# **What makes a successful GitHub project: Delving into the behaviour of contributors**

Ranjani Amrapali Vishwanath, Yanqing Zhou

## Link to our slides

[Github presentation](https://docs.google.com/presentation/d/1pl-2Yocco9QReSAj-XKMtwu0_AxgGnB6OJ8QrtyYlTo/edit#slide=id.g29d0ecc799a_0_106)

## Introduction

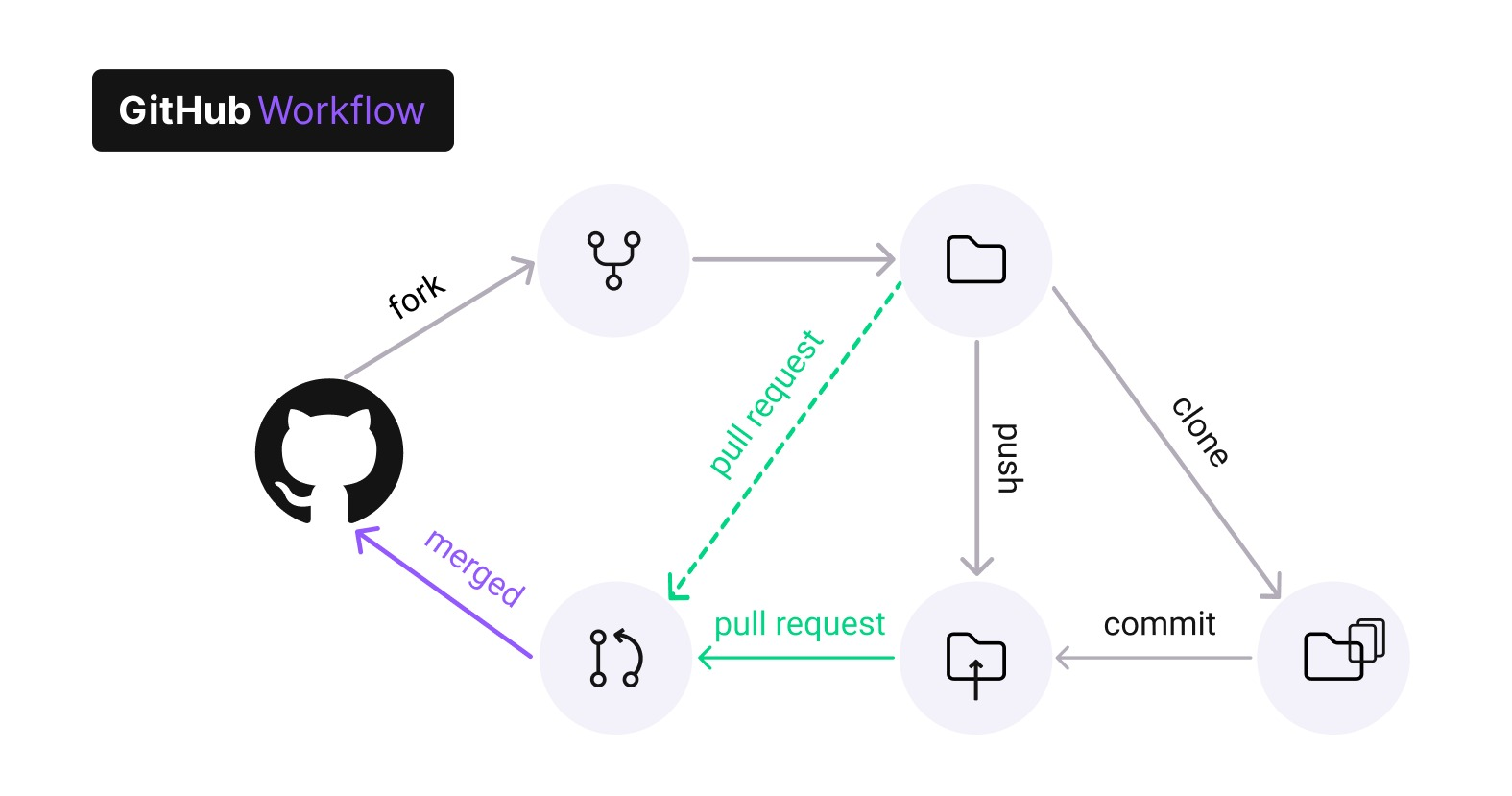
**What is GitHub?**

**GitHub** is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere.

* A **repository** is usually used to organise a single project. Repositories can contain folders and files, images, videos, spreadsheets, and data sets -- anything your project needs.
* **Starring** makes it easy to find a repository or topic again later. You can star repositories to discover similar projects on GitHub. Many of GitHub's repository rankings depend on the number of stars a repository has. GitHub shows popular repositories based on the number of stars they have.
* **Branching** lets you have different versions of a repository at one time. By default, your repository has one branch named main that is considered to be the definitive branch. You can create additional branches off of main in your repository. You can use branches to have different versions of a project at one time. This is helpful when you want to add new features to a project without changing the main source of code. The work done on different branches will not show up on the main branch until you merge it.
* Similar to saving a file that's been edited, a **commit** records changes to one or more files in your branch.

**Github workflow:**

When collaborating on a project, contributors typically **fork** the original repository to create their copy. They make changes in their forked repository, creating branches for specific features or fixes. Once changes are ready, contributors submit a **pull request** to the original repository, proposing their modifications. This pull request acts as a bridge for collaboration, allowing maintainers and other contributors to review the changes, provide feedback, and discuss any potential adjustments. If the changes are accepted, the pull request is **merged** into the original repository, incorporating the contributions from the forked repository into the main project.



## Research objective

Understanding the impact of contributors’ action on the popularity of github projects

## Data processing

* What features we extract:
  + Features of the whole project :
    - No. of forks “Forks”
    - No. of pulls "Pull"
    - No. of contributors "contributor\_numbers\_in\_total"
    - No. of commits "Commit"
    - No. of active days "no\_of\_days"
  + Features of the behaviour of original contributor(referred to as oc below):
    - 1 month, commits in total "first\_month\_commits"
    - 6 months, commits in total "six\_months\_commits"
    - 1 year, commits in total "one\_year\_commits"
    - 2 years, commits in total "two\_year\_commits"
    - The total number of commits by oc "no\_of\_commits\_original"
    - The average time interval between two commits by oc "original\_contributor\_average\_duration"
  + Features of the behaviour of non-original contributors:
    - No. of active days where no. of commits by not-leader exceeds no. of commits by leader. "no\_of\_surpass\_days"
    - Median no of commits by not-leader in the 1st yr by the “username”"one\_year\_median\_non"
    - Median no of commits by not-leader in the 2nd yr by the “username” "two\_year\_median\_non"
* The data that we obtained for many of the features follows a skewed distribution. That is why we do log transformation. The log transformation is the most popular among the different types of transformations used to transform skewed data to approximately conform to normality.
* **Normalisation**: standardising the variables to have equal variance is important because it means they get weighted equally with respect to each other. It's reasonable to weigh the variables the same if you have no reason to think one is more important than the others.

## Data Analysis

* **k-Means Clustering**
  + Choose optimal k
    - The elbow method is used in clustering analysis to determine the optimal number of clusters. It involves plotting the within-cluster sum of squares (WCSS) for different cluster numbers and identifying the “elbow” point where WCSS starts to level off.
    - The average silhouette approach measures the quality of a clustering. That is, it determines how well each object lies within its cluster. A high average silhouette width indicates a good clustering. The average silhouette method computes the average silhouette of observations for different values of k. The optimal number of clusters k is the one that maximises the average silhouette over a range of possible values for k.
    - Though these two methods, we choose k=3
* **Heatmap**
  + A heatmap is a graphical representation of data where values are depicted by colour. Brighter colours are used to represent larger values, while cooler or darker colours represent smaller values.
  + Correlation heatmaps are a type of plot that visualise the strength of relationships between numerical variables. Correlation plots are used to understand which variables are related to each other and the strength of this relationship.
* If we want to see how the project turns out to be after clustering, we use a prediction algorithm to understand whether a project becomes ‘popular’ based on the cluster’s behaviour, we use a prediction algorithm.
* **Linear Regression**
  + Linear regression analysis is used to predict the value of a variable based on the value of another variable. The variable you want to predict is called the dependent variable. The variable you are using to predict the other variable's value is called the independent variable.
  + We select the independent variable differently for different clusters.
  + For all, we want to predict the number of stars of the project.
  + After selecting a list of features from the scatter plot, We look at their correlation between the independent variables. In Linear Regression, we need to be careful about correlation between independent variables.
    - Correlation Coefficient Magnitude

0.7≤∣r∣≤1.0 indicates a strong correlation.

0.4≤∣r∣<0.7 suggests a moderate correlation.

0.0≤∣r∣<0.4 indicates a weak correlation.

* + We remove features that have high correlation with many of the selected features.
  + Selected features for Linear Regression:
    - Cluster 0:
      * surpass not selected because not much info
      * No of contributors 0.19
      * No. of commits original 0.08
      * Correlation between commits and contributors, surpass days and no of days is high, so commits is omitted
    - Cluster 1:
      * No of contributors 0.56
      * OC average duration -0.15
      * Surpass days 0.17
      * OC and surpass have moderate/high correlation with contributors, so no of contributors omitted.
    - Cluster 2:
      * No. of contributors 0.41
      * 1 yr commits -0.04
      * Surpass days 0.32 omitted since it has high correlation with no. of contributors
      * Number of days not taken because in most clusters, for relevant independent variables, inter-correlation is high.
  + To make sure that we have the right model, we need to see to it that the Mean Squared Error(MSE) is not too high, lower it is, better the model. Therefore, we can say that the current model to predict the popularity of the project is not the best model.

## What next?

* Look for a better model for popularity prediction.
* Creating new and more relevant features could be helpful out of the dataset if the current features do not fit the model.