PYTHON FOR DATA ANALYSIS – PROJECT 2022

Sarushan Sathananthan and Françoise Ruch

Dataset:
https://archive.ics.uci.edu/ml/datasets/Facebook+Comment+Volume+Dataset

DATASET

- The dataset represents the Facebook comment volume.

- **Shape of the data:**
  - We have 40 949 rows and 54 columns.

- **Objective:**
  - Our goal is to predict how many comments a post is expected to receive in next H hours **->** Column= 'Nb_comments_h_hour'

COLUMNS

- **54 columns:** (all the columns are in int or float)

'Popularity', 'Nb_views', 'Daily_interest', 'Category',

'Min Page 0', 'Max Page 0', 'Average Page 0', 'Median Page 0', 'Sd Page 0',

'Min Page 1', 'Max Page 1', 'Average Page 1', 'Median Page 1', 'Sd Page 1',

'Min Page 2', 'Max Page 2', 'Average Page 2', 'Median Page 2', 'Sd Page 2',

'Min Page 3', 'Max Page 3', 'Average Page 3', 'Median Page 3', 'Sd Page 3',

'Min Page 4', 'Max Page 4', 'Average Page 4', 'Median Page 4', 'Sd Page 4',

'Nb_comments_total_CC1', 'Nb_comments_last_24h_CC2', 'Nb_comments_last_48-24h_CC3',

'Nb_comments_first_24_CC4', 'Diff_CC2-CC3', 'Selected_Time', 'Nb_characters', 'Nb_shares', 'Promoted',

'H_Hour', 'Sunday_published', 'Monday_published', 'Tuesday_published', 'Wednesday_published',

'Thursday_published', 'Friday_published', 'Saturday_published', 'Sunday_selected_time',

'Monday_selected_time', 'Tuesday_selected_time', 'Wednesday_selected_time',

'Thursday_selected_time', 'Friday_selected_time', 'Saturday_selected_time', **'Nb_comments_h_hour'**

CLEANING

- 1) Delete the column Promoted (all the values were equal to 0)
- 2a) Create a column **Day_published** (where 0=Sunday, 1=Monday, 2=Tuesday, 3=Wednesday, 4=Thursday, 5=Friday, 6=Saturday)
- 2b) Delete all the columns (Sunday-Monday-…)_published (all the information of theses columns are in the column Day_published)
- 3a) Create a column **Day_Selected_Time** (where 0=Sunday, 1=Monday, 2=Tuesday, 3=Wednesday, 4=Thursday, 5=Friday, 6=Saturday)
- 3b) Delete all the columns (Sunday-Monday-…)_selected_time (all the information of theses columns are in the column Day_Selected_Time)
- 4) Drop columns min_page, max_page, average_page, median_page, and sd_page for page=0,1,2,3,4.

PREPROCESSING

- **Scaling:**

In order to better visualize our data, we will do a **copy of our data (datascale)** and we will scale some columns with a **standard scaler** method.

Name of scaled columns: 'Popularity', 'Nb_views', 'Daily_interest', 'Nb_comments_total_CC1', 'Nb_comments_last_24h_CC2', 'Nb_comments_last_48-24h_CC3' 'Nb_comments_first_24_CC4', 'Diff_CC2-CC3', 'Nb_characters', 'Nb_shares'

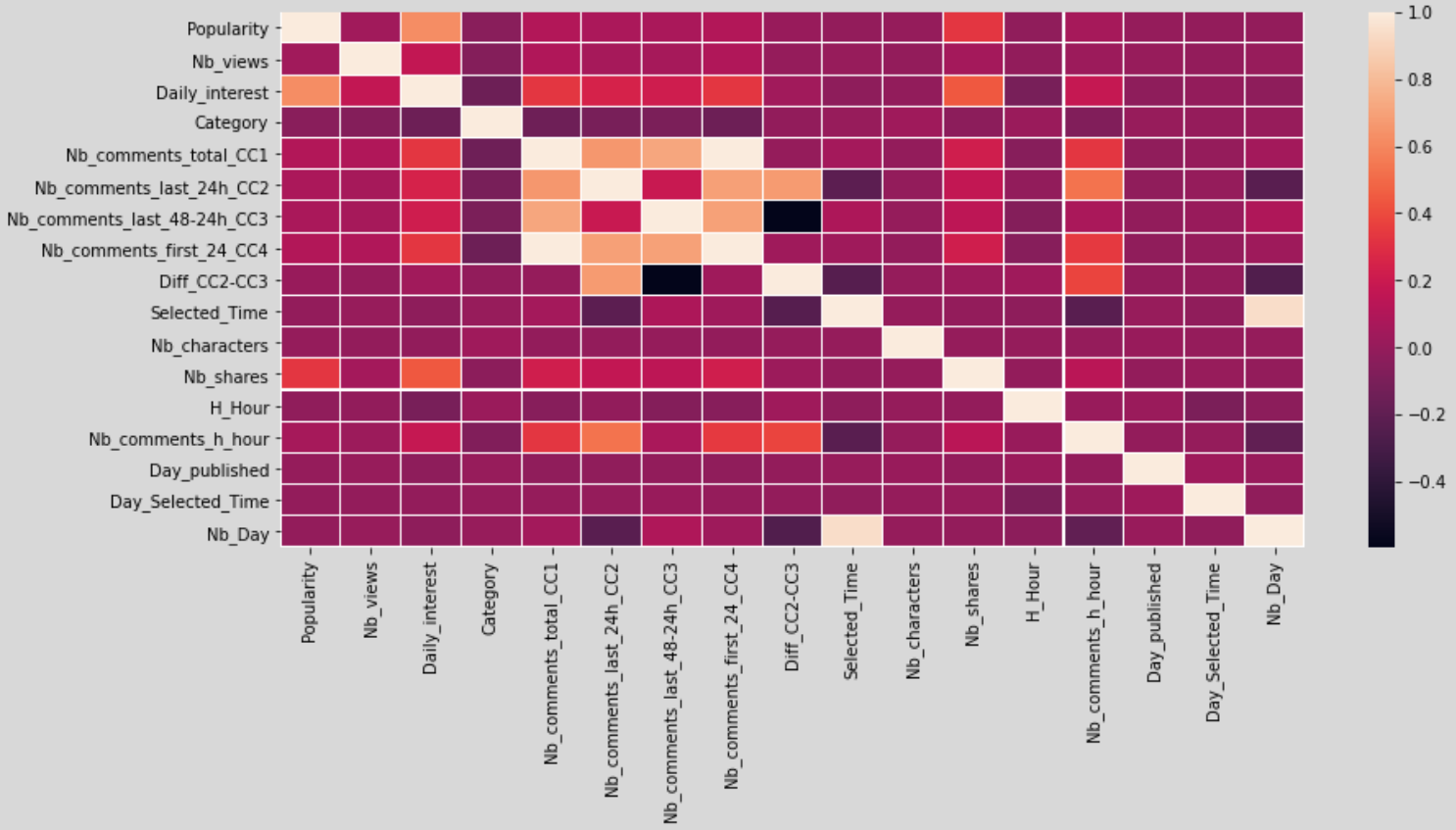
PREPROCESSING

- We have added the column Nb_Day on datascale in order to do comparison depending on the number of simulation days through graphics.
- Finally, the data we will use for dataviz looks like the following.

#	Column	Non-Null Count	Dtype
0	Popularity	40949 non-null	float64
1	Nb_views	40949 non-null	float64
2	Daily_interest	40949 non-null	float64
3	Category	40949 non-null	int64
4	Nb_comments_total_CC1	40949 non-null	float64
5	Nb_comments_last_24h_CC2	40949 non-null	float64
6	Nb_comments_last_48-24h_CC3	40949 non-null	float64
7	Nb_comments_first_24_CC4	40949 non-null	float64
8	Diff_CC2-CC3	40949 non-null	float64
9	Selected_Time	40949 non-null	int64
10	Nb_characters	40949 non-null	float64
11	Nb_shares	40949 non-null	float64
12	H_Hour	40949 non-null	int64
13	Nb_comments_h_hour	40949 non-null	int64
14	Day_published	40949 non-null	int64
15	Day_Selected_Time	40949 non-null	int64
16	Nb_Day	40949 non-null	int64

DATAVISUALIZATION

HEATMAP



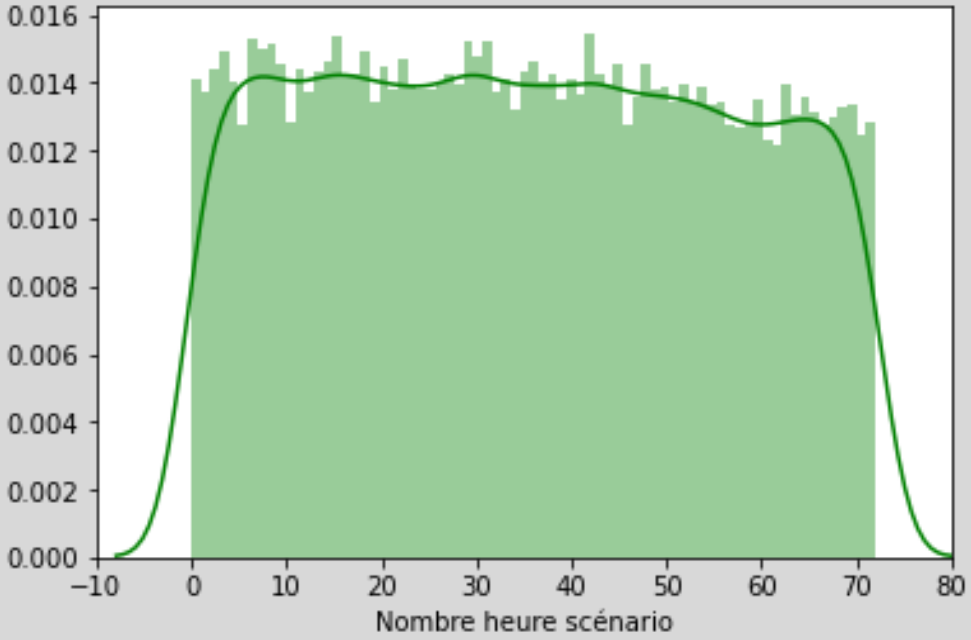
- With this heat map we can see that with the column target the most correlated variables are:
  - Selected_Time / Nb_Day
  - Nb_comments_total_CC1,
  - Nb_comments_last_24h_CC2,
  - Nb_comments_first_24h_CC4,
  - Diff_CC2_CC3,
  - Nb_shares,
  - Daily_interest.

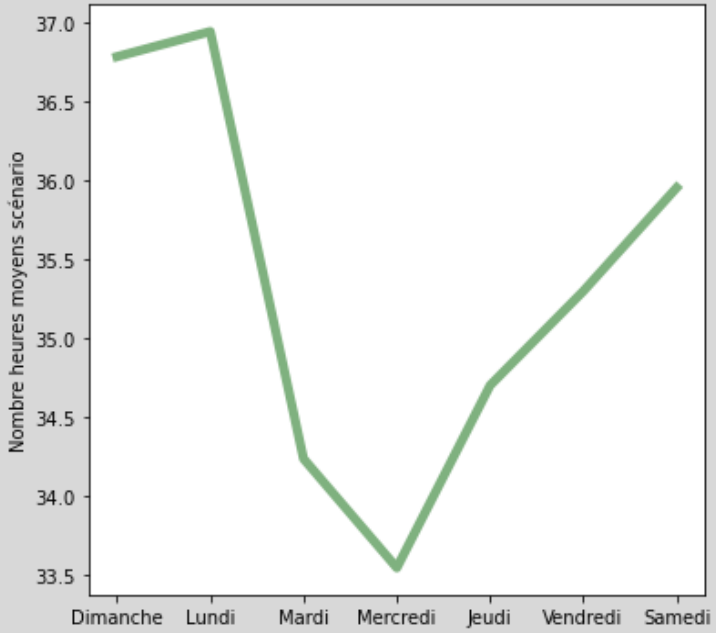
HEATMAP

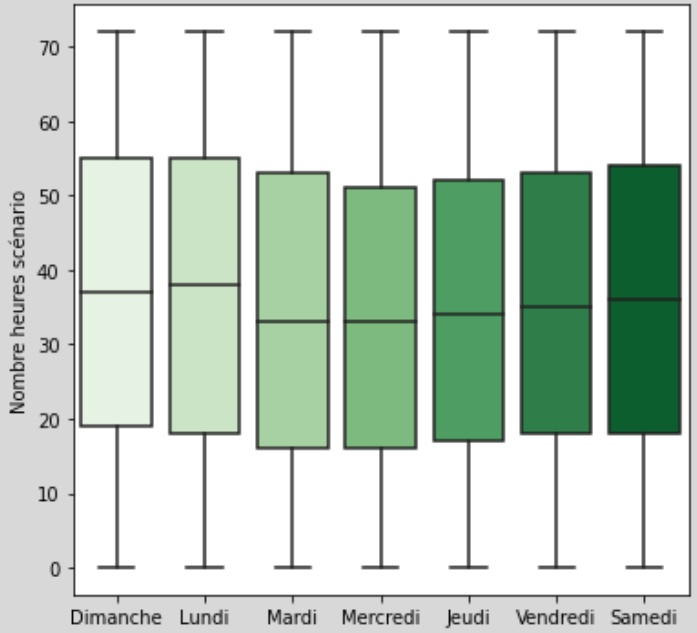
- We can verify this information with a more complete correlation matrix.

	Popularity	Nb_views	Daily_interest	Category	Nb_comments_total_CC1	Nb_comments_last_24h_CC2	Nb_comments_last_48-24h_CC3
Popularity	1.000000	0.044839	0.623436	-0.042167	0.105624	0.077773	0.071448
Nb_views	0.044839	1.000000	0.166850	-0.060181	0.098352	0.061610	0.064703
Daily_interest	0.623436	0.166850	1.000000	-0.148685	0.329139	0.251529	0.217939
Category	-0.042167	-0.060181	-0.148685	1.000000	-0.145932	-0.103961	-0.094728
Nb_comments_total_CC1	0.105624	0.098352	0.329139	-0.145932	1.000000	0.657492	0.713641
Nb_comments_last_24h_CC2	0.077773	0.061610	0.251529	-0.103961	0.657492	1.000000	0.193922
Nb_comments_last_48-24h_CC3	0.071448	0.064703	0.217939	-0.094728	0.713641	0.193922	1.000000
Nb_comments_first_24_CC4	0.104064	0.101214	0.329399	-0.148661	0.996736	0.689478	0.699315
Diff_CC2-CC3	0.009764	0.001620	0.041487	-0.013638	-0.000170	0.672384	-0.595761
Selected_Time	-0.005717	0.004760	-0.024592	0.003876	0.055679	-0.211857	0.087907
Nb_characters	-0.003509	-0.005189	-0.011251	0.037214	-0.005036	-0.005820	-0.003411
Nb_shares	0.331114	0.052569	0.443296	-0.028564	0.225711	0.170726	0.151366
H_Hour	-0.021676	-0.014528	-0.107347	0.017831	-0.049533	-0.011621	-0.063407
Nb_comments_h_hour	0.058918	0.022982	0.177330	-0.073673	0.326882	0.528696	0.072088
Day_published	-0.002329	0.007423	-0.022330	0.006056	-0.020310	-0.019978	-0.013365
Day_Selected_Time	-0.004764	-0.005007	-0.003604	0.001134	0.000281	-0.001770	0.009623

ANALYSIS ON SELECTED TIME





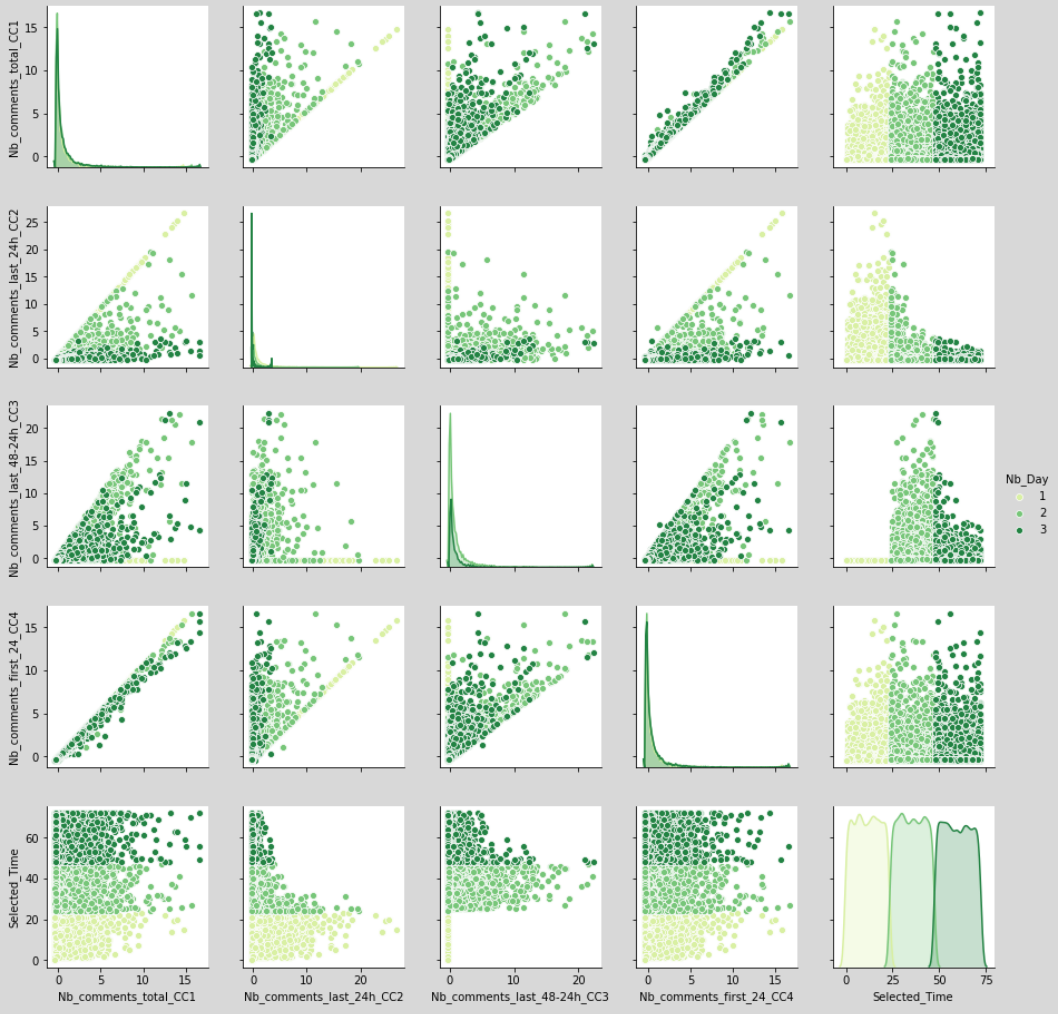


The duration of the scenarios is evenly distributed and sensitivaly similar for each day of the week. So it can be possible to do studies without taking into account specific cases depending on the day.

ANALYSIS ON SELECTED TIME

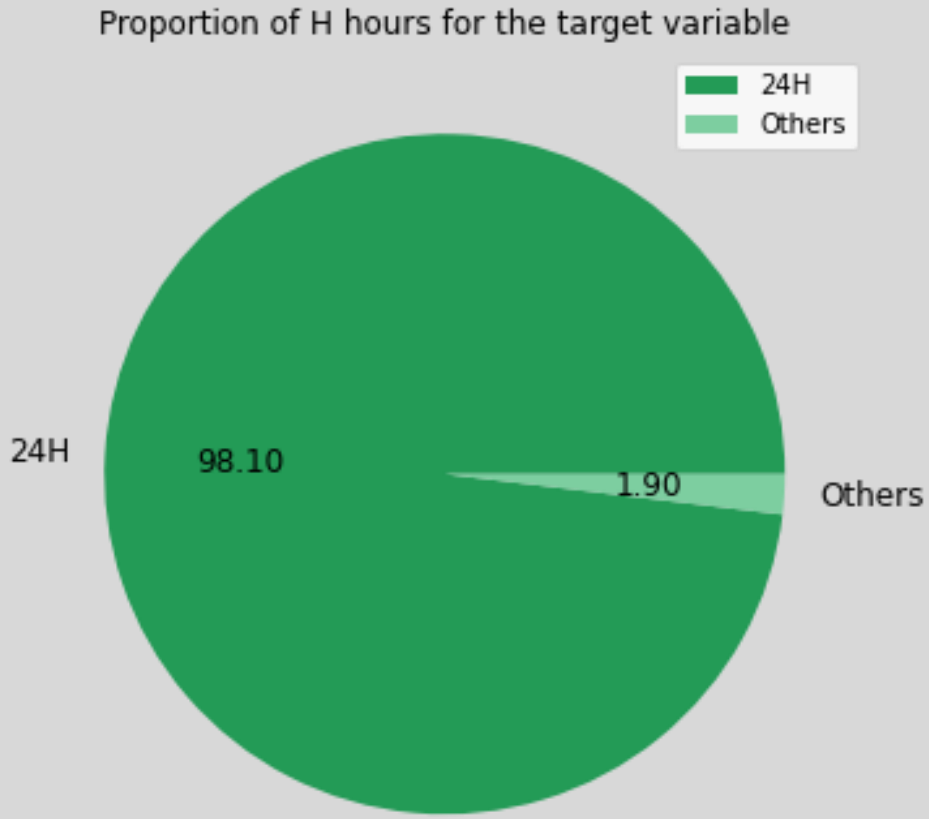
This pairplot shows:
1) The number of comments decreases with the duration of the simulation
2)Most comments are made within the first 24h

So, there is a strong correlation between simulation duration and the number of comments during the last 24h (the correlation is visible in the heatmap)



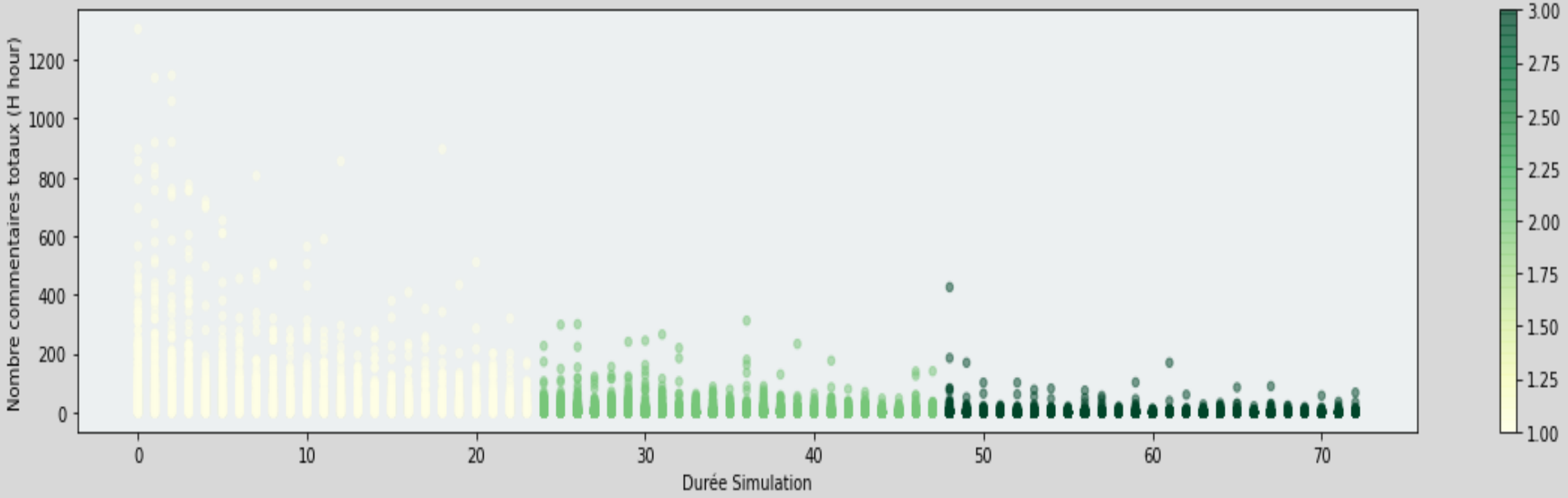
This pie chart allows us to see over how many hours we will predict the number of future comments of a post.

Mostly the prediction covers 24 hours.

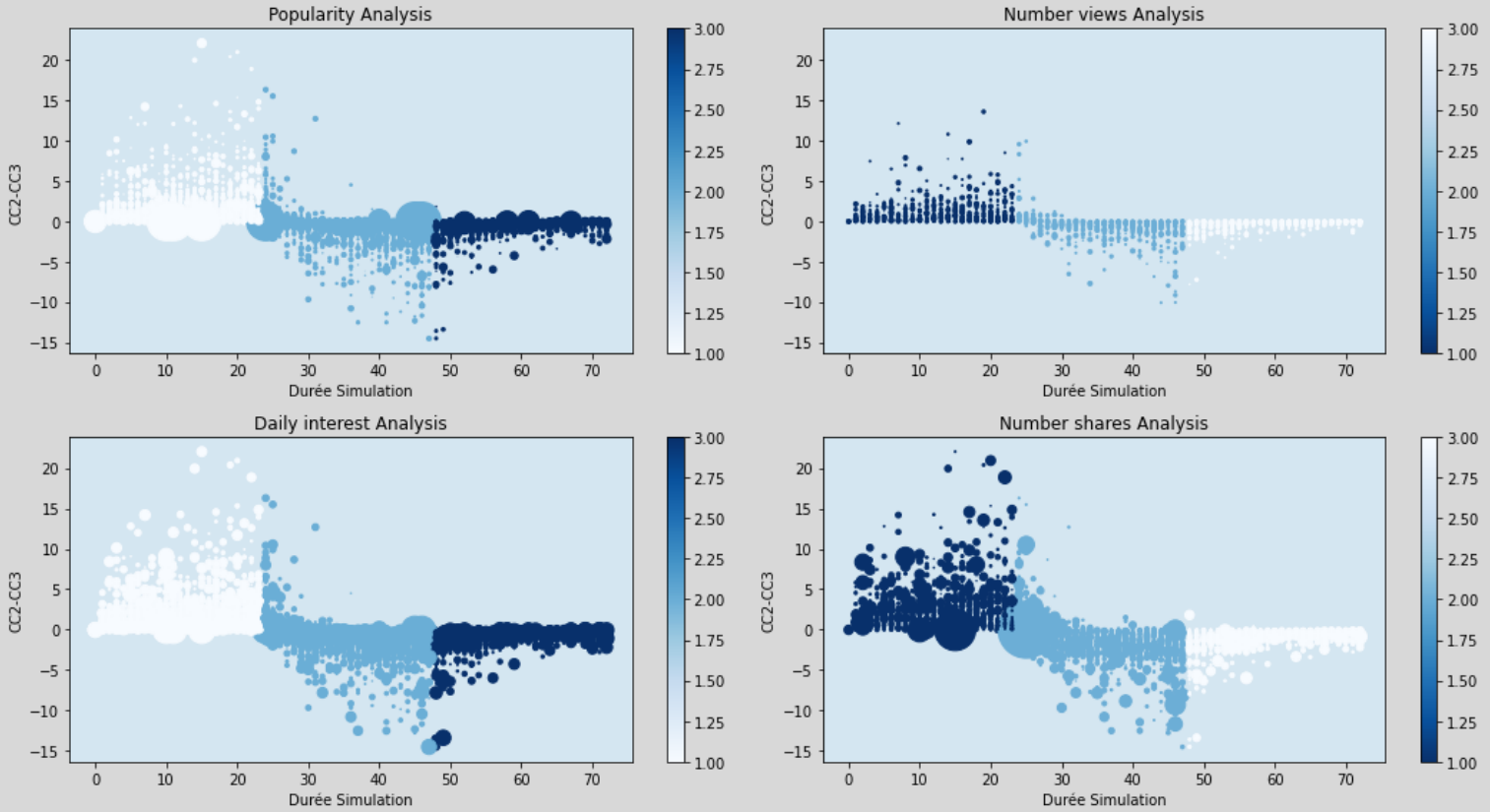


ANALYSIS ON SELECTED TIME

This plot shows that there is negative correlation between the target variable and the simulation duration (this correlation is visible on the heatmap



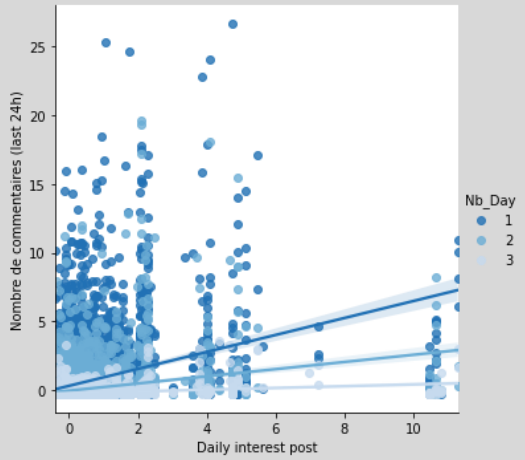
ANALYSIS OF COMMENTS

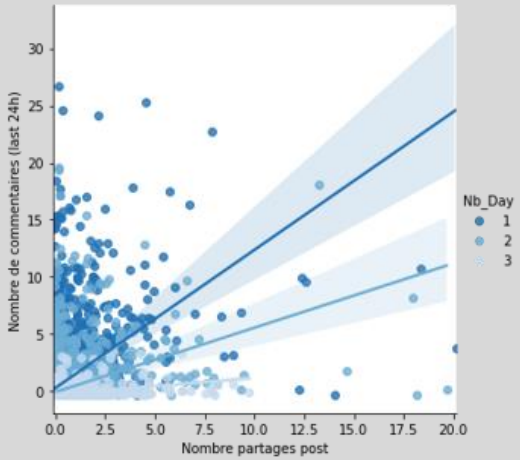


These scatter plots give us some information the reason for the decrease in the number of comments depending on the number of days:

1) A post will tend to be the most shares on the first day
2) The more a post is shared, the higher the number of comments they will be
3)The number of unique views and the popularity of a post has a low impact on reducing the number of comments

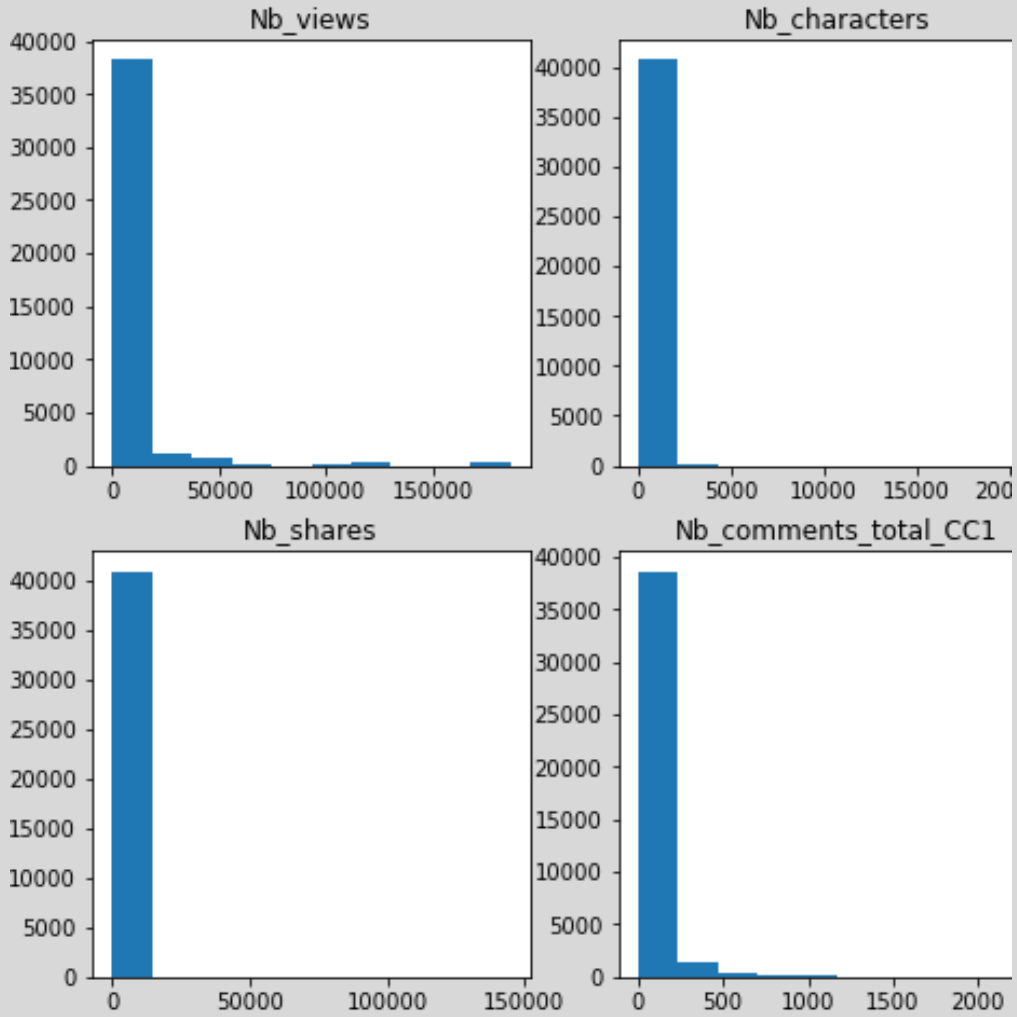
ANALYSIS OF COMMENTS





This linear model plot shows:
- a decrease in the difference between the regression lines which indicates a decrease in the number of daily comments over time
- a increase in the number of comments during the last 24h depending on daily interest (figure 1)
- a increase in the number of comments during the last 24h depending on number of shares (figure 1)

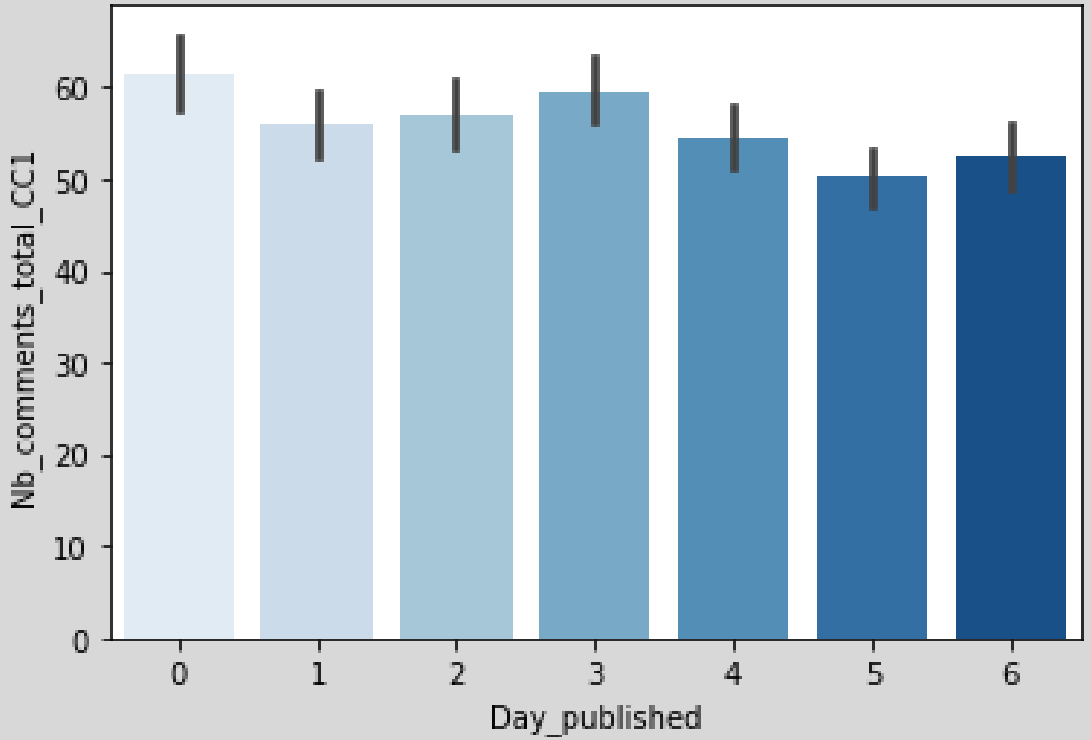
OTHER PLOTS



- We can see that most of publication have a number of views, number of characters, number of shares and number of total comments that are lower than 500.

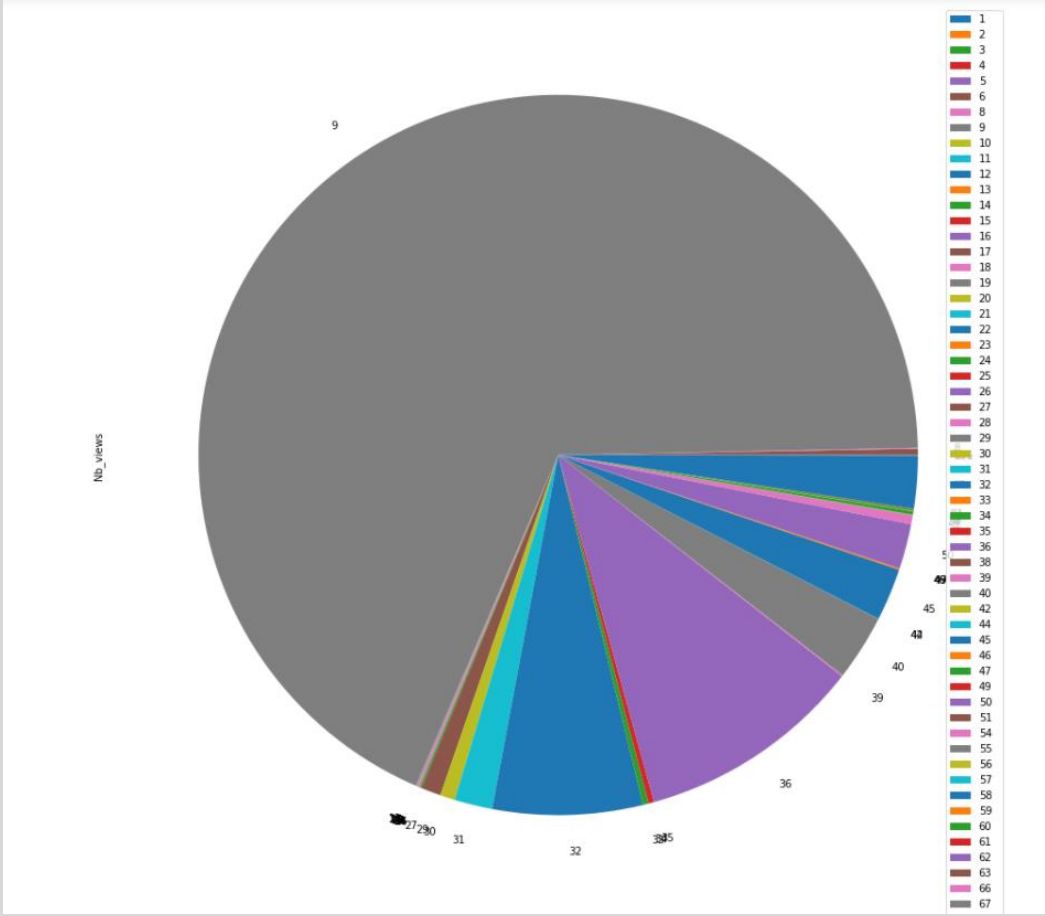
OTHER PLOTS

- With this bar chart we can see that the number of total comment is different dependind the day of the publication but the difference is very small. For all the day the number of comments is near to 2000.



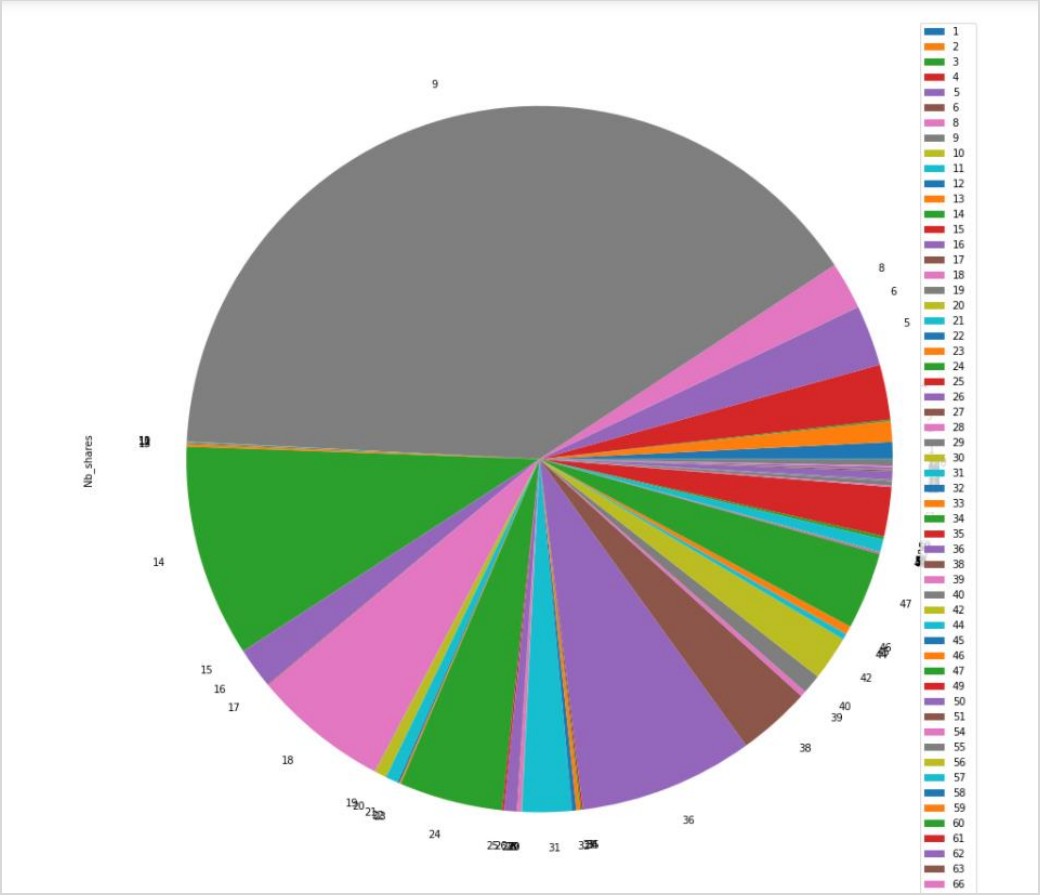
OTHER PLOTS

- In this pie chart we can see depending on which category the number of views.
- The category 9 is the most viewed category.



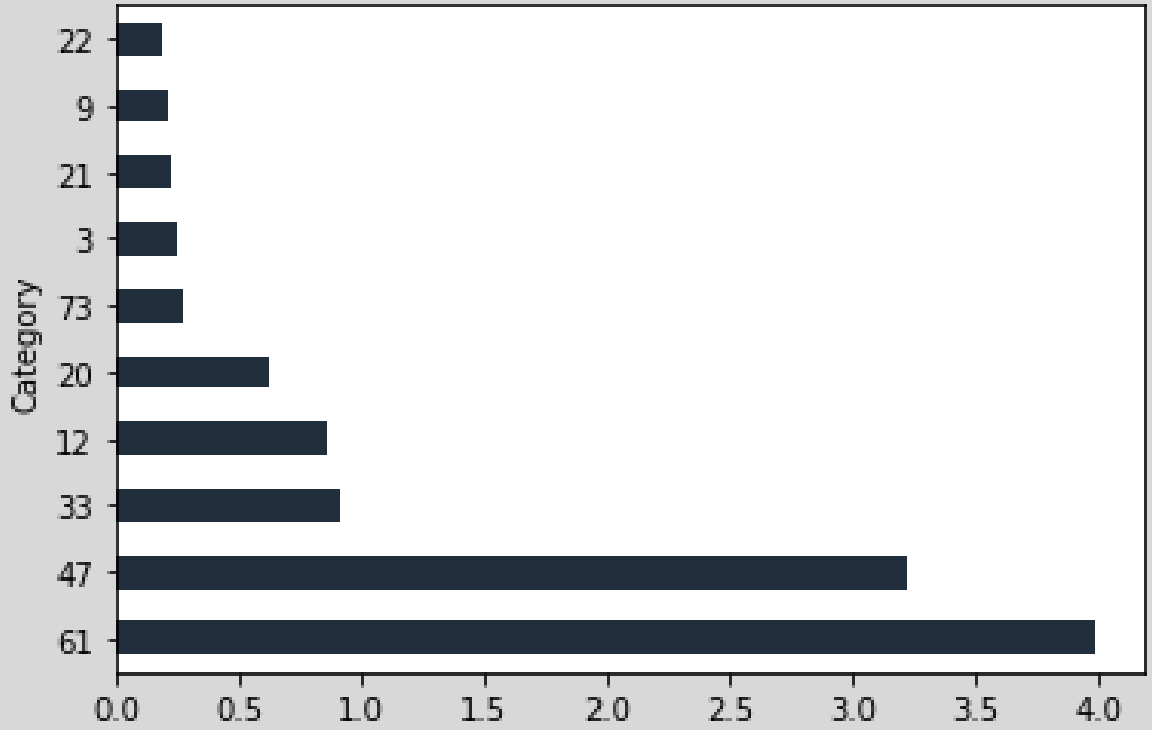
OTHER PLOTS

- In this pie chart we can see depending on which category has the higher number of shares.
- The category 9 is the most shared category.



OTHER PLOTS

- In this horizontal barchart we can see which category is the most popular .
- The category 61 is the most popular category.



COLUMNS SELECTED FOR MODEL

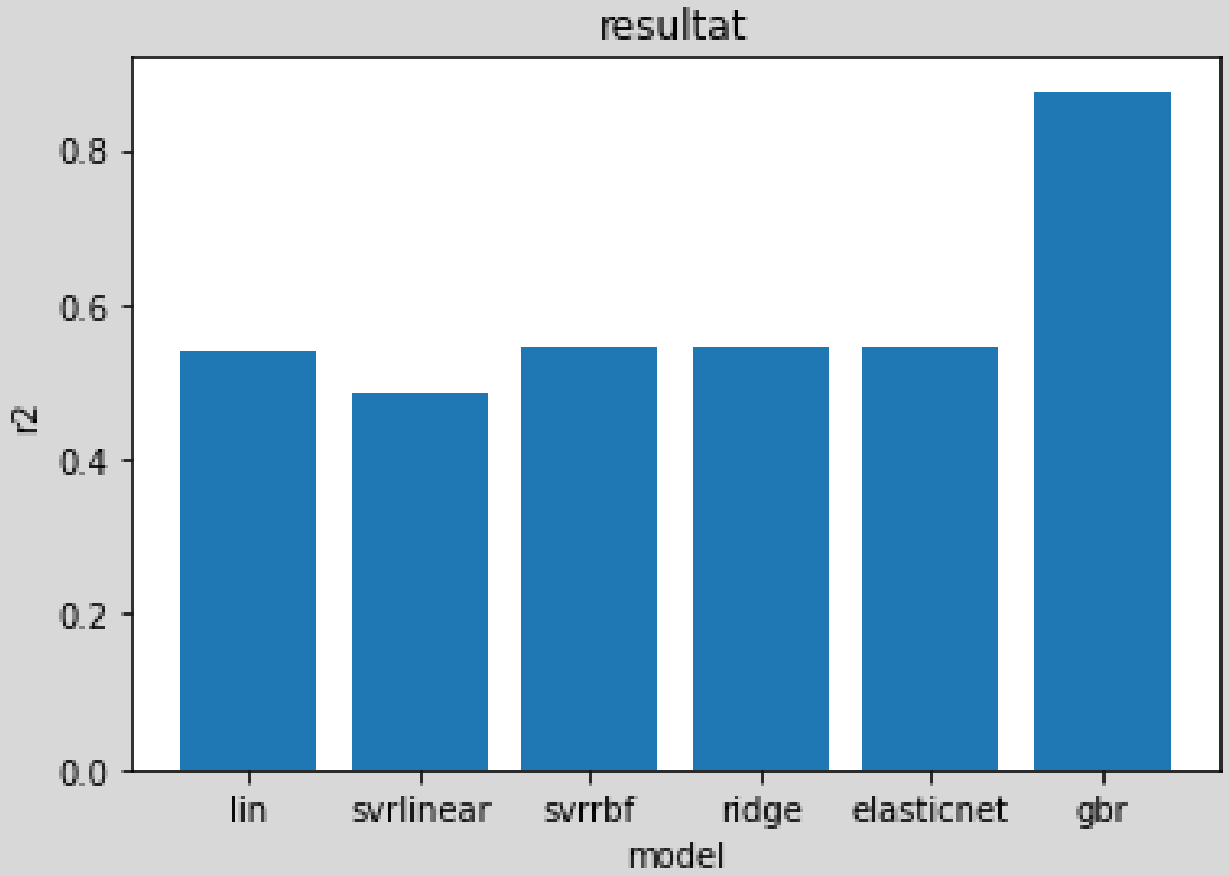
- X columns : 'Popularity', 'Nb_views', 'Daily_interest', 'Category',
- 'Nb_comments_total_CC1', 'Nb_comments_last_24h_CC2',
- 'Nb_comments_last_48-24h_CC3', 'Nb_comments_first_24_CC4',
- 'Diff_CC2-CC3', 'Selected_Time', 'Nb_characters', 'Nb_shares', 'H_Hour','Day_published', 'Day_Selected_Time'

- Y column : "Nb_comments_h_hour"

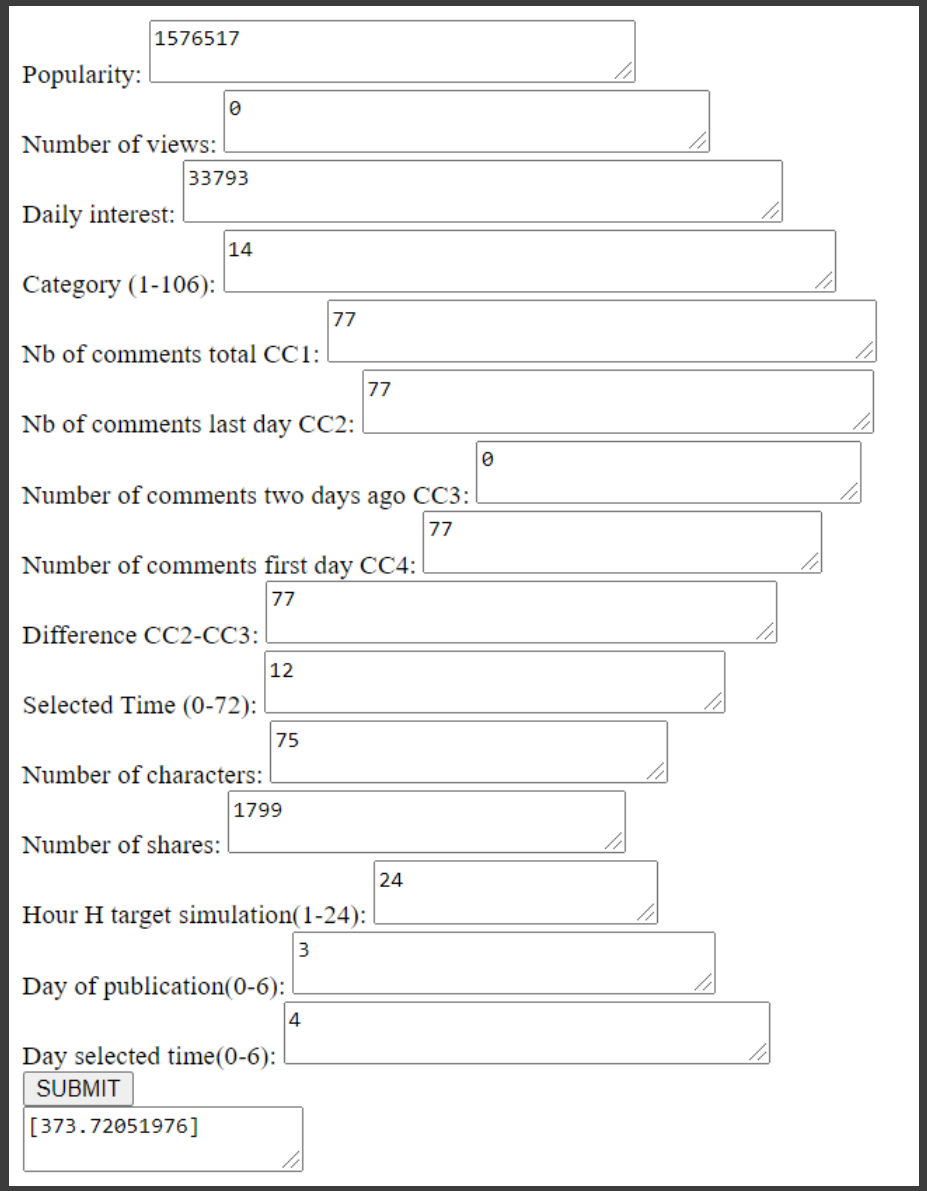
We decided to take only 10 000 observations because the time running for all model was very slow with all the observation.

- X size: (8 000,15)
- Y size: (2 000,)

MODEL



- LinearRegression() : 0,5405
- SVR(linear): 0,4831
- SVR(rbf): 0,5554
- Ridge(): 0,5429
- ElasticNet(): 0,5420
- GradientBoostingRegressor(): 0.8754



API WITH GRADIENT BOOSTING REGRESSOR

CONCLUSION

- The best model is Gradiant Boosting Regressor with a accuracy around 87%.

- We can have a better results by training with more observations but it will take too long time to be executed.