

Finally, it's time to get started! Please complete the following for next week (I recommend you don't save this for last minute):

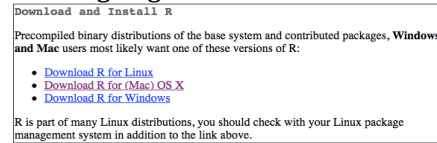
1. Send us a photo of yourself and a quick 3-sentence blurb, one sentence is your education (major, etc.), one is your goal in AQM, the last is a fun fact. Please use the following filename convention and format: **last_first.png**, **last_first.doc**. Send it to us [here \(https://dropitto.me/AQM_pic_and_blurb\)](https://dropitto.me/AQM_pic_and_blurb) – password is **BrownianMotion**. This will be displayed on our public website, provided to our industry partners, and academic supporters.
2. Accept the AQMchat Slack invite and send us a message indicating you have it up and running! You may be required to sign-up. You can begin making groups and discuss whatever you want until you're hearts content. **Email will no longer be used come next class, so get familiar with Slack!** We recommend downloading the application so you can be notified easily. If you are resistant to that and would prefer the web-version, just make sure you have desktop notifications turned on. Adding a photo would be nice, so we know who you are and get more comfortable knowing each other 😊
3. Get R installed and ready to go! If you have not done this before, please follow along with the detailed instructions below.
4. Set up your Github account and install git. Remember that there's a difference for Windows and OSX/Linux users, so follow the instructions below and be aware of your platform. **Send us your Github account name on AQMchat on Slack so we can add you into the new private AQM organization repo.**
5. Connect to your Github account via HTTPS/SSH (SSH recommended). We cover this in more detailed in the instructions following this page.
6. Do the assignment on the last page! I know we said no assignments, but this is an exception.

Objective is for you to get familiar with git and github through connecting, committing, pushing, and pulling. This is a vital tool, so please take your time if you aren't familiar already. We will focus on programming next week.

If you have any issues, do your best to figure them out yourself. Remember, you can also ask your team on AQMchat on Slack anything you'd like, and I'm sure they'd be able to help out. We will go over the command interface in more detail next week.

1. **Install R** [here](http://cran.stat.sfu.ca) (<http://cran.stat.sfu.ca>)

According to your computer system, choose downloading R from one of these three: Linux, (Mac) OS X, Windows. R is the *language* itself.



2. **Install Rstudio** [here](http://www.rstudio.com/products/rstudio/download/) (<http://www.rstudio.com/products/rstudio/download/>)

Choose one of the Installers for Supported Platforms based on the computer system you use. Rstudio is the Integrated development environment (IDE), which only makes programming in R easier – it is not the language itself.

Installers for Supported Platforms

Installers
RStudio 0.99.441 - Windows XP/Vista/7/8
RStudio 0.99.441 - Mac OS X 10.6+ (64-bit)
RStudio 0.99.441 - Ubuntu 12.04+/Debian 8+ (32-bit)
RStudio 0.99.441 - Ubuntu 12.04+/Debian 8+ (64-bit)
RStudio 0.99.441 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (32-bit)
RStudio 0.99.441 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (64-bit)

(For Linux user, please download the installer for Linux server by following the instructions online.)

3. **Install Git: OSX/Linux** [here](http://git-scm.com) (<http://git-scm.com>), Windows users please use **this one:** [here](https://git-for-windows.github.io/) (<https://git-for-windows.github.io/>)

choose Downloads→select your computer system→download should be started

4. **Set-up Github account** [here](https://github.com) (<https://github.com>)

finish three steps to sign up

5. **Get Github free student account** [here](https://education.github.com) (<https://education.github.com>)

sign in with your Github account→request a discount

To upload the identification as a student→get your UBC enrolment letter: (go to SSC→Grades & Records→proof of enrolment letter→download PDF letter)

Setup Git/Github through the command line (recommended and the way we will handle most things, so try to get familiar with it now if you haven't):

Follow this helpful link and follow the links accordingly:

<https://help.github.com/articles/set-up-git/>

If you are a Windows user, use the command interface provided in the git download you already downloaded above via <https://git-for-windows.github.io/>.

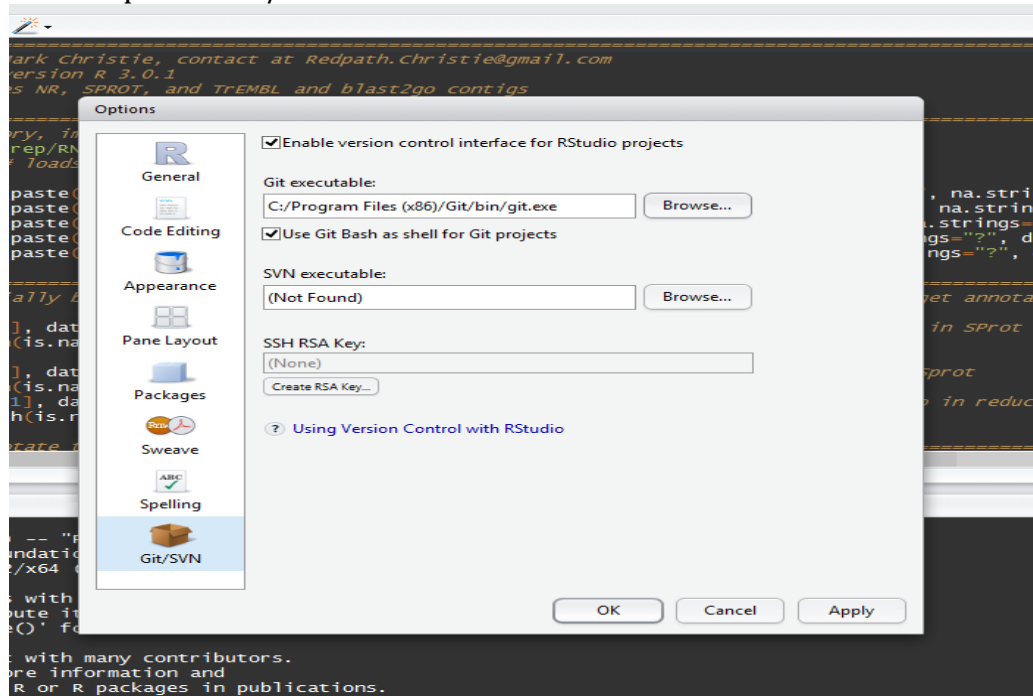
Follow the [git tutorial](#) on how to use git to commit, push and pull using the command line. OSX/Linux users use Terminal. Windows users use the command interface that you already downloaded [here](#).

Setup Git/Github through Rstudio (alternative for the time being if you are having trouble with the above method):

6. → Configure Git with global commands, so Github knows you and your computer. Open up the bash version of Git and type the following:

```
git config --global user.email "GitHub account name"
git config --global user.name "GitHubEmail@something.com"
```

→ Open Rstudio and set the path to Git executable. Go to Tools>Options>Git/SVN



Make sure you know where your git.exe file is located.

7. Pushing to Github with ssh-authentication:

If RStudio prompts you for a username and password every time you try to push your project to Github, open the shell (Git menu: More/Shel...) and do the following:

- Set username and email(if you did not do that before)
To check with Rstudio:

```
git config --global user.name "your_username"
```

```
git config --global user.email "your_email@example.com"
```

- Create SSH key: go to menu Tools / Global options / Git SVN / View public key and copy the key to your Github account setting (Edit profile / SSH keys / Add SSH key).

To check with that ssh-authentication works, try to run

```
ssh -T git@github.com
```

and you should get something like:

```
Hi your_username! You've successfully  
authenticated, but GitHub does not provide shell  
access.
```

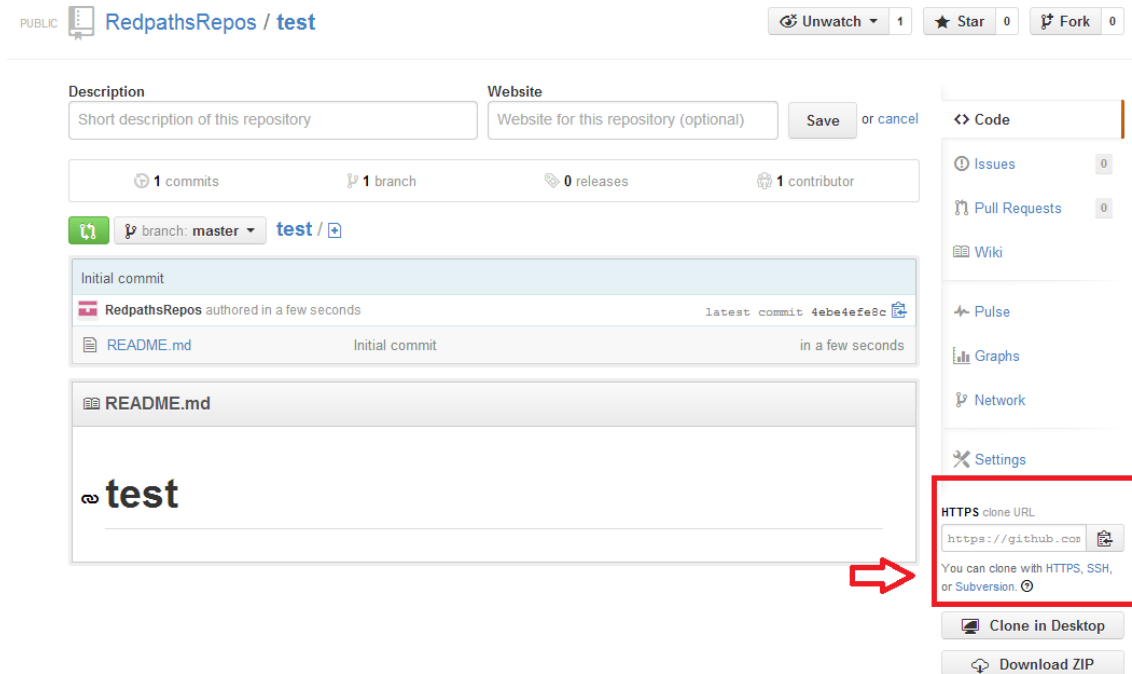
- Change remote.origin.url from HTTPS to HTTP
It might be Windows specific, but after 1)+2) RStudio still asks me for user name and password. After a long Google search, I have found a solution [here](http://stackoverflow.com/questions/1595848/configuring-git-over-ssh-to-login-once) (<http://stackoverflow.com/questions/1595848/configuring-git-over-ssh-to-login-once>) and that is

```
git config remote.origin.url  
git@github.com:your_username/your_project.git
```

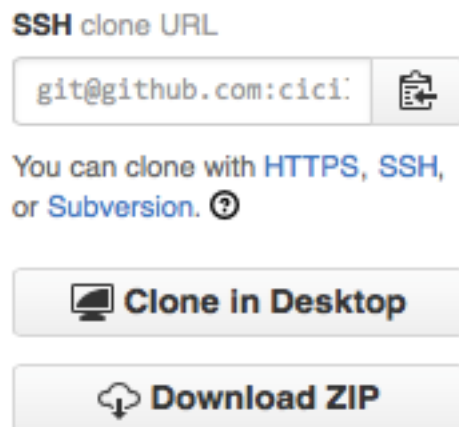
8. Connect R Studio with Github's repositories:

Once you have R studio and Github, we are going to connect these two. So you can upload your file to GitHub each time you create a new project and make some changes.

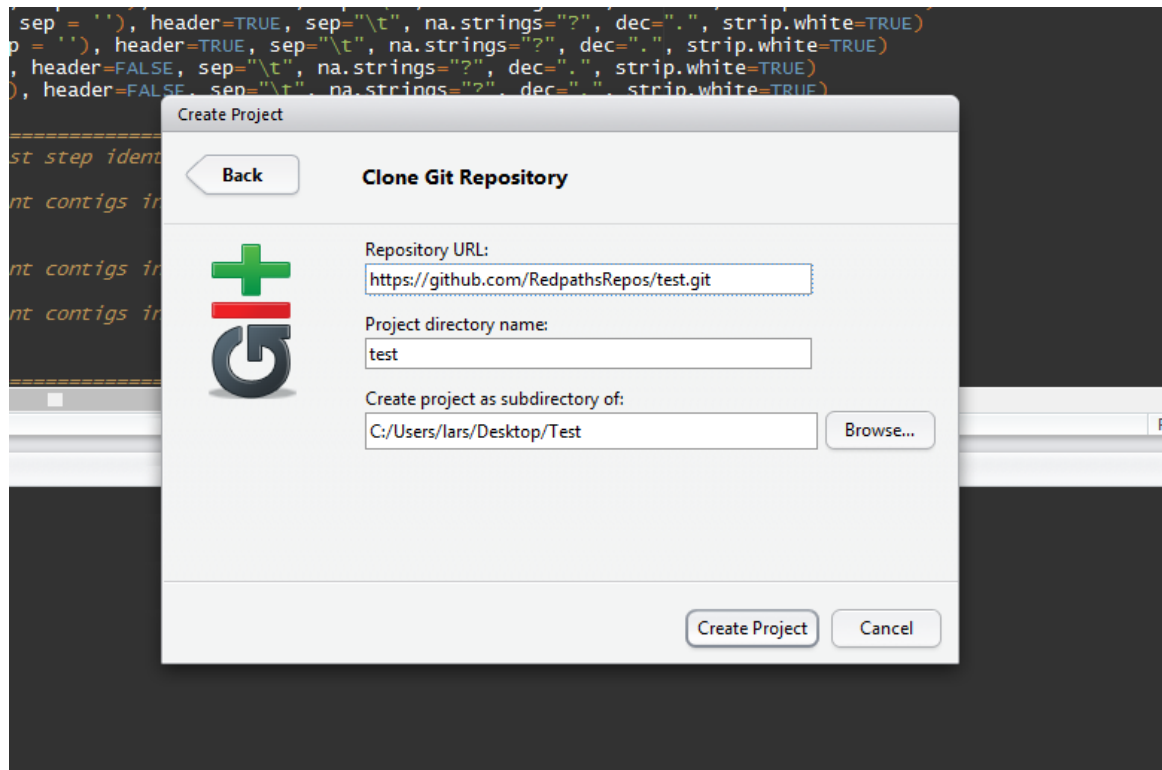
→log into your GitHub account and create a new repository. Make sure you check the box 'Initialize this repository with a README.' When you are done, you should be able to view the Repository like below:



*Notice the box highlighted in red. Change HTTPS to SSH as shown below.

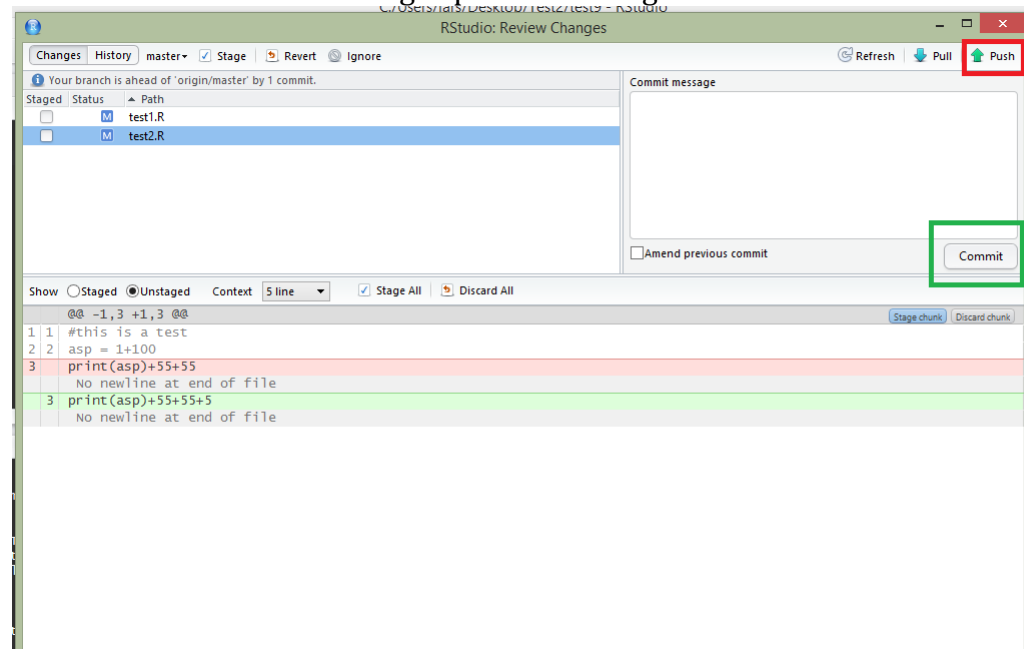


→Now, open Rstudio and go to Project > Create Project > Version Control > Git and you should see a screen like below:



→ In the Repository URL box, you should copy and paste the URL indicated in the 'red box' above.

→ The next step is to 'commit' your work. To do this go to Tools > Version Control > Commit. This brings up the following window:



→ Now I simply check the files that I want to commit and press the 'commit' button, highlighted with the green box.

→ If I want to also move these files onto the GitHub servers, I will click on the red box, marked **'push'**.

Assignment – Create a Markdown Profile for us to visit:

For more details, follow [Week #1 from last year's AQM](#). For samples of last year, please visit our public repos at <https://github.com/AQM-2014>. Your objective is to create a nice landing page on your personal repo like the examples from last year.

Definition: **R Markdown** is an authoring format that enables easy creation of dynamic documents, presentations, and reports from R. It combines the core syntax of markdown (an easy-to-write plain text format) with embedded R code chunks that are run so their output can be included in the final document.

→ Open Rstudio and create a new markdown project on R (top left with blank paper with green plus icon). You may be prompted to install packaged – that's fine.

→ Once you save the file, you have an R markdown file ending with **.rmd**

→ You also want to change it to html format. By doing this, press “Knit to html” and you will have two files with same content but in different format (rmd. and html)

→ However, R automatically deletes the original **.md** markdown file after you change the file to html format. We want to keep all of the three files (rmd, md, html), to do this, write the code in the “[YAML](#) front matter” highlighted below:

```
1 ---
2   title:"Assignment 1"
3   author:"first name_last name"
4   date:"xxx, xxx"
5   output:
6     html_document:
7       keep_md:true
```

Follow the [git tutorial](#) on how to use git to commit, push and pull using the command line. OSX/Linux users use Terminal. Windows users use the command interface that you already downloaded [here](#). Otherwise, follow the Rstudio method above.

→ Commit .rmd, .md and .html files. Commit essentially saves a snapshot of your work in the folder at that point in time locally, and puts its on the launch pad for the next step.

→ Push rmd, md and html files to upload unto GitHub. Push pushes your commits to the repo specified

→ You will have the file in three formats(md, rmd, html)

- * We recommend to have all three files saved on Github. They will come in handy later on.
- * To clarify the **differences among .rmd, html, .md**: rmd is a symbol of R Markdown which is an authoring format that enables easy creation from R. Compared to html that only read code chunks, R markdown combines R code with the core syntax of markdown (an easy-to-write plain text format); md and rmd are the format that R can read, but .md is another format which occurs in Github.