**Data:** ECLS data, that consists of various behavioral and socio-economic attributes of 3143 unique children, taken in 7 waves of data collection

**Target Attributes:** Attributes that define Cognitive abilities, and Self-efficacy

**Attributes that define Cognitive abilities**: Math, Science, and Reading scores

**Attributes that define Self-efficacy:** ANGRY, LIKRD, ENJRD, LIKMTH, ENJMTH, FLGOOD, SADLON

* Step1:
* Understand the problem statement and enough of domain
* Data Preprocessing steps if necessary
* Omit all the attributes whose type is “Weights”, and whose values are “SUPPRESSED
* From the dataset thus created, find any interesting correlations/patterns and visualize them
* Step 2:
* Subset the data pertaining to each wave to get 7 mini data sets, for children in poverty, and for children not in poverty.
* **The below process is to be done for both categories of children in , and not in poverty, independently**
* Step 3:

For understanding cognitive abilities, the following analyses is to be done on Math, Science, and Reading scores as target variables, for each wave independently

* Apply various feature engineering techniques that can potentially find out important attributes for each wave
* Visualize any interesting patterns
* Get those attributes that are consistently chosen as important ones, from all the above techniques, across all the waves
* Apply regression analyses on the selected important attributes with the above-mentioned targets, for each wave, and save the regression coefficients
* Step 4:
* For each independent significant attribute, compare its regression coefficients of all the waves.
* Visualize and report the result; This enables us to understand how the weights of regression coefficients are changing across the waves
* Step 5:
* For self –efficacy related attributes, data is sparse, and many of those attributes are not present in all the waves of data collection. So, comparative analyses is done of those attributes that are present at least in 3 or more waves.
* In both the cases, for understanding the Self-efficacy, implement rules-based/tree-based algorithms to come up with rules and report insights
* Step 6:
* Organize the entire code ; source, functionize and comment it wherever necessary
* Fine tune all the visualizations/graphs with properly labelled axes, titles, and stuff, so that they readily tell the story you intend to tell