**Estimating the Number of Active GitHub Users Using Random Sampling**

Yash Patel

Big Data Analytics, Worcester Polytechnic Institute, [yppatel@wpi.edu](mailto:yppatel@wpi.edu)

# 1 Introduction

The number of active github users is very large and counting them by normal means is impractical because of the time and resources it takes. To be able to solve this, we can use a sampling methodology to be able to estimate the number of active github users. With my proposed sampling method of using bins, I can say with 95% confidence that the true number of active users is between 119931537 and 120202479.

# 2 Proposal

Estimating the total number of active GitHub users is a challenge because there are over 100 million valid GitHub ids and checking each id on whether it is valid or not can take a ton of time. Not only does it take time to extract ids from the GitHub API, but you can also only receive a maximum of 100 users per request and there is a limit on the number of requests to be made. With these limitations, a sampling method is needed to estimate the total number of active GitHub users.

The estimation method I propose is to get the average number of active users in a set range. Then multiply the average by the number of total sets to get the estimation for the total number of active GitHub users

# 3 Methodology

## 3.1 Sampling / Estimation method

Understanding the GitHub user data is important to understand how we will be estimating the number of active users. Every user is given a numerical id. A new id is generated by adding 1 to the previous new id. This means that ids are added sequentially. The entire id space ranges from 1 to the newest id. As of writing this report, we found the latest GitHub id to be 124937369. We can see from this that there are probably 124 million total users however, not all ids within this space are active. To determine which users are active we use the GitHub API. We can receive a list of ids when requesting all of the users since a certain id. There is a limit on the total number of users it returns which we can set to 100. The list of user ids that it returns are all of the active users.

For example, if I request 100 users since id number 200. It will give me a list of ids: 201, 202, 217, 219, etc. Notice that it did not return 203 meaning that id 203 is not an active user. This query is the K - Nearest query where we get the K number of users since our id. This is a right K - Nearest Query so it will only return users after our specified id.

The sampling and estimation method is similar to how to determine the number of Youtube videos there are. We are going to create b bins each with a set length of l. The length s indicates the range for the ids. Each bin has a starting id and an ending id to determine whether a GitHub user belongs to the bin. For example, if s = 1000 then we can have bins that hold ids 0-999, 2000 - 2999, and 1031 - 2030.

b = number of sampling bins

s = length of each bin

N = total number of ids

B = total number of bins

x = number of active users in a bin

To generate our b bins, we will generate a random number to be the starting id. The ending id is the starting id + s. We ensure that the bins do not overlap by checking if the random number falls within any of the bins that we have already created. We fill each bin with all of the valid active users using the GitHub API. We then take the total number of active users in each bin. After that, we sum the number of active users and divide it by the number of bins to get the average number of active users per bin. We calculate the total number of bins, B, by taking the total number of ids and dividing it by the length of each bin, N / s. Here is how we do the calculations but in a more readable format.

B = N / s

X\_avg = (x1 + x2 +...xb) / b

Total\_github\_users = B \* X\_avg

## 3.2 Prood of unbiasedness

The number of Github active users is not uniformly distributed across the entire id space. There is a higher chance that there are fewer active users as the id approaches 0 and there are more active users as the id approaches the latest user. This is because the new user is less likely to simply delete their account as soon as they make it.

This estimation method uses random bins made from the entire dataset to generate the average active users. When generating the random bins, we randomly pick a starting id from the entire dataset. This ensures a uniform distribution of our bins. But we also need to pick a large number of bins to encapsulate the wide range of the data. We also need to ensure that the bins are large enough to gather enough range. For example, if our bin size was only 10 then our active users will most likely be at around 100%.

# 4 Results and Evaluation

## 4.1 Validation set

Here are the average number of active users with a validation set of size 60,000 using 20 bins and 100 users per bin. The true average is 56066.

55800

55710

55980

55800

56430

56670

56220

55620

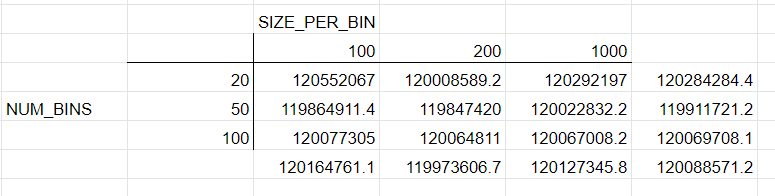
56070

56190

From our estimation method, the true average falls within the range and is not too far off. To prove this I took the average of all of these estimations and computed the Average, Std dev, and confidence interval of 95%. Our 95% confidence interval is (55687.70598, 56620.29402). The true mean falls within this range which helps prove that our estimation method is valid and unbiased.

## 4.2 Entire id space

We sampled using 20, 50, and 100 bins with 100, 200, and 1000 ids in each bin. The purpose of this is to check which set is optimal in determining the number of active GitHub users. I ran each scenario 5 times because of the limitation on the amount of requests I can do in the API. The main results can be found in the excel spreadsheet attached but here are the averages for all of the scenarios.



We used the last GitHub id to be 124,937,369 and the average of each scenario to be 120,088,571 therefore we can say that there are 4,849,797 inactive accounts. Another way to say it is that GitHub has a 0.9611 active to total id GitHub user percentage. Also, the 95% confidence interval for each of the scenarios lines up with each other with the Number of bins = 100 and Size\_per\_bin = 1000 having the smallest interval range. Since that was our largest sample size, we can say with 95% confidence that the true mean is between

| 119931537 | 120202479.4 |
| --- | --- |

# 5 Conclusion

Using our non-biased random bin sampling method, we can get an estimation for the number of active users in GitHub. From our different scenarios we found the number of active users to be around 120 million users which indications around 5 million users are not considered active. This is to note that we defined active and nonactive users on whether their github id is still being used even if the account itself may have not been used in a long time. To further improve the estimation, we can run the Number of bins = 100 and size per bin = 1000 with more iterations rather than just 5. This was constrained due to time especially since the api has a limit on the number of requests to be made in an hour.