Data Scientist's Toolbox

Nicola Davide D'Avanzo

12/12/2014

This document is a generic overview about the data scientist's skills and attitudes.

What do data scientists do?

* Define the **questions of interest**
* Define the **ideal data set**
* Determine what **data you can access**
* **Obtain** the data
* **Clean** the data
* Exploratory **data analysis**
* **Statistical prediction/modeling**
* **Interpret** results
* **Challenge** results
* **Synthesize/write up** results
* Create **reproducible code**
* **Distribute** results to other people

Data Scientists tools:

* R (statistical programming environment)
* GitHub (Git repository web-based hosting service)
* Terminal Linux

R questions:

* What steps will reproduce the problem?
* What is the expected output?
* What do you see instead?
* What version of the product are you using?
* What operating system?

Data analysis questions:

* What is the question you are trying to answer?
* What steps/tools did you use to answer it?
* What did you expect to see?
* What do you see instead?
* What other solutions have you thought about?

Data analysis files:

* Data (Raw Data, Processed Data)
* Figures (Explorator figures, Final figures)
* R code (Raw scripts, Final scripts, R markdown files)
* Text (Readme files, Text of analysis)

Command Line Interface (CLI):

* **Navigate** folders
* **Create** file, folders and programs
* **Edit** file, folders and programs
* **Run** computer programs

CLI commands: **command** **flags** **arguments**

* pwd
* clear
* ls -al
* cd
* mkdir
* touch
* cp -r
* rm -r
* mv new\_file renamed\_filed
* echo
* date

Git: open-source version control system

* most popular
* **local** repository
* command line
* git config - -global user.name " your\_user\_name "
* git config - -global user.email " [your\_email@example.com](mailto:your_email@example.com) "
* git config - -list

GitHub: web-based hosting service for software development project that use the Git revision control system

* **Remote** repository (on the web)
* Homepage repository **display**
* **Backup**
* **Follow** (access) and **share**

Creating GitHub repository:

* from **Scratch**: " create a new repo "
* **Local copy**:

git init

git remote add origin <https://www.github.com/YourUsernameHere/test_repo.git>

* **Fork** another user's repository: "Fork"
* **Clone the repo**:

git clone <https://www.github.com/YourUsernameHere/RepoNameHere.git>

Pushing and Pulling on GitHub:

* git add. (add all files to track on local repository)
* git add -u (update file to track on local repository)
* git add -A (both previous operations)
* git commit -m "massage" (commit index)
* git push (load files on remote repository)
* git checkout -b branchname (create a branch)
* git branch (to see what branch you are on type)
* git checkout master (to switch back to the master branch type)