Interrogate the GitHub API to build visualisation of data available that elucidates some aspect of the software engineering process, such as a social graph of developers and projects, or a visualisation of individual of team performance. Provide a visualisation of this using the d3js library.

In order to build visualisation of data available that elucidates some aspect of the software engineering process, I looked at 3 aspects;

1. **Developer’s Productivity**
   1. Measured by number of Commits
   2. Number of Branches
   3. Number Repositories
2. **Developer’s Performance**
   1. Measured by number of followers
3. **Developer’s Happiness**
   1. Measured by organization location and World Happiness Rates.

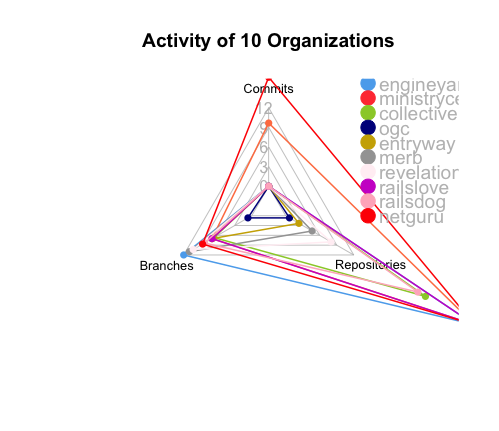
**Graph 1: Assessing the productivity of each of the 10 organizations by measuring their activity:**

**Information:**

* I took a sample of 10 organizations that have at least 3 members. From the 10 organizations, I took a sample of at least 10 members from each organization.
* Taking Number Commits, Branches and Repositories to measure productivity of each of the organizations
* The number of Branches for each organization was taken from the organization’s first 10 repositories
* The number of organization repositories was recorded (max 200)
* The number of Commits for each organization was taken from the organization’s first 10 repositories

**Outliers:**

* The 10th organization has 827 commits. I changed the value to 50 but this will not compromise the interpretation of the visual.



**Modifying Code:**

* The functions getOrgsWithMembers(), getMembers() and list10ReposOfOrg() can be easily modified to take in an increased number of organizations and number of members from each organization by replacing the number 10 in ‘?per\_page=10’ in the ‘GET’ function.

**Result:**

* From the plot it can be seen that the organizations Netguru and Engineyard are productive. While OGC is less productive.

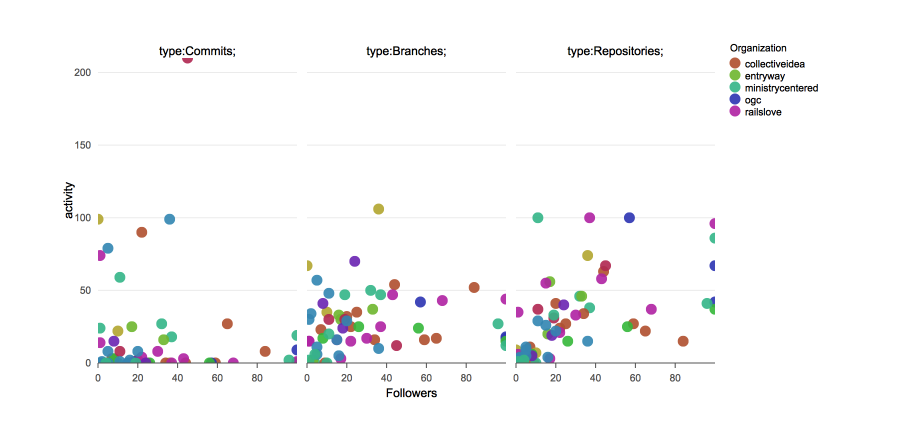
**Graph 2 assesses;**

1. **Does productivity of users influence their performance?**
2. **Are the users who are more ‘productive’ a member of more the more ‘productive’ organizations?**

**Information:**

* Measuring **individual productivity** with Commits, Branches and Repositories
* Measuring Performance with number of Followers
* The number of Branches for each user was taken from the user’s first 10 repositories
* The number of member repositories was taken (max 200)
* The number of Commits for each user was taken from the user’s first 10 repositories
* User’s followers were recorded (maximum 200)

**Interactive Graphs of User productivity (measured in Commits, Branches and Repositories) VS. performance (measured in number of Followers):**

****

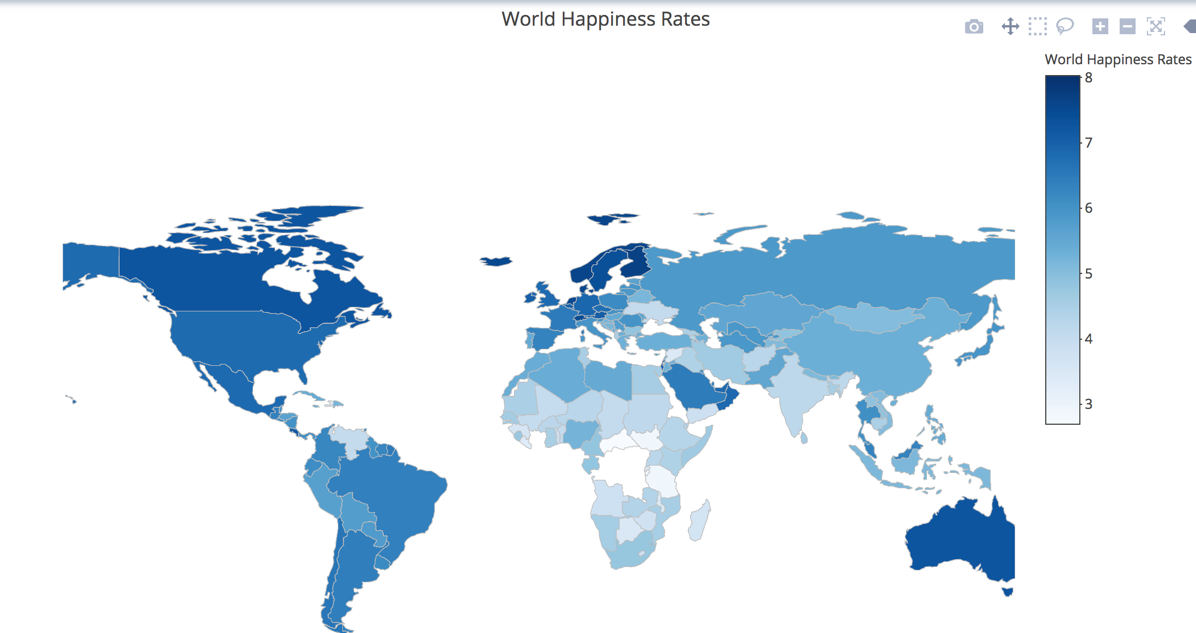
\***Run the R code to see the interactivity, using rchart library**

**Result:**

* There is a correlation between User Activity (Productivity) and the number of followers they have.
* There is no clear correlation between the productivity of a user and the productivity of their organization. ‘Ministrycentred’ can be noted to be productive from Graph 1. However, its members are not noticeably productive.

**Graph 3: Can you measure a developer’s productivity by attempting to quantify their happiness?**

* The argument has been made that one can measure the software engineering process by attempting to quantify a developer’s happiness as happiness has been shown to have a direct correlation with worker productivity.
* Below is a screenshot of a Chloropleth map of world happiness. This is the link to the map on my plotly account <https://plot.ly/~beltonn/1>.

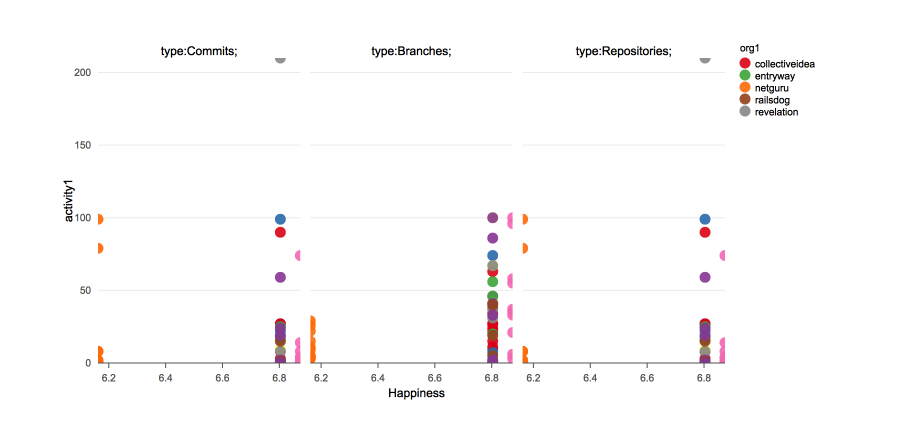


\*Interactive map available at <https://plot.ly/~beltonn/1>

**Graph 4: Productivity Vs. Happiness**

* Taking productivity Vs. Happiness
  + Using number of Commits, Branches and Repositories to measure productivity
  + Using World Happiness Scores (Dataset uploaded also as ‘d3-world happiness’. Assigning each user a happiness score based on the location of their organisation used in the data.
* The number of Branches for each user was taken from the user’s first 10 repositories
* The number of user repositories was taken (max 200)
* The number of Commits for each user was taken from the user’s first 10 repositories

**Interactive graph of User productivity (measured in Commits, Branches and Repositories) Vs. Happiness:**

****

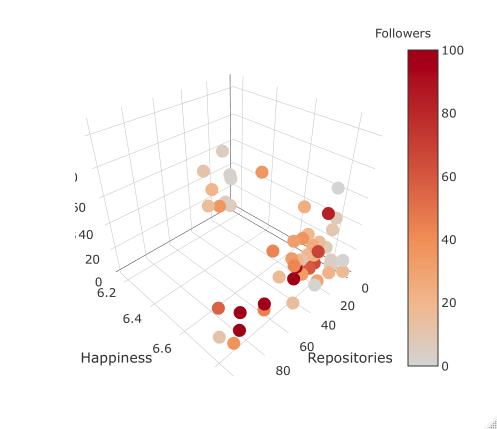
**\*Run the R code to see the interactivity, rchart library**

**Result:**

* A slight correlation can be seen when measuring activity types ‘Commits’ and ‘Branches’. This means if a user’s country scores higher on the happiness scale, developer’s would be more productive. Hence, this could be used to measure the software engineering process.

**Graph 4, Conclusion:**

Considering the 3 main aspects; Productivity ~ Performance ~ Happiness. The following graph has the number of Repositories on the x axis and the number of Branches on the z axis (both measures of productivity), Happiness on the y axis. The number of followers of each user are different colours depending on the amount of followers they have. This is an attempt to visualize all 3 aspects mentioned earlier.



**Results:**

* As expected there is a cluster of users in the origin that have small number of repositories, branches (low productivity) and a low happiness rating. Hence, they are low on the colour scale showing that low Productivity and low Happiness ratings can be linked to low performance (number of followers).
* Also, it can be seen that users with a high happiness score and high number of repositories have a higher amount of followers.