Fairness in protease specificity

Changpeng Lu, Weiting Lyu

1 Introduction

In biology, molecular recognition is a key interaction between host and guest biomolecules, e.g., DNA – protein [1], protein – protein, to drive affinities of the host-guest complex. It is an essential biological process to specify functions of enzymes, such as proteases. Every enzyme only recognizes specific substrate sequences over the substrate landscape, like locks are only fit for correct keys. Reversely, substrate sequences in the landscape, that turn out to be recognized/cut, or not recognized/cut by the enzyme, could indicate specificity of each enzyme. Therefore, instead of intensive biological experiments, we could take advantage of the outstanding ability of supervised learning that could explore main feature information implicitly from the massive feature space. Hence, we encapsulate it as a classification problem that predict cleavage information of unknown substrates on a specific protease to discover protease specificity without long cycle experiments for each individual unknown substrate.

Current methods for predicting protease specificity landscapes rely on learning sequence patterns by using machine learning (ML) models in experimentally derived data [2]. Specifically, specific amino acid types of all residues of the protein represent features based on sequence information. Since there are entirely 20 amino acid types in nature, a 20-binary-numbers-long one hot vector encodes sequential features of each position of the sequence. Another four total energy terms of the protease/peptide/interface qualify the problem regarding discovering protein-protein interaction.

Recent work addressed outstanding performances of ML models on protease specificity prediction [3-11]. Scientists discovered meaningful and unique patterns from results of ML models using their biology knowledge to imply the fitness of ML results with biology insights. However, reports on checking ML fairness before biological analysis are rare. Those reported models, though resulted in extremely high accuracy of ML models on a specific biology data, like the prediction of hepatitis C virus (HCV) wild type protease specificity, and well indication of the research purpose, it is worth mentioning that the false positive rate was high and the predicted patterns among different protein systems were volatile at the meantime. For example, most ML models highly suggested most of partially cleaved substrates into cleaved (or uncleaved) class when ML models that were trained to test cleaved/uncleaved substrates, test partially cleaved substrates. Those sequences that were partially recognized/cut exists, which means those sequences do not belong to either cleaved or not cleaved class in nature, should be equally classified into cleaved and uncleaved class, randomly. Taking into account that such conflicts of ML results with prior biology knowledge hindered the path to real natural landscape discovery, we would like to implement ML fairness technique to adjust ML models along with the sensitive feature direction, to remove bias of models on some obvious biological insights as much as possible. Herein, we used partially cleaved data as the sensitive subspace of the enzyme molecular recognition problem as an example. Moreover, we could consider more subspaces related to biology insights in the future. For example, sequences that contain cystine (one type of amino acid) are invalid data for this problem, but when we predicted new sequences using ML models based on experimental data, models indicated several sequences that contain cystine to be cleaved by the enzyme, which contradicted with biological prior knowledge. Hence, it is also essential to check whether ML models made biases to cystine-contained sequences.

2 Methods

Herein, we implement the work from Mikhail, Amanda et. al [12], who develops a distributionally robust optimization approach to enforce individual fairness during training.

2.1 Machine learning fairness

Machine learning (ML), widely used in decision making problems, is able to consider all characteristics comprehensively and capture inherent features of objects to predict the whole object landscape, based on only a small set of data (training data). Nevertheless, ML models fail to make appropriate decisions by training data so that they bolster or even aggravate objectionable biases arisen by humans. Therefore, we should evaluate fairness of ML models based on prior knowledge of how appropriate decisions look. In most cases, ML models is not fair in a specific sensitive direction (such as gender, ethnicity, sexual orientation, disability, etc.), so the goal of fairness is to validate the performance of ML models is invariant under certain perturbations in a sensitive subspace.

There are two types of formal definitions of algorithmic fairness: group fairness and individual fairness. People evaluate group fairness by equalize two groups at the level of the outcome. Group fairness make sure statistical outcomes be fair, but individuals might still be discriminated against. Past study related to individual fairness is somewhat impractical for many ML tasks, since fairness is a task-specific problem. Thus, in this paper, they introduce a data-driven optimization approach based on distributionally robustly fair (DRF) that is easy to implement for different ML tasks, to formulate the fair metric.

2.2. Distributionally Robustly Fair (DRF) optimization problem

For a classification task, they define the fair metric d_x as the form,

$$d_x(x_1, x_2)^2 \triangleq \langle x_1 - x_2, \sum (x_1 - x_2) \rangle^{\frac{1}{2}} \tag{1}$$

where $\sum \in \mathcal{S}_+^{d \times d}$, which is the orthogonal complement projector of the span of the sensitive directions. Define $\mathcal{X}$ and $\mathcal{Y}$ are the spaces of inputs and outputs. Then they equip the input with the fair metric, so the square root of the transport cost function on $\mathcal{Z} \triangleq \mathcal{X} \times \mathcal{Y}$ is denoted as,

$$d_z((x_1, y_1), (x_2, y_2)) \triangleq d_x(x_1, x_2) + \infty \cdot 1\{y_1 \neq y_2\} \tag{2}$$

They denote the probability distributions on $\mathcal{Z}$ using the fair Wasserstein distance,

$$W(P, Q) = \inf_{\prod \in \mathcal{C}(P,Q)} \int_{\mathcal{Z} \times \mathcal{Z}} c(z_1, z_2) d \prod (z_1, z_2) \tag{3}$$

where $\mathcal{C}(P, Q)$ is the set of couplings between P and Q.

Therefore, the optimization problem is shown below,

$$\max_{P: W(P, P_n)} \leq_\epsilon \int_{\mathcal{Z}} \ell(z, h) dP(z) \tag{4}$$

We consider a ML model is $(\epsilon, \delta) - distributionally\ robustly\ fair\ (DRF)$ WRT the fair metric $d_x\ iff$

$$\max_{P:W(P,P_n)} \leq_\epsilon \int_{\mathcal{Z}} \ell(z,h)dP(z) \leq \delta \tag{5}$$

And (5) has an existed solution by Blanchet & Murthy, which is the dual of equation shown below,

$$\sup_{P:W(P,P_n)\leq\epsilon} \mathbb{E}_P\,[l(Z,h)] = \inf_{\lambda\geq 0}\{\lambda\epsilon + \mathbb{E}_{P_n}[l_\lambda^c(Z,h)]\},$$
$$l_\lambda^c\big((x_i,y_i),h\big) \triangleq \sup_{x\in\chi} l\big((x,y_i),\theta\big) - \lambda d_x(x,x_i). \tag{6}$$

They propose the sensitive subspace robustness (SenSR) algorithm to solve this DRF optimization problem.

2.3. Sensitive subspace robustness algorithm (SenSR)

From (6), the author proposed solving the minmax problem for individual fairness, that is to minimize the equation

$$inf_{h\in\mathcal{H}} \sup_{P:W(P,P_n)\leq\epsilon} \mathbb{E}_P\,[l(Z,h)] = inf_{h\in\mathcal{H}} \inf_{\lambda\geq 0} \lambda\epsilon + \mathbb{E}_{P_n}[l_\lambda^c(Z,h)] \tag{7}$$

which means learning a classifier that is insensitive to pertubations along the horizontal (i.e. sensitive) direction. Here they used adversarial training (Madry et al., 2017) to solve equation 7 (see algorithm 2).

Algorithm: Sensitive Subspace Robustness (SenSR)

Require: starting point $\hat{\theta}_1$, step sizes $\alpha_t, \beta_t > 0$

1. **Repeat**
2. Sample mini-batch $(x_1,y_1), \ldots, (x_B,y_B) \sim P_n$
3. $x_{t_b}^* \leftarrow argmax_{x\in\mathcal{X}} l\left(\left(x,,y_{t_b}\right),\theta\right) - \hat{\lambda}_t d_x\left(x_{t_b},x\right), b \in [B]$
4. $\hat{\lambda}_{t+1} \leftarrow \max\,[0, \hat{\lambda}_t - \alpha_t(\epsilon - \frac{1}{B}\sum_{b=1}^{B} d_x(x_{t_b}, x_{t_b}^*))\}$
5. $\hat{\theta}_{t+1} \leftarrow \hat{\theta}_t - \frac{\beta_t}{B}\sum_{b=1}^{B} \partial_\theta l\left(\left(x_{t_b}^*, y_{t_b}\right), \hat{\theta}_t\right)$
6. **Until** converged

3 Experiments

Here, we implemented SenSR into two steps. First, we replicated one of experimental results in the Mikhail, Amanda et. al 2020 paper. Since ML models don't even need to see sensitive information as features, so they considered two situations: one for problems in which the sensitive attribute is reliably observed (income prediction), and another for problems in which the sensitive attribute is unobserved (sentiment analysis). It turned out that SenSR is able to handle both situations. Here we only showed sentiment analysis result. Second, we built a pipeline to generate a test set appropriate for fairness validation based on partially cleaved samples. We would like to test the partially cleaved samples randomly classified into cleaved and uncleaved class by the trained ML model and test invariance among natural groups of the partially cleaved samples.

3.1 Sentiment Analysis

Sentiment analysis refers to classifying the sentiment of words using positive and negative words. Taking into account that names should have no sentiment intuitively, we could evaluate whether SenSR helps satisfy the individual fairness by analyzing test results on names using the sentiment ML model. In the paper, they trained words with 300-dimensional GloVe word representations using a one layer neural network. First, we trained with the same model architecture on the same data, splitting into 90% training data, 10% validation data and the model reached 95% accuracy, which matched the statement in the paper. Afterwards, we tested the trained model on 94 names and reached 94% test accuracy, which also matched the result in the paper. Based on the test result, race and sex biases showed in **Figure 1**. It turned out that black people will be less likely recommended by the system than white people, while female will be more likely recommended by the system than male people.

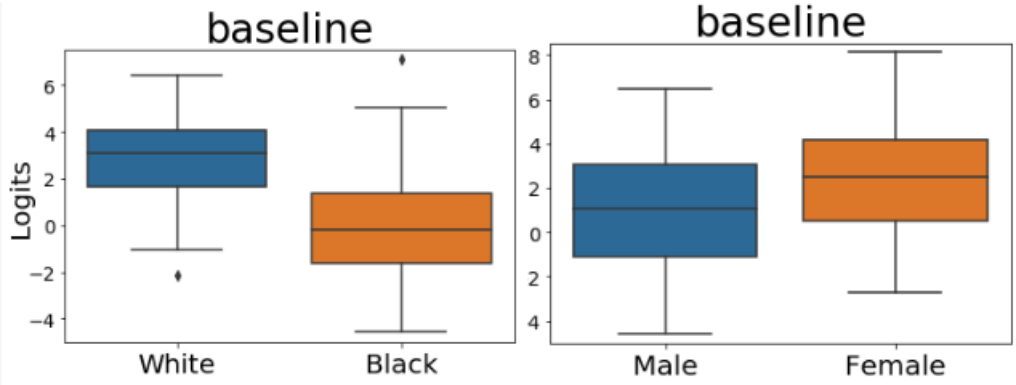

Figure 1. Test logits for the race group and the sex group of the names.

Next, we considered GloVe embeddings of 94 names as sensitive directions, indicating prior expert knowledge, that should be treated equally. Fair metric was then defined using an orthogonal complement projector of the span of sensitive directions as we discussed in **Section 2. Method**. Furthermore, a broader set of baby names in New York City was also used as sensitive directions to simulate the situation that we do not know expert knowledge. **Table 1** shows train and test accuracies of these two cases for SenSR. Results of SenSR when expert knowledge is accessible and not accessible were presented in **Figure 1** and **2**, respectively. We could see that accuracy dropped from 95% to around 93% when we compared SenSR with the baseline model in **Table 1**, but either race or sex group had similar sentiment values, from which we could conclude that SenSR further improved both group and individual fairness indeed. An interesting phenomenon to address is that, using baby names in NYC as the sensitive direction reached higher accuracy than using expert knowledge. Since expert knowledge only contains 94 embeddings, while the large amount of baby names, we could guess whether sensitive data is rich or not matters SenSR's performance.

Table 1. Train and test accuracy results with/without expert knowledge.

	Train Accuracy	**Test Accuracy**
94 Names (SenSR)	92.76	92.31
Baby Names in NYC (SenSR)	93.85	93.21

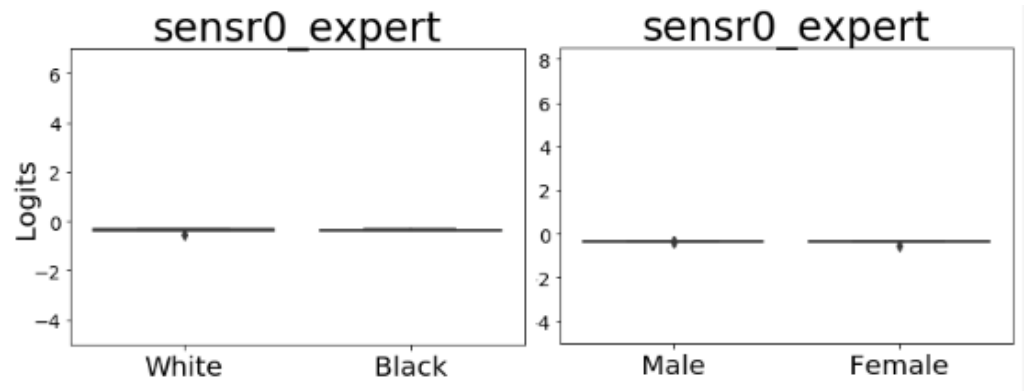

Figure 2. SenSR results when expert knowledge is accessible.

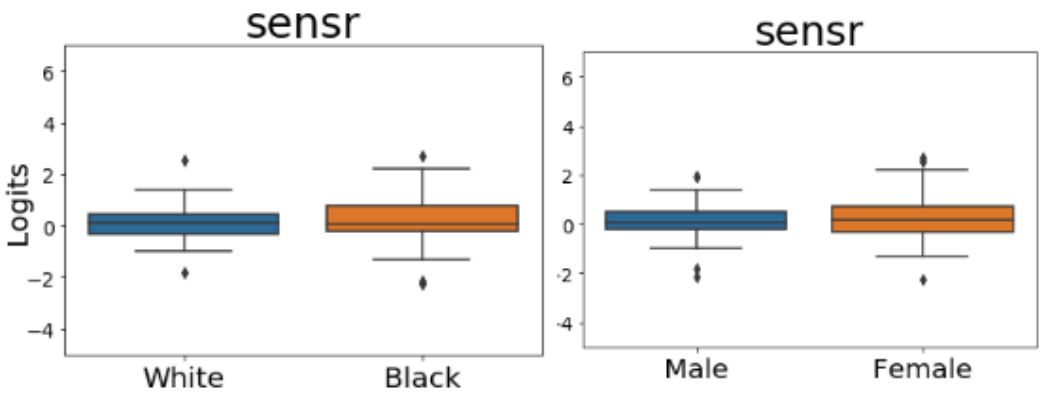

Figure 3. SenSR results when expert knowledge is not accessible.

4. Real Experiments

After validating the SenSR algorithm, we would like to further implement it to our biology problem. We could introduce a simple analogy of partially cleaved problem with sentiment analysis. Partially cleaved substrates could be naturally classified into two groups, according to 'biological ethnicity'. Based on biology sense, they do not belong to either cleaved or uncleaved class and ML should treat either natural group the same if ML model is designed for binary classification of cleavage. Therefore, 'sentiment score' of two groups should be equivalent.

The embedding of sample features is a 20-binary-numbers-long one hot vector multiplying the number of positions of the sequence. Here, all substrates are 5-amino-acid long, so sequential embedding is a 100-binary-numbers vector. Again, another four total energy terms of the protease/peptide/interface qualify the problem regarding discovering protein-protein interaction. The trickiest part of partially cleaved data is that we do not know what the natural groups of the data is. Therefore, we simply use KMeans to cluster partially cleaved samples into two clusters, indicating the natural groups of partially cleaved samples. KMeans clustered samples into two groups with the ratio of nearly 3:1, as shown in **Figure 4**. In order to see fairness on the partially cleaved direction, we first trained the one layer neural network model on cleaved and uncleaved samples, along with the partially cleaved direction of 104-dimensional embeddings. Then we tested partially samples using the trained model to achieve logits information, as shown in **Figure 5**. In this setting, we followed the same SenSR implementation procedure in **Section 3**, and compared SenSR result with the baseline model, as shown in **Figure 6**. We could see SenSR improves variances of either class significantly.

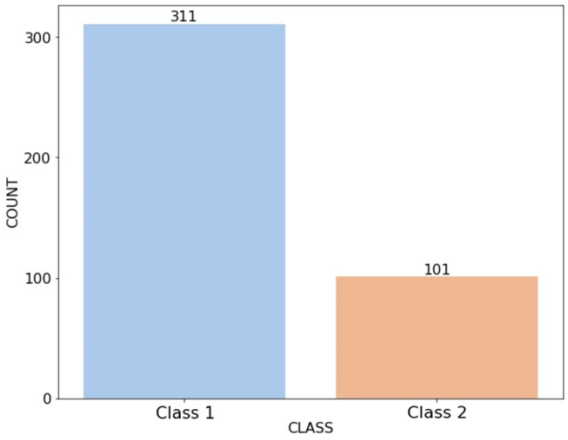

Figure 4. Barplot of clustering results for partially cleaved samples

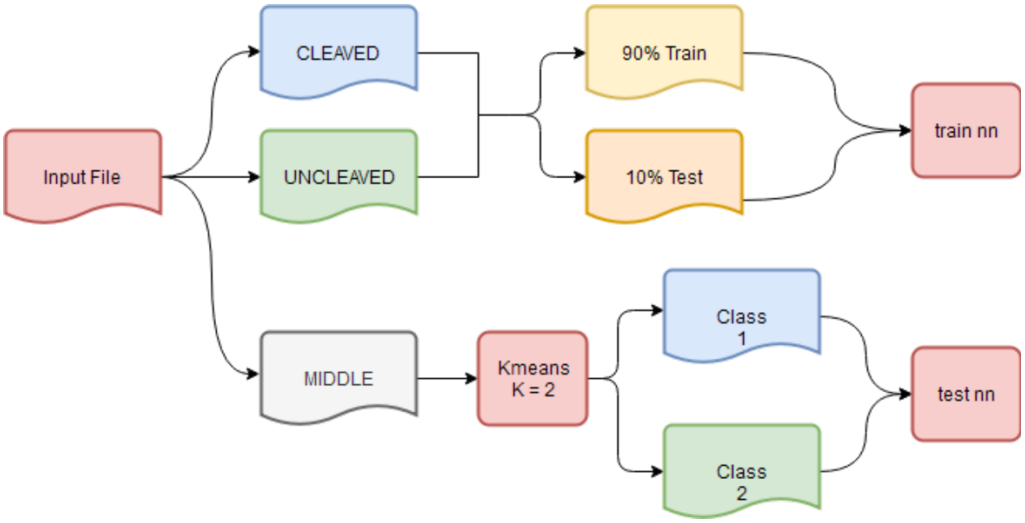

Figure 5. Flowchart of biology implementation

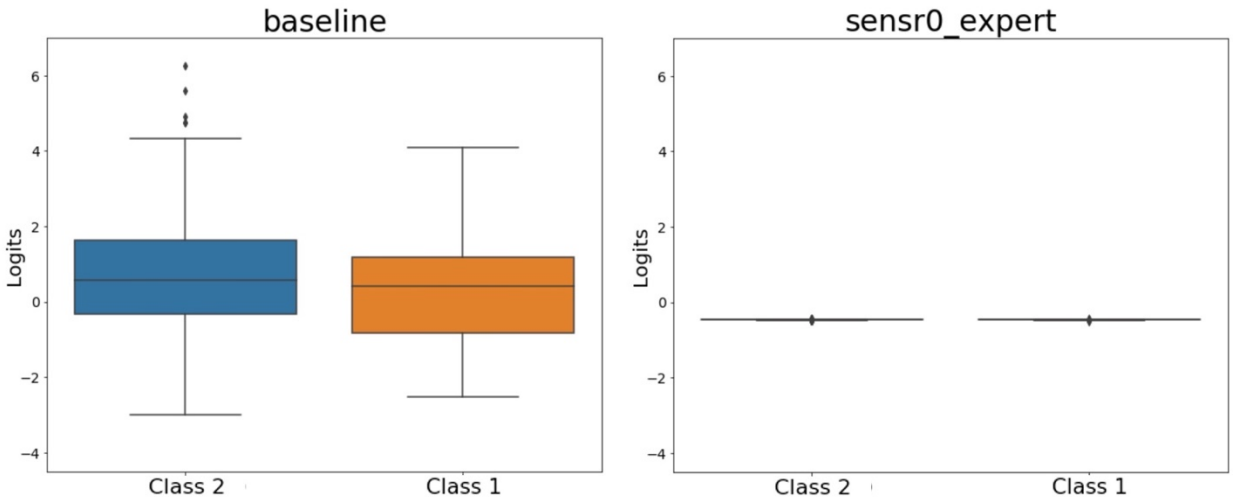

Figure 6. SenSR result compared with baseline model on partially cleaved samples

5 Conclusion

Herein, we point out the importance of ML fairness validation using biology knowledge, i.e., protease specificity prediction. ML fairness is an important feature to avoid high false positive rate and volatile predicted patterns among different protein systems. Inspired by SenSR, a distributionally robustly fair optimization approach to operate fair metric along with the sensitive direction, which indicates utilizing the expert knowledge on prediction to some extent, we first replicated one of the experiments in the paper and concluded that we built the algorithm well and reached exactly the same results provided in the paper. Next, we formalized our problem into SenSR and reached significantly good results.

In the present study, the highlight of our work is fairness on a structural biology problem. Most of fairness work focus on race, sex, ethnicity biases, however more biases are hidden in biology, physics, etc. areas. It is challenging to define biases in these areas and most importantly, data of these areas are far more complicated to analyze. Nevertheless, this makes fairness on biology/physics more interesting and unique so that we could put forward some questions like, how to improve fairness model when expert knowledge is hidden inside the data, etc.

6 Availability and Implementation

All codes are available at https://github.com/Nucleus2014/fairness_for_protease_specificity.

References

[1] John A. Tainer and Richard P. Cunningham. Molecular recognition in dna-binding proteins andenzymes. Current Opinion in Biotechnology, 4:474–483, 08 1993.

[2] Fuyi Li, Yanan Wang, Chen Li, Tatiana T. Marquez-Lago, Andr´e Leier, Neil D. Rawlings, Gho-lamreza Haffari, Jerico Revote, Tatsuya Akutsu, Kuo-Chen Chou, Anthony W. Purcell, Robert N.Pike, Geoffrey I. Webb, A. Ian Smith, Trevor Lithgow, Roger J. Daly, James C. Whisstock, andJiangning Song. Twenty years of bioinformatics research for protease-specific substrate and cleav-age site prediction: A comprehensive revisit and benchmarking of existing methods. Briefings in Bioinformatics, 20:2150–2166, 11 2019.

[3] Michael D. Tyka, Daniel A. Keedy, Ingemar Andr´e, Frank DiMaio, Yifan Song, David C. Richard-son, Jane S. Richardson, and David Baker. Alternate states of proteins revealed by detailed energylandscape mapping. Journal of Molecular Biology, 405:607–618, 01 2011.

[4] Matej Vizovi˘sek, Robert Vidmar, Marcin Drag, Marko Fonovi´c, Guy S. Salvesen, and Boris Turk. Protease specificity: Towards in vivo imaging applications and biomarker discovery. Trends in Biochemical Sciences, 43:829–844, 10 2018.

[5] Lawrence J. K. Wee, Tin Wee Tan, and Shoba Ranganathan. Casvm: Web server for svm-basedprediction of caspase substrates cleavage sites. Bioinformatics, 23:3241–3243, 12 2007.

[6] Jiangning Song, Hao Tan, Hongbin Shen, Khalid Mahmood, Sarah E. Boyd, Geoffrey I. Webb, Tat-suya Akutsu, and James C. Whisstock. Cascleave: Towards more accurate prediction of caspasesubstrate cleavage sites. Bioinformatics (Oxford, England), 26:752–760, 03 2010.

[7] Jiangning Song, Hao Tan, Andrew J. Perry, Tatsuya Akutsu, Geoffrey I. Webb, James C. Whisstock,and Robert N. Pike. Prosper: An integrated feature-based tool for predicting protease substratecleavage sites. PLoS ONE, 7:e50300, 11 2012.

[8] Jiangning Song, Fuyi Li, Andr´e Leier, Tatiana T Marquez-Lago, Tatsuya Akutsu, Gholamreza Haf-fari, Kuo-Chen Chou, Geoffrey I Webb, and Robert N Pike. Prosperous: High-throughput pre-diction of substrate cleavage sites for 90 proteases with improved accuracy. Bioinformatics, 34:684–687, 10 2017.

[9] Jiangning Song, Yanan Wang, Fuyi Li, Tatsuya Akutsu, Neil D Rawlings, Geoffrey I Webb, andKuo-Chen Chou. Iprot-sub: A comprehensive package for accurately mapping and predictingprotease-specific substrates and cleavage sites. Briefings in Bioinformatics, 20:638–658, 04 2018.
[10] Manasi A. Pethe, Aliza B. Rubenstein, and Sagar D. Khare. Large-scale structure-based predictionand identification of novel protease substrates using computational protein design. Journal of Molecular Biology, 429:220–236, 01 2017.
[11] Manasi A. Pethe, Aliza B. Rubenstein, and Sagar D. Khare. Data-driven supervised learning of a viral protease specificity landscape from deep sequencing and molecular simulations. Proceedingsof the National Academy of Sciences, 116:168–176, 12 2018.
[12] Mikhail Yurochkin, Amanda Bower, Yuekai Sun. Training individually fair ML models with sensitive subspace robustness, ICLR 2020.

Appendix 1 Table for quartiles of real experiments

	Class 1				Class 2			
	Q1	Q2	Q3	IQR	Q1	Q2	Q3	IQR
Baseline	-0.33	0.54	1.63	1.96	-0.72	0.40	1.29	2.01
SenSR	-0.38	-0.38	-0.38	0.01	-0.39	-0.39	-0.38	0.01