**(Documentation) - Git Hub Repository Analysis**

**Abstract**:

Team sizes like small and large make a huge difference in contribution of new scientific Ideas. In this project we want to find out if this is true. There is a belief that smaller teams produce most scientific Ideas than the larger teams as their research goes way into the past. Small Teams are Known for their Disruptive Work, Whereas Large teams for their developing work. There is very little proof that large teams do knowledge discovery and technological Inventions.

**Previous Work**:

Previously, the research showed that large article and research teams received more citations. Well, there is little known that citations alone will not decide the contribution, this is represented in two well known articles called BTW Model and Bose – Einstein Condensation. BTW Model had cited only the model itself without mentioning the references from the article. Bose Einstein Condensation is almost always cited by Bose, Einstein and other antecedents. Researchers Consistently saw from past 60 years that larger teams produce articles, patents and software with a disruption score that markedly and monotonically declines with each additional team member. Differences in disruption between works produced by small and large teams are magnified as the impact of the work increases high-impact papers produced by small teams are the most disruptive, and high-impact papers produced by large teams are the most developmental. Disrupting (*D* > 0) or developing (*D* < 0) articles. Small teams may generate more theoretical innovations and large teams more empirical analyses. Review articles with fewer authors are more disruptive than those with more, just as with original research articles. Teams of between 1 and 10 authors account for 98% of articles, 99% of patents and 99% of code repositories. Bootstrapped 95% confidence intervals are shown as grey zones. Another possible explanation for our results is that the team effect that we observe occurs because the scientists, inventors and software designers involved in larger teams are qualitatively different from those comprising smaller teams.

**Problem Statement**:

Part 1:

1. How may unique contributors for each group? What’s the distribution of the number of contributors among all projects here? For example, what is the mean, and the standard deviation? Is it a normal distribution?
2. Is there any patterns for the contributors activities, such as top 10 contributors contribute 90% development activities?
3. Using check in plus check out as the development activities, how these activities distributed among the who projects?

Part 2:

1. Find the Total Team contributors and the Core contributors.

**Analysis**:

**Core Contributor of a repository:** If the number of edits or pushes of the contributor is greater than the average number of edits or pushes of all the contributors to a project. He is said to be a Core Contributor of that particular repository.

**Active period on a repository:** The time period between the first and last edit of the core contributor on a repository.

Part 1: Performed Analysis on a single day 2015-01-01.csv GitHub data scrapped from GitHub archive.

Question 1: Contributor's distribution in the data set?

Answer:

|  |  |  |
| --- | --- | --- |
|  | **repositories** | **contributors** |
| **0** | **1272** | **0** |
| **1** | **1270** | **1** |
| **2** | **14** | **2** |
| **3** | **2** | **3** |

* Out of 2558 unique repositories, we have 1272 repositories having no contributors
* We also see Repositories with 1 contributor the most.
* There are very few repositories with 2 and 3 contributors

Chart, scatter chart

Description automatically generated

Question 2: Patterns for the contributor's activities?

Answer:

* 1 contribution is done in more than half of the repositories.
* only 1 repository has more than 70 contributions.

Chart, histogram

Description automatically generated

Question 3: Activities distribution of push events?

Answer:

* (21481110: 79 > 6058234: 30 > 25173910: 29 > 28650038: 20) are the repositories with maximum contribution done.

Part 2:

The below code helps to find the total contributors or the Team Total size:

A picture containing timeline

Description automatically generated

To connect to local repository where the data is stored in the SQL database we use the below code. This has the database: “github”, password: “Root@123”.



Found Core contributors and Team Total contributors of the below repositories.

Graphical user interface, application, table, Excel

Description automatically generated

**Results**:

Found the Core Contributors and the Number of total contributors of the given 26,898 repositories.

**Future Work**:

Finding the Disruption Values using the below formula.

*ni*– *nj*

Disruption =      ------------------------------

*ni*+ *nj*+ *nk*

*ni = Number of subsequent works citing the focal paper*

*nj = Number of subsequent works citing both focal paper and references*

*nk =Number of subsequent works citing references*

In Repository terms:

ni = Number of subsequent works citing repository.

nj = Number of subsequent works citing both repository and references of a repository

nk= Number of subsequent works citing references.

To implement the formula, we need to recognize the subsequent works, focal paper and references.

The code can be made efficient by making it as a function and sending all the repositories in a loop and getting back the data in a .csv format.