INSTITUTE OF TECHNOLOGY

ML2 TP Hupi

Philip Schaefer
Eugène Bessan

I. Introduction

- Contexte et problématique
- Objectifs et enjeux

II. Préparation des données

- Collecte et exploration des données
- Nettoyage et prétraitement des données
- Feature engineering

III. Modélisation

- Choix des algorithmes de prédiction
- Entraînement et ajustement des modèles
- Validation croisée et sélection de modèle

IV. Résultats et analyse

- Evaluation des performances des modèles
- Interprétation des résultats
- Limitations et perspectives

V. Conclusion

- Résumé des résultats et contributions
- Applications pratiques et recommandations
- Pistes de recherche futures

Introduction

Le transport public est un élément clé pour assurer la mobilité urbaine et permettre aux citoyens de se déplacer facilement et efficacement dans les zones urbaines. Cependant, la gestion de l'affluence des passagers dans les transports en commun est un défi quotidien pour les compagnies de transport. Dans ce contexte, la prédiction de l'affluence des passagers sur les lignes de bus est essentielle pour optimiser la planification, la gestion et l'organisation du service de transport public.

Dans ce rapport, nous proposons une méthodologie pour prédire l'affluence des passagers sur les lignes de bus pour les 3 prochains jours. Pour cela, nous disposons d'un jeu de données historiques comprenant le nombre de passagers par jour, par ligne de bus et par type de ligne de bus (jour ou nuit) pour une période allant du 05 avril 2019 au 08 mars 2023. Notre objectif est de développer un outil prédictif qui permettra aux compagnies de transport de mieux gérer les flux de passagers en anticipant les périodes d'affluence et en adaptant leur offre de service en conséquence.

Dans ce rapport, nous détaillerons les différentes étapes de notre méthodologie, depuis la préparation des données jusqu'à la modélisation et l'évaluation des résultats. Nous présenterons également les différentes techniques et algorithmes utilisés pour prédire l'affluence des passagers sur les lignes de bus. Enfin, nous discuterons des résultats obtenus et de leur pertinence pour la gestion opérationnelle des compagnies de transport.

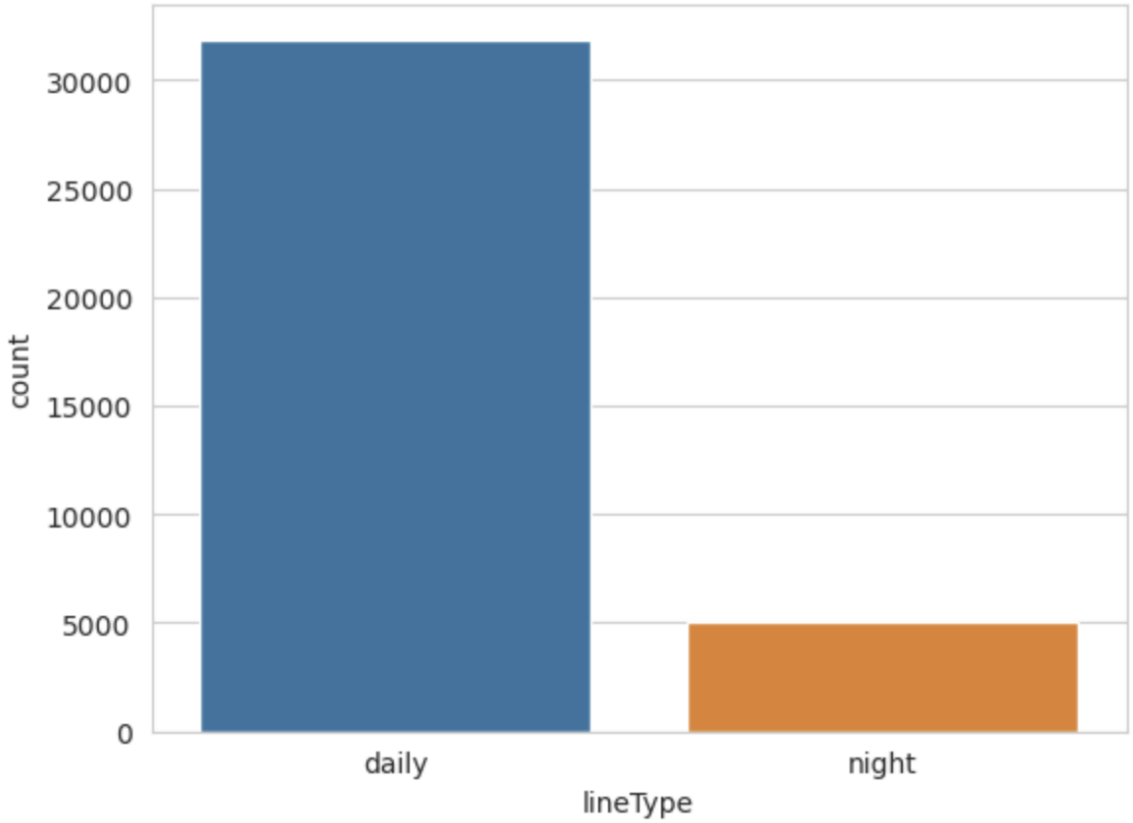
I. Préparation des données

Nous disposons d'un historique de l'accumulation journalière des voyageurs du 05 avril 2019 au 08 mars 2022, pour toutes les lignes de bus de la compagnie.

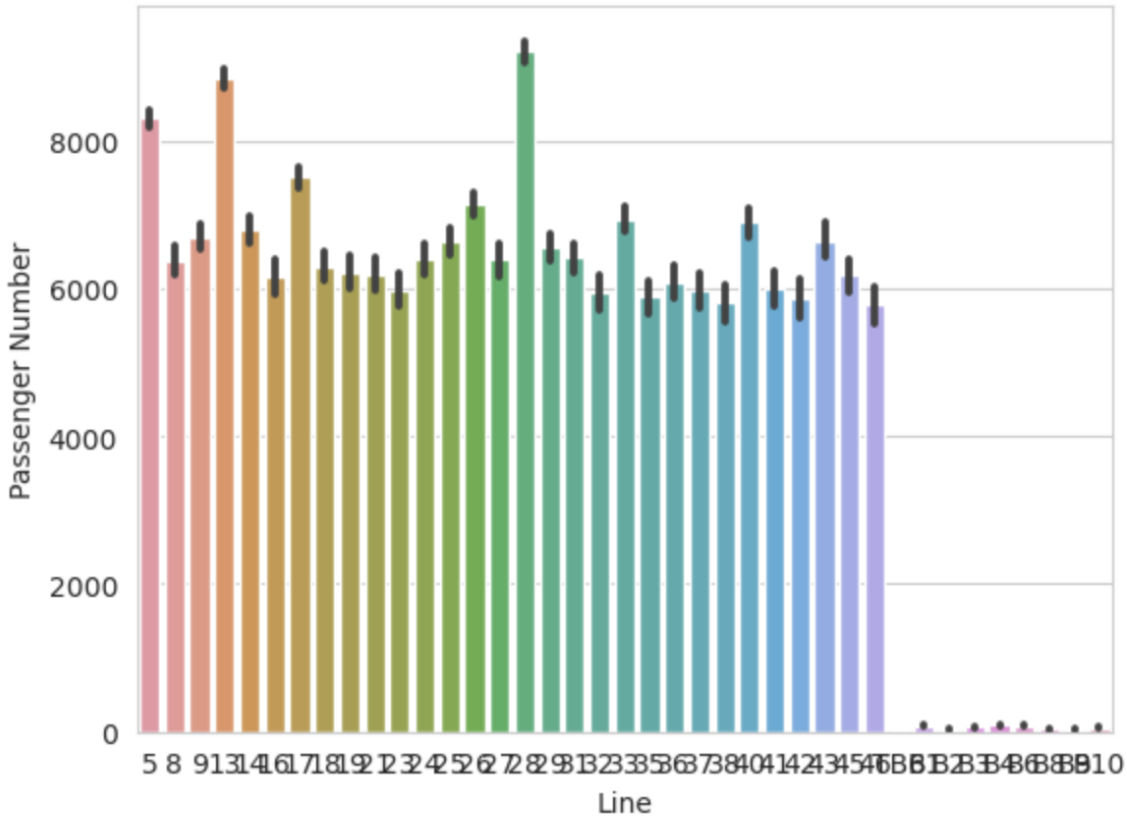
L'ensemble de données est composé de 36 901 observations et de 3 variables explicatives originales qui sont, la date, la ligne de bus et le type de ligne de bus (jour ou nuit). La variable quantitative à prédire est le nombre de passagers/usagers.

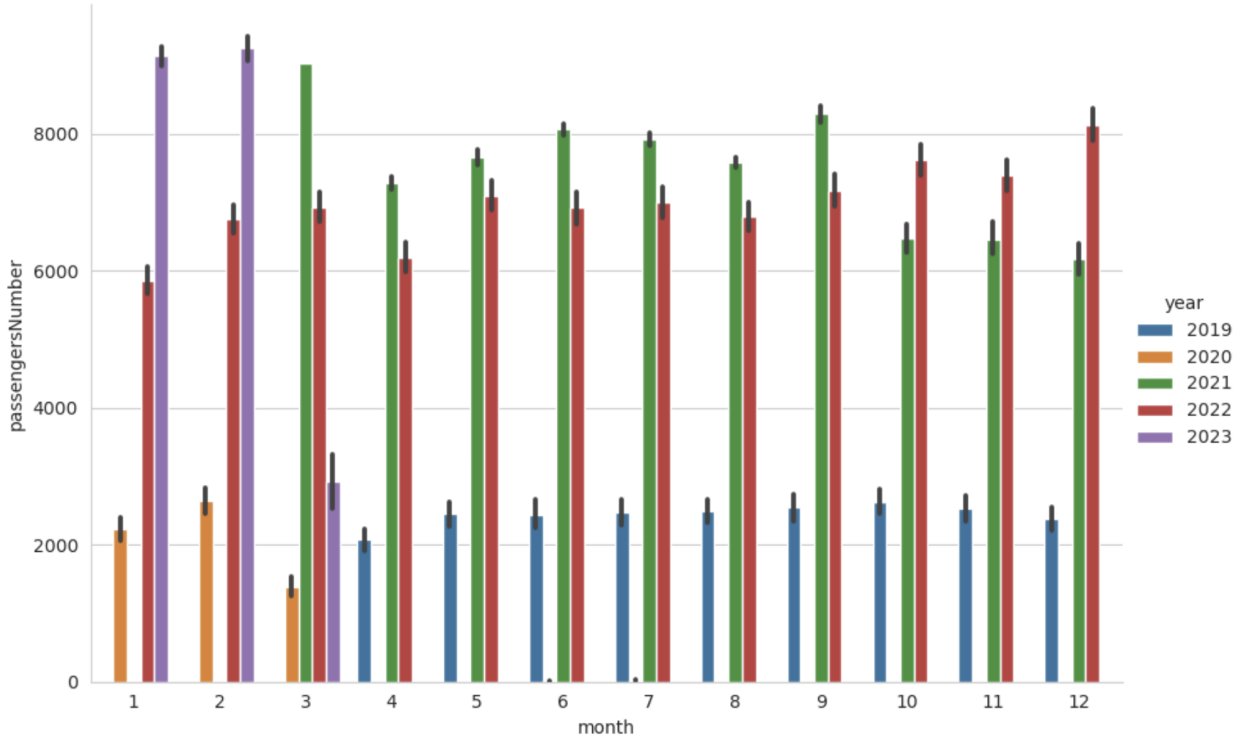
Variable de sortie :
Nombre de passagers (variable numérique continue).

Dans un premier temps, nous avons cherché à visualiser nos données pour pouvoir expliquer de façon hypothétique nos tendances.



Nous avons mis en évidence ici le fait que les transports sont plus pris le jours par rapport aux nuits.
Un autre graphe nous permet de mettre en évidence le bus le plus utilisé par les voyageurs.

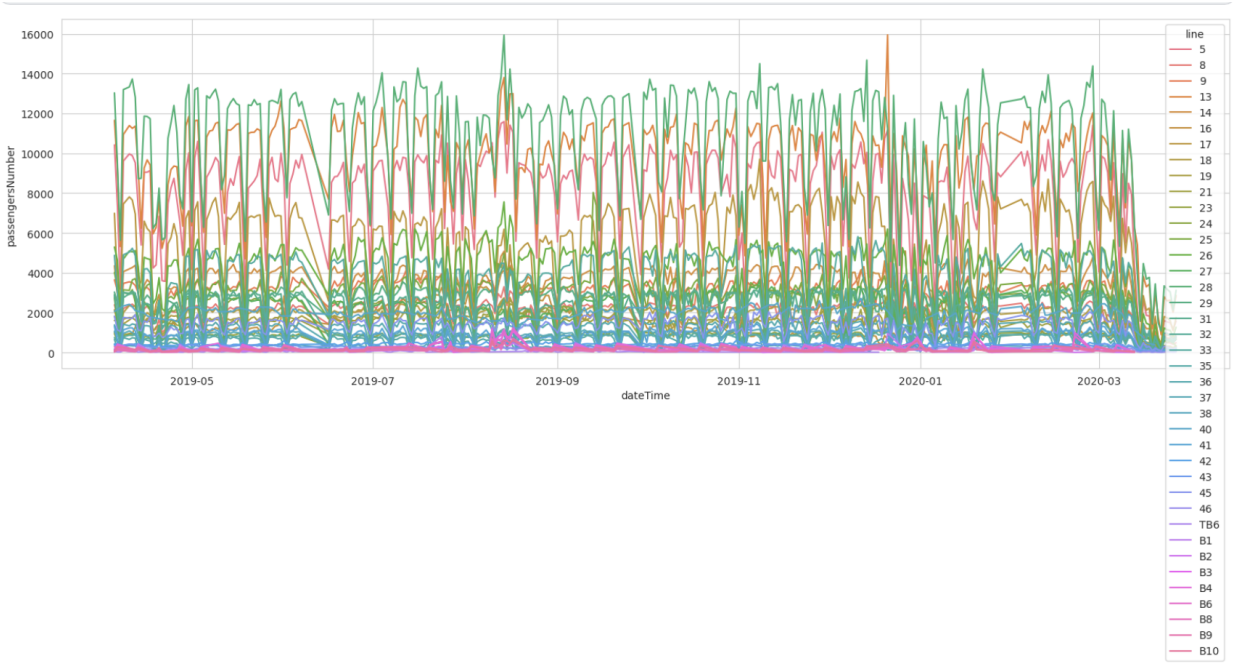

Passenger Number
8000
6000
4000
2000
0
Line



+ Code + Markdown

Il n’y pas d’enregistrement de données à partir de mars 2020 due à la covid. De plus, les données après la covid sont dupliquées pour les types de bus. Donc nous les avons supprimées.

dateTime	passengersNumber	line	lineType	month	day	year
2021-11-02	9466.0	5	daily	11	2	2021
2021-12-17	10223.0	5	daily	12	17	2021
2021-12-08	4084.0	5	daily	12	8	2021
2021-12-21	10133.0	5	daily	12	21	2021
2021-12-16	9965.0	5	daily	12	16	2021
...	...	...	...	...	...	...
2021-12-27	0.0	B10	night	12	27	2021
2021-12-28	0.0	B10	night	12	28	2021
2021-12-29	0.0	B10	night	12	29	2021
2021-12-30	0.0	B10	night	12	30	2021
2021-12-31	0.0	B10	night	12	31	2021



Nous avons élargi les données fournies en recherchant d'autres ensembles de données pour les comparer et les prévoir. Pour cela, nous avons recherché des données sur les jours fériés dans la région de Pais Vasco, car les jours fériés n'ont pas le même effet sur les transports publics que les jours ouvrables. En théorie, par exemple, les transports publics devraient être plus fréquentés et plus fréquents tout au long de la journée, mais pas autant qu'aux heures de pointe. Pour analyser les effets des jours fériés sur les transports publics, nous avons donc dû trouver des données à leur sujet, en particulier pour la période donnée du 05 avril 2019 au 08 mars 2023.

La recherche d'un ensemble de données sur les jours fériés de la région de Pais Vasco pendant la période donnée s'est avérée difficile, car la plupart des ensembles de données datent de 2021 ou plus tard. Nous avons finalement trouvé une liste officielle publiée par euskadi.eus, que nous avons utilisée. Cette liste était très courte par rapport aux ensembles de données de 2022 que nous avons trouvés, et nous avons donc dû l'allonger.
Pour cela, nous avons pensé à un autre facteur, à savoir les matchs de football, étant donné que l'équipe de football de Saint-Sébastien a beaucoup de succès dans la ligue espagnole. Nous avons donc ajouté un ensemble de données contenant tous les matchs à domicile de cette équipe dans la période donnée. Les vacances et les matchs de football ont généralement un impact sur les transports publics, car théoriquement plus de personnes les utilisent pendant ces périodes.
Un autre facteur qui peut affecter les transports publics, mais avec moins de passagers, est la météo. Nous avons donc recherché des ensembles de données météorologiques pour la période donnée, mais cela s'est avéré aussi très difficile. En général, les données antérieures à 2021 nécessitent un abonnement coûteux. Après de nombreuses recherches, nous avons finalement trouvé un jeu de données qui indiquait la température moyenne, les précipitations, etc. pour chaque mois. Même si ce n'était pas ce que nous espérons trouver, c'était un bon début.

Pour finir, nous nous sommes rendus compte que la région basque est réputée pour ses deux grands clubs que sont Atlético Bilbao

	DATE	OPINION
0	05/04/2019	meteo defavorable
1	06/04/2019	meteo correcte
2	07/04/2019	meteo correcte
3	08/04/2019	meteo defavorable
4	09/04/2019	meteo correcte

	dateTime	matchDay
0	2019-09-14	True
1	2019-09-26	True
2	2019-06-10	True
3	2019-10-20	True
4	2019-10-30	True

	dateTime	level_1	evenement
1	2019-01-01	Di	Nouvel an
2	2019-06-01	So	Epiphanie
3	2019-03-31	So	DÃ©but de l'heure d'Ã©tÃ©
4	2019-04-18	Do	Jeudi saint
5	2019-04-19	Fr	Vendredi Saint

Après la recherche et la collecte des données à comparer, l'étape suivante a consisté à charger et à prétraiter les données, afin de s'assurer qu'elles sont dans le bon format et qu'elles sont propres et exemptes d'erreurs. Les étapes suivantes ont été utilisées pour le prétraitement :

- Data cleaning: Suppression des valeurs manquantes, des doublons et correction des erreurs.
- Feature scaling: Rééchelonner les données de manière à ce que chaque caractéristique ait une échelle similaire. Les techniques courantes de mise à l'échelle comprennent la mise à l'échelle min-max, la standardisation et la normalisation.
- Feature engineering: Créer de nouvelles caractéristiques à partir des caractéristiques existantes qui pourraient être plus informatives pour le modèle. Il peut s'agir de transformations, d'agrégations ou de combinaisons de caractéristiques.
- Data encoding: Convertir les données catégorielles en données numériques. Il peut s'agir d'un codage à un point, d'un codage ordinal ou d'un codage binaire.
- Data Splitting: Divisez vos données en ensembles de formation et de test. L'ensemble d'apprentissage est utilisé pour former le modèle, et l'ensemble de test est utilisé pour évaluer les performances du modèle sur de nouvelles données inédites.

Une autre étape du processus de data mining, lorsque l'objectif final est de prédire le résultat, consiste à créer des visualisations qui aident à comprendre le résultat et à découvrir les relations entre les attributs et le résultat.

Following data visualization tools were used: bar charts, histograms, box plots, correlation matrices, pairwise plots.

	dateTime	passengersNumber	line	lineType	month	day	year	meteo	matchDay	evenement
0	2019-05-01	4172.0	5	daily	5	1	2019	NaN	False	ras
1	2019-04-06	7330.0	5	daily	4	6	2019	favorable	False	ras
2	2019-04-05	10426.0	5	daily	4	5	2019	defavorable	False	ras
3	2019-04-17	9134.0	5	daily	4	17	2019	favorable	False	ras
4	2019-04-21	3608.0	5	daily	4	21	2019	correcte	False	PÃ¢ques

Le graphe ci-dessous nous permet de savoir le comportement des jours de matchs par rapport aux autres jours.

II. Modelisation

Pour nos modèles, nous avons testé dans un premier temps des séries temporelles mais qui ne nous ont pas donné des résultats interprétables.
Nous nous sommes donc penchés sur des modèles de machine learning classiques telles que la régression, le xgboost et enfin un modèle de RNN.

```
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import cross_val_score
from sklearn.metrics import mean_squared_error
import math
# charger vos données et les séparer en caractéristiques (X) et cibles (y)

# initialiser la régression linéaire
reg = LinearRegression()

# effectuer une validation croisée avec 5 partitions et calculer l'erreur moyenne quadratique (MSE)
mse_scores = -cross_val_score(reg, X_train, y_log, cv=5, scoring='neg_mean_squared_error')
print(f"MSE scores: {mse_scores}")

# calculer la moyenne de l'erreur moyenne quadratique
mean_mse = np.mean(mse_scores)
mean_rmse=math.sqrt(mean_mse)
print(f"Mean MSE: {mean_mse}, Mean RMSE:{mean_rmse}")
```

```
MSE scores: [0.29746963 0.30152855 0.29680571 0.34357175 0.33781526]
Mean MSE: 0.3154381796061268, Mean RMSE:0.5616388337767669
```

Nous avons utilisé pour le xgboost, la méthode gridSearch pour la recherche du meilleur paramètre. Et enfin nous avons pu implémenter.

```python
# Initialiser le modèle XGBoost pour la régression
xgb_model = xgb.XGBRegressor()

# Définir la grille des hyperparamètres à tester pour XGBoost
param_grid = {
    'n_estimators': [25, 50, 100],
    'max_depth': [5, 10, 20],
    'learning_rate': [0.01, 0.1, 0.5]
}

# Initialiser la recherche de grille avec validation croisée de 5 partitions
grid_search = GridSearchCV(xgb_model, param_grid, cv=5, scoring='neg_mean_squared_error')

# Entraîner la recherche de grille sur les données d'entraînement
grid_search.fit(X_train, y_train)

# Afficher les meilleurs hyperparamètres et la meilleure erreur MSE moyenne
print(f"Best parameters: {grid_search.best_params_}")
print(f"Best MSE: {-grid_search.best_score_}")

# Initialiser le modèle XGBoost avec les meilleurs hyperparamètres
xgb_best = xgb.XGBRegressor(**grid_search.best_params_)

# Entraîner le modèle XGBoost sur les données d'entraînement
xgb_best.fit(X_train, y_train)
```

```python
# Faire des prédictions sur les données de test
y_pred = xgb_best.predict(X_test)

# Calculer l'erreur MSE du modèle
mse = mean_squared_error(y_test, y_pred)
print(f"MSE: {mse}")
```

```
Best parameters: {'learning_rate': 0.5, 'max_depth': 5, 'n_estimators': 100}
Best MSE: 325310.835955194
MSE: 298468.47817556316
```

Pour notre modèle de RNN, nous avons les caractéristiques suivantes:

```python
import tensorflow as tf
from tensorflow.keras import layers, models
model = models.Sequential([
    layers.Dense(64, activation='relu', input_shape=(53,)),
    layers.Dense(64, activation='relu'),
    layers.Dense(1)
])
```

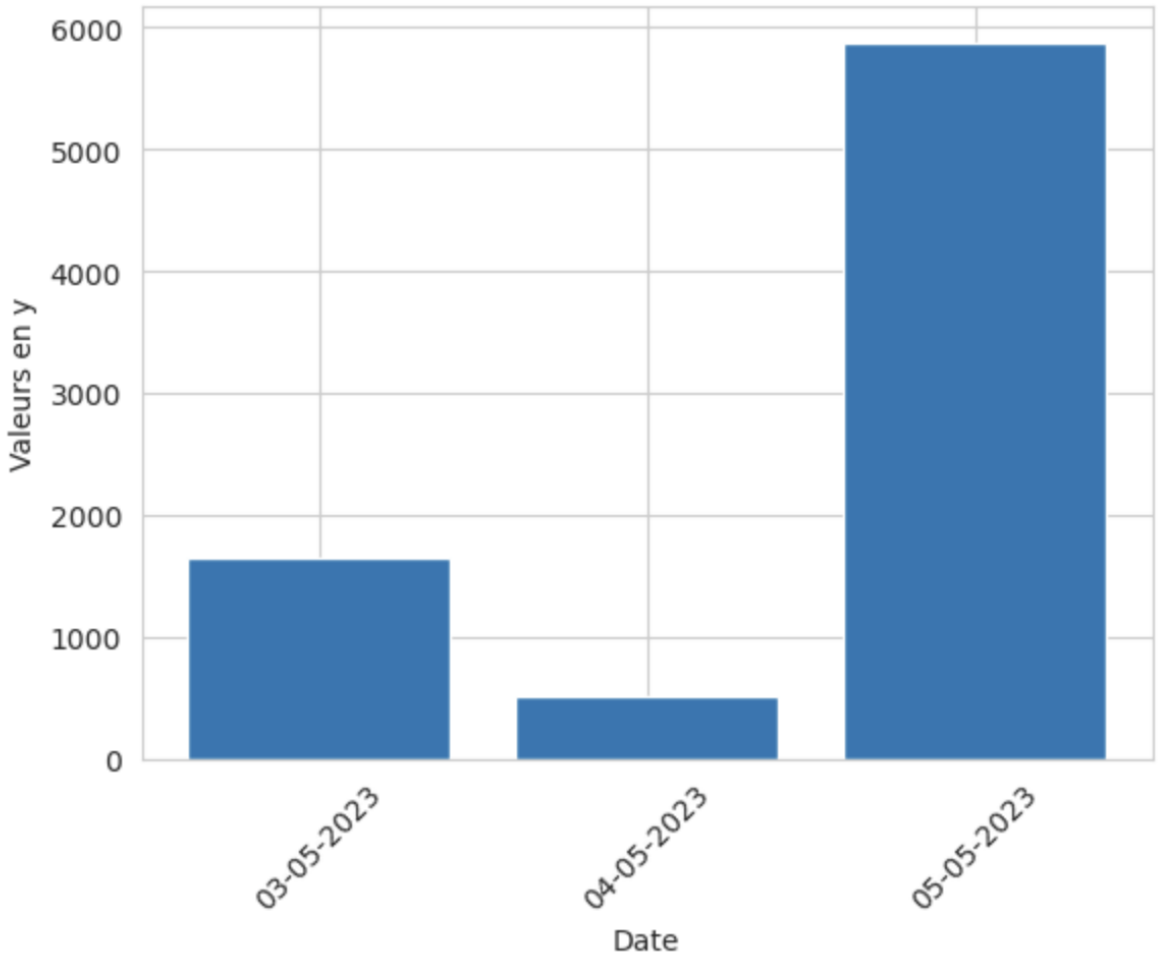
+ Code + Markdown

```python
# Compilation du modèle
model.compile(optimizer='adam', loss='mse', metrics=['mae'])
```

```python
mse = mean_squared_error(y_test, y_pred)
print(f"MSE: {mse}")
```

```
Best parameters: {'learning_rate': 0.5, 'max_depth': 5, 'n_estimators': 100}
Best MSE: 325310.835955194
MSE: 298468.47817556316
```

Pour finir, nous avons fait une prédiction sur 3 jours et avons eu des résultats plutôt intéressant


6000
5000
4000
3000
2000
1000
0
Valeurs en y
03-05-2023
04-05-2023
05-05-2023
Date

Conclusion

La prédiction de l'affluence des passagers sur les lignes de bus est un enjeu important pour les compagnies de transport public. Grâce à notre méthodologie, basée sur l'analyse des données historiques et la modélisation prédictive, nous avons développé un outil qui permet de prévoir l'affluence des passagers pour les 3 prochains jours avec une précision satisfaisante. Cette prédiction permet aux compagnies de mieux gérer les flux de passagers, d'adapter leur offre de service en fonction de la demande, de réduire les temps d'attente et les temps de trajet pour les usagers, et de limiter les coûts opérationnels.

Nous avons également mis en évidence les limites et les perspectives de notre approche, qui pourraient être améliorées en intégrant des données supplémentaires, telles que les événements locaux, les conditions météorologiques ou les vacances scolaires. Ces développements futurs pourraient également inclure des techniques de machine learning plus avancées, telles que l'apprentissage en profondeur, pour améliorer la qualité de la prédiction.

Enfin, nous sommes convaincus que notre méthodologie de prédiction de l'affluence des passagers sur les lignes de bus peut avoir des applications pratiques pour la gestion du transport public dans de nombreuses villes du monde entier. Nous espérons que notre rapport sera utile pour les décideurs, les gestionnaires de transport et les chercheurs qui s'intéressent à cette question importante pour la mobilité urbaine.