**Sergio Andrés García Bernal**

**Causal Inference and Research Design**

**Assignment 2 – Empirical workflow**

**Due date: Wednesday, June 10th, 2020**

1. **Summarize briefly the point of chapters 2- 8 in less than one page.**

Gentzkow and Shapiro want to show us that there is a solution for the problems that researchers have suffered during the years in the process of handle with code and data.

They gave some rules or elements to take always into account when you are going to carry out an investigation. If you implement them you can minimize the possibilities of making silly mistakes, it’s evident that today there is a lot of specialized softwares to do tasks that we are used to do manually, so here we have an excellent idea, less mistakes and in less time.

There are 8 principles mentioned in this text (automation, version control, directories, keys, abstraction, documentation and management) that will be explained in detail soon. The principal idea is to make the life of the researcher easier, I think this is a lecture of advices that if you implement them with a good level of organization you will never fail.

They use a very interesting example thorough all the 8 chapters, they want to test the hypothesis that the introduction of television to the US increased sales of potato chips. Thereby, they just don´t give advice for getting everything more clear for the researcher, the purpose is also to make easier for the collaborators, research assistant, coauthors or anybody who could potentially read your information, you have to understand yourself but the rest of the people too.

1. **Why do Gentzkow and Shapiro think these elements of modern empirical work are so important? What problems does each element solve?**

* **Automation:** With this technique we can avoid the “interactive mode” of repeating every step so many times. Costs tend to be lower; we can replicate our directories and it’s more efficient.
* **Version Control:** It’s a tool for forgetting the date and initial method, the researcher can have a bigger control of the versions that have been done of the code without the need of creating a new document for each change. It’s possible to “undo” any version that you don’t want in the word processing.
* **Directories:** We have to separate directories by function because if we have just one for everything it will be difficult when we want to change some specific thing or if we have multiple projects that use the same data file and we could skip the tedious process of doing it all again.
* **Keys:** When you are going to use some data you have to be sure you can understand it, that’s why you can not have missing keys and the best way of storing your data is when you keep it normalized.
* **Abstraction:** Defined as the action of turning the specific instances of something into a general-purpose tool is very important because it eliminates redundancy and it makes code more readable. But you must be very careful and use it only when it’s necessary, it has no sense to introduce it if you are going to use that code only once.
* **Documentation:** It’s not a good idea to have excessive documentation (comments, notes, readmes), just add the strictly necessary, sometimes it can be more confusing. The best option is to make your code self-documenting, where the commands, the variables and the labels that you use are so clear that comments are non-essential.
* **Management:** The key aspect in here is the organization, no matter how nor with which software, you just have to be organized and it’s recommendable to have a task management system to have responsibilities clear. When you work alone there is no such a big deal but the most people you work with the most difficult it is to organize jobs.

1. **Give an example of the sort of problem that could arise in the course of an empirical project if someone were to fail to adopt these principles.**
2. **How do you plan to incorporate these solutions into your own work?**
3. **Briefly explain what git and github are used for, how they are similar and how they are different.**

It´s important to clarify that Git and GitHub are different.

Git is a distributed version control system, the most popular in software development. GitHub is an online hosting platform that provides an array of services built on top of the Git system. So when you use Git you can find other services apart from GitHub, like Bitbucket and GitLab. As I have explained before with these tools you can have a better control of the changes that are made in the documents, codes, data, or anything you need to transform. You can undo changes and you don’t need to create a new version for each person that makes a change or with the date of each modification.

1. **Name a benefit of using git to organize your empirical research. What types of common problems can occur if you don’t use git?**
2. **What about using git is challenging for you for right now? What steps can you take to minimize those challenges such that you can adopt git for this class?**
3. **Name the four main Git operations. What does each operation do and how are is each operation different from one another?**

* **Stage:** Tell Git you want to add changes to the repository history
* **Commit:** To be sure that you want to do these changes
* **Pull:** Get any new changes made on the GitHub repository, it can be from you or from your collaborators. Load all new commits from the repo.
* **Push:** Push any local changes to the GitHub repository. Save all your commits to the repo.