* Discard irrelevant or obviously erroneous data
  + Most of the variable names should be self-explanatory, however data is deeply nested and will require detailed review in order to select the most appropriate data elements
* Complete thorough EDA to identify which variables you can use to complete your analysis
  + Any poorly populated or duplicate variables should be discarded
* What is the **timeline** of the data? Do you see significant peaks and valleys?
  + Do you see any data collection gaps?
  + Do you see any outliers? Remove obvious outliers before plotting the timeline
  + Do you see any spikes? Are these spikes caused by real activities / events?
* What are the most popular **programming languages** on GitHub?
  + Did the trend of most popular programming languages change over time?
* What is the distribution of **licenses** across GitHub repositories?
  + Any certain programming languages that are more likely to be associated with a particular license?
* What can you tell about the most popular and most rapidly growing **repositories**?
  + Is there certain technology that is driving popularity or explosive growth?
  + Are these associated with Big TechLinks to an external site., who are open sourcing the technology?
  + Are there any technological breakthroughs that are driving this brisk adoption?
* Identify what **technologies** are most frequently associated with Data Science or AI projects. Did these technologies change over time?
* What are the **most frequent reasons** for committing into GitHub repositories?
  + Is this new technology development, bug fix, etc.
* Identify the most prolific / influential **Committers**
  + By commit volume
  + Visualize the distribution of these commits
* How unique are the **“subject” and “message”** values?
  + Are they mostly unique? Or are people usually just copy-pasting the same text?
  + You can use LSH to measure uniqueness / similarity
  + Visualize “subject” and “message” duplication across all programming languages
  + Visualize “subject” and “message” duplication for each of the top 5 programming languages
* Presentation guidelines:
  + Executive Summary
  + Methodology and source data overview
  + Conclusions and actionable recommendations
  + Roughly 8-12 pages is reasonable for this kind of project