CS 234 Winter 2022
HW 4
Due: Feb 27 at 6:00 pm (PST)

For submission instructions please refer to website. For all problems, if you use an existing result from either the literature or a textbook to solve the exercise, you need to cite the source.
This time, there is no submission script for you to run in the starter code of Problem 1. Please only zip "main.py" into a zip file when you upload to Gradescope.

1 Estimation of the Warfarin Dose [60pts]

1.1 Introduction

Warfarin is the most widely used oral blood anticoagulant agent worldwide; with more than 30 million prescriptions for this drug in the United States in 2004. The appropriate dose of warfarin is difficult to establish because it can vary substantially among patients, and the consequences of taking an incorrect dose can be severe. If a patient receives a dosage that is too high, they may experience excessive anti-coagulation (which can lead to dangerous bleeding), and if a patient receives a dosage which is too low, they may experience inadequate anti-coagulation (which can mean that it is not helping to prevent blood clots). Because incorrect doses contribute to a high rate of adverse effects, there is interest in developing improved strategies for determining the appropriate dose (Consortium, 2009).

Commonly used approaches to prescribe the initial warfarin dosage are the *pharmacogenetic algorithm* developed by the IWPC (International Warfarin Pharmacogenetics Consortium), the *clinical algorithm* and a *fixed-dose* approach.

In practice a patient is typically prescribed an initial dose, the doctor then monitors how the patient responds to the dosage, and then adjusts the patient's dosage. This interaction can proceed for several rounds before the best dosage is identified. However, it is best if the correct dosage can be initially prescribed.

This question is motivated by the challenge of Warfarin dosing, and considers a simplification of this important problem, using real data. The goal of this question is to explore the performance of multi-armed bandit algorithms to best predict the correct dosage of Warfarin for a patient *without* a trial-an-error procedure as typically employed.

Problem setting Let T be the number of time steps. At each time step t, a new patient arrives and we observe its individual feature vector $X_t \in \mathbb{R}^d$: this represents the available knowledge about the patient (e.g., gender, age, …). The decision-maker (your algorithm) has access to K arms, where the arm represents the warfarin dosage to provide to the patient. For simplicity, we discretize the actions into $K = 3$

- Low warfarin dose: under 21mg/week
- Medium warfarin dose: 21-49 mg/week
- High warfarin dose: above 49mg/week

If the algorithm identifies the correct dosage for the patient, the reward is 0, otherwise a reward of -1 is received.
Lattimore and Szepesvári have a nice series of blog posts that provide a good introduction to bandit algorithms, available here: BanditAlgs.com. The Introduction and the Linear Bandit posts may be particularly of interest. For more details of the available Bandit literature you can check out the Bandit Algorithms Book by the same authors.

1.2 Dataset

We use a publicly available patient dataset that was collected by staff at the Pharmacogenetics and Pharmacogenomics Knowledge Base (PharmGKB) for 5700 patients who were treated with warfarin from 21 research groups spanning 9 countries and 4 continents. You can find the data in `warfarin.csv` and metadata containing a description of each column in `metadata.xls`. Features of each patient in this dataset includes, demographics (gender, race, . . .), background (height, weight, medical history, . . .), phenotypes and genotypes.
Importantly, this data contains the true patient-specific optimal warfarin doses (which are initially unknown but are eventually found through the physician-guided dose adjustment process over the course of a few weeks) for 5528 patients. You may find this data in mg/week in `Therapeutic Dose of Warfarin`[1] column in `warfarin.csv`. There are in total 5528 patient with known therapeutic dose of warfarin in the dataset (you may drop and ignore the remaining 173 patients for the purpose of this question). Given this data you can classify the right dosage for each patient as *low*: less than 21 mg/week, *medium*: 21-49 mg/week and *high*: more than 49 mg/week, as defined in Consortium (2009) and Introduction.
The data processing is already implemented for you

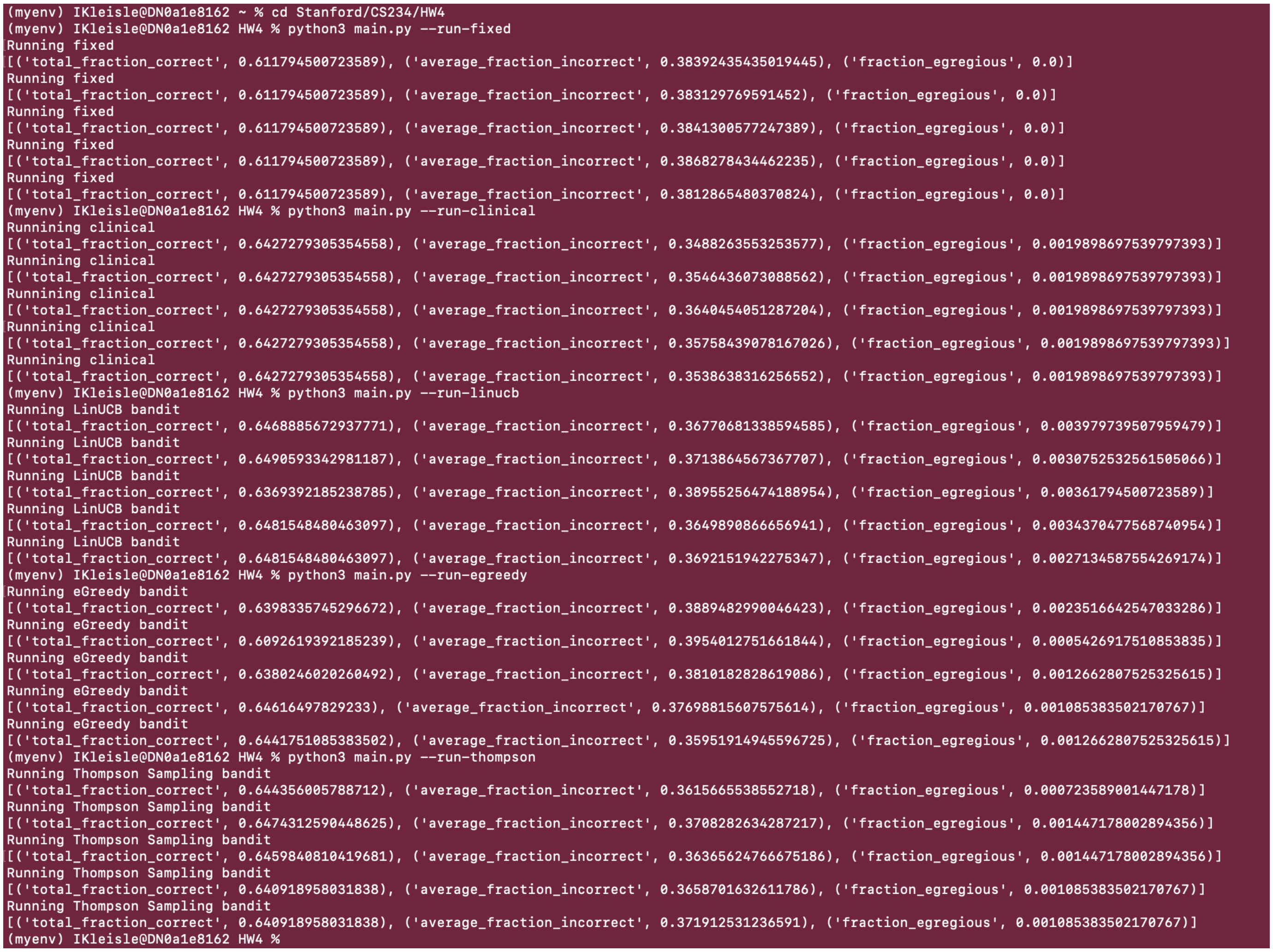
1.3 Implementing Baselines [10pts]

Please implement the following two baselines in main.py

1. *Fixed-dose*: This approach will assign 35mg/week (medium) dose to all patients.
2. *Warfarin Clinical Dosing Algorithm*: This method is a linear model based on age, height, weight, race and medications that patient is taking. You can find the exact model is section S1f of `appx.pdf`.

   Full results for all runs are provided below:

[1] You cannot use `Therapeutic Dose of Warfarin` data as an input to your algorithm.



```
(myenv) IKleisle@DN0a1e8162 ~ % cd Stanford/CS234/HW4
(myenv) IKleisle@DN0a1e8162 HW4 % python3 main.py --run-fixed
Running fixed
[('total_fraction_correct', 0.611794500723589), ('average_fraction_incorrect', 0.38392435435019445), ('fraction_egregious', 0.0)]
Running fixed
[('total_fraction_correct', 0.611794500723589), ('average_fraction_incorrect', 0.383129769591452), ('fraction_egregious', 0.0)]
Running fixed
[('total_fraction_correct', 0.611794500723589), ('average_fraction_incorrect', 0.3841300577247389), ('fraction_egregious', 0.0)]
Running fixed
[('total_fraction_correct', 0.611794500723589), ('average_fraction_incorrect', 0.3868278434462235), ('fraction_egregious', 0.0)]
Running fixed
[('total_fraction_correct', 0.611794500723589), ('average_fraction_incorrect', 0.3812865480370824), ('fraction_egregious', 0.0)]
(myenv) IKleisle@DN0a1e8162 HW4 % python3 main.py --run-clinical
Runnining clinical
[('total_fraction_correct', 0.6427279305354558), ('average_fraction_incorrect', 0.3488263553253577), ('fraction_egregious', 0.0019898697539797393)]
Runnining clinical
[('total_fraction_correct', 0.6427279305354558), ('average_fraction_incorrect', 0.3546436073088562), ('fraction_egregious', 0.0019898697539797393)]
Runnining clinical
[('total_fraction_correct', 0.6427279305354558), ('average_fraction_incorrect', 0.3640454051287204), ('fraction_egregious', 0.0019898697539797393)]
Runnining clinical
[('total_fraction_correct', 0.6427279305354558), ('average_fraction_incorrect', 0.35758439078167026), ('fraction_egregious', 0.0019898697539797393)]
Runnining clinical
[('total_fraction_correct', 0.6427279305354558), ('average_fraction_incorrect', 0.3538638316256552), ('fraction_egregious', 0.0019898697539797393)]
(myenv) IKleisle@DN0a1e8162 HW4 % python3 main.py --run-linucb
Running LinUCB bandit
[('total_fraction_correct', 0.6468885672937771), ('average_fraction_incorrect', 0.36770681338594585), ('fraction_egregious', 0.003979739507959479)]
Running LinUCB bandit
[('total_fraction_correct', 0.6490593342981187), ('average_fraction_incorrect', 0.3713864567367707), ('fraction_egregious', 0.0030752532561505066)]
Running LinUCB bandit
[('total_fraction_correct', 0.6369392185238785), ('average_fraction_incorrect', 0.38955256474188954), ('fraction_egregious', 0.00361794500723589)]
Running LinUCB bandit
[('total_fraction_correct', 0.6481548480463097), ('average_fraction_incorrect', 0.3649890866656941), ('fraction_egregious', 0.0034370477568740954)]
Running LinUCB bandit
[('total_fraction_correct', 0.6481548480463097), ('average_fraction_incorrect', 0.3692151942275347), ('fraction_egregious', 0.0027134587554269174)]
(myenv) IKleisle@DN0a1e8162 HW4 % python3 main.py --run-egreedy
Running eGreedy bandit
[('total_fraction_correct', 0.6398335745296672), ('average_fraction_incorrect', 0.3889482990046423), ('fraction_egregious', 0.0023516642547033286)]
Running eGreedy bandit
[('total_fraction_correct', 0.6092619392185239), ('average_fraction_incorrect', 0.3954012751661844), ('fraction_egregious', 0.0005426917510853835)]
Running eGreedy bandit
[('total_fraction_correct', 0.6380246020260492), ('average_fraction_incorrect', 0.3810182828619086), ('fraction_egregious', 0.0012662807525325615)]
Running eGreedy bandit
[('total_fraction_correct', 0.64616497829233), ('average_fraction_incorrect', 0.37698815607575614), ('fraction_egregious', 0.001085383502170767)]
Running eGreedy bandit
[('total_fraction_correct', 0.6441751085383502), ('average_fraction_incorrect', 0.35951914945596725), ('fraction_egregious', 0.0012662807525325615)]
(myenv) IKleisle@DN0a1e8162 HW4 % python3 main.py --run-thompson
Running Thompson Sampling bandit
[('total_fraction_correct', 0.644356005788712), ('average_fraction_incorrect', 0.3615665538552718), ('fraction_egregious', 0.000723589001447178)]
Running Thompson Sampling bandit
[('total_fraction_correct', 0.6474312590448625), ('average_fraction_incorrect', 0.3708282634287217), ('fraction_egregious', 0.001447178002894356)]
Running Thompson Sampling bandit
[('total_fraction_correct', 0.6459840810419681), ('average_fraction_incorrect', 0.36365624766675186), ('fraction_egregious', 0.001447178002894356)]
Running Thompson Sampling bandit
[('total_fraction_correct', 0.640918958031838), ('average_fraction_incorrect', 0.3658701632611786), ('fraction_egregious', 0.001085383502170767)]
Running Thompson Sampling bandit
[('total_fraction_correct', 0.640918958031838), ('average_fraction_incorrect', 0.371912531236591), ('fraction_egregious', 0.001085383502170767)]
(myenv) IKleisle@DN0a1e8162 HW4 %
```

Run the fixed dosing algorithm and clinical dosing algorithm with the following command:

```
python main.py --run-fixed --run-clinical
```

You should see the total_fraction_correct to be fixed at about 0.61 for fixed dose and 0.64 for clinical dose algorithm. You can run them individually as well. Just use one of the command line arguments instead.

Indeed, we see fixed dose hovering around .611, and clinical dose at .6427.

1.4 Implementing a Linear Upper Confidence Bandit Algorithm [15pts]

Please implement the Disjoint Linear Upper Confidence Bound (LinUCB) algorithm from Li et al. (2010) in main.py. See Algorithm 1 from paper. Please feel free to adjust the –alpha argument, but you don't have to. Run the LinUCB algorithm with the following command:

```
python main.py --run-linucb
```

You should see the total_fraction_correct to be above 0.64, though the results may vary per run.

Again, the UCB method achieves the stated goal: of the five runs, four sit above .64 (maximum .649), while one is just under at .6369.

1.5 Implementing a Linear eGreedy Bandit Algorithm [5pts]

Is the upper confidence bound making a difference? Please implement the e-Greedy algorithm in main.py. Please feel free to adjust the –ep argument, but you don't have to. Does eGreedy perform better or worse than Upper Confidence bound? (You do not need to include your answers here) Run the ε-greedy LinUCB with the following command:

```
python main.py --run-egreedy
```

You should see the total_fraction_correct to be above 0.61, though the results may vary per run.

The e-greedy method is a bit all over the place: using default configurations, the smallest reward is just under the .61 measure, at .609. However, one run returns a value of .646.

1.6 Implementing a Thompson Sampling Algorithm [20pts]

Please implement the Thompson Sampling for Contextual Bandits from Agrawal and Goyal (2013) in main.py. See Algorithm 1 and section 2.2 from paper. Please feel free to adjust the –v2 argument, but you don't have to. (This actually v squared from the paper) Run the Thompson Sampling algorithm with the following command:

```
python main.py --run-thompson
```

You should see the total_fraction_correct to be **around** 0.64, though the results may vary per run.

All five of the Thompson runs sit above the desired .64, with a minimum of .6409 and a maximum of .6474.

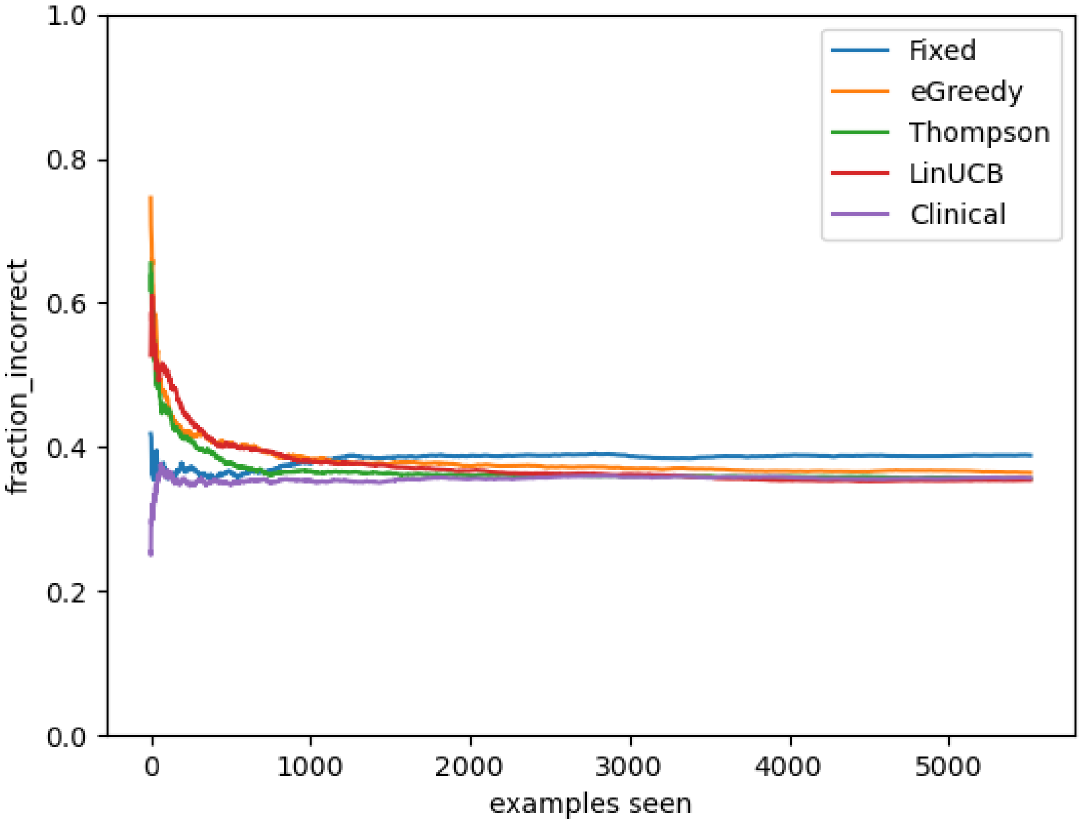
1.7 Results [10pts]

At this point, you should see a plot in your results folder titled "fraction_incorrect.png". If not, run the following command to generate the plot:

```
python main.py
```

Include this plot in for this part. Please also comment on your results in a few sentences. How would you compare the algorithms? Which algorithm "did the best" based on your metric?

Analysis of each method is proffered above; not unexpectedly, the two clear front-runners are LinUCB and Thompson sampling, both of which were consistently at or above .64. Using average run score as the “did the best” criteria, we have:



- AVG(linUCB) = .6458
- AVG(Thompson) = .6439

Hence, in the long run (i.e. > 2000 examples), it appears that the UCB has a minute edge – this bears out in the plot above, where the red line (UCB) has slipped just below the purple and green (Clinical and Thompson) lines.

Of course, we see that over the first ~ 1000 examples, the clinical and fixed methods have the clear edge. As such, we would want to stick with these *en masse* in the early stages for ethical reasons, until we had collected sufficient examples to be confident that the Thompson and/or ECB methods were better.

2 Learning a Policy From an Approximated MDP [20pts]

Consider an infinite-horizon, discounted MDP $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T}, \gamma \rangle$ with $\gamma \in [0,1)$ and bounded rewards such that $R_{\max} = \max\limits_{(s,a)\in\mathcal{S}\times\mathcal{A}} \mathcal{R}(s,a) < \infty$. In practical settings, we rarely know the true model of the agent-environment interaction. Here, we are interested in the case where the model is estimated from experience data in the real world; scarcity of data then implies that our model will only be approximate.

Recall certainty-equivalence and model-based reinforcement-learning where we attempt to compute an optimal policy by first estimating an approximate MDP $\widehat{M} = \langle \mathcal{S}, \mathcal{A}, \widehat{\mathcal{R}}, \widehat{\mathcal{T}}, \gamma \rangle$ from the experience data which is identical to $\mathcal{M}$ except for the approximate reward function $\widehat{\mathcal{R}} : \mathcal{S} \times \mathcal{A} \to [0, R_{\max}]$ and approximate transition function $\widehat{\mathcal{T}} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0,1]$.

Naturally, we are interested in controlling the performance difference between policies computed in $\widehat{\mathcal{M}}$ but deployed in $\mathcal{M}$. Let $\pi^\star_{\mathcal{M}}$ and $\pi^\star_{\widehat{\mathcal{M}}}$ denote the optimal policies of $\mathcal{M}$ and $\widehat{\mathcal{M}}$, respectively. We define the planning loss as follows:

$$||V_{\mathcal{M}}^{\pi^*_{\mathcal{M}}} - V_{\mathcal{M}}^{\pi^*_{\widehat{\mathcal{M}}}}||_\infty$$

Here, $V^\pi_{\mathcal{M}}$ indicates the value function obtained by running policy π in the MDP $\mathcal{M}$

If $\Pi = \{\mathcal{S} \to \mathcal{A}\}$ denotes the class of all stationary, deterministic policies, let $\overline{\Pi} \subseteq \Pi$ be a restricted policy class.

(a) **[5pts]** Prove that

$$||V_{\mathcal{M}}^{\pi^*_{\mathcal{M}}} - V_{\mathcal{M}}^{\pi^*_{\widehat{\mathcal{M}}}}||_\infty \leq 2 \max_{\pi\in\Pi} ||V^\pi_{\mathcal{M}} - V^\pi_{\widehat{\mathcal{M}}}||_\infty.$$

For any function $f : \mathcal{X} \to \mathbb{R}$ and any two sets $A \subseteq B \subseteq \mathcal{X}$, we have that $\max\limits_{x\in A} f(x) \leq \max\limits_{x\in B} f(x)$. This fact immediately implies the following corollary of part (a) whenever $\pi^*_{\mathcal{M}}, \pi^*_{\widehat{\mathcal{M}}} \in \overline{\Pi}$:

$$||V_{\mathcal{M}}^{\pi^*_{\mathcal{M}}} - V_{\mathcal{M}}^{\pi^*_{\widehat{\mathcal{M}}}}||_\infty \leq 2 \max_{\pi\in\overline{\Pi}} ||V^\pi_{\mathcal{M}} - V^\pi_{\widehat{\mathcal{M}}}||_\infty.$$

We have

$$\begin{aligned}
||V_M^{\pi^*_M} - V_M^{\pi^*_{\hat{M}}}||_\infty &= ||V_M^{\pi^*_M} - V_{\hat{M}}^{\pi^*_M} + V_{\hat{M}}^{\pi^*_M} - V_M^{\pi^*_{\hat{M}}}||_\infty \\
&\leq ||V_M^{\pi^*_M} - V_{\hat{M}}^{\pi^*_M}||_\infty + ||V_{\hat{M}}^{\pi^*_M} - V_M^{\pi^*_{\hat{M}}}||_\infty \\
&= \underbrace{||V_M^{\pi^*_M} - V_{\hat{M}}^{\pi^*_M}||_\infty}_{(i)} + \underbrace{||V_M^{\pi^*_{\hat{M}}} - V_{\hat{M}}^{\pi^*_M}||_\infty}_{(ii)}.
\end{aligned}$$

First, looking at (i), it follows by definition of the inf norm and the max operator that

$$\begin{aligned}||V_M^{\pi^*_M} - V_{\hat{M}}^{\pi^*_M}||_\infty &= \max_s |V_M^{\pi^*_M}(s) - V_{\hat{M}}^{\pi^*_M}(s)| \\ &\leq \max_\pi \max_s |V_M^\pi(s) - V_{\hat{M}}^\pi(s)| \\ &= \max_\pi ||V_M^\pi - V_{\hat{M}}^\pi||_\infty.\end{aligned}$$

Then, looking at (ii), first recall the fact that rewards are bounded in $[0, R_{\max}]$. This, coupled with the definition of optimality, yields for any π and s

$$V_{\hat{M}}^\pi(s) \leq V_{\hat{M}}^{\pi^*_{\hat{M}}}(s)$$

and

$$V_M^\pi(s) \leq V_M^{\pi^*_M}(s).$$

Now break into two cases. **First**, suppose that the s' that satisfies the inf-norm $||V_M^{\pi^*_{\hat{M}}} - V_{\hat{M}}^{\pi^*_M}||_\infty$ induces

$$V_M^{\pi^*_{\hat{M}}}(s') - V_{\hat{M}}^{\pi^*_M}(s') < 0$$

By the bounded and positive rewards, this would give

$$V_{\hat{M}}^{\pi^*_M}(s') > V_M^{\pi^*_{\hat{M}}}(s') > 0 \implies V_{\hat{M}}^{\pi^*_{\hat{M}}}(s') > V_{\hat{M}}^{\pi^*_M}(s') > V_M^{\pi^*_{\hat{M}}}(s') > 0,$$

by the first inequality above. Hence, it must be that (as we are now subtracting off a "bigger" term)

$$\begin{aligned}||V_M^{\pi^*_{\hat{M}}} - V_{\hat{M}}^{\pi^*_M}||_\infty &= |V_M^{\pi^*_{\hat{M}}}(s') - V_{\hat{M}}^{\pi^*_M}(s')| \\ &< |V_M^{\pi^*_{\hat{M}}}(s') - V_{\hat{M}}^{\pi^*_{\hat{M}}}(s')| \\ &\leq ||V_M^{\pi^*_{\hat{M}}} - V_{\hat{M}}^{\pi^*_{\hat{M}}}||_\infty \\ &\leq \max_\pi ||V_M^\pi - V_{\hat{M}}^\pi||_\infty.\end{aligned}$$

Second, suppose alternatively that the s' that satisfies the inf-norm $||V_M^{\pi^*_{\hat{M}}} - V_{\hat{M}}^{\pi^*_M}||_\infty$ induces

$$V_M^{\pi^*_{\hat{M}}}(s') - V_{\hat{M}}^{\pi^*_M}(s') > 0$$

By the bounded and positive rewards, this would give

$$V_M^{\pi^*_{\hat{M}}}(s') > V_{\hat{M}}^{\pi^*_M}(s') > 0 \implies V_M^{\pi^*_M}(s') > V_M^{\pi^*_{\hat{M}}}(s') > V_{\hat{M}}^{\pi^*_M}(s') > 0,$$

by the first inequality above. Hence, it must be that (as we are now subtracting off a "bigger" term)

$$\begin{aligned}||V_M^{\pi^*_{\hat{M}}} - V_{\hat{M}}^{\pi^*_M}||_\infty &= |V_M^{\pi^*_{\hat{M}}}(s') - V_{\hat{M}}^{\pi^*_M}(s')| \\ &< |V_M^{\pi^*_M}(s') - V_{\hat{M}}^{\pi^*_M}(s')| \\ &\leq ||V_M^{\pi^*_M} - V_{\hat{M}}^{\pi^*_M}||_\infty \\ &\leq \max_\pi ||V_M^\pi - V_{\hat{M}}^\pi||_\infty.\end{aligned}$$

So **no matter what**, for (ii)

$$||V_M^{\pi^*_{\hat{M}}} - V_{\hat{M}}^{\pi^*_M}||_\infty < \max_\pi ||V_M^\pi - V_{\hat{M}}^\pi||_\infty.$$

Hence, our original inequality becomes

$$\begin{aligned}
||V_M^{\pi^*_M} - V_M^{\pi^*_{\hat{M}}}||_\infty &= ||V_M^{\pi^*_M} - V_{\hat{M}}^{\pi^*_M} + V_{\hat{M}}^{\pi^*_M} - V_M^{\pi^*_{\hat{M}}}||_\infty \\
&\leq ||V_M^{\pi^*_M} - V_{\hat{M}}^{\pi^*_M}||_\infty + ||V_{\hat{M}}^{\pi^*_M} - V_M^{\pi^*_{\hat{M}}}||_\infty \\
&= \underbrace{||V_M^{\pi^*_M} - V_{\hat{M}}^{\pi^*_M}||_\infty}_{(i)} + \underbrace{||V_M^{\pi^*_{\hat{M}}} - V_{\hat{M}}^{\pi^*_M}||_\infty}_{(ii)} \\
&\leq \max_{\pi \in \prod} ||V_M^\pi - V_{\hat{M}}^\pi||_\infty + \max_\pi ||V_M^\pi - V_{\hat{M}}^\pi||_\infty \\
&= 2 \max_{\pi \in \prod} ||V_M^\pi - V_{\hat{M}}^\pi||_\infty,
\end{aligned}$$

as desired. As set forth above, we thus have

$$||V_{\mathcal{M}}^{\pi^*_{\mathcal{M}}} - V_{\mathcal{M}}^{\pi^*_{\widehat{\mathcal{M}}}}||_\infty \leq 2 \max_{\pi \in \overline{\Pi}} ||V_{\mathcal{M}}^\pi - V_{\widehat{\mathcal{M}}}^\pi||_\infty.$$

(b) **[7pts]** Prove that for any $\pi \in \Pi$,

$$||Q_{\mathcal{M}}^\pi - Q_{\widehat{\mathcal{M}}}^\pi||_\infty \leq \frac{1}{1-\gamma} \max_{(s,a) \in \mathcal{S} \times \mathcal{A}} \left| \widehat{\mathcal{R}}(s,a) + \gamma \mathbb{E}_{s' \sim \widehat{\mathcal{T}}(\cdot|s,a)} \left[V_{\mathcal{M}}^\pi(s') \right] - Q_{\mathcal{M}}^\pi(s,a) \right|.$$

Hint: Define the approximate Bellman operator $\widehat{\mathcal{B}}^\pi : \{\mathcal{S} \to \mathbb{R}\} \to \{\mathcal{S} \times \mathcal{A} \to \mathbb{R}\}$ such that

$$\widehat{\mathcal{B}}^\pi Q(s,a) = \widehat{\mathcal{R}}(s,a) + \gamma \mathbb{E}_{s' \sim \widehat{\mathcal{T}}(\cdot|s,a)} \left[V(s') \right],$$

and consider the sequence of value functions $\{Q_0, Q_1, \ldots, Q_m, ..., Q_\infty\}$ where

$$\begin{cases}
Q_0(s,a) = Q_{\mathcal{M}}^\pi(s,a) & \forall (s,a) \in \mathcal{S} \times \mathcal{A} \\
V_0(s) = V_M^\pi & \forall s \in \mathcal{S} \\
Q_m(s,a) = \widehat{\mathcal{B}}^\pi Q_{m-1} & \forall (s,a) \in \mathcal{S} \times \mathcal{A} \\
Q_\infty(s,a) = Q_{\widehat{M}}^\pi(s,a) & \forall (s,a) \in \mathcal{S} \times \mathcal{A}
\end{cases}.$$

Hint: Can you show that $\widehat{\mathcal{B}}^\pi$ is a contraction? You may find the telescoping sum and triangle inequality proof techniques you used in Assignment 1 Q2 to be useful here.
Now, suppose it is the case that s', a' satisfy the inf-norm $||Q_M^\pi - Q_{\hat{M}}^\pi||_\infty$. As proffered in the problem setup, we know that $Q_{\hat{M}}^\pi$ is a fixed point of $\hat{\beta}^p i$, that is

$$\hat{\beta}^{pi} Q_{\hat{M}}^\pi = Q_{\hat{M}}^\pi.$$

Thus, we get

$$\begin{aligned}
||Q^\pi_M - Q^\pi_{\hat{M}}||_\infty &= |Q^\pi_M(s', a') - Q^\pi_{\hat{M}}(s', a')| \\
&= |Q^\pi_M(s', a') - \hat{\beta}^\pi Q^\pi_M(s', a') + \hat{\beta}^\pi Q^\pi_M(s', a') - Q^\pi_{\hat{M}}(s', a')| \\
&\leq |Q^\pi_M(s', a') - \hat{\beta}^\pi Q^\pi_M(s', a')| + |\hat{\beta}^\pi Q^\pi_M(s', a') - Q^\pi_{\hat{M}}(s', a')| \\
&\leq ||Q^\pi_M - \hat{\beta}^\pi Q^\pi_M||_\infty + ||\hat{\beta}^\pi Q^\pi_M - Q^\pi_{\hat{M}}||_\infty \\
&\leq ||Q^\pi_M - \hat{\beta}^\pi Q^\pi_M||_\infty + ||\hat{\beta}^\pi Q^\pi_M - \hat{\beta}^\pi Q^\pi_{\hat{M}}||_\infty \\
&\leq ||Q^\pi_M - \hat{\beta}^\pi Q^\pi_M||_\infty + \gamma||Q^\pi_M - Q^\pi_{\hat{M}}||_\infty \\
&\leq ||\hat{\beta}^\pi Q^\pi_M - Q^\pi_M||_\infty + \gamma||Q^\pi_M - Q^\pi_{\hat{M}}||_\infty.
\end{aligned}$$

The steps are justified by: (i) definition of s', a'; (ii) a cheap insertion/uninsertion; (iii) triangle inequality; (iv) definition of an inf-norm; (v) the fixed point set forth above; (vi) the fact that $\hat{\beta}^\pi$ is a contraction (**proven below**); (vii) simple rearrangement. So then we solve for $||Q^\pi_M - Q^\pi_{\hat{M}}||_\infty$ in the above to get

$$\begin{aligned}
||Q^\pi_M - Q^\pi_{\hat{M}}||_\infty &= \frac{||\hat{\beta}^\pi Q^\pi_M - Q^\pi_M||_\infty}{1-\gamma} \\
&= \frac{\max_{(s,a)\in S\times A} |\hat{\beta}^\pi Q^\pi_M(s, a) - Q^\pi_M(s, a)|}{1-\gamma} \\
&= \frac{\max_{(s,a)\in S\times A} |\hat{R}(s, a) + \gamma E_{s'\sim\hat{T}(\cdot|s,a)} V_M(s') - Q^\pi_M(s, a)|}{1-\gamma},
\end{aligned}$$

where the final step is just substitution of the approximate Bellman.

Proof of Contraction

As promised, I prove the contraction step above. Specifically, let s', a' satisfy $||\hat{\beta}^\pi Q^\pi_M - \hat{\beta}^\pi Q^\pi_{\hat{M}}||_\infty$. This expands out to

$$\begin{aligned}
||\hat{\beta}^\pi Q^\pi_M - \hat{\beta}^\pi Q^\pi_{\hat{M}}||_\infty &= |\hat{\beta}^\pi Q^\pi_M(s', a') - \hat{\beta}^\pi Q^\pi_{\hat{M}}(s', a')| \\
&= |\hat{R}(s', a') + \gamma E_{s''\sim\hat{T}(\cdot|s',a')} V^\pi_M(s'') - \hat{R}(s', a') - \gamma E_{s''\sim\hat{T}(\cdot|s',a')} V^\pi_{\hat{M}}(s'')| \\
&= \gamma|E_{s''\sim\hat{T}(\cdot|s',a')}[V^\pi_M(s'') - V^\pi_{\hat{M}}(s'')]| \\
&= \gamma\left|\sum_{s''} \pi_{\hat{T}}(s''|s', a')[V^\pi_M(s'') - V^\pi_{\hat{M}}(s'')]\right| \\
&\leq \gamma \sum_{s''} \pi_{\hat{T}}(s''|s', a')\left|V^\pi_M(s'') - V^\pi_{\hat{M}}(s'')\right| \\
&\leq \gamma \sum_{s''} \pi_{\hat{T}}(s''|s', a') \max_{s''', a'''}\left|Q^\pi_M(s''', a''') - Q^\pi_{\hat{M}}(s''', a''')\right| \\
&= \gamma \sum_{s''} \pi_{\hat{T}}(s''|s', a')||Q^\pi_M - Q^\pi_{\hat{M}}||_\infty \\
&= \gamma||Q^\pi_M - Q^\pi_{\hat{M}}||_\infty \sum_{s''} \pi_{\hat{T}}(s''|s', a')
\end{aligned}$$

$$= \gamma ||Q^\pi_M - Q^\pi_{\hat{M}}||_\infty \cdot 1,$$

as desired.

(c) **[8pts]** Assume that the restricted policy class is finite ($|\overline{\Pi}| < \infty$) and that we have observed n independent samples of rewards and next-state transitions from each state-action pair $(s, a) \in \mathcal{S} \times \mathcal{A}$. Prove for any $\delta \in [0, 1]$ that the following bound on planning loss holds with probability at least $1 - \delta$:

$$||V_{\mathcal{M}}^{\pi^\star_{\mathcal{M}}} - V_{\mathcal{M}}^{\pi^\star_{\widehat{\mathcal{M}}}}||_\infty \leq \frac{2R_{\max}}{(1-\gamma)^2}\sqrt{\frac{1}{2n}\log\frac{2|\mathcal{S}||\mathcal{A}||\overline{\Pi}|}{\delta}}.$$

Hint: We are sampling from $\widehat{\mathcal{T}}$ in $\widehat{\mathcal{M}}$ to compute $Q^\pi_{\widehat{M}}$, while the true mean is Q^π_M.

Hint: Recall Hoeffding's inequality – let $X_1, X_2, \dots, X_n$ be a finite sequence of i.i.d., real-valued random variables such that $X_i \in [a, b]$ for all $i \in \{1, \dots, n\}$. Denote the sample mean and empirical mean as $\overline{X} = \frac{1}{n}\sum_i X_i$ and $\mu = \mathbb{E}[X_1]$, respectively. Then, the following concentration inequality holds for any $\varepsilon > 0$:

$$\mathbb{P}(\left|\overline{X} - \mu\right| \leq \varepsilon) \geq 1 - 2\exp\left(\frac{-2n^2\varepsilon^2}{\sum_{i=1}^n (b_i - a_i)^2}\right).$$

First, let's get the b_i, a_i and the $\sum_{i=1}^n (b_i - a_i)^2$ part out of the way. We can show straightforwardly, using the definition of Q and V that for any a, s, π, for which rewards are bounded in $[0, R_{\max}]$

$$\begin{aligned}
Q^\pi(s,a) &= R(s,a) + \gamma \sum_{s'} V^\pi(s')P(s'|s,a) \\
&\leq R_{\max} + \gamma \sum_{s'} P(s'|s,a)V^\pi(s') \\
&= R_{\max} + \gamma \sum_{s'} P(s'|s,a)E_\pi\left[\sum_{t=0}^\infty \gamma^t r_t\right] \\
&\leq R_{\max} + \gamma \sum_{s'} P(s'|s,a)E_\pi\left[\sum_{t=0}^\infty \gamma^t R_{\max}\right] \\
&= R_{\max} + \gamma \sum_{s'} P(s'|s,a)R_{\max}E_\pi\left[\sum_{t=0}^\infty \gamma^t\right] \\
&= R_{\max} + \gamma \sum_{s'} P(s'|s,a)R_{\max}\sum_{t=0}^\infty \gamma^t \\
&= R_{\max} + \gamma \left(R_{\max}\sum_{t=0}^\infty \gamma^t\right)\sum_{s'} P(s'|s,a)
\end{aligned}$$

$$
\begin{aligned}
&= R_{\max} + \gamma \left(R_{\max} \sum_{t=0}^{\infty} \gamma^t \right) \cdot 1 \\
&= R_{\max} \sum_{t=0}^{\infty} \gamma^t \\
&= \frac{R_{\max}}{1-\gamma}.
\end{aligned}
$$

Hence, $(b_i - a_i)^2 = \left(\frac{R_{\max}}{1-\gamma} - 0\right)^2$ in this setup. We will use this shortly.

Now, examine for whatever π, a, s

$$\left| \hat{R}(s,a) + \gamma E_{s' \sim \hat{T}(\cdot|s,a)}[V_M^\pi(s')] - Q_M^\pi(s,a) \right|.$$

Thinking in terms of certainty equivalence in conjunction with Hoeffding notation, it is easy to think of the estimate as

$$\hat{R}(s,a) + \gamma E_{s' \sim \hat{T}(\cdot|s,a)}[V_M^\pi(s')] \approx \text{``}\bar{X}_n\text{''},$$

as it is constructed off of some number n offline samples. We liken the estimate to a mean, because "hat" estimates are (i) taken via sampling (akin to sampled averages) and (ii) in the certainty-equivalence setting, often taken via MLE, which amounts to sample averaging[2]. Similarly, it is easy to think of the true value – the thing we wish to approximate – as

$$Q_M^\pi(s,a) \approx \text{``}X\text{''}.$$

Hoeffding then gives us for $\varepsilon > 0$ (note ε implies a $\delta > 0$, which we leverage for notational convenience) that

$$P(|\bar{X}_n - X| > \varepsilon) \leq 2 \exp\left(\frac{-2n^2\varepsilon^2}{\sum_{i=1}^n (b_i - a_i)^2} \right) = \delta,$$

so in our case, this amounts to

$$P\left(\left| \hat{R}(s,a) + \gamma E_{s' \sim \hat{T}(\cdot|s,a)}[V_M^\pi(s')] - Q_M^\pi(s,a) \right| > \varepsilon \right) \leq 2 \exp\left(\frac{-2n^2\varepsilon^2}{n\left(\frac{R_{\max}}{1-\gamma}\right)^2} \right) = \delta,$$

using our finding about bounded rewards from the outset of the problem. We then solve for ε,

[2] The method of estimation for the "hat" estimates is not provided by the problem setup.

i.e.

$$\begin{aligned}
2\exp\left(\frac{-2n^2\varepsilon^2}{n\left(\frac{R_{\max}}{1-\gamma}\right)^2}\right) &= \delta \\
\implies \frac{-2n\varepsilon^2}{\left(\frac{R_{\max}}{1-\gamma}\right)^2} &= \log\left(\frac{\delta}{2}\right) \\
\implies 2n\varepsilon^2 &= \left(\frac{R_{\max}}{1-\gamma}\right)^2 \log\left(\frac{2}{\delta}\right) \\
\implies \varepsilon^2 &= \left(\frac{R_{\max}}{1-\gamma}\right)^2 \frac{1}{2n}\log\left(\frac{2}{\delta}\right) \\
\implies \varepsilon &= \left(\frac{R_{\max}}{1-\gamma}\right)\sqrt{\frac{1}{2n}\log\left(\frac{2}{\delta}\right)}.
\end{aligned}$$

Again, choice of π, a, s were arbitrary here. Accordingly, we have by the complement and then basic algebra that for arbitrary π, a, s that:

$$P\left(\left|\hat{R}(s,a) + \gamma E_{s'\sim\hat{T}(\cdot|s,a)}[V_M^\pi(s')] - Q_M^\pi(s,a)\right| > \left(\frac{R_{\max}}{1-\gamma}\right)\sqrt{\frac{1}{2n}\log\left(\frac{2}{\delta}\right)}\right) \leq \delta$$

$$\implies$$

$$P\left(\left|\hat{R}(s,a) + \gamma E_{s'\sim\hat{T}(\cdot|s,a)}[V_M^\pi(s')] - Q_M^\pi(s,a)\right| \leq \left(\frac{R_{\max}}{1-\gamma}\right)\sqrt{\frac{1}{2n}\log\left(\frac{2}{\delta}\right)}\right) > 1-\delta$$

$$\implies$$

$$P\left(\frac{1}{1-\gamma}\left|\hat{R}(s,a) + \gamma E_{s'\sim\hat{T}(\cdot|s,a)}[V_M^\pi(s')] - Q_M^\pi(s,a)\right| \leq \left(\frac{R_{\max}}{(1-\gamma)^2}\right)\sqrt{\frac{1}{2n}\log\left(\frac{2}{\delta}\right)}\right) > 1-\delta$$

$$\implies$$

$$P\left(\frac{1}{1-\gamma}\left|\hat{R}(s,a) + \gamma E_{s'\sim\hat{T}(\cdot|s,a)}[V_M^\pi(s')] - Q_M^\pi(s,a)\right| \leq \left(\frac{R_{\max}}{(1-\gamma)^2}\right)\sqrt{\frac{1}{2n}\log\left(\frac{2|S||A||\bar{\prod}|}{\delta}\right)}\right) > 1-\delta$$

$$\implies$$

$$P\left(\frac{1}{1-\gamma}\max_{a,s,\pi}\left|\hat{R}(s,a) + \gamma E_{s'\sim\hat{T}(\cdot|s,a)}[V_M^\pi(s')] - Q_M^\pi(s,a)\right| \leq \left(\frac{R_{\max}}{(1-\gamma)^2}\right)\sqrt{\frac{1}{2n}\log\left(\frac{2|S||A||\bar{\prod}|}{\delta}\right)}\right) > 1-\delta,$$

where steps are justified by: (1) our original finding; (2) taking a complement; (3) multiplication on both sides by $(1-\gamma)^{-1}$; (4) taking a union bound to ensure coverage over all s, a, π; (5) the fact that a, s, π were arbitrary, and hence it should hold over any state and action.

At this point, we can put everything together: in part a., we showed that

$$||V_M^{\pi^*_M} - V_M^{\pi^*_{\hat{M}}}||_\infty \leq 2\max_{\pi\in\bar{\prod}}||V_M^\pi - V_{\hat{M}}^\pi||_\infty.$$

By a subset argument, it is then clearly the case that

$$||V_M^\pi - V_{\hat{M}}^\pi||_\infty \leq ||Q_M^\pi - Q_{\hat{M}}^\pi||_\infty,$$

and we showed in b. that

$$||Q_M^\pi - Q_{\hat{M}}^\pi||_\infty \leq \frac{1}{1-\gamma}\max_{a,s}\left|\hat{R}(s,a) + \gamma E_{s'\sim\hat{T}(\cdot|s,a)}[V_M^\pi(s')] - Q_M^\pi(s,a)\right|.$$

Thus, altogether, we have

$$\begin{aligned}
||V_M^{\pi^*_M} - V_M^{\pi^*_{\hat{M}}}||_\infty &\leq 2\max_{\pi\in\bar{\prod}}||V_M^\pi - V_{\hat{M}}^\pi||_\infty \\
&\leq 2\max_{\pi\in\bar{\prod}}||Q_M^\pi - Q_{\hat{M}}^\pi||_\infty \\
&\leq 2\max_{\pi\in\bar{\prod}}\left(\frac{1}{1-\gamma}\max_{a,s}\left|\hat{R}(s,a) + \gamma E_{s'\sim\hat{T}(\cdot|s,a)}[V_M^\pi(s')] - Q_M^\pi(s,a)\right|\right) \\
&= \underbrace{2\left(\frac{1}{1-\gamma}\max_{a,s,\pi\in\bar{\pi}}\left|\hat{R}(s,a) + \gamma E_{s'\sim\hat{T}(\cdot|s,a)}[V_M^\pi(s')] - Q_M^\pi(s,a)\right|\right)}_{\leq\frac{R_{\max}}{(1-\gamma)}\sqrt{\frac{1}{2n}\log\left(\frac{2|S||A||\bar{\prod}|}{\delta}\right)}\ w.p.\ >1-\delta}
\end{aligned}$$

Hence,

$$||V_M^{\pi^*_M} - V_M^{\pi^*_{\hat{M}}}||_\infty \leq \frac{R_{\max}}{(1-\gamma)}\sqrt{\frac{1}{2n}\log\left(\frac{2|S||A||\bar{\prod}|}{\delta}\right)}$$

with probability $> 1-\delta$, as desired.

3 Analysing features used in Warfarin dosage algorithms [4pts]

1. **[2pts]** As a recent survey of dosing algorithms suggest, most existing clinical algorithms have not been evaluated for their efficacy in under-served populations. How big of a disparity would warrant intervention? How would you decide on a threshold? In a scenario in which your disparity threshold was met, what constraint(s) (Thomas et al. (2017)) could you introduce to your algorithm?

   In general, whenever the disparity between (i) the Warfarin evaluation population and (ii) the performance within each subgroup of the population*i.e. it works for one race but not another* differs non-trivially from that of the deployment population (which, under basic notions of equality/fairness, should be the *entire* US or global population – not just a couple of races or subgroups), I'd start to get pretty uncomfortable. In general, I feel that procedures/advancements like these should be made available equally (both with respect to access and efficacy), so it would not seem right to tailor things to one (possibly more privileged) population. So my bar for intervention is probably pretty low – any statistically significant disparity (think intro Stat tests), either in terms of inclusion with respect to the demographics of the broader population, or performance across demographics, is more/less my standard.

   To correct for these two sources of disparity, I would:

(a) Encourage some form of stratification within the training/evaluation population, to address the fundamental disparity in who is being considered/evaluated under these dosings, and;

(b) Use, per Thomas et al., some combination of soft-constraints, hard-constraints, and/or other full/quasi-Seldonian algorithms to ensure that performance is comparable (say, up to statistical significance), across cross-sections of the population. This would nudge the models towards equal performance across groups, which I think is fair and reasonable.

2. Many current Warfarin dosage algorithms rely on race as a feature. When race is used in medicine, it is often intended as a proxy for genetic difference (Vyas et. al 2020). However, as (Goodman & Brett 2021) note, prevalence of genetic markers relevant to a particular disease might vary more within a group than between groups. Observed racial differences in health outcomes are at least as likely to be caused by social and environmental determinants of health (such as differential access to lead-free water) as they are by genetic factors.

   (a) **[OPTIONAL]** Run at least one of your algorithms again without the "race" or "gender" feature. How does accuracy change?

   (b) **[2pts]** In addition to knowing whether average accuracy increases or decreases, what else might you measure or investigate in order to determine whether to use race as a feature in your algorithm? First, if we know there are other factors that confound race – e.g. access to lead-free water, access to produce/food deserts, etc. – but we are still interested in the *genetic* aspect of race, we may try to feature engineer those factors (e.g. some proxy for water quality, distance to nearest grocery store, etc.) into the feature set. This would allow us to isolate the genetic component of race while controlling for the social and environmental aspects; in turn, we might better be able to model Warfarin behavior based on race without penalizing or mis-dosing people based on misleading social/environmental features.

References

S. Agrawal and N. Goyal. Thompson sampling for contextual bandits with linear payoffs. In *International Conference on Machine Learning*, pages 127–135, 2013.

I. W. P. Consortium. Estimation of the warfarin dose with clinical and pharmacogenetic data. *New England Journal of Medicine*, 360(8):753–764, 2009.

L. Li, W. Chu, J. Langford, and R. E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670, 2010.

P. S. Thomas, B. C. da Silva, A. G. Barto, and E. Brunskill. On ensuring that intelligent machines are well-behaved, 2017.