# Prep Notes - Meeting 02

Dylan McDowell, Andrew Benson

April 19, 2018

# 1 Python

## 1.1 Sylabii for CS 101 & 241 Courses

Here is an online CS101 Syllabus. Some of the topics covered in CS101 include:

- Use of Variables
- Mathematical Expressions
- If statements
- For Loops
- While Loops
- Basic Input/Output commands to display (ie `print()`)
- Basic Input/Output to file

Things **Not** covered in CS101 that could potentially be needed for Econ381:

- Use of Pandas (reading in data, manipulating data.frames)
- Use of Numpy (performing mathematical functions on data)
- Use of Matplotlib (plotting data)
- Working in Jupyter Labs Notebooks

Here is an online CS241 Syllabus. The focus of this course will be to help students develop an understanding of object oriented programming and data structures. These concepts are monumental in the foundation for *Software Development*. Given the broad nature of python and it's fairly recent adoption in the data analysis community. We should ask the question, **"What exactly should the students learn to maximize the utility gained from learning a language like Python?"**

Surely, knowing object oriented programming and data-structures could aid in developing an understanding of working with data, but given the current exercises and what student will students will be expected to know, there is a disconnect. This doesn't mean that the exercises are pointless in teaching python but having a correct vision of the goals of the course will be paramount.

*Reference examples from Chapter 1 Exercise 4*

## 1.2 Proposed Homework Work Flow

- Students will begin the semester by completing the Data Camp Course **"Intro to Python for Data Science"**. This should get students up to speed on the tools that will be needed to complete the assignments for the class. *Free Data Camp accounts can be setup for students*
- Students will download the latest version of Anaconda. This comes with everything the students will need to complete the assignments
- Give students .pynb files for the assignments and have them submit .html files on the due date.

## 1.3 Pros & Cons

| **Pros** | **Cons** |
|---|---|
| Broad language used across many industries | Early stages of data analytics (cleaning, plotting, etc...) |
| Powerful scripting and data manipulation libraries | Paradigm based more in software development |
| High ROI regarding compensation[1] | Data analysis libraries inherently assume OOP experience |
| Gaining popularity in academics | Lacking a strong IDE for data focused projects |
| Overall useful language | Broad language meaning, focused learning is difficult |

Table 1: Pros and Cons of Python

[1] StackOverflow Developer Survey

# 2 R and R Studio

## 2.1 Comments on R

R is great for many applications. Because it is open source, many people have built up tools around it to get almost any task done. R is great at modeling and then making data into charts and visualizations. In addition, because it has so many tools, this would allow students to focus more on applications of economics rather than focusing on learning a foreign skill from the ground up.

## 2.2 Proposed Homework Work Flow

- Students will complete a tutorial that will cover the basics in R. The assignment will be designed using an interactive document that allows students to code along in the assignment itself made possible with a package called *learnr*.[2] This will be accompanied by doing an additional tutorial called *swirl*. Alternatively, one or two data camp courses could be taken titled **"Intro to R"** or **"Intermediate R"**. [3]

- Students will download both R and R Studio. All of the assignments will be completed in an R markdown document though the R studio platform.

- A LaTeX, .HTML, or Rmarkdown document will be pre-designed where students will then fill in the questions with code to complete the assignment.

## 2.3 Pros & Cons

| Pros | Cons |
|---|---|
| R Markdown Environment<br>Superior packages for graphics<br>Data Wrangling driven by the Tidyverse paradigm<br>Blends well with statistics classes (Econometrics, Interm. Statistics (Math 325), and Data Wrangling and Visualization (Math 335) [4]<br>Can be utilized in the same capacity as Python | Does not teach basics of programming as well as Python<br>Debugging can be harder than other languages<br>More of a niche language<br>Relies heavily on Tidyverse and Rstudio |

Table 2: Pros and Cons of R

## 3 Conclusion

Before moving forward with the rest of the assignments we'd like to address some of the aforementioned pros and cons, and nail down a vision for what exactly the students should walk away from the class knowing. Although Python is becoming more data science driven, Python is still a general purpose programming language. It was designed first to build software, and just recently has become adopted into the arena of data munging. It's usefulness is in process automation and prototyping.

Given the requirements of the first assignment, we fear that this course will require students to learn tools that are not taught in CS101 or CS241, albeit useful tools to learn indeed. That isn't to say that Python is a lost cause for this course, rather, a clear curriculum will be required that might mix CS101, CS241, and topics outside those courses so that students can maximize the utility Python provides.

On the flip side, we fear that R may not effectively teach the basics of programming. R is unlike many programming languages given that it was built by statisticians and not programmers. This means that it's very strong in data-cleaning, visualization, and statistical tests but falters in general purpose programming and wide scale applications.

Regardless of the language chosen as a best fit for the course. Students should know why they are learning a different language at all. When I (*Dylan*) was at Goldman Sachs, I was in the minority of people advocating for more automation using Python despite having a script that completed what normally took an hour to do.

[2] Library learnr
[3] R Data Camp
[4] Math 335 Syllabus