Python for Data Analysis

Guillaume BOULEY, Ariane HUCKEL

Dataset Description

The Dataset **here** was created in 2016 by Kamaljot Singh and studies the networking activities that occur in a famous social media, whenever posts are published. The **influence of social media** is, and now even more than in 2016, so big, that Facebook ads are more and more used since they can affect the **2,74 billions** active users per month.

This project aims to predict the number of comments a Facebook post can get at different periods of time after publishing it.

Dataset Description

The data originates from Facebook pages. The raw data is crawled using crawler that is designed for this research work : designed using JAVA and Facebook Query Language (FQL). The raw data is crawled by crawler and cleaned off.

53 features are used to describe the data, while a **target** value estimates how much comments the post is getting. The inputs describe the page as its popularity, the amount of likes it has, and other describe when the post was published, the amount of comments in 24, 48 hours, …

Dataset Description

5 datasets have been saved, where each corresponds to a different base date/time. The difficulty here, was to take in consideration all of the 5 variants for our prediction, and not just base our analysis on the first one.

TABLE II. TRAINING SET VARIANTS.

Training set Variant	Instance Count
Variant - 1	40,949
Variant - 2	81,312
Variant - 3	121,098
Variant - 4	160,424
Variant - 5	199,030

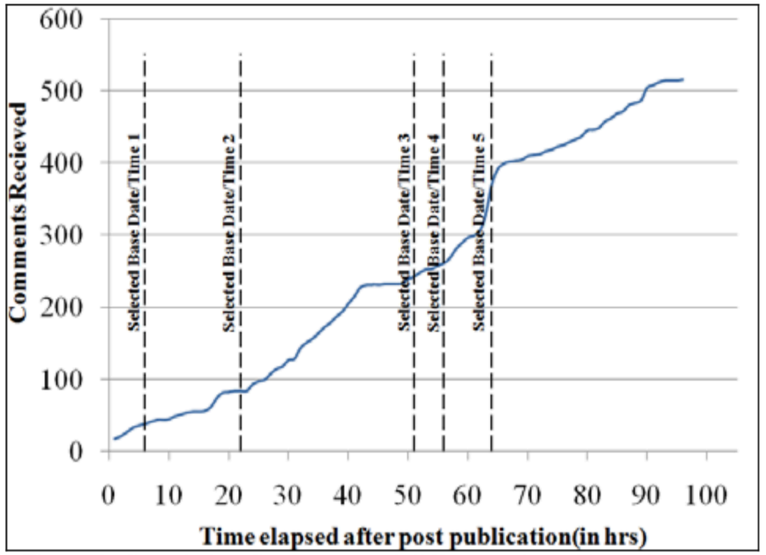


Figure 3. Cumulative Comments and different selected base date/time.

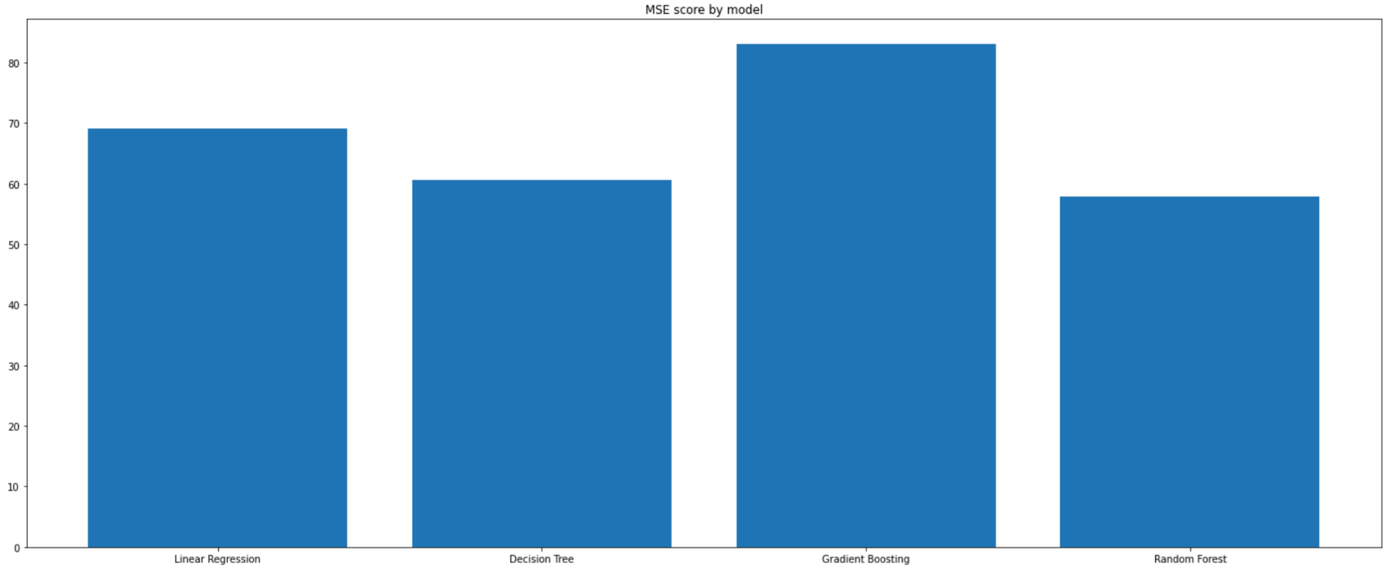
Dataset Pre-processing

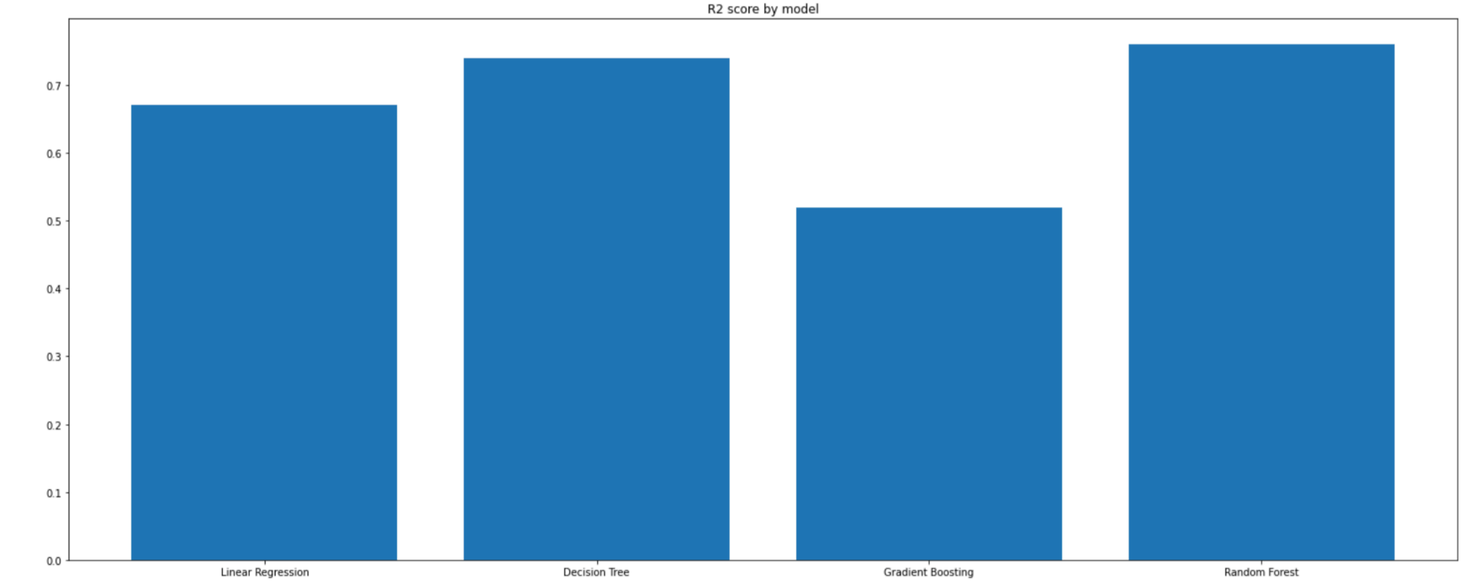
In order to understand the dataset, we needed to see the relationships between all the features, as well as the possible correlations. We plotted some matrices and renamed the columns so we could have a better understanding of this complex dataset.

After manipulating the datas for a moment, we decided to train different models, as **Linear Regression, Decision Tree, Gradient Boosting and Random Forest**, to meet our goal of prediction.

Dataset Modeling

We based our studies on the Mean Square Error method, as well as on the r^2 score.





The Random Forest model took the longest but was the best-fitting algorithm.

API

Then, we decided to turn our best model into an API, using the Pickle module.

Therefore, we could predict new values in one click !

```
Copyright (C) Microsoft Corporation. Tous droits réservés.

Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6
PS C:\Users\boule\Desktop\Python projet> & C:/Users/boule/AppData/Local/Programs/Python/Python38-32/python.exe "c:/Users/boule/Desktop/Python p
 * Serving Flask app "testapi" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
PS C:\Users\boule\Desktop\Python projet> & C:/Users/boule/AppData/Local/Programs/Python/Python38-32/python.exe "c:/Users/boule/Desktop/Python p
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 297-411-659
 * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
 * Detected change in 'c:\\Users\\boule\\Desktop\\Python projet\\app.py', reloading
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 305-709-796
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [10/Jan/2021 22:58:51] "GET / HTTP/1.1" 404 -
127.0.0.1 - - [10/Jan/2021 22:58:56] "GET / HTTP/1.1" 404 -
127.0.0.1 - - [10/Jan/2021 22:59:16] "GET / HTTP/1.1" 404 -
127.0.0.1 - - [10/Jan/2021 22:59:21] "GET /api HTTP/1.1" 405 -
127.0.0.1 - - [10/Jan/2021 22:59:59] "GET /api HTTP/1.1" 405 -
127.0.0.1 - - [10/Jan/2021 23:00:19] "GET /api/ HTTP/1.1" 405 -
127.0.0.1 - - [10/Jan/2021 23:01:50] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [10/Jan/2021 23:01:59] "GET /api/ HTTP/1.1" 405 -
```

```
C:\Users\boule\Desktop\Python projet>python request.py
53
<Response [200]> "[0.26953423]"
```