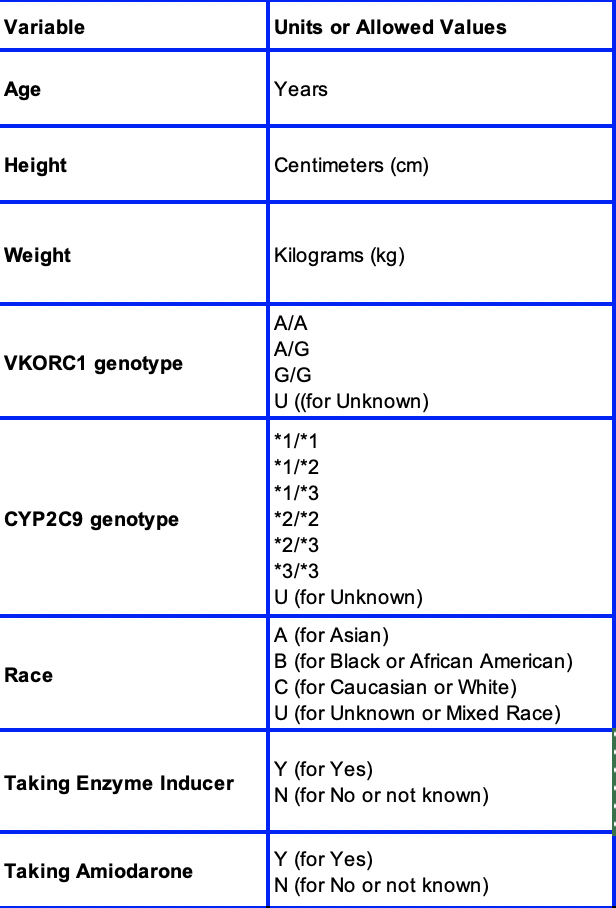
**Instruction: Completed homework should be typed (e.g., using LaTeX or word document) or hand-written clearly and scanned and uploaded into Moodle. You can discuss about how to use certain tools for data collection and analysis, but no collaboration is permitted to solve the problems.**

1. The IWPC Warfarin dosage equation is provided below:

* Age in decades should be entered as 1 for 10–19 years, 2 for 20–29 years, 3 for 30–39 years
* Enzyme-inducer status = 1 if patient taking carbamazepine, phenytoin, rifampin, or rifampicin, otherwise 0
* For VKORC1, CYP2C9, race and amiodarone status enter 1 if present, otherwise 0. For example, if the patient is VKORC1 A/G then only the coefficient for VKORC1 A/G is 1 and the coefficient for all other VKORC1 genotype is 0. Similarly, for any **nonmatching coefficient assign such coefficient to 0** (e.g., if the race of a person is Caucasian then you simply assign ‘Asian race’ =0 and ‘Black or African–American’ =0). For any missing value simply assume its contribution is **zero** to the dosage (e.g., VKORC1 G/G contributes nothing to the dosage).



Assume Bob is your neighbor who is 56-year-old Caucasian male. Assume you are gym buddies with Bob and hence you know his height (5 feet 10 inch) and weight (around 72 kg). Also let us assume that Bob has bipolar disorder (i.e., takes carbamazepine) and heart rhythm problem (i.e., takes amiodarone). One day you saw Bob’s prescription for Warfarin dose to be around 21mg/week then what is his VKORC1 and CYP2C9 genotypes? Write a code to compute the genotype using all the auxiliary information you have.

Submit your **code** and README if there are any special instructions. **[points 20+10]** [code+answer]

1. You are given a skeleton for testing Federated Learning based approach. Create 10 worker nodes. Assign them ids like “node1”, “node2” etc. Then distribute the data across the nodes (the *syft* package has a function to do this easily; take a close look at the code for **TODOs inside the code**). Next, randomly select **X** nodes to participate in the learning process and observe the test accuracy after **N** epochs. Change the seed for random number generation to your student ID number at line 17.
2. Vary the value of **X** as 3, 5, 7, 10 (setting N to 3) and report the test accuracy at the end of the last iteration.Your code can assume to take input X, in which case you need to run it multiple times to generate the table below. **[points 15]**
3. Vary the value of **N** as 3, 5, 10 (setting X to 5) and report the test accuracy at the end of the last iteration. **[points 15]**

Output should look like the following tables:

|  |  |
| --- | --- |
| X | Accuracy (when N=3) |
| 3 |  |
| 5 |  |
| 7 |  |
| 10 |  |

|  |  |
| --- | --- |
| N | Accuracy (when X=5) |
| 3 |  |
| 5 |  |
| 10 |  |

1. Also comment on what you see when you vary X and N from the tables listed above**. [points 10]**

Submit your code and README if there are any special instructions. If want to use separate code for the different parts you can do so, but your README should tell us how to generate the results shown tables.

1. You are given the ‘adult’ dataset which will be used to predicts if a given adult has an income greater than >$50K. Determine whether the training data has statistical parity for the following protected groups:
   1. Female **[points 15]**
   2. Asian people **[points 15]**

If the difference in probability for the protected and non-protected group is less than 0.05 then you can assume they are very close and thus not biased (i.e., fair) towards the protected group. You are given a skeleton code (python3) to read and parse the input data. Please look at the code comment to understand the structure of the data. Categorical values have been flattened to generate one-hot encoding.

Submit your code and README if there are any special instructions.

**Submission:**

You have to submit three files:

1. Merge all the written parts into a single pdf file named <your unity id>\_HW4.**pdf**.

2. Rename the program file you used for as <your unity id>\_HW4\_QX.extension (e.g., .c/.cpp/.java/.py for question X).

3. Add a README file regarding how to run your code.

Zip all files into <your unity id>\_HW4.zip and submit the zip file on Moodle.