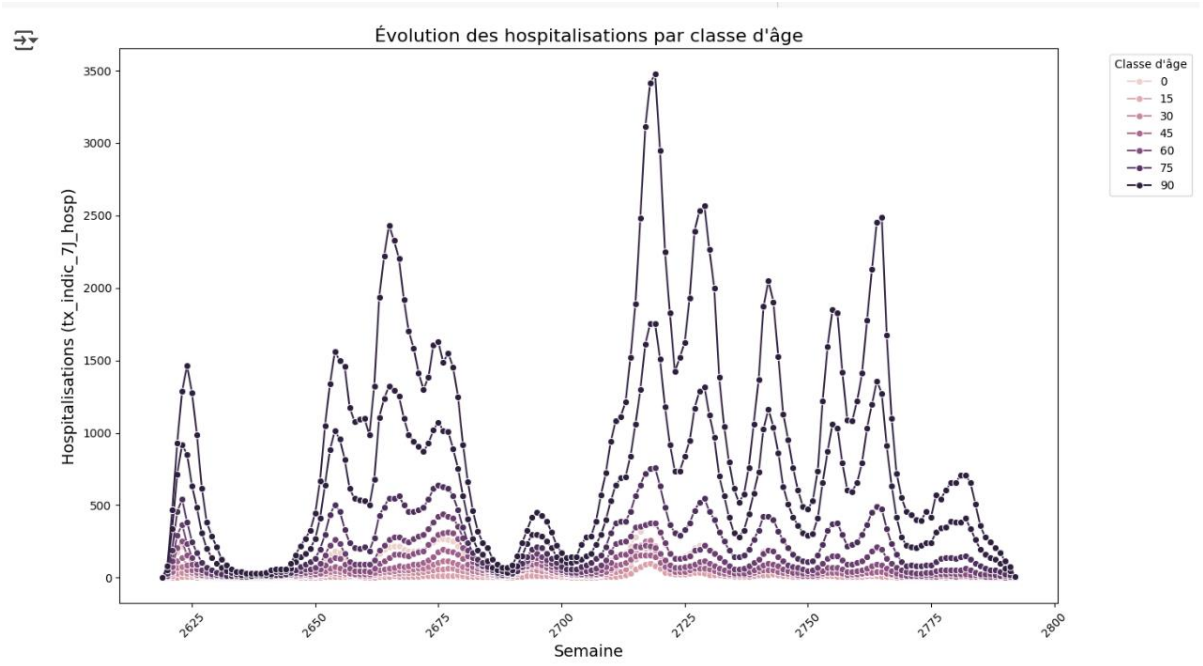
Les données ont été agrégées par semaine et par classe d'âge (`clage_90`) afin d'obtenir des indicateurs plus pertinents pour l'analyse.

```
# Agrégation par semaine et classe d'âge
df_grouped = df.groupby(['semaine', 'clage_90']).agg({
    'tx_indic_7J_hosp': 'sum',
    'tx_indic_7J_SC': 'sum',
    'tx_prev_hosp': 'sum',
    'tx_prev_SC': 'sum'
}).reset_index()

# Afficher les résultats
print(df_grouped.head())
```

3. Analyse exploratoire (EDA)

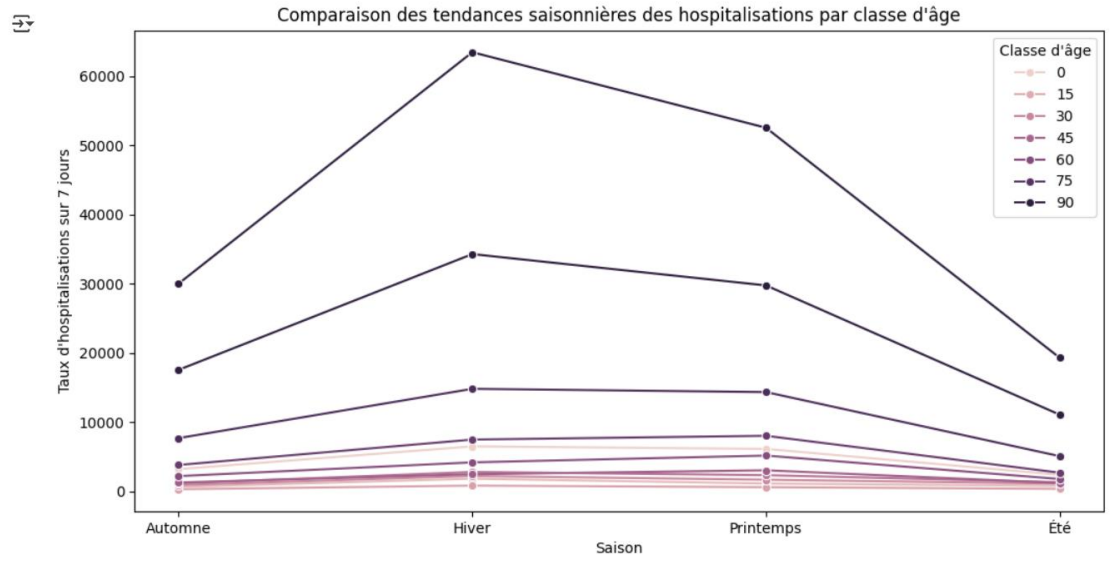
3.1 Visualisation de l'évolution des hospitalisations par classe d'âge

J'ai réalisé une graphiques montrant l'évolution des hospitalisations en fonction des classes d'âge pour identifier les groupes les plus touchés par le COVID-19.



3.2 Comparaison des tendances saisonnières

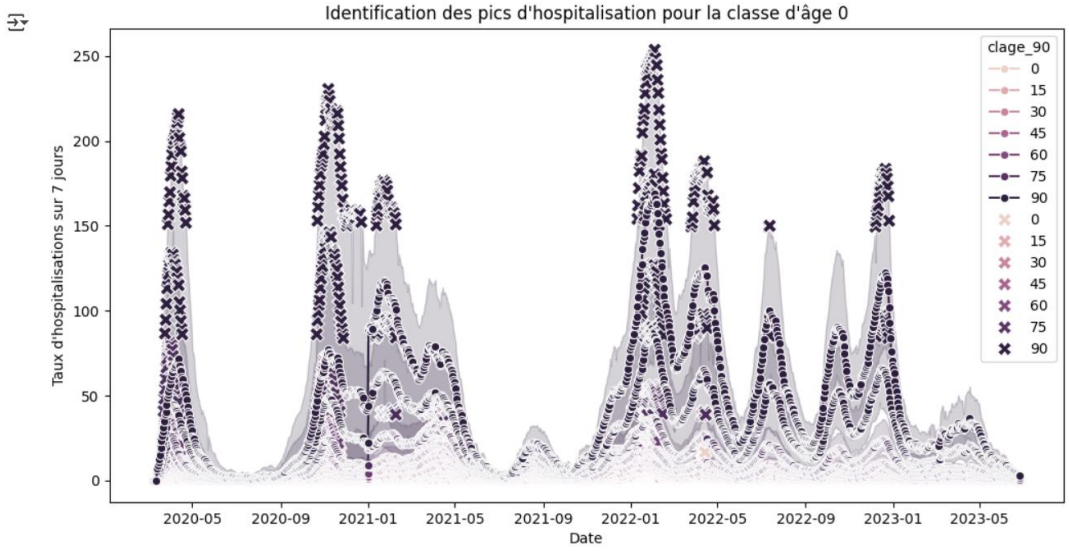
Des analyses saisonnières ont été effectuées pour observer d'éventuelles fluctuations en fonction des périodes de l'année.



3.3 Identification des pics d'hospitalisation

J'ai identifié les périodes où le nombre d'hospitalisations atteignait des sommets, ce qui permet de mieux comprendre l'impact des vagues épidémiques.

	fra	jour	clage_90	tx_indic_7J_hosp
48	FR	2020-03-23	0	18.471282
51	FR	2020-03-24	0	20.876110
54	FR	2020-03-25	0	23.444835
57	FR	2020-03-26	0	25.894362
60	FR	2020-03-27	0	28.548017



3.4 Analyse de la corrélation entre hospitalisations et soins critiques

Un heatmap de corrélation a été généré pour examiner le lien entre le taux d'hospitalisation et les admissions en soins critiques.

```
Corrélation entre hospitalisations et soins critiques : 0.5686092700220383
```

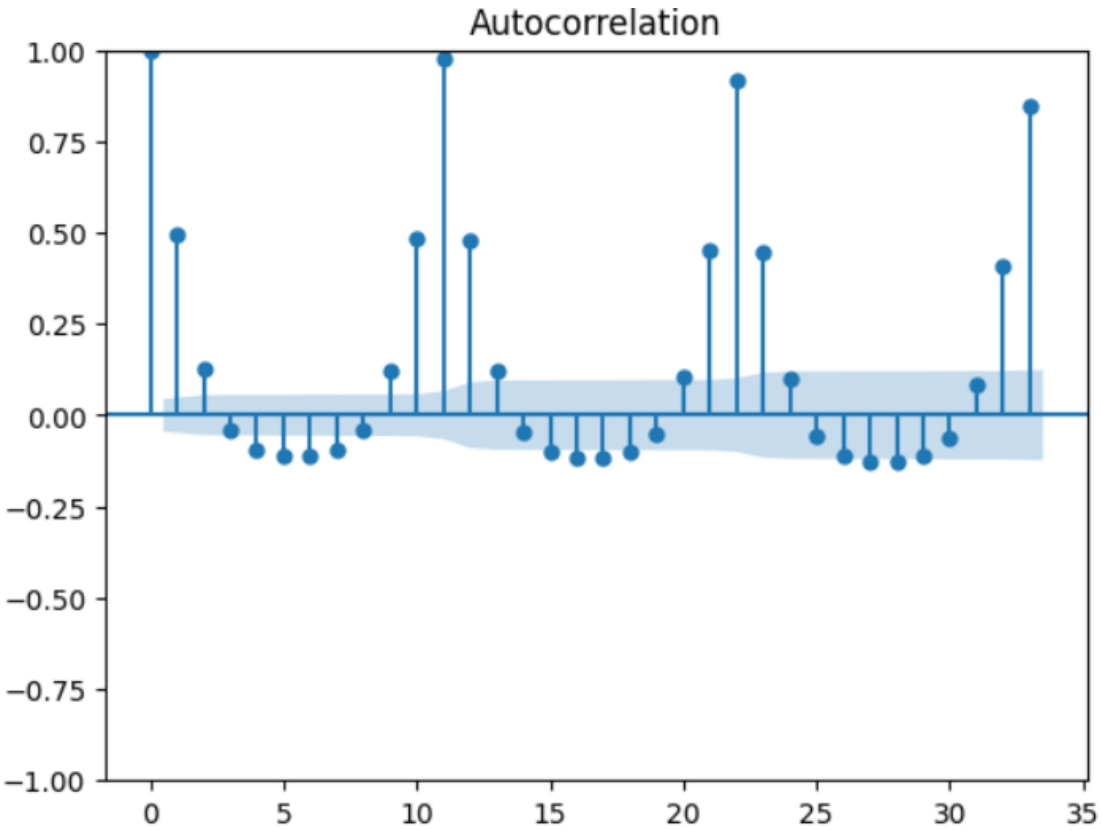
```
[ ] ## les hospitalisations et les soins critiques sont liés, mais d'autres facteurs peuvent également influencer les soins critiques
    ## indépendamment des hospitalisations
```

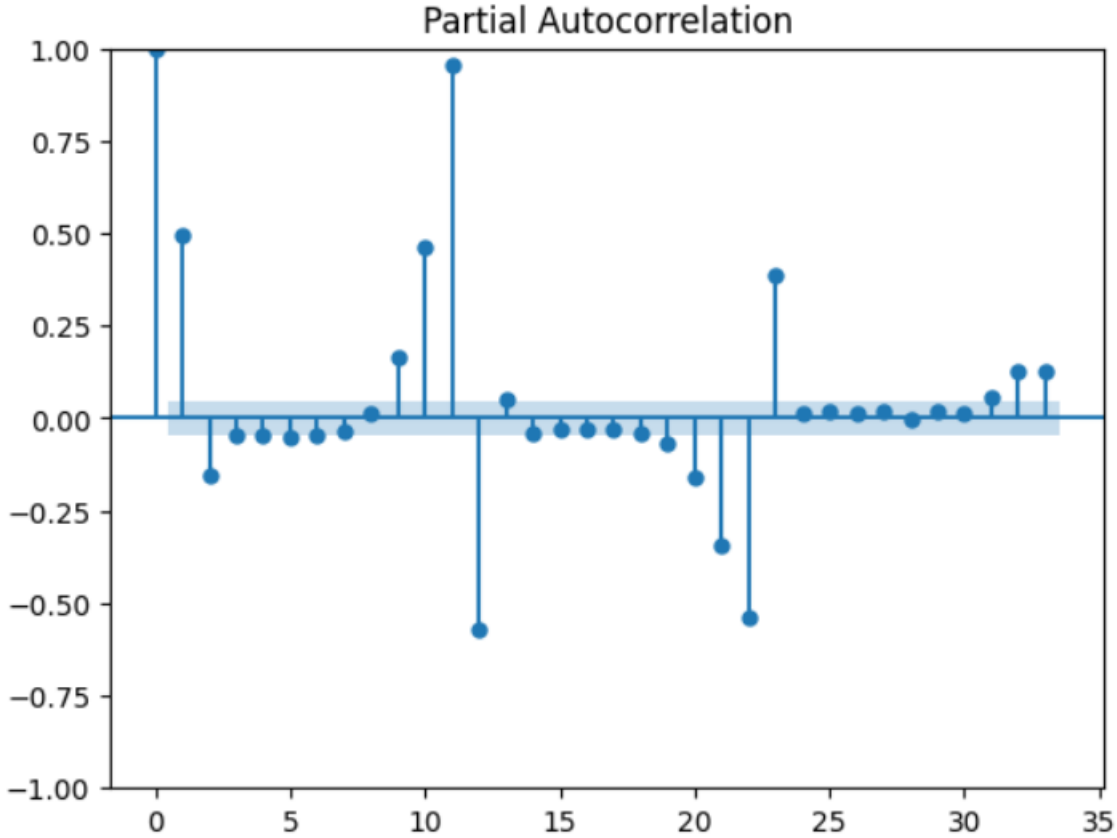
4. Modélisation prédictive

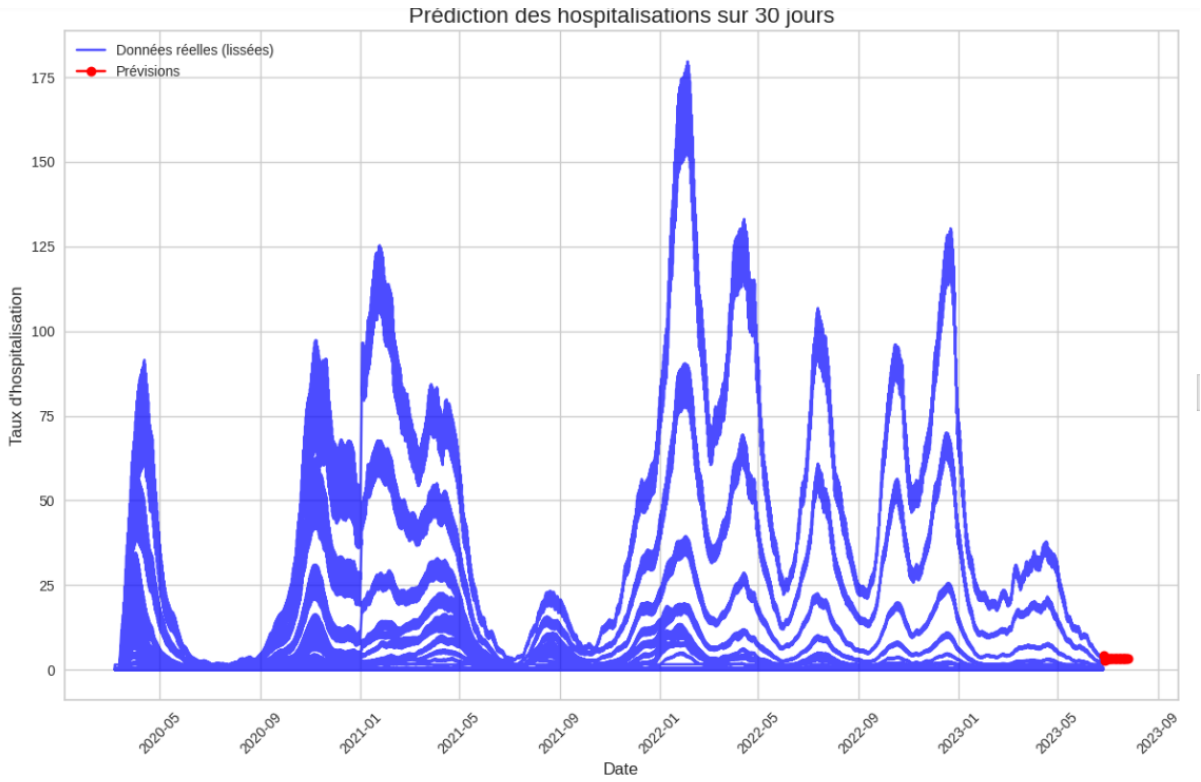
Plusieurs modèles ont été testés pour prédire les hospitalisations à court terme :

4.1 Modèle ARIMA

L'ARIMA a été utilisé pour analyser les séries temporelles et détecter les tendances sous-jacentes.

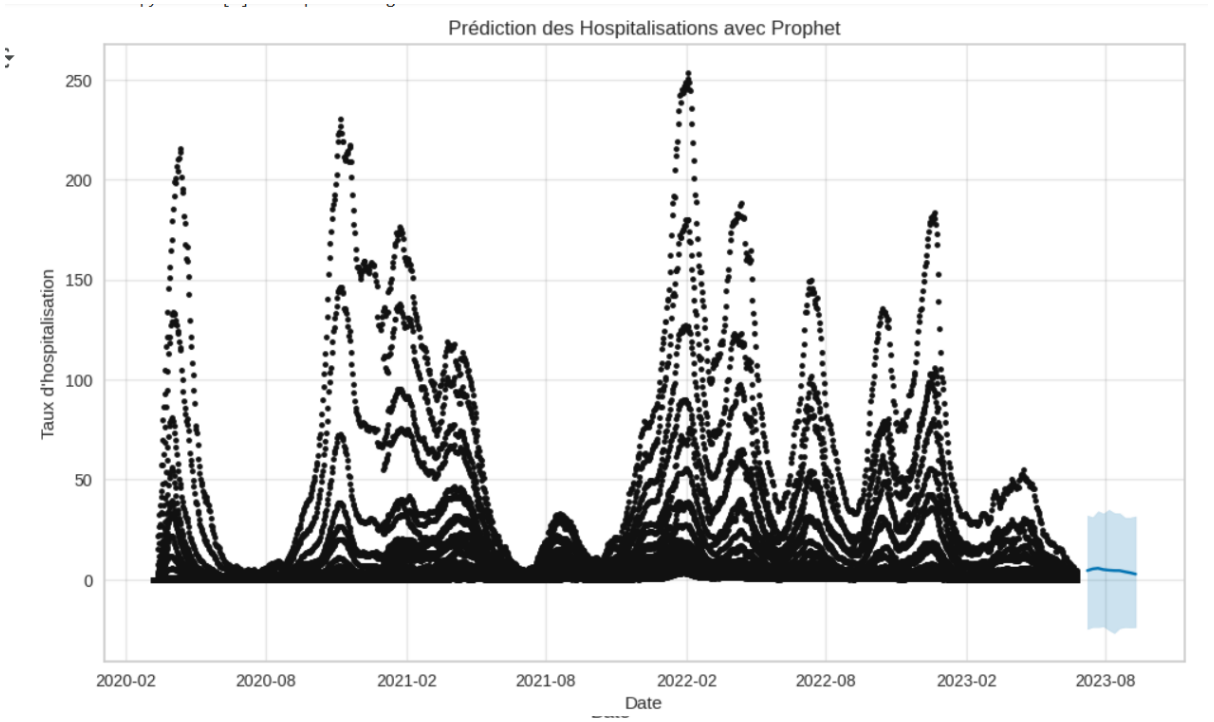

Autocorrelation
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
5
10
15
20
25
30
35

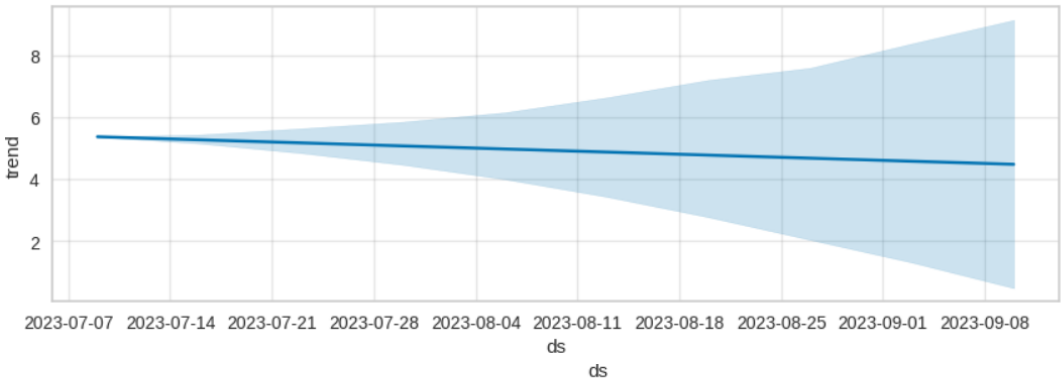

Partial Autocorrelation
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
5
10
15
20
25
30
35

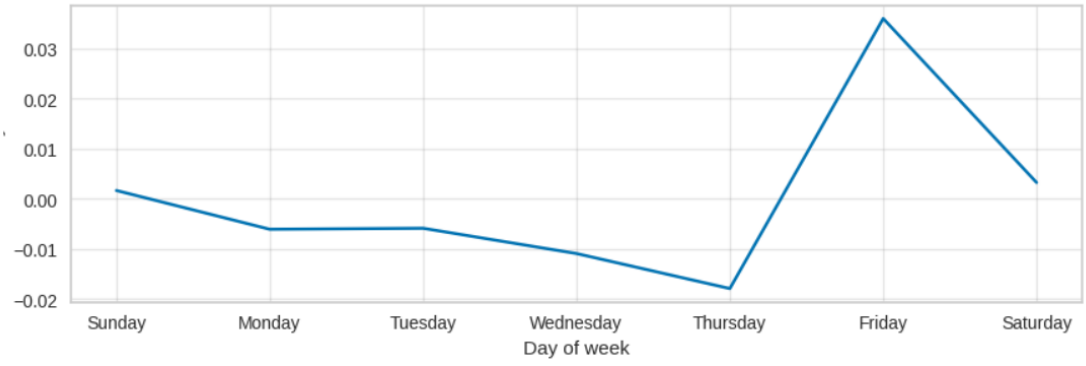


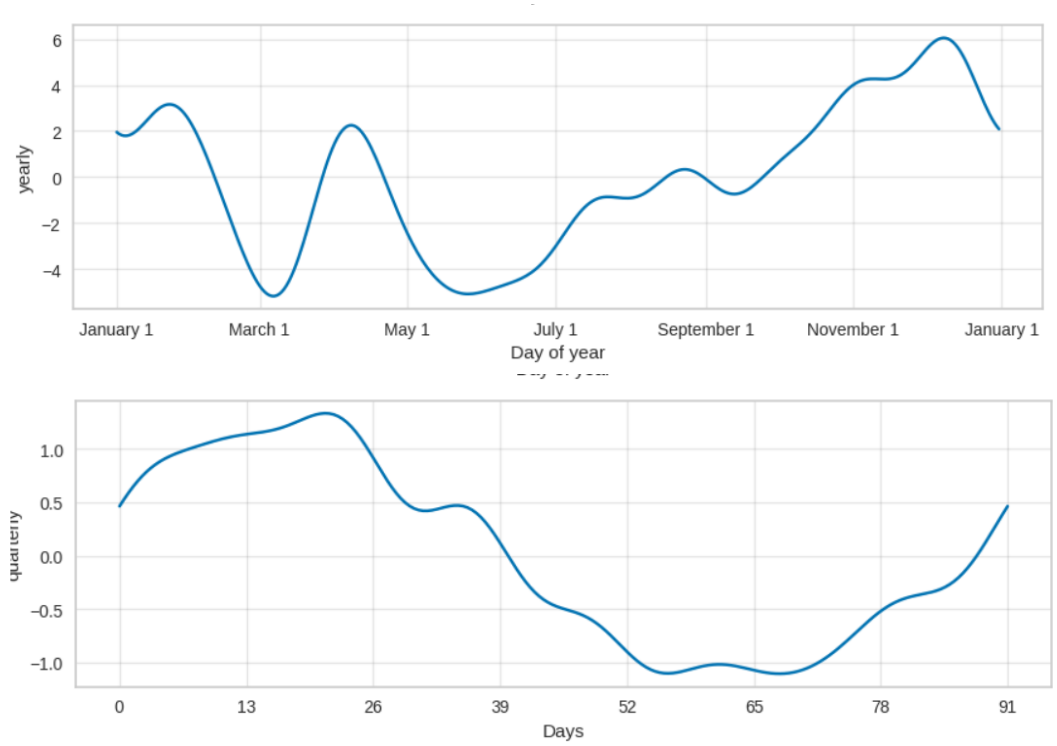
4.2 Modèle Prophet

Prophet a été testé pour intégrer la saisonnalité et améliorer la précision des prévisions.


Prédiction des Hospitalisations avec Prophet
Taux d'hospitalisation
250
200
150
100
50
0
2020-02
2020-08
2021-02
2021-08
2022-02
2022-08
2023-02
2023-08
Date


trend
8
6
4
2
2023-07-07
2023-07-14
2023-07-21
2023-07-28
2023-08-04
2023-08-11
2023-08-18
2023-08-25
2023-09-01
2023-09-08
ds
ds


0.03
0.02
0.01
0.00
−0.01
−0.02
Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Day of week



4.3 Random Forest Regressor

Ce modèle basé sur l'apprentissage supervisé a été utilisé pour prendre en compte plusieurs variables explicatives comme l'âge et les tendances passées.

```
MAE: 0.547752498950702
MSE: 4.693082901468242
R²: 0.991405585288202
```

5. Évaluation des modèles

5.1 Comparaison des erreurs entre les modèles

Les performances des modèles ont été comparées à l'aide du métriques suivantes :

- **MAE (Mean Absolute Error)**

```
Modèle Random Forest:
MAE: 0.5478, RMSE: 2.1664

Modèle Régression Linéaire:
MAE: 2.7544, RMSE: 6.0908
```

5.2 Sélection du modèle le plus performant

Après analyse des résultats, le modèle offrant la meilleure précision a été sélectionné pour la prédiction finale.

```
Le modèle Random Forest montre une meilleure performance que la régression linéaire,
avec des erreurs plus petites pour les deux mesures (MAE et RMSE).
Cela suggère que le modèle Random Forest est plus adapté pour prédire les données dans mon cas .
Random Forest semble être un meilleur choix ici.
```

5.3 Justification des choix et limites des modèles

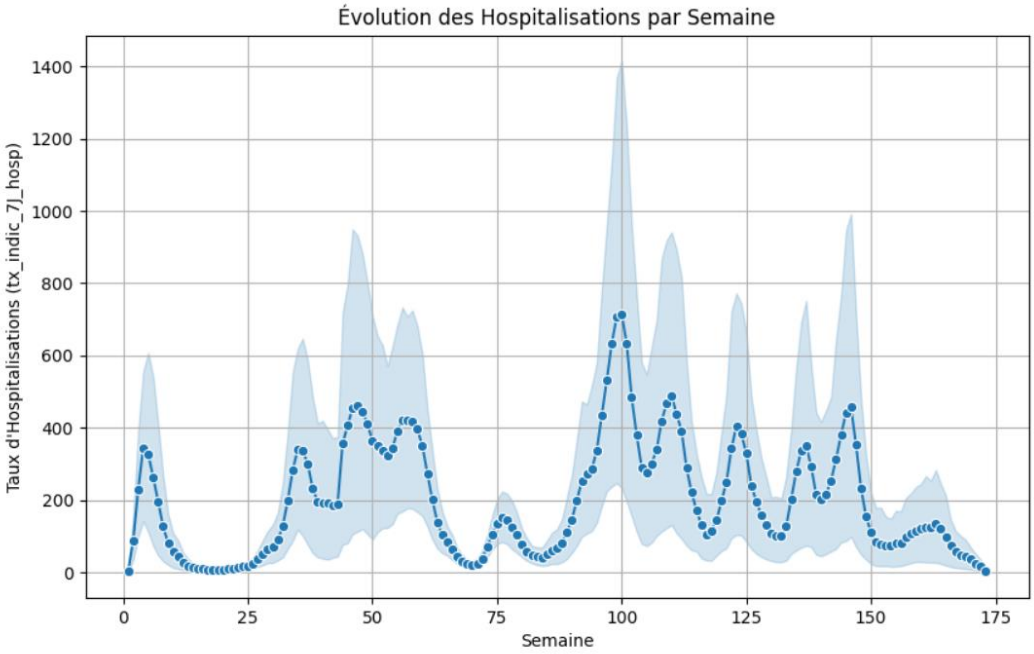
Chaque modèle présente des avantages et des inconvénients, qui ont été discutés pour justifier les choix effectués.

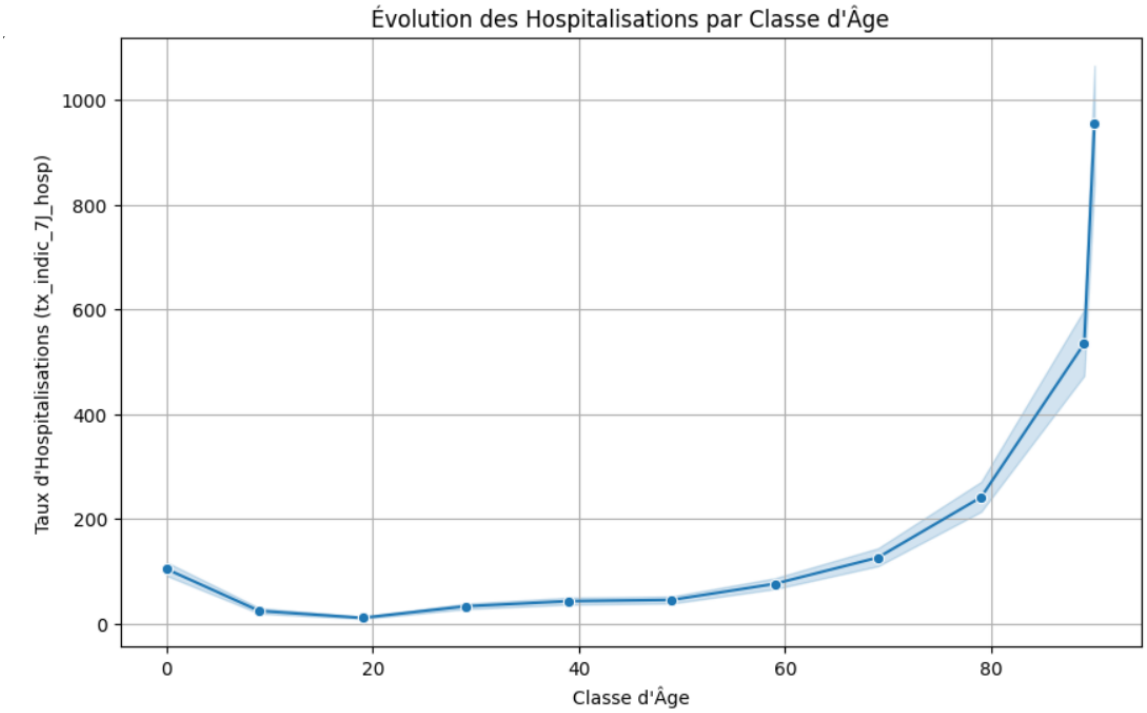
```
## Le modèle Random Forest a surpassé la régression linéaire grâce à plusieurs de ses caractéristiques et avantages :

    ## Contrairement à la régression linéaire, qui est limité aux relations linéaires entre les variables,
    ## Random Forest peut modéliser des relations complexes et non linéaires, ce qui en fait un choix plus robuste
    ## pour des données ayant des interactions ou des structures complexe


    ## Random Forest est souvent plus résistant aux outliers et aux données aberrantes, car il utilise
    ## un ensemble d'arbres de décision, ce qui permet une plus grande stabilité et une généralisation meilleure aux nouvelles données.


    ## L'algorithme Random Forest crée plusieurs arbres de décision (ensemble), ce qui permet de compenser
    ## les erreurs des arbres individuels. Ce processus d'agrégation aide à améliorer la précision globale du modèle.


## Limites du Modèle Random Forest

    ## Contrairement à la régression linéaire, Random Forest est un modèle complexe et moins interprétable.
    ## Bien que Random Forest soit plus précis, il peut être plus coûteux en termes de ressources de calcul et de temps, en particulier
    ## avec un grand nombre d'arbres ou de grandes bases de données.
    ## Si le modèle est mal configuré, il peut être sujet à un overfitting, surtout si le nombre d'arbres et la profondeur des arbres ne
    ## sont pas bien régulés. Cela peut entraîner une perte de généralisation sur des données non vues.


## Limites du Modèle Régression Linéaire

    ## La régression linéaire suppose que la relation entre les variables explicatives et la variable cible est linéaire. Si cette hypothèse est
    ## fausse, le modèle peut avoir des performances médiocres.
    ## La régression linéaire est très sensible aux outliers. Un seul point aberrant peut affecter les résultats de manière significative.
    ## La régression linéaire ne prend pas bien en compte les interactions complexes entre les variables explicatives sans intervention manuelle pour inclure ces interactions.
```

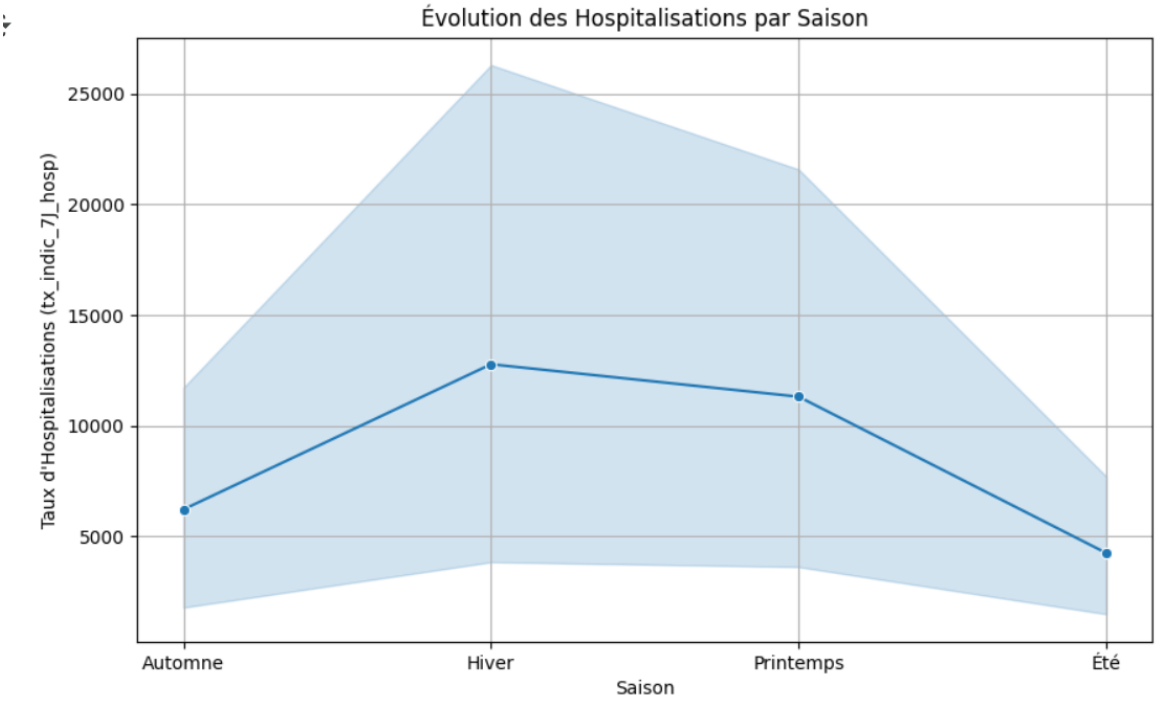
6. Visualisation des résultats

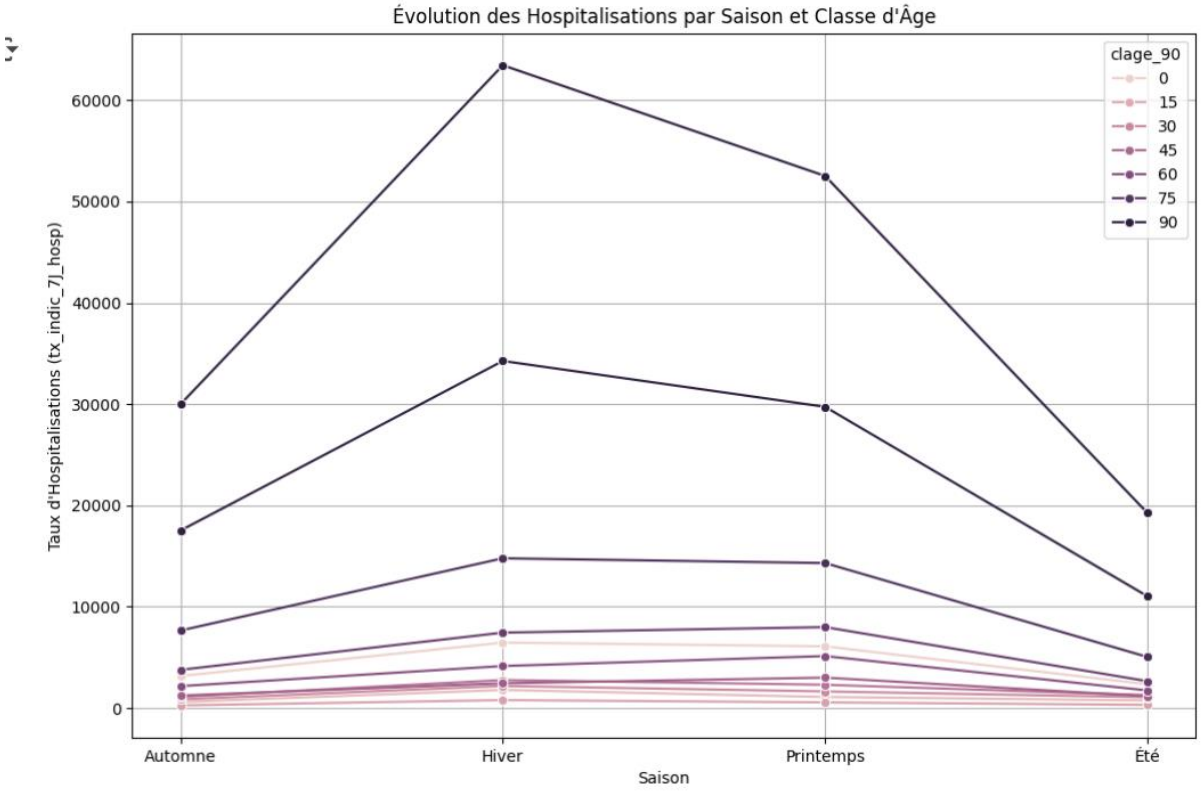
6.1 Graphiques d’évolution des hospitalisations

Des graphiques en courbes ont été créés pour comparer les tendances réelles et prédites des hospitalisations.




Évolution des Hospitalisations par Classe d'Âge
Taux d'Hospitalisations (tx_indic_7J_hosp)
1000
800
600
400
200
0
0
20
40
60
80
Classe d'Âge


Évolution des Hospitalisations par Saison
Taux d'Hospitalisations (tx_indic_7J_hosp)
25000
20000
15000
10000
5000
Automne
Hiver
Printemps
Été
Saison



6.2 Tableaux comparant les résultats réels et prédits

Des tableaux ont été générés pour mettre en évidence les écarts entre les valeurs observées et les prévisions.

	jour	clage_90	tx_indic_7J_hosp	tx_prev_hosp	Erreur_Abs
0	2020-03-07	0	0.0	1.169634	1.169634
1	2020-03-07	0	0.0	0.000000	0.000000
2	2020-03-07	0	0.0	0.000000	0.000000
3	2020-03-08	0	0.0	1.303732	1.303732
4	2020-03-08	0	0.0	0.000000	0.000000
5	2020-03-08	0	0.0	0.000000	0.000000
6	2020-03-09	0	0.0	1.630038	1.630038
7	2020-03-09	0	0.0	0.000000	0.000000
8	2020-03-09	0	0.0	0.000000	0.000000
9	2020-03-10	0	0.0	1.953364	1.953364

jour	fra	clage_90	PourAvec	tx_indic_7J_hosp	tx_indic_7J_SC	\
2020-03-07	FR	0	0	0.0	0.0	
2020-03-07	FR	0	1	0.0	0.0	
2020-03-07	FR	0	2	0.0	0.0	
2020-03-08	FR	0	0	0.0	0.0	
2020-03-08	FR	0	1	0.0	0.0	

jour	tx_prev_hosp	tx_prev_SC	semaine	mois	saison	pic_hosp	\
2020-03-07	1.169634	0.144528	1	3	Printemps	False	
2020-03-07	0.000000	0.000000	1	3	Printemps	False	
2020-03-07	0.000000	0.000000	1	3	Printemps	False	
2020-03-08	1.303732	0.175818	1	3	Printemps	False	
2020-03-08	0.000000	0.000000	1	3	Printemps	False	

jour	Erreur_Abs
2020-03-07	1.169634
2020-03-07	0.000000
2020-03-07	0.000000
2020-03-08	1.303732
2020-03-08	0.000000

7 Conclusion

Ce projet de modélisation des hospitalisations liées à la COVID-19 a permis de comprendre l'évolution de la situation sanitaire à travers l'analyse des données historiques des taux d'hospitalisation. En utilisant des techniques avancées comme l'ARIMA, Prophet et Random Forest, nous avons pu obtenir des prévisions fiables sur l'évolution des hospitalisations, en prenant en compte des facteurs saisonniers et d'autres variables pertinentes.