Assignment: PSQI cleaning

Context

The Pittsburgh Sleep Quality Assessment asks participants to list specific times when they wake up and fall asleep, and how long different elements of the sleep process take them. Based on those answers, it provides a standardized way to compute meaningful scores and subscores. It is widely administered at research institutions across the United States.

In this case, the administering site committed two errors. First, it allowed participants to write their answers in free-form entry fields without format validation. Second, it allowed Microsoft Excel to affect the data prior to transferring them to the database, causing further distortion.

Problem statement

You are an analyst who needs to accomplish three objectives:

1. Clean the raw PSQI data and use them to compute components 2, 3 and 4.
2. Provide cleaning and processing code reusable for the next batch of PSQI data, and
3. Create an environmental specification that allows easy deployment and re-deployment.

(In the real world, we might do some manual alterations to the dataset. *You should not do that in this assignment.* Assume that while we're trying to correctly clean up and process as much PSQI data as we can, we're getting it much faster than we can manually correct it.)

Please complete the assignment in Python or R. (You can use Jupyter Notebook or RMarkdown to prototype your functionality and/or demonstrate your results, but your deliverables should be plain R/Python files that can be invoked from the command line.)

Assume that your code will be read, used, re-written and built upon by others. Assume that it will be run repeatedly in a larger system.

You will need to make choices about pattern interpretation, record exclusion, and other cases that don't have a single clear right answer. Whatever decisions you make, please document and justify them in a comment accompanying the implementation of this decision.

Input files

- This file, which provides the specifications for your deliverables.
- `PSQI.pdf` has both the questionnaire and the description of the calculation needed for each score and subscore.
- `psqi_dirty.csv` contains the PSQI variables needed to compute the scores.

Deliverables

1. A git repository containing all of the files below. Well-structured commits with clear commit messages will be appreciated.
2. `clean_psqi` should do the following:
    - Process the first two command-line arguments as the names of input CSV file and output CSV file, so that it can be called with `clean_psqi psqi_dirty.csv psqi_clean.csv`, or equivalent. (As a reminder, both R and Python have an `argparse` library, which may make things easier.)
        - *Optional:* If the second argument is not received, it should direct the CSV output to `stdout`. If the first argument is not received, it should read the CSV input from `stdin`.
    - Verify that the input CSV files contains *at least* the following columns: `ID`, `psqi1`, `psqi2`, `psqi3`, `psqi4`, and `psqi5a`. Exit with a non-zero exit code if it does not contain these columns.
    - Use the values from each `psqi#` column to create a `psqi#.clean` column, and include both in the output.
    - Do not modify any values "by hand" / by index address. Scrambling rows / IDs shouldn't affect how messy values translate into clean ones.
    - Maintain the number of rows in the data frame (i.e. not drop any rows).
    - Include a short description of functionality and sample usage at the top of the file.
3. `score_psqi` should:
    - Process the first two command-line arguments as the names of input CSV file and output CSV file, with a similar invocation and bonus as `clean_psqi`.
        - *Optional:* Implement the CLI argument processing so that the `clean_psqi` script can pipe into the `score_psqi` script, e.g. `clean_psqi psqi_dirty.csv | score_psqi > psqi_scored.csv`.
    - Verify that the input file has *at least* the columns you expect it to have from cleaning. Exit with a non-zero exit code if it does not contain these columns.
    - Use the cleaned data to compute the sleep duration and sleep efficiency PSQI scores.
    - Output a CSV file that has the dirty columns (if available), the clean columns, and the scored columns.
    - Include a short description of functionality and sample usage at the top of the file.
4. `psqi_scored.csv` should contain the original data, the cleaned data, and the required PSQI scores.
5. *Optional:* An environment specification of your choice. A `Dockerfile` is preferred, but we're

equally fine with a conda `environment.yml`, `requirements.txt`, or another reasonable equivalent that will allow us to replicate your environment in up to two commands.

6. *Optional:* Unit and/or integration tests that verify the correctness of (subsets of) your code. If you write them, please provide a short note on how to run them.