RiskLens Data Science Candidate Task

This is the Data Science candidate take-home task for candidates interviewing to join the Data Science team at RiskLens in Spokane, WA.

The purpose of this task is to allow the candidate to explore some industry-relevant cybersecurity incident data, and allow for the hiring manager to see a sample of the candidate's work using data that the candidate has probably not seen before.

Instructions

See the data directory for the data set and data description. We are using data from the VERIS Community Database (VCDB) that has been filtered and transformed to be more easily workable in a few hours. The data has been filtered to be only incidents from the healthcare industry.

You should further filter the data to the years 2010-2018 (inclusive of both).

Please create a **reproducible report*** answering the following questions. Note: the Lead Data Scientist at RiskLens *loves* data visualizations :) :

1. Complete an **exploratory analysis** of the data set and answer at least the following questions:
   a. How many total incidents are in the database for each year?
   b. Grouping by `action` and year, what are some trends in the action types that you notice over time? Please note that since this is an open source database, the total number of incidents in a given year is more of a function of community involvement in incident reporting than a good representation of the total number of incidents. Given this fact, for your trend analysis it may be better to look at proportions of actions for each action type in a given year rather than total number of incidents for a given action.
   c. Repeat step `b` but for `actor`.
   d. Repeat step `b` but for `asset`.
   e. Repeat step `b` but for the three `attribute` variables.
   f. Do we see a trend for the proportion of incidents within the US versus outside of the US?
   g. Please feel free to share any other notable findings as you explore the data.
2. **Modeling questions**. Note: The RiskLens Platform uses PERT distributions for users to report minimum, most likely, and maximum values for estimates in FAIR analyses. However, you don't need to restrict yourself to reporting values in this manner if you discover a different distribution for your data.
   a. Let's assume that you work for a large healthcare employer (1001 employees or larger), and you are scoping a risk scenario where you are worried about an *insider threat* (actor: internal) compromising the confidentiality of medical records. Assuming that you *will* have a cybersecurity incident this year, based on the data you have can you come up with a model that will estimate, with 90% confidence, the range of the counts of medical records that will be compromised in such an incident (*minimum* and *maximum*)? Within that 90% confidence interval, what is the *most likely* count of the breached records? You may ignore those employers with *unknown* employee counts for the sake of time. Bonus points if you integrate the year of breach into your model -- are the trends changing?
   b. If you have time: How does your model change if you estimate *total* record count instead of just *medical* records?

3. Optional and bonus: Can you do anything fun and interesting with the text in the `summary` column?

***A reproducible report will allow us to re-run your code with the data set and obtain the same results and figures, given dependencies and relevant instructions. The most popular formats for this type of report are R Markdown notebooks and Jupyter Notebook (aka IPython Notebook).**

Submission Notes

The candidate may either fork this repository and complete the exercise, adding their own reproducible document to their fork of this repository, and send the repository URL to the hiring manager. Alternatively, the candidate may simply `git clone` the repo or download the data and complete the exercise locally, emailing the hiring manager when complete.