Final Project Notes

Ideas

- Impact of type of event (review, commit, issue, pull, comment)
- Which types of ecosystems most successful
- Impact of indiv v org
- Impact of size of community

Problems:

- Change index of >=100 from 0+ to 100+ (in oracle ds)

Notes from paper:

- **Why studying this**
  - Sustainability problem with open source software
  - Need to find a way to quickly determine which ones are good projects to pick up (not even an easy way to see which ones are active v dead)
  - Can make ML predicting future evolution of a project
  - Want to show risks to orgs that rely heavily on open source projects w/o paying enough attn to their health or sustainability
- **Main theme:** Survival of different github repositories
- **Findings**
  - More than 50% of projects in ds die w/in first 4 years
    - Highest # of deaths occur w/in the first year
    - R has highest survival rates over time but suddenly drops at 5yr
  - Repos by orgs have better chance of surviving
  - Big impact of coding on a project (as opposed to non coding activity) on its development process
  - Once a project moves into zombie, more likely to move to dead for npm, wordpress, and laravel ecosystems
  - About 16% of zombie to dead transitions are the final one (this happened for 21% of projects in this ds)
  - R had the highest number of transitions
  - Average state time in running is higher in orgs and avg dead time is lower for them
    - Same patter as tier # increase
  - Typical evolution for each ecosystem is short period of lots of activity followed by long periods of no activity
  - Org projects have higher chance of surviving (same with higher tiers)
    - All repos that die w/in 1st year in laravel ecosystem are tier 1
    - 59% of repos that survive more than 5yrs are tier 3

    - Entering zombie status is red flag for project, could be used to send warning to those projects
    - Non coding contributors don't have much impact on survivability of project
- **Dictionary**
    - **Bots** = provide services for repositories (automatically create issues or launch external tools each time an event occurs)
    - **Time series dbs =** dbs tailored to the analysis of time series data. Offers support for storing and retrieving time series data, scaling, and providing viz tools or easy integration components with third party viz tools
    - **Project survives** when they have not been abandoned by end of study
        - Right censor when project has not failed by the end but later died
    - **Tiers of projects (community size)**
        - 1 if project has # of contributors lower than min value of interquartile
        - 2 if project has # of contributors inside interquartile
        - 3 if project has # of contributors greater than max value of interquartile
    - **Status of repositories**
        - Alive : some kind of activity w/in 6mo (not including bot activity)
        - Running: coding activity
        - Zombie: only non coding activity
        - Dead
- **Studying four ecosystems (laravel, npm, r, and wordpress) from 2016-2022**
    - Laravel is a free and open-source PHP- based web framework for building high-end web applications
    - Node package manager (npm) is a package manager and a software register but it's also a place where developers can find, build and manage code packages.
    - WordPress is an open-source content management system (CMS). It's a popular tool for individuals without any coding experience who want to build websites and blogs
- **RQ1:** How does the project activity change over time?This research question analyzes the different status (or phases) a project may transition during their lifespan (e.g., periods of time with high activity and others when the project may look dead). By studying the evolution of the project activity, we can better understand the dynamism and common evolution patterns of projects' lives.
    - **Time is in months**
    - Ex: running-running-zombie-dead would be coded as running(2)-zombie(1)-dead(1)
- **RQ2:** What is the survival rate? In this research question we focus on identifying projects that have died and when they have died (i.e., been abandoned and with no human activity). We then calculate the survival rate (i.e., dead vs. alive projects) which will allow us to understand the survivability of the projects across a number of dimensions.

- **Future**
    - Did not consider alive if a fork was still active, so would be interesting to see survival of forks
    - Did not consider bot activity in survival
- **Other considerations**
    - Data pulled from apis may not be complete