REPORT

Kaggle Challenge : Predicting Cyclist Traffic in Paris

Master Data Science for Business X-HEC, 2023
Schroeder Nicolas, Benslimane Mohamed (https://github.com/schronic/predicting_bike_traffic)

SUMMARY

BRIEF CASE PRESENTATION

The City of Paris's open data initiative provides two datasets related to bike usage: one dataset contains data from bike counters across the city, while the other records individual bike usage instances. The task is to use this data, along with any external datasets that cover a period from September 1, 2020, to September 9, 2020 and to forecast the log-transformed bike count (log(bike_count)) for the upcoming months.

EXPLORATORY DATA ANALYSIS

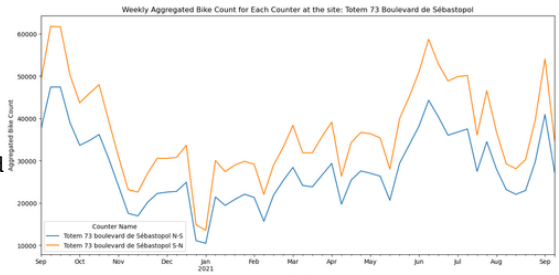
Discovery and visualization our first dataset : train.parquet

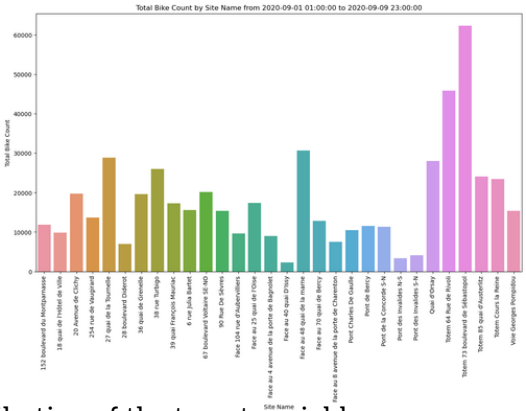
The first rows of the dataset are displayed, offering a glimpse into its structure. This step is fundamental as it reveals the dataset's format, variable names, and types of values, laying the groundwork for more in-depth analysis.

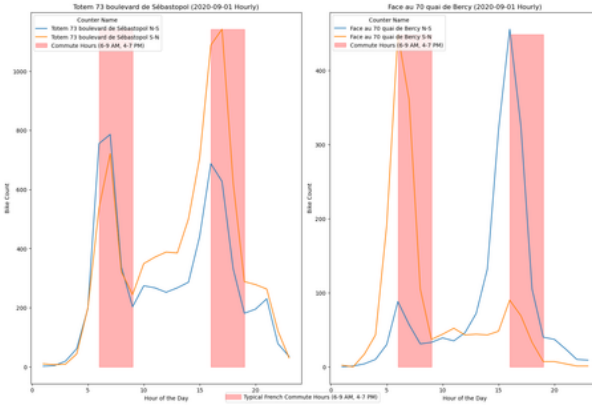
	counter_id	counter_name	site_id	site_name	bike_count	date	counter_installation_date	coordinates	counter_technical_id	latitude	longitude	log_bike_count
48321	100007049-102007049	28 boulevard Diderot E-O	100007049	28 boulevard Diderot	0.0	2020-09-01 02:00:00	2013-01-18	48.846028,2.375429	Y2H15027244	48.846028	2.375429	0.000000

When plotting the counters site, we see that we have 30 sites where sometimes multiple counters are installed per site. Our data is taken from 2020-09-01 01:00:00 to 2020-09-09 23:00:00. Mapping counter locations in Paris and plotting site visit frequencies offers valuable spatial and frequency insights. This visualization helps in understanding how counters are distributed geographically and which sites are more popular.

In the next phase of our analysis, we shifted focus to identifying the most frequented sites, as this was key to understanding the factors influencing the total 'bike_count'. This exploration led to the revelation of the predominant role of certain sites in attracting high bike traffic. Intrigued by these findings, our attention was particularly drawn to "Totem 73 boulevard de Sébastopol," identified as the most frequented site. We delved deeper to examine the distribution of 'bike_count' at this location. Our investigation sought to understand the nature of this distribution - was it linear, or did it exhibit peaks at certain times? We were also curious about the number of counters present at this site. Analyzing the 'bike_count' trends revealed interesting patterns, including peak periods during summer and a distribution that shouwed patterns . This deeper analysis provided valuable insights into the dynamics of bike traffic at this popular site.

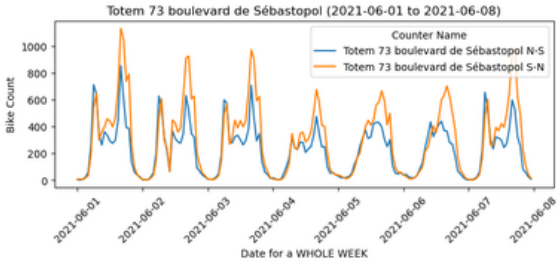


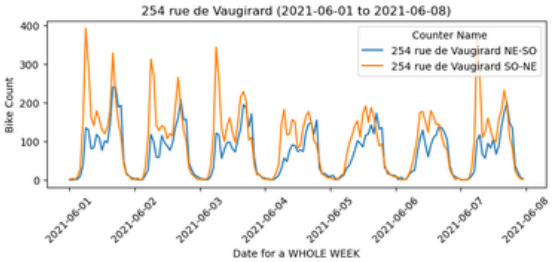




Skewed distribution of the target variable

Upon plotting the 'bike_count' variable on a daily basis, we made an interesting observation: the variable exhibited a skewed behavior. This skewness posed a challenge because it indicated a non-normal distribution of the data. In statistical modeling and analysis, normal distribution of variables is often a key assumption, especially in linear models. A skewed distribution, with its asymmetry and potential for extreme values, can lead to biases in model predictions and affect the accuracy of the analysis. Recognizing this issue is crucial as it guides the subsequent steps in data processing and model





These two graphs indicate that the bike_count data is highly skewed to the right, with a large number of counts close to zero and few counts reaching higher values. The KDE line, which is the smooth curve on the histogram, confirms this skewness by showing the peak near the lower end of the bike_count range and a long tail stretching towards the higher counts.

In the following, we use `log_bike_count` as the target variable as otherwise we will be restricted with `bike_count` that would be sometimes inappropriate to model due to it's distribution.

The stationarity of our new target is a good thing, why?

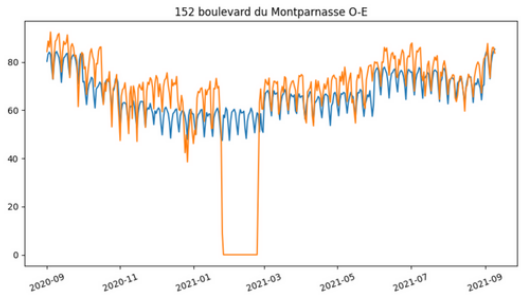
```
Results of Dickey-Fuller Test:
Test Statistic                -1.183813e+01
p-value                        7.703468e-22
#Lags Used                     2.000000e+00
Number of Observations Used    4.970000e+02
Critical Value (1%)           -3.443576e+00
Critical Value (5%)           -2.867373e+00
Critical Value (10%)          -2.569877e+00
dtype: float64
```

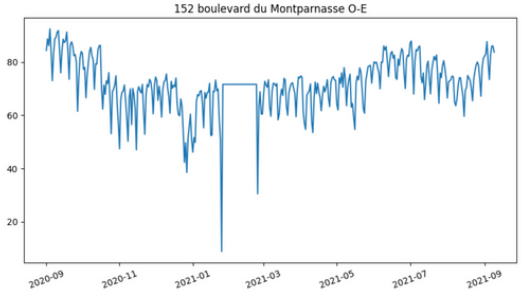
In the subsequent phase of our analysis, we turned our attention to the stationarity of our target variable, 'log(bike_count)'. To ascertain this, we employed the Dickey Fuller test, which confirmed the stationary nature of our variable. The results of the Dickey-Fuller test were quite telling, with a p-value significantly less than 5%. This allowed us to confidently reject the null hypothesis, which suggests that the 'log_bike_count' time series is non-stationary and contains a unit root. As a result, we could assert with certainty that our target variable is indeed stationary.
Why is this finding important and beneficial? (1)Consistency in Mean and Variance: The stationarity of a time series implies consistent mean, variance, and autocorrelation over time. This consistency is advantageous for developing models that can accurately predict future values, as these predictions are based on the premise that future statistical properties will reflect those of the current series. (2)Long-term Forecasting: Stationary data typically does not exhibit long-term trends, enhancing the reliability of long-term forecasting. Predictions made under stationary conditions are less likely to be influenced by evolving trends over time. In summary, it validates our approach and allows us to confidently proceed with time series analysis, modeling, and forecasting

Handling null values

Now, let's address the topic of missing values, or more accurately, hidden missing values, which are essentially zeros in the 'log(bike_count)' variable. In our dataset, it's important to recognize that while there are no explicit missing values, there's a significant portion (about 8%) of values for 'log(bike_count)' that are zero for certain counters, likely indicative of a malfunction or bug in the counter system. Initially, we aimed to visualize and demonstrate the extent of this issue in our code. We chose to illustrate this by examining two specific sites above: "152 Boulevard du Montparnasse" and "20 Avenue de Clichy". The results from this investigation revealed a real loss of information: we observed null values for 20 days at the Montparnasse site and for 3 months at Avenue Clichy, which is abnormal and suggests potential issues such as faulty counters or periods of construction.

To address these null values, we considered four different approaches. Option 1: Replace them by the mean for each counter. Option 2: Predict them using a new model this approach involves building a model to estimate the missing values. Option 3: Not doing anything - an option that entails leaving the dataset as is, without any intervention for the null values. Option 4: Deleting them.
In the modeling section of our analysis, we will examine the outcomes of each of these approaches to determine their effectiveness and impact on our results. This comprehensive approach allows us to evaluate the best method for dealing with these hidden missing values, ensuring the integrity and reliability of our analysis.





FEATURE ENGINEERING AND DATA PREPROCESSING

Feature Selection and Data Cleaning

- **(1) Feature Selection and Data Cleaning on the given train.parquet:** Initially, we scrutinized the correlation matrix, and even considered an embryonic stage of Principal Component Analysis (PCA), though our mastery of PCA isn't yet perfect, we intended to include it in our code. The heatmap generated from the correlation matrix for numerical variables revealed a crucial insight: there were no particularly strong correlations among the different variables, with the notable exception of 'bike_count' and its logarithmic transformation 'log_bike_count'. This strong correlation was anticipated, as 'log_bike_count' is directly derived from 'bike_count'. The absence of strong correlations among the other variables indicated that multicollinearity, which can adversely affect certain models, was not a significant issue for this dataset. Consequently, there was no immediate need to remove any explanatory variables based on correlation analysis alone, with the exception of the 'bike_count' variable. When it comes to cleaning the outliers, as you will see in the modelling part, we decided to keep these outliers after the Interquartile Range (IQR) method.
- **(2) Feature Selection and Data Cleaning on the New Paris weather dataset:** Initially, we faced a **challenge with the granularity** of the date column in our first weather dataset, external_data.csv. The data in this dataset was aggregated in 3-hour intervals, creating a level of imprecision that was unsuitable for our analysis. This was especially problematic when trying to align it with other datasets recorded on an hourly basis. The disparity in data resolution could potentially result in misaligned insights and, as a consequence, less accurate predictions. To address this issue, we sought a new weather dataset that offered a finer level of detail, matching the temporal resolution of our existing datasets. Our search led us to the "paris 2020-09-01 to 2021-10-31.csv" dataset, which provides hourly data points. This enhanced granularity is key for a more accurate and detailed analysis, allowing us to align our weather data more precisely with other hourly datasets. That's why we chose **not to use the given external_data.csv.**

	datetime	temp	humidity	precip	precipprob	windspeed	cloudcover
0	2020-09-01 00:00:00	0.512563	0.631085	0.0	0.0	0.150000	0.333
1	2020-09-01 01:00:00	0.497487	0.677387	0.0	0.0	0.152174	0.100

Now, let's discuss the feature selection process we employed on this new dataset. Firstly in the new dataset, we encountered variables with excessive missing values, which we opted to remove outright. Secondly, we decided to retain variables with outliers, even those with as high as 14% outliers, like the 'precipprob' variable. Lastly, in terms of feature selection, we addressed the issue of highly correlated variables. The correlation matrix heatmap provided a clear visual representation of the relationships between various meteorological variables. We observed that certain variables, such as 'solarradiation' and 'uvindex', exhibited a high degree of correlation. In machine learning models, highly correlated variables can lead to redundancy and increase the risk of overfitting. Overfitting occurs when a model captures not only the underlying patterns but also the noise in the training data, which hinders its performance on new, unseen data. To mitigate the risk of overfitting and streamline our model, we have chosen to reduce the dimensionality in our refined dataset, `scaled_weather_data.csv`.

Moreover, we applied a numeric conversion to the entire DataFrame to ensure all possible numeric types were appropriately typed. We removed any duplicate rows based on the 'datetime' column to maintain data integrity.
For the feature Selection on the New Velib Dataset we invite you to see our repository (~ same processes)

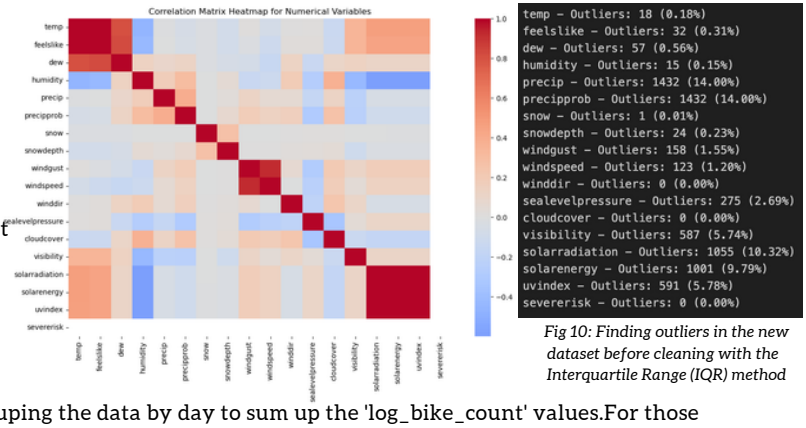


Fig 10: Finding outliers in the new dataset before cleaning with the Interquartile Range (IQR) method

Feature Transformation, Construction, and Merging our final dataset

- 1. Feature Transformation and Construction for train.parquet Dataset: How did we proceed concretely to build our ridge model to predict the null values of the target variable ? We identified critical counters and dates where 'log_bike_count' values were zero.

This was achieved by iterating through each unique counter name and grouping the data by day to sum up the 'log_bike_count' values.For those counters and dates where 'log_bike_count' was zero, we replaced these values with NaN (Not a Number). This approach helped in distinctly marking the data points that needed prediction. Ridge regression was chosen as the predictive model for several reasons: it incorporates L2 regularization, which penalizes large coefficients. This regularization helps prevent overfitting, making the model more robust, especially when dealing with multicollinearity.Given the complexity and potential multicollinearity in our dataset, Ridge regression was suitable for managing these aspects while still providing reliable predictions.

2. Feature Transformation and Construction for the Paris Weather Dataset: The Paris weather dataset is loaded, with datetime values standardized. Missing timestamps are identified and imputed to ensure continuity. Redundant or irrelevant columns are removed, leaving only essential weather variables. This step simplifies the dataset and focuses on the most impactful features for the analysis. Finally, numeric features are scaled using MinMaxScaler. This normalization standardizes the range of continuous initial variables, which is beneficial for our algorithms that are sensitive to the scale of input features.

3. Feature Transformation and Construction for the Velib Dataset: We create a new variables after Data Aggregation and Distance Calculation. We compute distances between Velib station coordinates and site locations from the 'train.parquet' dataset. This geographical information is crucial for understanding the proximity of Velib stations to the bike counters. Descriptive statistics (mean, std ...) are calculated for the distances, and only relevant summary metrics are retained. This reduces the complexity of the data, focusing on key features like mean and standard deviation of distances.

Merging: The final dataset is obtained by merging these transformed and processed datasets. The merge process involves aligning the datasets on common keys or indices (likely timestamps and locations) and consolidating the features from each dataset into a unified structure.

MODEL PIPELINE

Model Selection Rationale

Hyperparameters tuning

Cross-Validation Strategy

Computational Considerations

MODEL EVALUATION

- Performance Metrics
- Results
- Comparison to Baselines and Previous Models
- Limitations and Assumptions

CONCLUSION

- Summary of Key Findings
- Recommendations and Next Steps
- Personal reflections