# Initial Posts

|  |
| --- |
| **ThinkStats…my way**  I actually really do like the book in this course and it has explained concepts that are new to me in a simple straightforward way. On the other hand though, I feel the author is mainly self-promoting his functions and doesn't go in-depth really about how the functions are created, what they're doing, and the underlying 'well-known' and widely used modules which his modules use. I find myself reading a concept in the book, reviewing the code behind his custom functions, and then trying to re-create them using the modules which I'll actually use in the future. I mean, it's awesome yeah, you created some cool functions, but I want to learn how to calculate things on my own. Ok, venting done 🙂. |
| **Project Ideas – Autism Spectrum Disorder**  As mentioned previously, I intend on doing EDA on Pediatric Autism Spectrum Disorder (ASD). I actually gave up trying to find individual level detail. Yes, I know, I wasn't expecting to find information that would identify a person which would be a HIPAA violation but was hoping to get individual level information to perform analysis on. Instead, what I was able to get was state level data. So instead of predicting ASD rate for a particular individual, I'll be predicting ASD for a state based on predictor variables also by state. Also, it was tough to pull data for specific years because surveys sometimes do not occur yearly but rather every 2 years, 5 years, 10 years, etc. Below are the variables I'm using.  Outcome Variable:   * Pediatric ASD rate for state. Testing data for ASD rate by state for children aged 3-17 for years 2019-2020. This would be for births between 2002-2017.   Predictor Variables:   * Estimated vaccination rate for children born 2014-2017 * Father age >= 40 rate by state for births in 2016 * Mother age >= 40 rate by state for births in 2016 * Premature birth rate by state for births in 2016 * BMI rate by state for year 2016 * Binge drinking rate by state for year 2015 |
| **Best Practices for Seeding in NumPy**  I saw some code online of someone using an interesting way to seed data that I've never seen before which made me do a little digging. What I found is that this peculiar way of seeding was actually NumPy's recommended best practice for seeding data. Historically I've been using numpy.random.seed() to seed my data and from the looks of it, a lot of others do the same. This method changes the global seed for the default RNG created when NumPy is imported. The bad thing about using this method is that it is prone to other modules and methods altering it. To avoid this, the best practice is to instead instantiate your own RNG then reference that throughout your code. Having your own custom RNG as a variable is less susceptible to unexpected alterations of the seed by other methods and modules.  [Good practices with numpy random number generators - Albert Thomas (albertcthomas.github.io)](https://albertcthomas.github.io/good-practices-random-number-generators/) |
| **Minimum Observations for Predictive Modeling**  For my project, as mentioned I'll be doing EDA on pediatric Autism Spectrum Disorder (ASD). I was not able to find data at the patient level but was able to find a lot of data pertaining to a state. I've compiled a data set that aggregates multiple years together since for some data, it is based on surveys that may be done yearly, every 2 years, 5 years, 10 years, etc. Because of this, the data set that I have is 50 observations, 1 for each state. I'm curious about the class's thoughts on whether this is too little observations. I've done some digging and there are various rules but there doesn't seem to be 1 rule for all scenarios. Like everything else, when determining if something is significant or insignificant in Data Science a lot of folks' answers seem to be, "It depends". |
| **Data Sources**  Just wanted to share some of the sources I've found so far regarding my project on pediatric Autism Spectrum Disorder (ASD). Maybe y'all can use them too hopefully or can use them in the future for another project. Has anyone else found any good public data sources? If so I would love to hear about them.   * [www.childhealthdata.org](https://www.childhealthdata.org) * <https://data.census.gov> * [CDC WONDER](https://wonder.cdc.gov/) |

# Replies

|  |
| --- |
| Very cool topic Saima. My company along with many others are actually currently having to deal with an issue related to this. Our time tracking software that we use, Kronos, was actually hit with a ransomware attack earlier this month and it has been a mess with payroll ever since. It would be awesome if there were some sort of indication that would maybe help predict an attack before it occurred. Have you been able to find any good data yet or have some predictor variables in mind?  Here's an article about the ransomware attack we're currently dealing with:  [Kronos hack update: Employers are suing as paycheck delays drag on : NPR](https://www.npr.org/2022/01/15/1072846933/kronos-hack-lawsuits) |
| Good topic Arti. I imagine this will be a topic that you can find a lot of data on as well. If you could get data that tells if the patient has a family history of heart problems I bet that would be a good variable. I used to work in a cardiologist office and patients would say all the time that it ran in their family. |
| Good find Madeleine. There seems to be a ton of public data on this site. Will definitely have to bookmark this one for future use. Thanks for the share! |

|  |
| --- |
| I definitely agree that an iterative methodology to project management for an analytics team is the way to go. From previous organizations I've worked at though, there is usually just one overarching methodology and the other IT teams are forced to abide by it. For example at a previous organization the data team used the same methodology as the application build team which was based around 4 month intervals where all requests from the organization would be compiled, accepted or denied, and completed within the 4 months. At the end of the 4 months, the completed tasks would then be released. This was not ideal for the data team and many customers were angry with this. We ended up creating multiple methodologies within our team and split the team into 2 groups which ended up not working either |
| Hi Robert, yes, that seems like a pretty good approach. Even if I was able to get another years worth of data for each state, that would double my observations giving me 100. I may have to go with that. I've also heard of people bootstrapping a sample if it is somewhat small. I've never done it before but that could be another thing to look into as well. Appreciate the input! |