Character Recruiting: Take Home Assignment for Sr Data Engineer

Background: At Character Bio, we turn high-dimensional, multi-modal data into potential drug targets for our Science team to pursue. One example workflow we follow looks like a Data or ML Engineer working with our Clinical and Science teams to take a raw data stream (e.g. patient medical images), process and model it, and test the derived signals (called phenotypes) against various genetic signatures (called Polygenis Risk Scores or PRS) that we've curated. And that's just what we'll do in this assignment!

Goal: To construct a robust, scalable data pipeline that takes as input raw demographic and phenotypic data and outputs genetic association results. We will use these association results to prioritize the phenotypic signatures we should develop further.

Data: You have 3 data files as input:

1. demographics.tsv: A TSV file with important information of patient demographics, e.g. age, sex, smoking status, etc.
2. phenotypes.parquet: A parquet file with the raw phenotypes we want to prioritize
3. prs.tsv: A TSV file with the PRS scores that we want to test association with each phenotype from 2)

General instructions:

- Please only use python to answer the questions below.
- Please do not use LLMs to respond to these prompts; any other resource you want to use is fine.
- Don't copy solutions from e.g. Stackoverflow; adapt the code to work with your solution.
- The assignment below is designed to require 3-4h of time. Try to limit your time to that window, and record how long it takes you.
- Don't worry about answering every question; we care more about building a correct, robust solution, than dotting every i.
- We will discuss your solutions and how you can extend it to much larger datasets in a follow up 45min interview.
- Have fun and enjoy!

Specific questions:

1. First, build a data pipeline in python that takes as input the 3 files described above and generates summary stats on each phenotype.
    a. These summary stats should include key information such as the number of patients with data for each phenotype, the missingness rate, the average age of patients with that phenotype, and the mean, median, and standard deviation of the phenotype.
    b. Your pipeline should output this dataset of summary statistics as a parquet file.

    c. Your pipeline should be able to run from the command line.
2. Now extend your data pipeline by adding a step to test for an association (using linear regression) between each phenotype and the PRS scores.
    a. Be sure to correct for the known covariates in the demographics data, particularly age, sex, and smoking status, by including them as additional features in your regression.
    b. Your pipeline should output a dataset in parquet format with one row per phenotype with information on how well the phenotype is associated with the PRS.
3. This pipeline will be critical for our science team, so we want it to be robust. Please write unit tests for the core functions in your pipeline.
    a. We will run pytest from the command line on your pipeline to execute your unit tests.
4. Not only do we want unit tests, but we want to be able to track the changes in the pipeline over time.
    a. Implement a simple semantic versioning scheme for your pipeline code.
    b. Also implement schema definitions (e.g. using a tool like pandera) for your output datasets.
5. Using your pipeline from step 2, which phenotypes are most associated with the PRS scores?
    a. **BONUS:** Produce a visual (PDF/PNG) showing the strength of association with the PRS for every phenotype.
6. **BONUS:** Adapt your pipeline to save the association dataset as a versioned database table, instead of as a parquet file.