Debiasing Algorithms for Protein Ligand Binding Data do not Improve Generalisation

Vikram Sundar and Lucy Colwell*

Department of Chemistry, University of Cambridge, Cambridge, UK

E-mail: ljc37@cam.ac.uk

Abstract

The structured nature of chemical data means machine learning models trained to predict protein-ligand binding risk overfitting the data, impairing their ability to generalise and make accurate predictions for novel candidate ligands. To address this limitation, data debiasing algorithms systematically partition the data to reduce bias. When models are trained using debiased data splits, the reward for simply memorising the training data is reduced, suggesting that the ability of the model to make accurate predictions for novel candidate ligands will improve. To test this hypothesis, we use distance-based data splits to measure how well a model can generalise. We first confirm that models perform better for randomly split held-out sets than for distant held-out sets. We then debias the data and find, surprisingly, that debiasing typically reduces the ability of models to make accurate predictions for distant held-out test sets. These results suggest that debiasing reduces the information available to a model, impairing its ability to generalise.

Introduction

The accurate identification of ligands that bind tightly and specifically to a given protein target is a crucial step in drug discovery. Experimental high-throughput screening is expensive, time-consuming, and far from comprehensive due to the numerous potential ligands in chemical space.[1,2] Physics-based methods such as docking and Molecular Dynamics can be inaccurate and are computationally expensive.[3–5] Databases such as ChEMBL[6] are growing rapidly, increasing the popularity of machine learning (ML)-based approaches to molecular property prediction.[7,8] In this paper, we focus on data-driven approaches to virtual screening, i.e. using data from high-throughput screening experiments to train models that predict whether ligands will have activity against a particular protein target.[9]

Many recent ML approaches have achieved outstanding success on benchmark datasets that are randomly partitioned into train and validation sets, with AUCs (area under the Receiver Operator Characteristic curve) routinely exceeding 0.9.[10–17] However, it is unclear whether this impressive performance indicates that a model that can truly generalize across chemical space, or instead simply overfits the training data.[18–23] Since chemical space contains clusters of molecules around scaffolds, memorizing the properties of a few scaffolds can be sufficient to perform well, masking the fact that the model may not generalize beyond close analogues.[24,25] Further, molecules tested experimentally are generally designed by humans and therefore likely to be easy to synthesize and similar to previously known binders.[19,26]

To counter this limitation, data bias definitions and corresponding debiasing algorithms have been introduced.[20,22,27,28] Two popular bias measures, Maximum Unbiased Validation (MUV) and Asymmetric Validation Embedding (AVE), are illustrated in Fig. 1.[20,22] In each case the bias measure is used by a genetic algorithm to rearrange the train/validation split such that the bias is reduced. Specifically, MUV ensures that active ligands are uniformly embedded among inactive ligands according to some distance metric, while AVE adds the requirement that inactive ligands are not tightly clustered. Across benchmark datasets for multiple protein targets, both bias metrics were shown to correlate to model performance, suggesting that heavily biased datasets provide a falsely optimistic picture of the predictive ability of the trained model. Furthermore, debiasing was shown

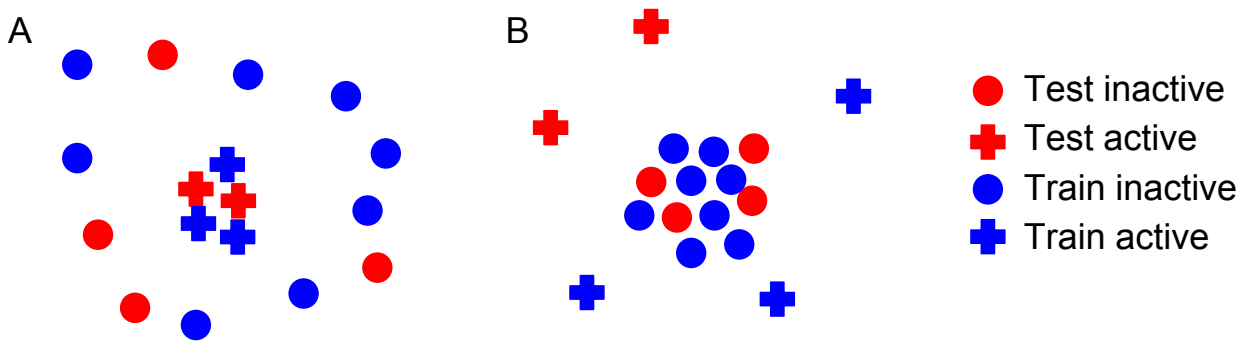

Figure 1: Definition of AVE and MUV Bias. (a) This set is considered biased by MUV because actives are clustered and not uniformly embedded in the inactive decoys. (b) This set is considered biased by AVE, because the inactives are tightly clustered relative to the active-inactive distance.

to decrease classification accuracy for the debiased validation set, presumably because the resulting models could not perform well by simply memorising the training data.[20,22]

The purported advantage of models trained using debiased data splits is that they overfit less, so generalise better to make accurate predictions for novel candidate ligands. In this paper, we develop a framework to measure generalisation ability, and explicitly test this hypothesis. We assemble data for 189 targets with > 500 reported active ligands, and split each dataset into a train set and a distant held-out test set used to define the far-AUC, a metric of model generalisation (see Fig. 2a). We then randomly split the train set to produce a random held-out validation set, which is used to measure the standard AUC. Despite achieving state-of-the-art AUCs on the held-out validation sets, our trained ML models struggle to generalise effectively when challenged with the distant held-out test sets. We then apply MUV and AVE debiasing to our 189 random train/validation splits, and use the resulting debiased train sets to build new models. We find that counter to the stated aim, the debiased models make less accurate predictions for novel candidate ligands, as illustrated by their performance on the distant held-out test sets.

Methods

We filter protein-ligand binding data from ChEMBL 24.1,[6,29] acquiring active ligands (IC_{50}, K_i, K_d, or EC_{50} of less than 1 µM) for each protein target. We acquired data for inactive ligands from PubChem indexed by UniProt Protein ID.[30,31] The handful of cases where ligands were marked both active and inactive by different assays were eliminated. Some targets have fewer inactives than actives, in which case we randomly drew inactives from ChEMBL to achieve an even split for every target. This procedure was repeated every time the algorithm was run, contributing to the error bars shown in the figures.

We use ECFP6 fingerprints with 2048 bits as the feature set for all models.[32] For consistency we use Tanimoto similarity as the distance metric throughout, computed as $1 - d$ where d is the Jaccard distance metric from Scipy[33] (note this differs from the specific metric used in MUV[20]). We first randomly split both the actives and inactives for each protein target into a 70% set and a 30% set. Each 30% set is filtered to build distant held-out test-sets that contain approximately 10% and approximately 25% of the total active and inactive ligands respectively, wherein all ligands are at least 0.4 from every ligand in the 70% set. Our far-AUC metric measured the performance of models on this distant held-out test set, which only contains molecules distant from any molecules (active or inactive) that the model has previously seen.

We further randomly split the remaining 70% set for each target into train (80%) and validation (20%) sets and train Naive Bayes, Logistic Regression, and Random Forest models using these data splits. All models used were implemented with scikit-learn.[34] We use no prior for Naive Bayes, $C = 1$ for Logistic Regression, and 100 trees with a maximum depth of 10 for Random Forest. We do not tune model hyperparameters since the focus of our work is the debiasing algorithms. The 3-way train/validation/test split and the requirement for data points far from the training set restricted us

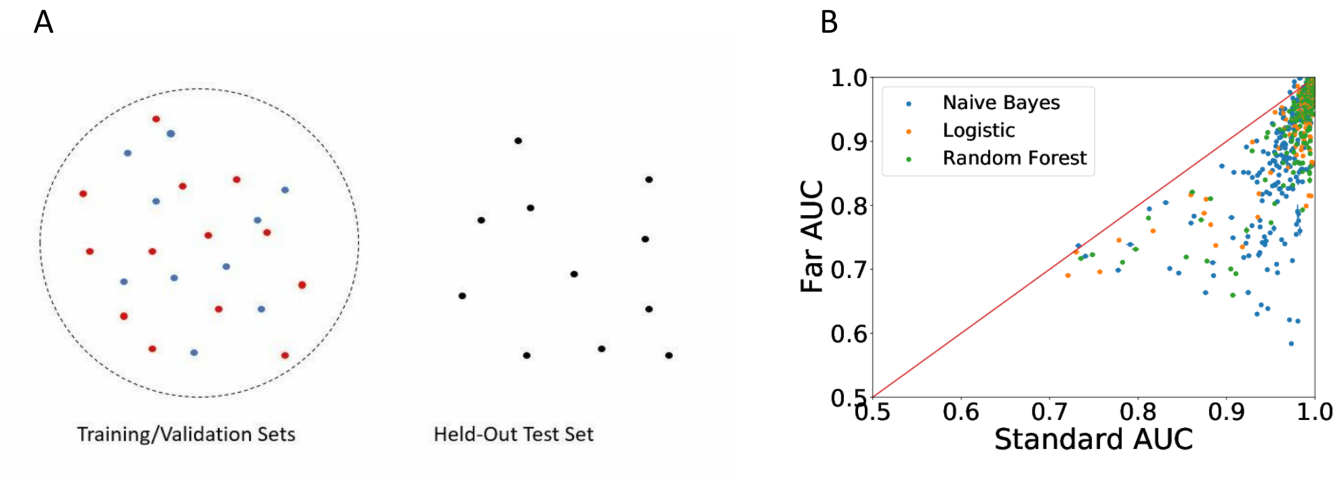

Figure 2: (a) Definition of the far-AUC, the AUC using a held-out test set with minimum distance of 0.4 from all elements of the training or validation set. (b) Comparison of the standard model AUC and the far-AUC, a measure of generalization. Model predictions are less accurate for the distant held-out test set than for the validation set, showing that these models do not generalise.

to the 189 protein targets with > 500 active ligands. All AUCs are measured over 50 replicates to determine error bars. Randomness comes from the train/validation/test split and (where needed) the selection of inactives. All error bars are SEM and indicated to 1 σ precision.

We follow the original definitions of AVE and MUV bias.[20,22] Specifically, given a set V of validation molecules and T of training molecules with a similarity threshold $d \in [0, 1]$, we define a nearest-neighbor function

$$S_{(V,T,d)} = \frac{1}{||V||} \sum_{v \in V} I_d(v, T) \tag{1}$$

where $I_d(v, T) = 1$ if the distance from the validation molecule v to its nearest neighboring training molecule is smaller than d. We then define a distance function on sets

$$H_{(V,T)} = \frac{1}{||D||} \sum_{d \in D} S(V, T, d) \tag{2}$$

with $D = \{0, 0.01, \ldots, 1\}$. For convenience, let V_a, V_i, T_a, T_i be the sets of validation actives, validation inactives, test actives, and test inactives, respectively. Then the AVE bias is defined as

$$B_{\text{AVE}} = H_{(V_a,T_a)} - H_{(V_a,T_i)} + H_{(V_i,T_i)} - H_{(V_i,T_a)} \tag{3}$$

and the MUV bias is defined as

$$B_{\text{MUV}} = H_{(T_a,T_a)} - H_{(T_a,T_i)} + H_{(V_a,V_a)} - H_{(V_a,V_i)}. \tag{4}$$

To minimise MUV or AVE bias we use the implementation developed in,[22] which largely follows that in.[20] A number of random initial train/validation splits are generated; then the mutation phase of the genetic algorithm involves randomly merging two train/validation splits, moving compounds between the training and validation sets, and deleting compounds from either set. The algorithm ran for 300 iterations or until bias was < 0.01, whichever occurred first, and then produced the least biased split.

Results

We first use our framework to test the extent to which ML models are able to generalise and make accurate predictions for novel candidate ligands. To assess this we compare the ability of each trained model to accurately classify ligands in (i) the random held-out validation set, reported by the standard AUC and (ii) the distant held-out test set, reported by the far-AUC. The distant held-out test sets mimic the real-world need to make accurate predictions for novel candidate ligands that are distinct from the training data. The results of our analysis for Naive Bayes, Logistic Regression and Random Forest models are shown in Fig. 2b. We find that the far-AUC is significantly lower than the

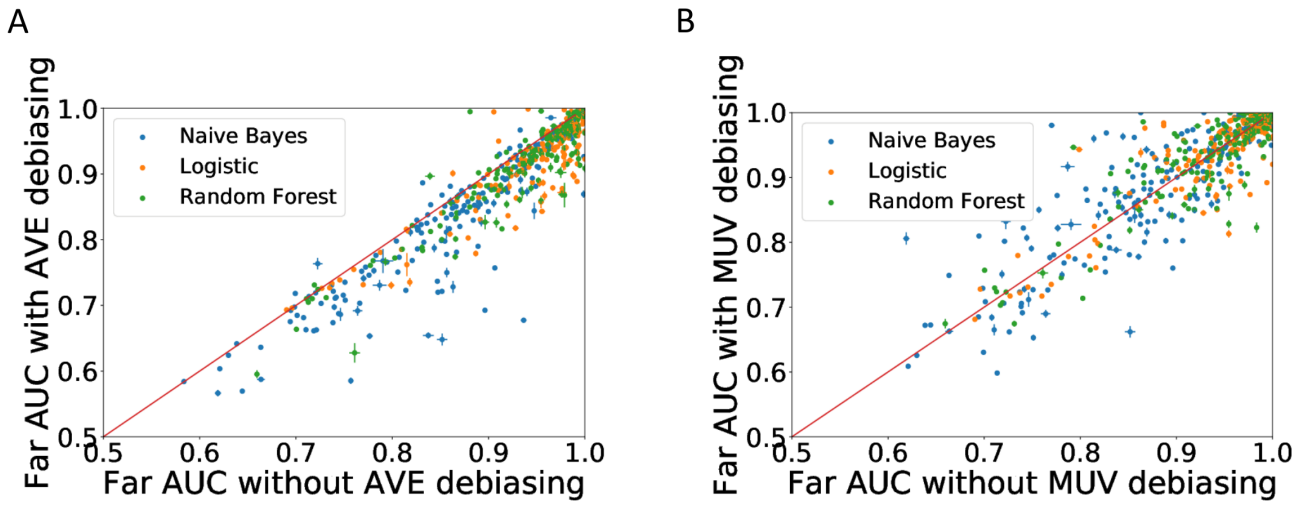

Figure 3: Impact of debiasing on the far-AUC; points below the diagonal indicate targets for which debiasing decreased the far-AUC. (a) AVE debiasing reduces the ability of the models to generalise. (b) MUV debiasing does not consistently help the models to generalise. These results suggest that debiasing does not improve the ability of the models to generalise to novel candidate ligands.

standard AUC for all models tested across all 189 target datasets, indicating that our models do not generalise well. This confirms the hypothesis that generalisation is a challenge for ML models trained on protein/ligand binding datasets. Although we have not tested more complex models, it is likely that as the complexity of the ML model increases, overfitting to the training data will increase.

We next assess the extent to which MUV and AVE debiasing alleviate this issue. The key question is whether these algorithms remove data biases that prevent the models from generalising or instead remove useful information that is necessary for the model to learn. This is a nontrivial question, as both MUV and AVE rely on the inherent assumption that the metrics being used to measure distances between ligands do not correlate strongly with the binding activity of the ligand. The ultimate goal of a ML model is to learn a function of the data features that distinguishes ligands that bind to a given protein from those that don't. In contrast, the goal of debiasing is to ensure that actives and inactives are well-mixed according to some distance metric, which is itself a function of the data features. If this distance metric happens to be correlated with the physico-chemical criteria required for binding, then debiasing will remove important information from the training data, potentially harming the performance of the model.

To address this question we use the distant held-out test sets to compare the existing models with versions trained using debiased train/validation splits. We evaluate the effect of debiasing by computing the change in the far-AUC score between the models trained on the debiased data, and the models trained on the original data for each target. As shown in Fig. 3, neither AVE nor MUV debiasing improve the generalisation ability of the trained models. On average, AVE decreases the far-AUC of Logistic Regression by 0.024 ± 0.030 (mean $\pm$ standard deviation) and that of Random Forests by 0.021 ± 0.029, while MUV debiasing decreases the far-AUC of Logistic Regression by 0.004 ± 0.034 and of Random Forest by 0.002 ± 0.036.

To check whether the far-AUC depends on the extent to which the data were debiased by either MUV or AVE, Figs. 4a and 4b show the change in far-AUC as a function of the final dataset bias achieved by AVE and MUV respectively. We find no correlation in each case. Supplementary Fig. S1 confirms that this still holds if we instead consider the change in MUV bias, to account for the initial MUV bias measured.

We further probe if the number of active ligands for a target indicates whether debiasing will prove effective. Figs. 4c and 4d show no correlation between the number of active ligands and the change in far-AUC achieved by debiasing. However, the magnitude of the change in far-AUC decreases slightly as the number of active ligands increases, suggesting that debiasing has a smaller effect for large datasets. In addition, we checked

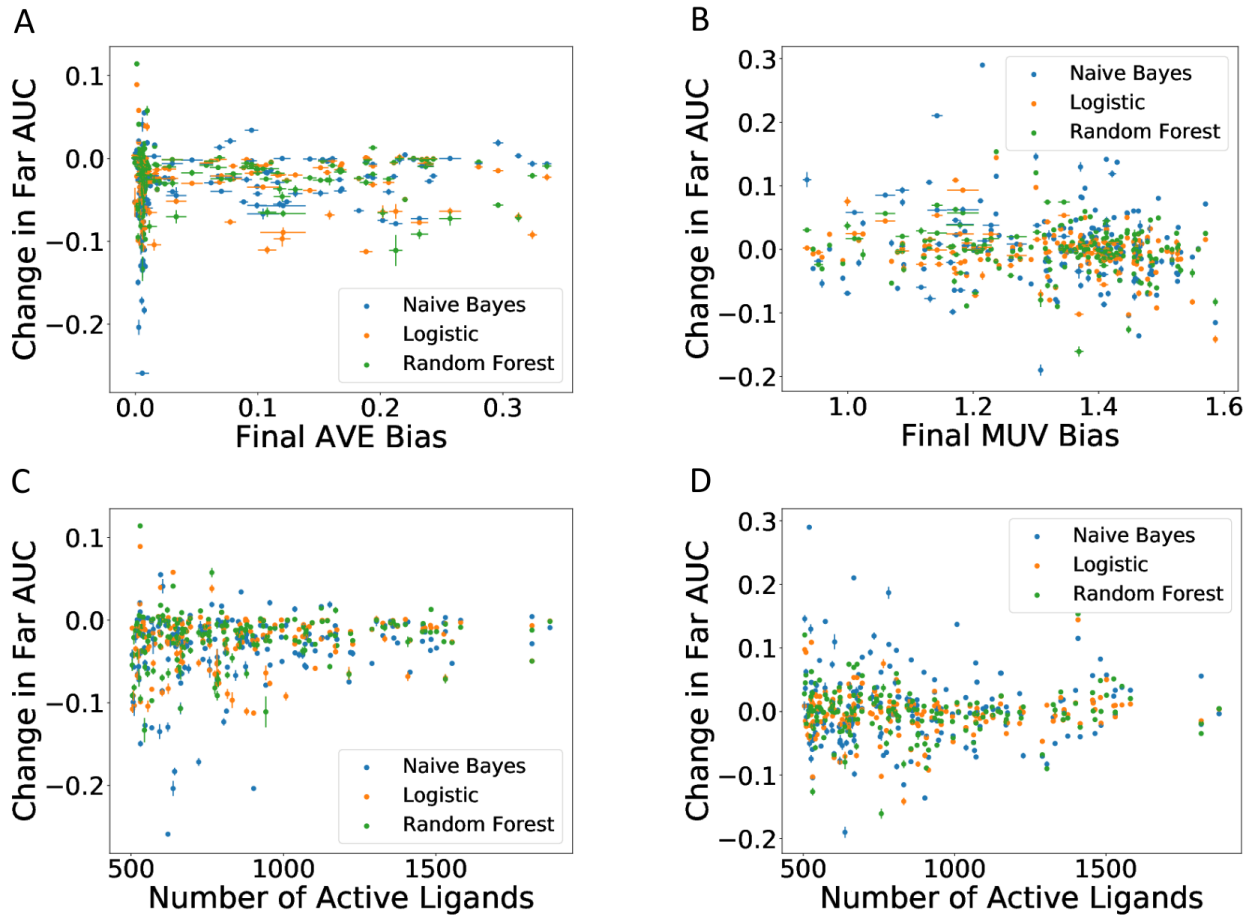

Figure 4: Relationship between the change in far-AUC achieved by debiasing and either (a) the final AVE bias or (b) the final MUV bias; we find no correlation. Furthermore we see no improvement in the resulting generalisability as a function of the number of active ligands per target for (c) AVE or (d) MUV debiasing. These results suggest that neither approach has much effect for large datasets.

whether there was any relationship between the change in the far-AUC and the number of decoy molecules added for a given target to achieve equal numbers of active and inactive ligands. Supplementary Fig. S2 confirms that there is no correlation.

To better understand these findings, and ask whether there are situations in which debiasing does improve the ability to generalise, we examine the data generated by our experiment in more detail. We first examine far-AUC trajectories measured during debiasing. Supplementary Figs. S3 and S4 suggest that there is no positive correlation between the far-AUC achieved and reduction of either the MUV or AVE bias. The jagged behaviour seen for both algorithms suggests that certain moves that significantly decrease the far-AUC are sometimes preferred.

We hypothesised that moves in these trajectories that decrease the far-AUC correspond to data deletion, a move allowed by the genetic algorithm. For example, Supplementary Fig. S5 shows how the size of the dataset for ChEMBL 5508 decreases during the debiasing process. To evaluate debiasing in the absence of data deletion, we modified the genetic algorithm to prevent data from being deleted. Supplementary Fig. S5 shows that the modified versions of both MUV and AVE debiasing still succeed at reducing the bias of the dataset. Could this provide an approach that better enables the models to generalise? To test this we repeated our earlier analysis with the modified debiasing algorithms.

As shown in Fig. 5, we find that forbidding deletion does slightly improve the generalisation ability of the resulting debiased models for both AVE and MUV compared to the versions with deletion. However, even without deletion, AVE decreases the

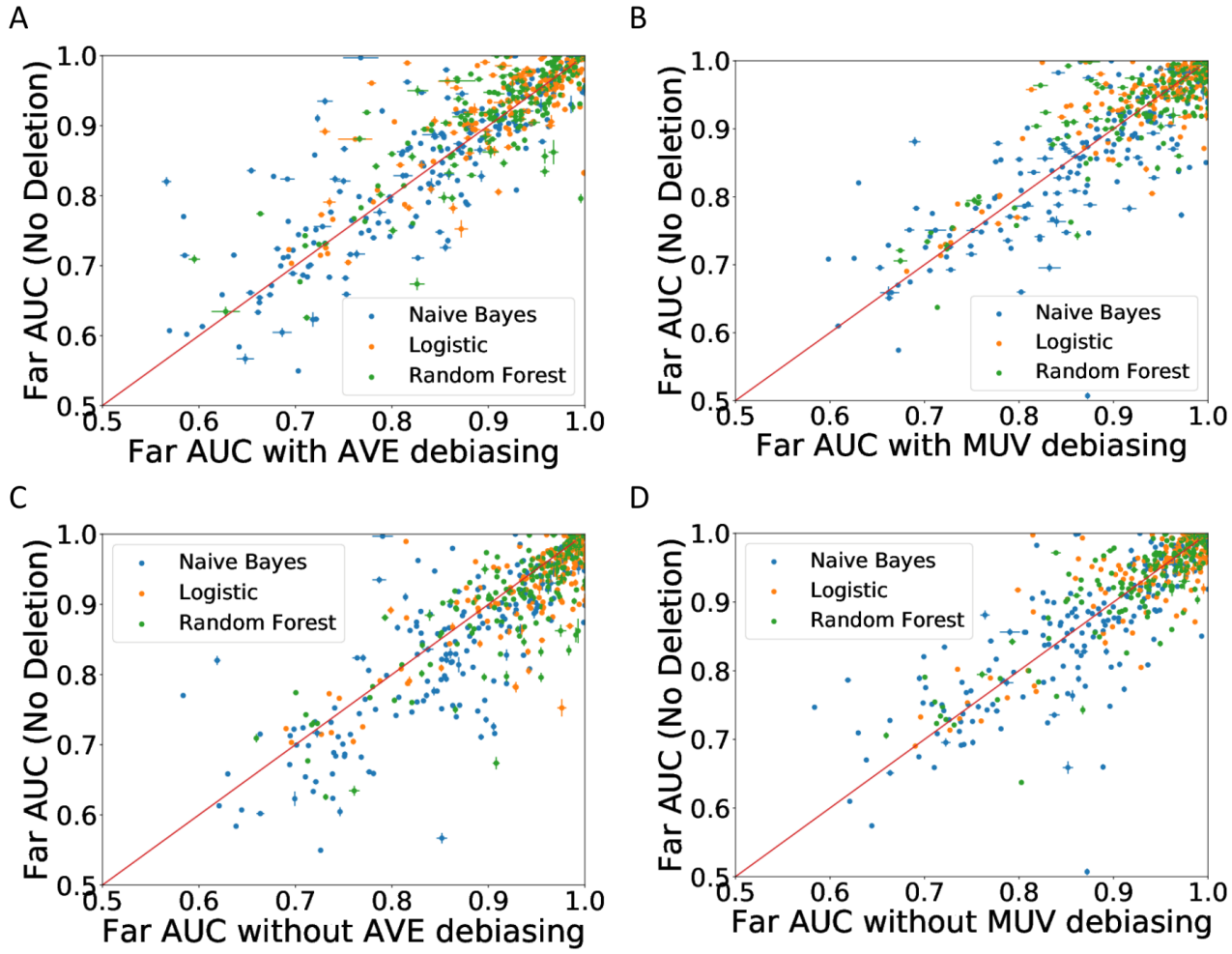

Figure 5: Debiasing without deletion. When deletion is forbidden for the (a) AVE or (b) MUV debiasing algorithms, performance improves slightly over that obtained with the standard version of each algorithm. However, (c) and (d) show that for both algorithms, debiasing without deletion is worse, on average, then the performance obtained by the models before debiasing.

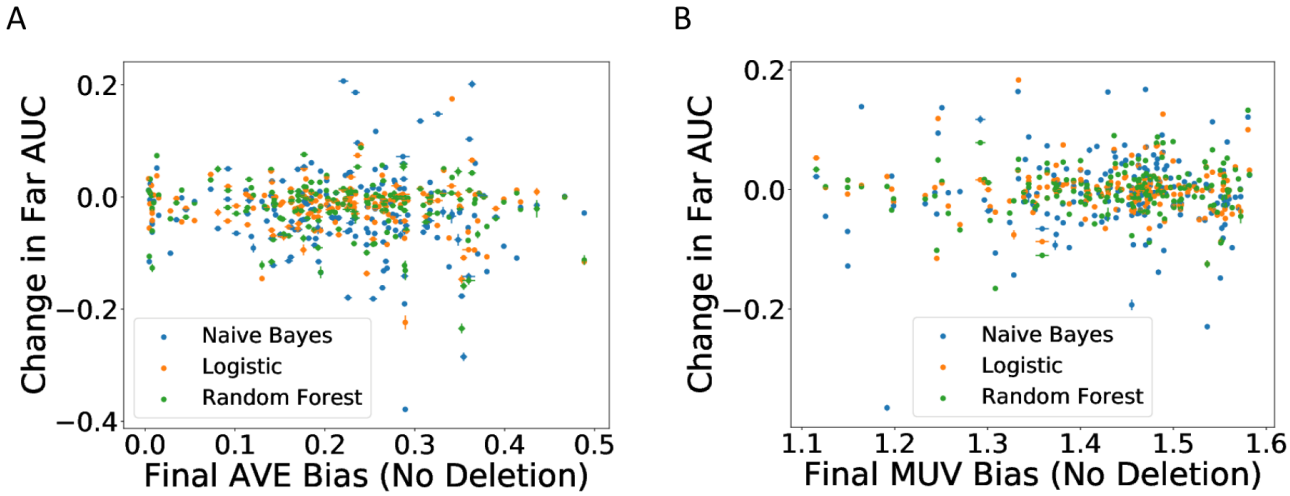

Figure 6: There is no clear relationship between the change in far-AUC achieved when deletion during debiasing is forbidden and either (a) the final AVE bias or (b) the final MUV bias.

far-AUC of Logistic Regression by 0.016 ± 0.041 and of Random Forest by 0.017 ± 0.044 on average across the 189 datasets, while MUV debiasing decreases the far-AUC of Logistic Regression by 0.0006 ± 0.037 and of Random Forest by 0.001 ± 0.038. As in the case where deletion is allowed, Figs. 6a and 6b confirm that there is no relationship between the extent to which the data were debiased, and the change in far-AUC obtained.

Overall, our results indicate that it is difficult to predict when AVE or MUV debiasing will improved generalisation. Further work is needed to determine the conditions under which debiasing has the potential to improve the ability of a model to generalise and make accurate predictions for novel candidate ligands.

Discussion

In this paper we develop a simple far-AUC metric that measures the ability of a protein-ligand binding model to generalise and make accurate predictions for novel candidate ligands. We use this metric to evaluate the AVE and MUV debiasing algorithms that were designed to reduce overfitting to the training data, and thus potentially improve the ability of models to generalise. Our analysis for both AVE and MUV debiasing in Fig. 3 finds that debiasing does not systematically improve the ability of the trained models to generalise, despite the fact that Fig. 2b shows there is significant room for improvement.

This suggests that debiasing algorithms are not able to accurately distinguish signal from bias, and in many cases remove relevant information from the training data. Our analysis of the debiasing trajectories suggests that the deletion operation used by the genetic algorithm in both MUV and AVE debiasing may exacerbate this loss of useful information or signal from the data. To address this we implemented versions that did not allow data points to be deleted. This did not result in models that were better able to generalise compared to those built without debiasing.

Dataset bias is clearly an important issue, particularly in chemistry; however, current debiasing approaches need to be applied carefully to ensure that they do not eliminate relevant information. Indeed, some clustering among actives is to expected in fingerprint space, since active ligands for a given protein can have structurally similar features. It is important to distinguish between this clustering and artificial clustering that may result from the fact that only portions of chemical space have been explored by synthetic chemists, or other potential sources of bias.

Another approach is to better understand the regions of chemical space in which protein-ligand binding models trained using a particular dataset are able to make accurate predictions. Generalisation is challenging for ML models across many contexts, even when trained with unbiased datasets, so given the highly biased nature of chemical data, expecting protein-ligand binding models to generalise may be ambitious. Methods that establish the domain of applicability for trained models need to be developed to provide confidence in those predictions that fall within this domain. This approach would have the advantage of avoiding the information/bias distinguishability problem described above while still allowing the resulting models to generalise to some degree.

Acknowledgement V.S. acknowledges support from the Winston Churchill Foundation of the USA. Computations for this project were run on the ziggy cluster at Centre for Molecular Informatics, Chemistry Department, University of Cambridge. L.J.C. acknowledges support from the Simons Foundation.

Supporting Information Available: The supplement includes figures showing the relationship between the change in far-AUC and the change in MUV bias, as well as the number of decoy molecules added. It also shows how the far-AUC changes during the process of running the debiasing algorithm and contains an analysis of how dataset bias depends on the size of the dataset and demonstrates the debiasing without deleting data points is possible. Code necessary for generating the dataset used in this paper, running the debiasing algorithm, and splitting the dataset to compute the far-AUC for an arbitrary model has also been provided. This material is available free of charge via the Internet at `http://pubs.acs.org/`.

References

(1) MacConnell, A. B.; Price, A. K.; Paegel, B. M. An integrated microfluidic processor for DNA-encoded combinatorial library functional screening. *ACS combinatorial science* **2017**, *19*, 181–192.

(2) Schneider, G. Automating drug discovery. *Nature Reviews Drug Discovery* **2017**, *17*, 97.

(3) Grinter, S. Z.; Zou, X. Challenges, applications, and recent advances of protein-ligand docking in structure-based drug design. *Molecules* **2014**, *19*, 10150–10176.

(4) Ballester, P. J.; Schreyer, A.; Blundell, T. L. Does a more precise chemical description of protein–ligand complexes lead to more accurate prediction of binding affinity? *Journal of chemical information and modeling* **2014**, *54*, 944–955.

(5) Chen, Y.-C. Beware of docking! *Trends in pharmacological sciences* **2015**, *36*, 78–95.

(6) Gaulton, A. et al. The ChEMBL database in 2017. *Nucleic Acids Research* **2017**, *45*, D945–D954.

(7) Gawehn, E.; Hiss, J. A.; Schneider, G. Deep learning in drug discovery. *Molecular informatics* **2016**, *35*, 3–14.

(8) Colwell, L. J. Statistical and machine learning approaches to predicting proteinligand interactions. *Current Opinion in Structural Biology* **2018**, *49*, 123–128.

(9) Ripphausen, P.; Nisius, B.; Bajorath, J. State-of-the-art in ligand-based virtual screening. *Drug discovery today* **2011**, *16*, 372–376.

(10) Unterthiner, T.; Mayr, A.; Klambauer, G.; Steijaert, M.; Wegner, J. K.; Ceulemans, H.; Hochreiter, S. Deep learning as an opportunity in virtual screening. Proceedings of the deep learning workshop at NIPS. 2014; pp 1–9.

(11) Ramsundar, B.; Kearnes, S.; Riley, P.; Webster, D.; Konerding, D.; Pande, V. Massively multitask networks for drug discovery. *arXiv preprint arXiv:1502.02072* **2015**,

(12) Duvenaud, D. K.; Maclaurin, D.; Iparraguirre, J.; Bombarell, R.; Hirzel, T.; Aspuru-Guzik, A.; Adams, R. P. Convolutional networks on graphs for learning molecular fingerprints. Advances in neural information processing systems. 2015; pp 2224–2232.

(13) Wallach, I.; Dzamba, M.; Heifets, A. AtomNet: a deep convolutional neural network for bioactivity prediction in structure-based drug discovery. *arXiv preprint arXiv:1510.02855* **2015**,

(14) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molecular graph convolutions: moving beyond fingerprints. *Journal of computer-aided molecular design* **2016**, *30*, 595–608.

(15) Ragoza, M.; Hochuli, J.; Idrobo, E.; Sunseri, J.; Koes, D. R. Protein–Ligand Scoring with Convolutional Neural Networks. *J. Chem. Inf. Model* **2017**, *57*, 942–957.

(16) Ramsundar, B.; Liu, B.; Wu, Z.; Verras, A.; Tudor, M.; Sheridan, R. P.; Pande, V. Is multitask deep learning practical for pharma? *Journal of chemical information and modeling* **2017**, *57*, 2068–2076.

(17) Gomes, J.; Ramsundar, B.; Feinberg, E. N.; Pande, V. S. Atomic Convolutional Networks for Predicting Protein-Ligand Binding Affinity. *arXiv preprint arXiv:1703.10603* **2017**,

(18) Verdonk, M. L.; Berdini, V.; Hartshorn, M. J.; Mooij, W. T.; Murray, C. W.; Taylor, R. D.; Watson, P. Virtual screening using protein- ligand docking: avoiding artificial enrichment. *Journal of chemical information and computer sciences* **2004**, *44*, 793–806.

(19) Cleves, A. E.; Jain, A. N. Effects of inductive bias on computational evaluations of ligand-based modeling and on drug discovery. *Journal of computer-aided molecular design* **2008**, *22*, 147–159.

(20) Rohrer, S. G.; Baumann, K. Maximum unbiased validation (MUV) data sets for virtual screening based on PubChem bioactivity data. *Journal of Chemical Information and Modeling* **2009**,

(21) Sheridan, R. P. Time-split cross-validation as a method for estimating the goodness of prospective prediction. *Journal of chemical information and modeling* **2013**, *53*, 783–790.

(22) Wallach, I.; Heifets, A. Most Ligand-Based Classification Benchmarks Reward Memorization Rather than Generalization. *Journal of Chemical Information and Modeling* **2018**,

(23) Liu, S.; Alnammi, M.; Ericksen, S. S.; Voter, A. F.; Ananiev, G. E.; Keck, J. L.; Hoffmann, F. M.; Wildman, S. A.; Gitter, A. Practical model selection for prospective virtual screening. *Journal of chemical information and modeling* **2018**,

(24) Hattori, K.; Wakabayashi, H.; Tamaki, K. Predicting key example compounds in competitors' patent applications using structural information alone. *Journal of chemical information and modeling* **2008**, *48*, 135–142.

(25) Good, A. C.; Oprea, T. I. Optimization of CAMD techniques 3. Virtual screening enrichment studies: a help or hindrance in tool selection? *Journal of computer-aided molecular design* **2008**, *22*, 169–178.

(26) Jain, A. N.; Cleves, A. E. Does your model weigh the same as a Duck? *Journal of computer-aided molecular design* **2012**, *26*, 57–67.

(27) Xia, J.; Jin, H.; Liu, Z.; Zhang, L.; Wang, X. S. An unbiased method to build benchmarking sets for ligand-based virtual screening and its application to GPCRs. *Journal of chemical information and modeling* **2014**, *54*, 1433–1450.

(28) Bolukbasi, T.; Chang, K.-W.; Zou, J. Y.; Saligrama, V.; Kalai, A. T. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. Advances in Neural Information Processing Systems. 2016; pp 4349–4357.

(29) Davies, M.; Nowotka, M.; Papadatos, G.; Dedman, N.; Gaulton, A.; Atkinson, F.; Bellis, L.; Overington, J. P. ChEMBL web services: Streamlining access to drug discovery data and utilities. *Nucleic Acids Research* **2015**, *43*, W612–W620.

(30) Mervin, L. H.; Bulusu, K. C.; Kalash, L.; Afzal, A. M.; Svensson, F.; Firth, M. A.; Barrett, I.; Engkvist, O.; Bender, A. Orthologue chemical space and its influence on target prediction. *Bioinformatics* **2018**, *34*, 72–79.

(31) Kim, S.; Thiessen, P. A.; Bolton, E. E.; Chen, J.; Fu, G.; Gindulyte, A.; Han, L.; He, J.; He, S.; Shoemaker, B. A.; Wang, J.; Yu, B.; Zhang, J.; Bryant, S. H. PubChem substance and compound databases. *Nucleic Acids Research* **2016**, *44*, D1202–D1213.

(32) Rogers, D.; Hahn, M. Extended-connectivity fingerprints. *Journal of chemical information and modeling* **2010**, *50*, 742–754.

(33) Jones, E.; Oliphant, T.; Peterson, P. {SciPy}: Open source scientific tools for {Python}. **2014**,

(34) Pedregosa, F. et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **2011**, *12*, 282 5–2830.

Graphical TOC Entry