Bike traffic counters installed by Paris city

Thomas Bordes, Alexandre Brun

17 décembre 2023

1 Summary of External Data for Bike Counter Predictions in Paris

First we wanted to merge external data to our original dataset to enhance the accuracy. We thought of several events such as holidays information, COVID-19 data, strikes and road accident for cyclists.

Here is a list of all the features for each dataset we believed could impact our predictions :

1.1 Meteorological Data

We used the following features from the external dataset given to us for the challenge :

— Wind speed (**ff**) impacts cycling by influencing the effort required to ride, especially in open areas or along riverbanks.
— Temperature (**t**) plays a crucial role as extreme temperatures can reduce the appeal of outdoor activities like cycling.
— Humidity (**u**) can affect comfort levels ; high humidity may discourage cycling.
— Visibility (**vv**) is critical for safety ; poor visibility conditions may lead to a decrease in cycling activity.
— Cloud cover (**n**) and sunlight presence may sway individuals' decisions to cycle, particularly for leisure.
— Precipitation (**rr3**) often deters cyclists due to the increased risk of accidents and discomfort from getting wet.
— Snowfall (**ssfrai**) usually leads to a significant decline in cycling due to safety hazards and the physical difficulty of riding in snow.

1.2 Holiday Data

We used the library `holidays` to implementate **public holidays**.

We manually generate **holidays** using data from holiday website Paris to the specific period we train and test our model.

Finally, we created the feature **week-end** manually.

— Holidays typically result in less commuting and can lead to either a decrease in cycling due to closed workplaces or an increase due to recreational use.
— Public holidays may attract tourists who use bikes as a convenient mode of sightseeing, potentially increasing bike counter numbers.

1.3 COVID-19 Data

Curfews primarily alter daily routines and reduce commuting during specific hours, possibly leading to shifts in peak bike traffic times. In contrast, lockdowns have a more profound impact by significantly reducing overall commuting due to the closure of public places and altering transportation preferences, with some people opting for bikes as a safer alternative to public transport. We manually generated the **curfews** and **lockdowns** to the specific period we train and test our model.

We thought as this period was uncommon, we could add information and thus we we added the external dataset DATA gouv COVID about hospital data.

— Hospitalization and intensive care statistics (**hosp** and **rea**) indicate the severity of the pandemic, which can alter public behavior and transportation preferences.
— Recovery numbers (**rad**) might signal a perceived reduction in risk, potentially encouraging more outdoor activities like cycling.
— Lockdowns and curfews significantly change movement patterns, possibly leading to more cycling as an alternative to public transportation or during permitted hours.

1.4 Road Accident Data

By examining the frequency, locations, timings, and characteristics of road incidents, we can infer cycling traffic trends. This data, therefore, is not only crucial for understanding the current state of cyclist safety in Paris but also serves as a proxy for bicycle traffic volume and patterns. The external dataset we added comes from DATA Gouv ROAD Accident.

- **Count_accidents** : The frequency and severity of accidents involving cyclists can reflect cycling traffic volumes and help identify high-risk areas and times for targeted safety improvements.
- **Max_Grav_accidents** : Understanding accident patterns may inform infrastructure planning, such as the need for protected bike lanes, which can influence cycling rates.

2 Preprocessing

Most of the machine learning models used for regression are dependant on numerical values to train and predict. This means we need to preprocess our data that can comes in different shape (categorical, strings...).

We apply a function to the **date** column to split it into several individual components such as year, month, day, weekday, and hour. This transformation is common in time series analysis to capture various temporal trends and patterns that could be significant for predictive models.

Categorical variables **counter_name** and **site_id** undergo `one-hot encoding`. This process converts these categorical variables into a series of binary columns, representing the presence or absence of each category.

3 Model selection

Initially, we started with linear regression models such as Ridge and Lasso for predicting bike counters in Paris. These models assume a linear relationship between the independent variables and the dependent variable. Ridge and Lasso are favored for their regularization capabilities, which reduce overfitting by penalizing large coefficients.

However, upon assessing the performance of these linear models, it became clear that they were not effectively capturing the complexity and patterns within our data.

The time series data showed non-linear behaviors, characterized by periodic peaks and troughs as shown in Figure 1, which linear models inherently cannot account for due to their linear assumption.

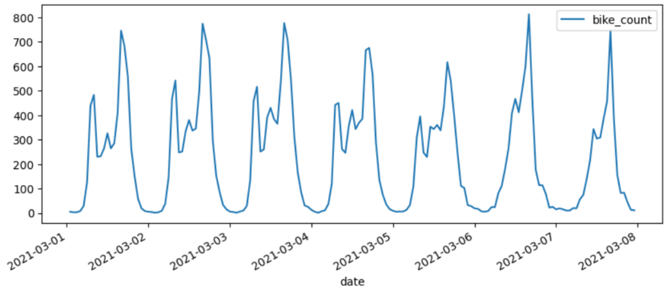


FIGURE 1 – Traffic for the site 'Totem 73 boulevard de Sébastopol' during the first week of March 2021

Realizing the shortcomings of linear models in this context, we transitioned to non-linear models, which are better equipped to manage the subtleties of such data :

- **Non-Linear Relationships** : Non-linear models like CatBoost, LightGBM, and XGBoost have the ability to capture non-linear relationships between features and the target. These models do not presuppose a linear correlation and are therefore adept at modeling the cyclical patterns observed in the bike counter data.
- **Feature Interactions** : These models inherently learn complex feature interactions, eliminating the need for manual feature engineering. This is especially valuable in datasets with temporal elements, where interactions between time-related components and other variables can be crucial.
- **Robustness** : Non-linear models are more robust to outliers and diverse data distributions, which suits real-world data that often includes noise and irregularities.

By adopting non-linear models, we observed a marked improvement in prediction accuracy. Our choice to use non-linear models was validated by their superior performance.

We conducted a simple test for the selected features on multiple models with default parameters on Table 1 :

Model	RMSE_Train	RMSE_Test	Train(s)
Lasso	1.56	1.33	10
Ridge	1.38	1.09	3
LightGBM	0.475	0.526	15
XGBoost	0.419	0.481	20
CatBoost	0.39	0.474	150

TABLE 1 – Default models performances on selected features

The model we chose to train after were **XGBoost**, **LightGBM** and **CatBoost** based on the results obtained.

After a few tries, for computation cost, we decided to continue with **XGBoost** especially when using our `feature selection process` and for `hyperparameters tuning`.

4 Feature selection

We designed a `feature selection process` designed to identify the best combination of features for predicting bike counters in Paris.

Independently of the data visualization done for each features relevant of our whole data, we separated all our features in two categories :

- **fixed_features** : Features that we believe are essential for training our model.

  ['date', "counter_name", "site_id", 't', 'u', 'rr3'].

- **chosen_fixed_features** : Features that we believe could have a role to play for the prediction of bike traffic.

  ['is_holiday', 'is_weekend', 'is_lockdown', 'is_curfew', 'Max_Grav_accidents', 'hosp', 'rea', 'incid_rea', 'rad', 'Count_accidents'].

Our process aims to test different combinations of features to determine which set provides the best model performance, measured by the **RMSE** on the test dataset.

The following are example of combinations of features used in the model :

- Example 1 : ['date', "counter_name", "site_id", 't', 'u', 'rr3', 'is_holiday', 'is_weekend', 'is_lockdown'].
- Example 1 : ['date', "counter_name", "site_id", 't', 'u', 'rr3', ”Max_Grav_accidents'].
- . . .

A total of more than 1000 fit on **XGBoost** with default parameters has been done.

The `feature selection process` returned us the following features : **['is_holiday', 'is_lockdown', 'hosp']** to add to our fixed_features.

Here is the correlation matrix for external data added to our original dataset in Figure 2 :

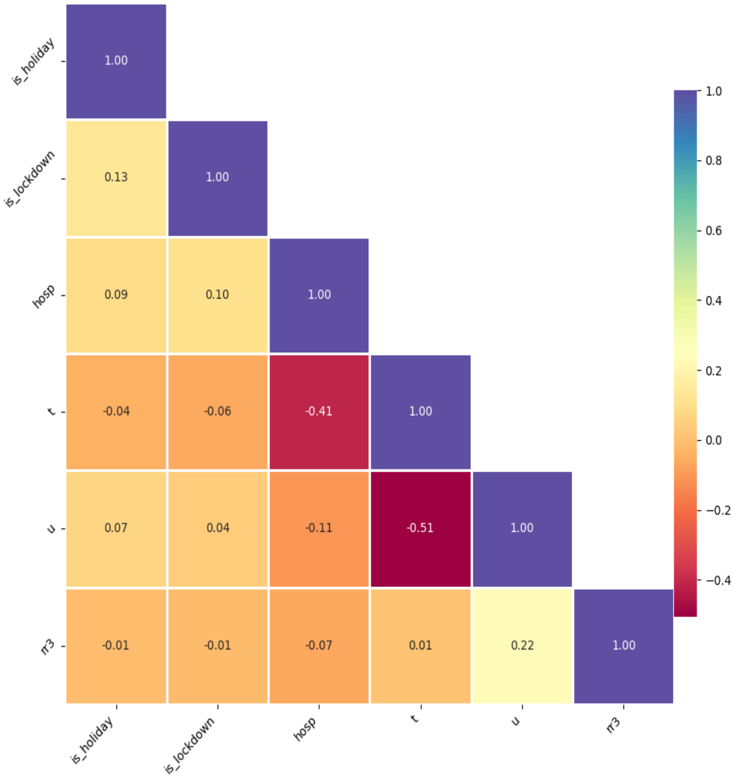


FIGURE 2 – Correlation matrix for external features added

Having redundant features can still make the model more complex than necessary. We tried to limit the amount of features added especially those correlated. Here only `t`, `u` and `hosp` have high correlation.

5 Hyperparameters tuning

Hyperparameter tuning is a crucial step in the machine learning pipeline, where the goal is to find the optimal configuration of hyperparameters that results in the best model performance.

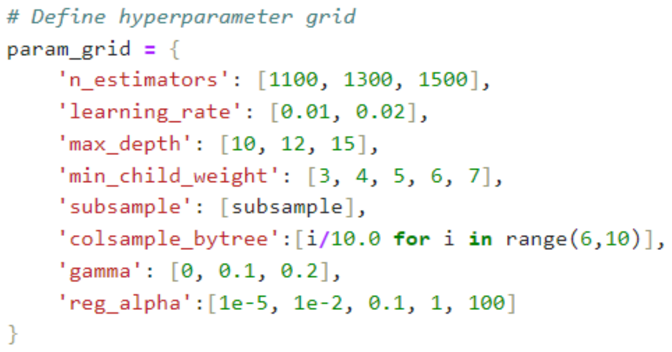
We used a `GridSearchCV` which is a systematic approach to tuning hyperparameters in which every combination of specified hyperparameter values is evaluated. We have the following parameters.

A **Hyperparameter Grid** was defined for :

- `n_estimators` : The number of trees in the ensemble.
- `learning_rate` : The step size shrinkage used to prevent overfitting.
- `max_depth` : The maximum depth of the trees.
- `min_child_weight` : The minimum sum of instance weight needed in a child node.
- `subsample` : The ratio of the training instance.
- `colsample_bytree` : The subsample ratio of columns when constructing each tree.
- `gamma` : The minimum loss reduction required to make a split.
- `reg_alpha` : The L1 regularization term on weights.

Before, launching the gridsearch, we trained our model for some random values of `n_estimators` and `learning_rate` to have some intuition.

Next we defined the grid for those parameters :



```
# Define hyperparameter grid
param_grid = {
    'n_estimators': [1100, 1300, 1500],
    'learning_rate': [0.01, 0.02],
    'max_depth': [10, 12, 15],
    'min_child_weight': [3, 4, 5, 6, 7],
    'subsample': [subsample],
    'colsample_bytree':[i/10.0 for i in range(6,10)],
    'gamma': [0, 0.1, 0.2],
    'reg_alpha':[1e-5, 1e-2, 0.1, 1, 100]
}
```

FIGURE 3 – Randomized GridSearch on XGBoost

We specified `RandomizedSearchCV` to navigate through the hyperparameter space, which is more efficient than `GridSearchCV` for large search spaces.

This gave us our final model we used to predict the `final test dataset` with the following results 2 :

Metric	Value
Leaderboard	8 / 43
RMSE	0.5899

TABLE 2 – Model Performance on the Leaderboard

6 Improvements

Analysis of bike count predictions across 32 sites shows varying levels of accuracy. For some sites, predictions align well with actual counts, while for others, especially those with pronounced peaks, there are significant discrepancies as shown in Figure 4 :

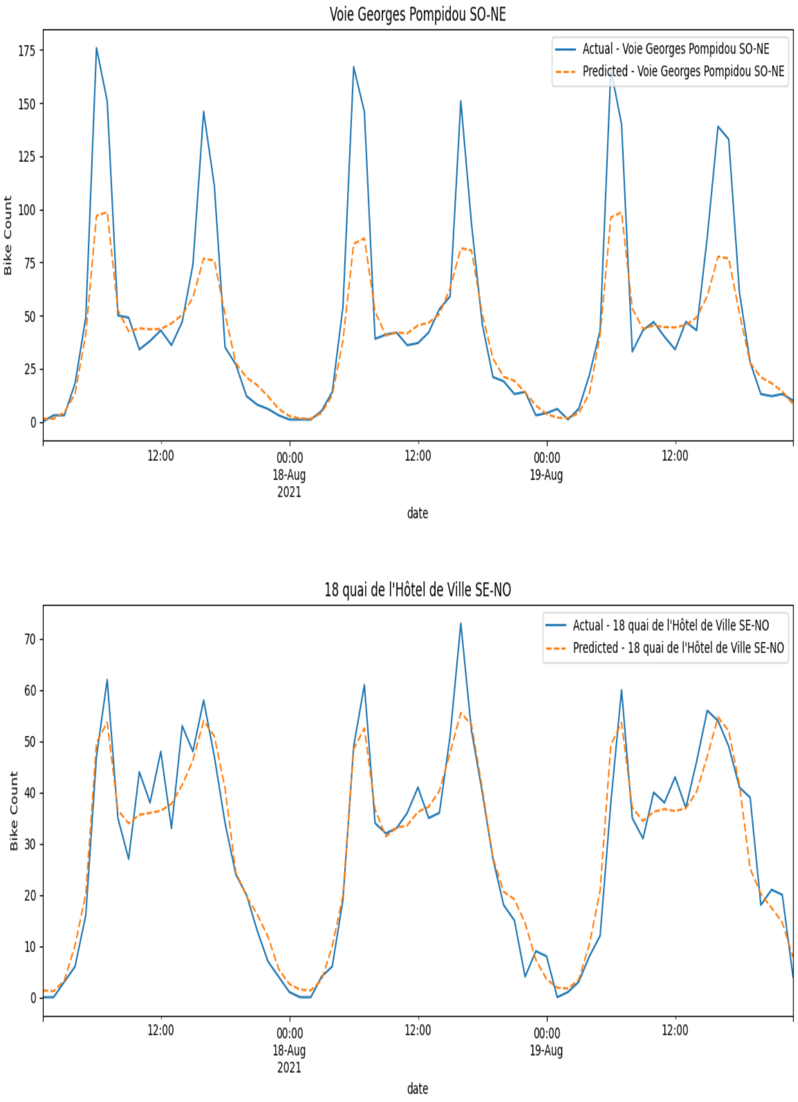


FIGURE 4 – bike_count vs predicted bike_count for two different locations on test set for the third week of August 2021

Each site has unique factors influencing bike count patterns, like local traffic and nearby attractions. The current model may not capture these nuances, leading to less accurate peak predictions.

Proposed Solution : Site-Specific Models

- Tailored to Local Dynamics : Models specific to each site would better capture local patterns and factors.
- Incorporating Local Data : Using data relevant to each location, like local events, can improve

predictions.

- Improved Accuracy : Focusing on distinct site characteristics can enhance model accuracy, especially for peaks and troughs.
- Flexibility in Model Complexity : Different sites may require models with varying complexity.

Implementation Considerations

- Data Availability : Ensure enough data for each site for effective model training.
- Feature Selection : Carefully select features relevant to each site.