Are there more GitHub R repositories created in Feburary then January?

-Shikha Dangi -Illinois State University

05 December 16

Introduction

GitHub is an open source code hosting service. Founded in April 2008, GitHub offers all of the distributed revision control and source code management functionality of git as well as adding its own features.It provides a web based graphical interface and desktop as well as mobile integration.

GitHub is a place where developers from around the world come together and work on the same project. It is mostly used for sharing source code, but is not limited to it. It support various formats and features such as documentation, pull request, commit, issue labels, milestones etc.

Having said that the purpose of this project is to determine the number of R repositories created on github in the month of January-2016 and Feburary-2016. R is a programming language for statistical computing and graphics. It is an open source, platform independent language which stores data externally in plain text format. The very reason for its popularity among researchers. Moreover, it is easy to reproduce or update data and R language can be integrated with other languages such as C/C++, python, and Java.

Obtaining the Data

To obtain the required data I have used GitHub API. It is standard for accessing github data which returns JSON(JavaScript Object Notation). To access GitHub API in R studio we have installed packae "JSONLITE" using install.package and then loaded the library in R. After which the data was obtained by passing the GitHub API URL along with search information.

```
install.packages("jsonlite")
install.packages("ggplot2")
install.packages("plyr")
library(jsonlite)
library (plyr)
library (ggplot2)
url  <- 'https://api.github.com/'
# the GitHub API
path <- 'search/repositories'
# Repositories path
search1 <- '?q=created%3A%222016-01-01+..+2016-01-31%22'
# Created in January of 2016
search2 <- '?q=created%3A%222016-02-01+..+2016-02-29%22'
# Created in February of 2016
searchR <- '+language:r'
# And language = R
pageNo <- '&page=1'
# We can specify a page number
pageSize <- '&per_page=100'
# 100 results per page is the max
```

```
URLJan <-paste0(url, path, search1, searchR, pageNo, pageSize)
#Here is what URLJan looks like all together
#URL <-'URL <- 'https://api.github.com/search/repositories
#?q=created%3A%222016-01-01+..+2016-01-31%22+language:r&page=1&per_page=1'
URLFeb <-paste0(url, path, search2, searchR, pageNo, pageSize)
#Here is what URLFeb looks like all together
#URL <-'URL <- 'https://api.github.com/search/repositories
#?q=created%3A%222016-02-01+..+2016-02-29%22+language:r&page=1&per_page=1'
# The above URL will create a page in json format
```

As the json file here is nested data we are using the "jsonlite" package to read data.If the data set consists of only one table, then we would have used "RJSONIO" package.

```
# Read the json and convert it to a list using the jsonlite package
j <- jsonlite::fromJSON(URLJan)
# The above URLFeb will create a page in json format
# Read the json and convert it to a list using the jsonlite package
f <- jsonlite::fromJSON(URLFeb)
```

Exploring Data

After converting data into a list we are exploring the data to find number of R repositories created each month. Once we get the count we create a dataframe by merging the two columns months and repository

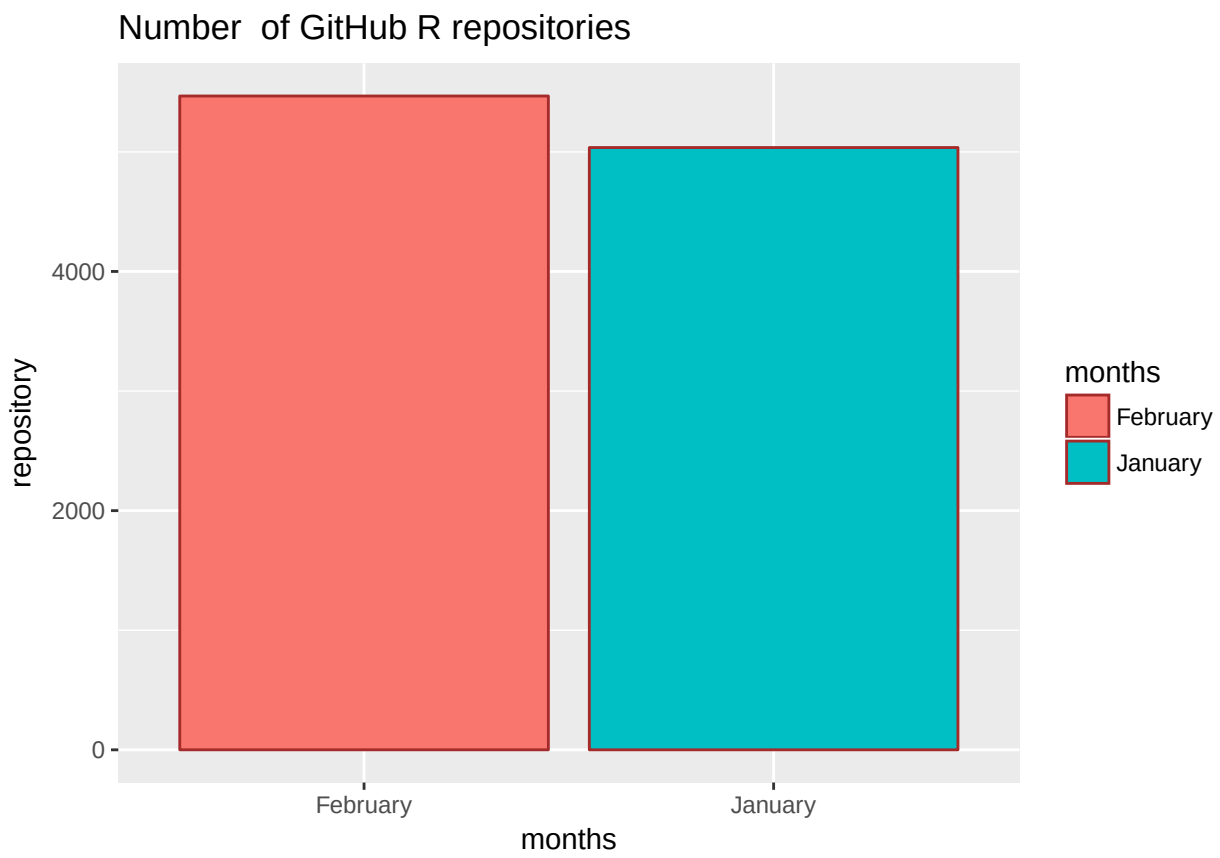
```
# Get number of Jan Repositories by capturing the total count
repoJan <-j$total_count
# Get the number of Feb Repositories by capturing the total count
repoFeb <-f$total_count
# Dataframe will have 2 columns -- months and repository
months <-c("January", "February")
repository <-c(repoJan, repoFeb)
#We use the data.frame function to bind the 2 columns together and create git
git<-data.frame(months, repository)
#To get month with More GitHub R repositaries
Highest<- months[which.max(repository)]
Lowest<-months[which.min(repository)]
```

February has more R repositories in GitHub than January.

Graphing Data

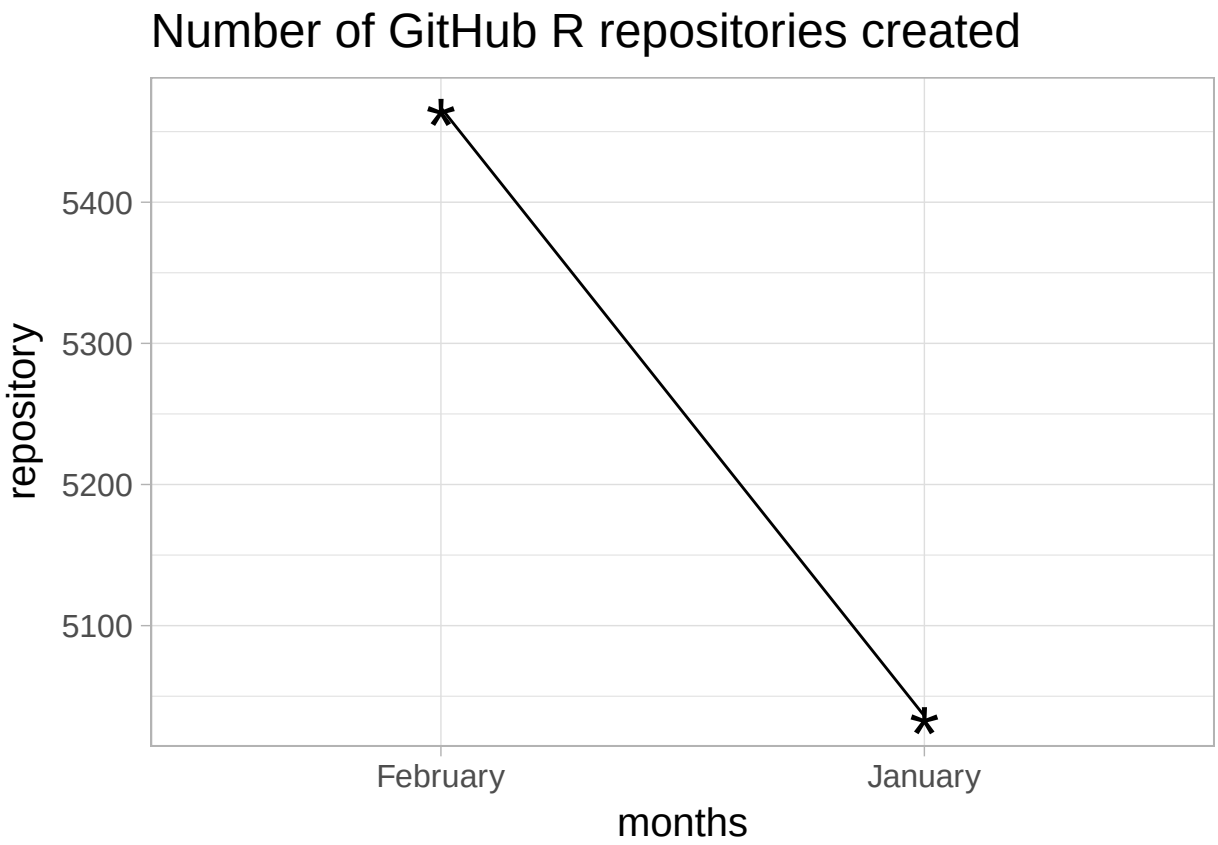
After obtaining all the information required, we will now intrpret this information with the help of graphs.

```
#Finally, we use ggplot2 to create a bar graph
ggplot(git, aes(x=months, y=repository, fill=months)) +
  geom_bar(stat="identity",color="Brown")+ggtitle ("Number  of GitHub R repositories ")
```



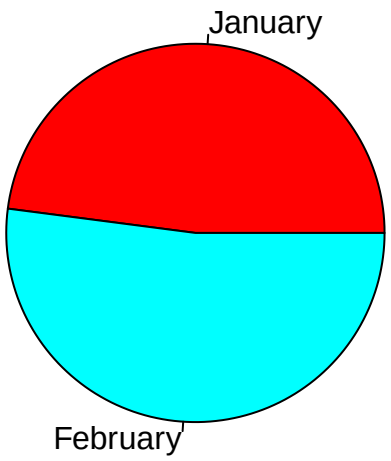
The above bar graph the x-axis represent months and y-axis represent number of repositories.

```
#line graph
ggplot(data=git, aes(x=months, y=repository, group=1)) +
    geom_line()+geom_point(shape=42,size=10)+ggtitle("Number of GitHub R repositories created")+
  theme_light(base_size = 15,base_family = "")
```



The above line graph the x-axis represent months and y-axis represent number of repositories.The graph is used to compare the two values denoted by *.

```
#pie chart
pie(repository, labels = months,col=rainbow(length(months)), main="Pie Chart of Months")
```



Summary

```
#summary() provides summary of data like min, max, mean, median.
Summary <- summary(git)
```

```
#Str() provides great info about the structure of object.
str(git)
```

```
## 'data.frame':    2 obs. of  2 variables:
##  $ months    : Factor w/ 2 levels "February","January": 2 1
##  $ repository: int  5036 5467
```

```
#class() provides info about what is the data type
#stored like integer, number, data frame etc.
gitclass <- class(git)
monthclass <- class(months)
reposclass <- class(repository)
```

```
#view() is used to see the table in grid
git
```

```
##     months repository
## 1  January       5036
## 2 February       5467
```