**R or Python: A Programmer’s Response**

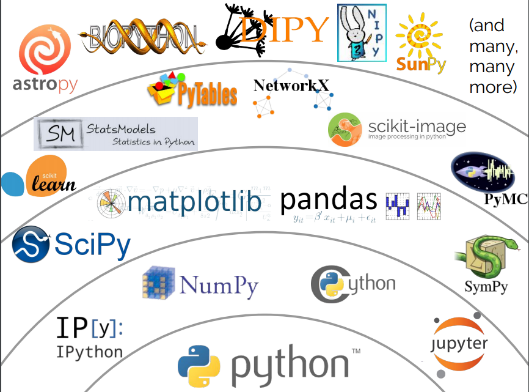
**Abstract/ Instruction:**

We hear lots of requests on porting R to Clinical Computing Platform. Before R really touches the ground, I’d like to share some thoughts on why R is a not a good choice, considering there is a strong alternative, Python.

The requests on R are mostly from statisticians inside clinical development units. Statisticians are not programmers; they might prefer R for some statistical tasks, like sample size estimate and visualization. But for heavy users for the Clinical Computing Platform, aka, clinical/statistical programmers, they need good language(s) to import data, clean data, transform data and do analysis and reporting. In this type of programming, I’d argue Python is far way better than R, from a programmer’s perspective.

First, Python is simply a better language, created by real computer scientists. R was mainly created by statisticians and for statisticians. There are lots of design drawbacks in R core language.

Second, for programming tasks mentioned above, Python offers more consistent and unified packages stack (Numpy, Pandas, Scikit-learn, etc), while in R, the packages are scattered. It’s true that there is an attempt to gather several core R packages to unify the process, but at least at this stage, it’s not satisfactory.



Third, Python is far way better at system integration, with current components residing in Clinical Computing Platform.

Fourth, considering Python’s more elegant syntax and unified ecosystem, I’d sure actually Python is easier to learn for SAS programmers, although it might not be well known in clinical programming world. R, instead, consists of lot of black tricks by statisticians.

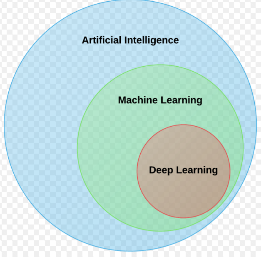
Last but not least, Python will make a happier programmer. Python will introduce SAS programmers to a more wider world of machine learning, deep learning which all cool kids are talking about…

The Marriage of Machine Learning and Clinical Research: A Proposal

**Abstract/ Instruction:**

The definitions we need to get the discussion started:

Artificial Intelligence (AI) is kind of intelligence learned by machines. Machine Learning is a technique within AI to give computers the ability to learn without explicit programming and handcraft rules. Deep Learning is a special technique of machine learning which utilizes neutral networks providing the most powerful machine learning technique in real life applications.



Machine learning, especially deep learning, is very successful in fields like computer vision. In the broader medical field, a very first successful application is within medical image diagnosing, which is very close to general image classification in computer vision. But on the clinical research side, the power of machine learning has not yet been realized.

Within clinical research, we handcraft lots of rules for various tasks. Considering the large number of manual rules and associated code we develop, it’s the perfect situation to think about the possibility of using machine learning techniques for the tasks without explicit programming. So, how can we get started?

As an example, I will demonstrate on how to use machine learning to perform clinical data de-identification.

1. What’s the problem?

To support the need for data transparency and to ensure patient privacy, all clinical data required for public use should be anonymized.

1. What’s the current approach to this problem?

Develop and write code for handcrafted rules. For example, we might shuffle the values of the variable “sex”

1. Can we translate it to machine learning problem?

A deep learning technique called Generative Adversarial Nets(GAN) is widely used for image replication. For example, with real dog images as inputs, GAN can learn though the images and then create images which look like dogs. Here is the idea to apply it to the de-identification problem:

We have original dataset (name it A) to be anonymized. At very beginning, we use GAN to generate fake dataset B, the B is sent to A for checking. If B doesn’t look like A, the model adjusts its parameters and creates a new B which will sent to A for check again and the process goes on, until B does look like A. So, the final B could serve as the final de-identified data for public use.

1. Do we have data?

Yes in this field, there are lots of data!