HHS Public Access

Author manuscript

Artif Intell Chem. Author manuscript; available in PMC 2023 August 15.

Published in final edited form as:

Artif Intell Chem. 2023 June ; 1(1): . doi:10.1016/j.aichem.2023.100004.

Evaluating point-prediction uncertainties in neural networks for protein-ligand binding prediction

Ya Ju Fan[a,*], **Jonathan E. Allen**[b], **Kevin S. McLoughlin**[b], **Da Shi**[c], **Brian J. Bennion**[d], **Xiaohua Zhang**[d], **Felice C. Lightstone**[d]

[a]Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, 7000 East Ave., Livermore, CA, USA

[b]Biological Science and Security Center, Lawrence Livermore National Laboratory, Livermore, CA, USA

[c]Biomedical Informatics and Data Science Directorate, Frederick National Laboratory for Cancer Research, Frederick, MD, USA

[d]Physical and Life Sciences Directorate, Lawrence Livermore National Laboratory, Livermore, CA, USA

Abstract

Neural Network (NN) models provide potential to speed up the drug discovery process and reduce its failure rates. The success of NN models requires uncertainty quantification (UQ) as drug discovery explores chemical space beyond the training data distribution. Standard NN models do not provide uncertainty information. Some methods require changing the NN architecture or training procedure, limiting the selection of NN models. Moreover, predictive uncertainty can come from different sources. It is important to have the ability to separately model different types of predictive uncertainty, as the model can take assorted actions depending on the source of uncertainty. In this paper, we examine UQ methods that estimate different sources of predictive uncertainty for NN models aiming at protein-ligand binding prediction. We use our prior knowledge on chemical compounds to design the experiments. By utilizing a visualization method we create non-overlapping and chemically diverse partitions from a collection of chemical compounds. These partitions are used as training and test set splits to explore NN model uncertainty. We demonstrate how the uncertainties estimated by the selected methods describe different sources of uncertainty under different partitions and featurization schemes and the relationship to prediction error.

Keywords

Uncertainty quantification; Neural networks; Drug discovery; Applicability domain

*Corresponding author. fan4@llnl.gov (Y.J. Fan).

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

1. Background and motivation

Traditional drug discovery examines bioactivities between drug and protein via a high throughput screening experiment [1,2]. The process is time-consuming and expensive. Deep learning (DL) models try to capture intricate nonlinear relationships between input data (such as drug compounds) and the associated output (such as protein inhibition) for large scale computational screens [3]. The models then predict the properties of new compounds that help determine a feasible set of compounds for synthesis and evaluation. Computational screens save time and effort, and provide the potential to speed up the drug discovery process and reduce its failure rates [4,5]. However, drug discovery requires exploring chemical space beyond the training data distribution. Predictions on unknown regions are prone to pathological failure. The success of adopting DL methods across a range of drug-design settings requires better communication of uncertainty [6,7]. Detecting regions of chemical space with high uncertainty could help design the experiments to expand a model's applicability domain.

Recent studies have emphasized the importance and challenges of uncertainty quantification (UQ) in deep learning for drug discovery. Mervin et al. (2020) [8] reviewed classic and modern UQ methods and discussed their usage for drug design (e.g, including the empirical, frequentist and Bayesian approaches). The study points out that the chemoinformatic data being modeled is overall heavily biased with respect to the amount, degree of diversity and distribution of data points. There are issues surrounding data quality and assay variability [9], as well as the skewed proportion of protein-target complexes [10,11] and the imbalance between active and inactive compound-target labels [12]. Hence, most models for drug discovery are not able to provide realistic probability estimates while providing single point predictions. A common misleading phenomenon is in the classification models where the predictions for two classes, such as labels of 'activity' and 'inactivity', come from probability-like fractions. One example is the softmax function used in the last layer of a neural network model, which gives an output value between zero and one. These function values only mean to separate output classes, not to provide confidence of the prediction. Applying explicit uncertainty quantification methods could provide an alternative confidence measure to the classification probability estimates.

Uncertainty of deep learning could come from two sources: data (aleatoric) uncertainty and model (epistemic) uncertainty [13]. The *data uncertainty* reflects a lack of confidence due to the imprecision of molecular measurements. The aleatory concept involves unknown outcomes that can differ each time one runs an experiment under similar conditions [14]. Data collected for the drug discovery process could have experimental variabilities due to natural biological changes in the samples, measurement fidelity, instrumentation, sampling procedures, etc. The data uncertainty is irreducible to the model and is an inherent property of their distribution.

In contrast, *model uncertainty* measures the uncertainty in model parameters given the training data. Particularly, inadequate knowledge contributes to the model uncertainty. This is often the case for the chemoinformatic data where there is a limited number of drug compounds available with unbalanced activity classes for a complex biological process. UQ

methods for deep neural networks (NN) that evaluate how changes in model parameters effect the NN predictions are aimed at capturing the model uncertainty. Bayesian neural networks (BNN) [15,16] quantify posterior uncertainty on NN model parameters (i.e. weights) and express predictions in terms of expectations with respect to this posterior distribution [17]. Other methods generate a set of point predictions and use its mean and variance to represent the optimal prediction and its corresponding uncertainty. Monte-Carlo dropout [18] passes a test sample multiple times through the NN model and generate a collection of such predictions. For each iteration, the model assigns the value zero to a fraction of randomly selected weights. Similarly, deep ensemble [19] and bootstrap [20] train multiple models and compute the mean and spread of the ensemble.

Apart from data uncertainty and model uncertainty, Malinin and Gales (2018) presented distributional uncertainty as a separate source of uncertainty [21]. *Distributional uncertainty* occurs when the test data is foreign to the model due to mismatch between the training and test distributions. Applicability domain (AD) estimates whether a model's prediction for a chemical compound is applicable based on the model's training set properties. Distance-based methods used to evaluate AD play a similar role as distributional uncertainty modeling methods [22].

Residual estimation with an I/O kernel (RIO) [23] directly estimates prediction residuals using modified Gaussian Processes (GP). GP models offer a mathematically grounded approach to reason about the predictive uncertainty [24]. RIO uses a new composite I/O kernel that makes use of both inputs and outputs of the NN, meaning that the method examines both the distance to the nearest training data and the prediction errors on the training set. It provides predictive uncertainty estimation without modifying the NN training or formulation. Hie et al., 2020 [7] demonstrated that RIO uncertainty estimation enables successful iterative learning across a broad spectrum of experimental scales for biological discovery and design.

It is important to have the ability to separately model the different types of predictive uncertainty, as the model can take assorted actions depending on the source of uncertainty [21]. In this article, we present a carefully designed experiment with publicly available drug data to gain insights into uncertainty quantification methods. We employ the calculated end-point binding free energy with MM/GBSA (the molecular mechanics generalized Born surface area) as the response values for the NN models. Note that MM/GBSA is one of a number of different methods used in computational drug discovery [25]. The simulated MM/GBSA scores are not from the experimental measurements, and hence reduce the aleatoric uncertainty, which allows the study to focus on modeling epistemic uncertainty. We select the UQ methods that examine different sources of uncertainty. Additionally, they can be directly applied to any standard NN without having to modify the model training formulation. We use the ATOM Modeling PipeLine (AMPL) [26] developed by the Accelerating Therapeutics for Opportunities in Medicine (ATOM) Consortium to provide a rigorous pipeline for training and evaluating drug discovery oriented NN models. We demonstrate how the selected UQ methods provide uncertainty estimations on point predictions of NN models. Since the goal of UQ is to detect unanticipated imprecision of model predictions, we inspect how the UQ methods reflect prediction errors.

This paper is organized as follows. First, we collect related work in Section 2. Next, we describe the selected uncertainty quantification methods and how they present different sources of uncertainty in Section 3. In Section 4, we explain the preparation of the chemical compound inputs, NN model building and the experimental design to evaluate the UQ methods. Finally, we present the experimental results in Section 5 and conclude our findings in Section 6.

2. Related work

There has been significant demand in quantifying prediction uncertainty for DNN models, especially for drug discovery where mistakes may be expensive. Hie et al. (2020) [7] leveraged Gaussian process-based uncertainty prediction to identify and validate experimental compounds with nanomolar affinity for diverse kinases and whole-cell growth inhibition of Mycobacterium tuberculosis. They showed that the GP uncertainty estimation enables successful iterative learning across a broad spectrum of experimental scales. Scalia et al. (2020) [27] compared scalable UQ methods, including MC-dropout, Deep Ensembles and bootstrapping, for graph convolutional neural networks (GCNN), designed for deep learning-based molecular property prediction. They introduced a set of quantitative criteria, including ranking-based methods and uncertainty calibration methods, to capture different uncertainty aspects. Wang et al. (2021) [22] combine both distance-based and Bayesian UQ approaches together for improved uncertainty quantification in QSAR (Quantitative Structure-Activity Relationship) regression modeling. The hybrid method quantitatively assesses the ranking and calibration ability of the selected UQ methods, including applicability domain (AD) methods, mean-variance estimation of the graph convolutional neural networks [27] and deep ensembles [19].

There has been notable progress made on predictive uncertainty for deep learning through the formulation of neural networks. One class of approaches stems from the combination of a Bayesian approach and neural networks [16,19,28–30]. All such methods require significant modifications to the model infrastructure and training procedure. Malinin and Gales (2018) developed Prior Networks for modeling predictive uncertainty, which explicitly models distributional uncertainty [21]. Tang and de Jong (2019) [31] developed marginalized graph kernel specifically for computing similarity between molecules. The framework employs GP regression to perform prediction on the atomization energy of molecules. Moreover, drug discovery tasks often evaluate novel molecules by entering areas in the feature space that are not previously represented in training data. Han et al. (2021) [32] measured the data distributional shift as the model reliability of graph neural networks (GNN).

In this work, we focus on measuring the point-prediction uncertainties of the standard neural network regression models with fully connected layers that require feature vectors as their input. We investigate a set of the UQ methods separately, which examine different sources of uncertainty. These methods also provide UQ estimations without requiring specific NN formulations, allowing broader applications.

3. Uncertainty quantification methods for neural networks

We select uncertainty quantification approaches for deep neural networks that do not limit the choice of model. Consider a neural network model with L layers. We denote $\mathbf{W}_i$ as the NN's weight matrices for each layer $i = 1, 2, \ldots, L$. We denote y_i as the observed output corresponding to input $\mathbf{x}_i, i = 1, 2, \ldots, n$, where n is the number of data points. Let $\mathbf{X}, \mathbf{y}$ be the input and output sets and let the training dataset $\mathcal{D} = (\mathbf{X}, \mathbf{y}) = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$. The predictive probability of a NN model can be parameterized as

$$P(y|x, \mathcal{D}) = \int \underbrace{P(y|x, \omega)}_{\text{Data}} \underbrace{p(\omega|\mathcal{D})}_{\text{Model}} d\omega . \tag{1}$$

In a Bayesian framework the predictive uncertainty of a NN model $P(y \mid x, \mathcal{D})$ trained on a finite dataset $\mathcal{D}$ will result from data (aleatoric) uncertainty and model (epistemic) uncertainty as shown in 1. The posterior distribution over responses y given a set of model parameters ω describes a model's estimates of data uncertainty, and the posterior distribution over the parameters given data describes model uncertainty [21].

3.1. Monte-Carlo dropout (MC-dropout)

Deep neural networks contain multiple nodes and layers that try to learn complicated relationships between the inputs and outputs. Dropout is a method used to prevent overfitting and lower generalization error for NN models [33]. Dropout means temporarily removing a node from the network along with all its incoming and outgoing connections, which we can simply set zero on the weight of the node. The choice of which nodes to drop is random. In most cases, we choose a fixed value to indicate the fraction of the nodes to drop. Gal and Ghahramani (2016) further applied the concept to generate ensembles for evaluating model uncertainty.

Let $\mathbf{W}_i$ be a random matrix for each layer i and set $\omega = \{\mathbf{W}_i\}_{i=1}^{L}$. In Equations 1 obtaining the true posterior $p(\omega \mid \mathcal{D})$ using Bayes' rule is intractable, and it is necessary to use either an explicit or implicit variational approximation $q(\omega)$ [34–36]. Monte-Carlo (MC) dropout utilizes the $q(\omega)$ as a distribution over matrices whose columns are randomly set to zero (called dropout). Furthermore, the integral in Equations 1 for the predictive probability is also intractable for neural networks. MC dropout employs sampling to approximate it. As shown in Equations 2 each term in the sum is approximated by Monte-Carlo integration with a single sample $\omega^{(i)} \sim q(\omega)$ to obtain an unbiased estimate[18].

$$P(y \mid x, \mathcal{D}) \approx \frac{1}{n} \sum_{i-1}^{n} P\left(y \mid x, \omega^{(i)}\right), \text{ where } \omega^{(i)} \sim q(\omega) . \tag{2}$$

Training a neural network with dropout is as training a collection of thinned networks with extensive weight sharing. The process does not change the dropout NN model itself. In testing we use the original model by simply cutting off the node's weights. MC

dropout estimates the predictive mean and predictive uncertainty by collecting the results of stochastic forward passes through the NN model.

3.2. Applicability domain (AD)

The Applicability Domain (AD) assessment is based on the numerical vector representation of chemical compounds in the training set. We select two AD methods that are distance-based and suitable for novelty (or outlier) detection.

3.2.1. Empirical distance distribution (AD-DD)—For distance-based novelty detection we need a threshold to decide whether or not the unseen object is actually novel [37]. The empirical distance distribution of the training set molecules can help establish a threshold, using the $1-\alpha$ quantile (for $\alpha = 0.01, 0.05$ or 0.1) of the k-nearest neighbors within the training set distance distribution [38]. We collect the mean distances from every data point to its k-nearest neighbors in the training set. We fit a normal distribution on these mean distances and obtain its total mean, μ_{knn}, and standard deviation, σ_{knn}. For a new chemical compound x, we compute its mean distance to its k-nearest neighbors in the training set $\bar{d}_{\text{knn}}(x)$. The score of the applicability domain using the empirical distance distribution is the rectified Z-score of the mean k-nearest neighbor distance of the new point based on the distribution of the training set:

$$\rho_D(x) = \max\left\{0, \frac{\bar{d}_{\text{knn}}(x) - \mu_{\text{knn}}}{\sigma_{\text{knn}}}\right\}. \tag{3}$$

3.2.2. Local density (AD-LD)—Denote the distance of a data point x to its k^{th} -nearest neighbor in the training set $NN_k^{train}(x)$ as $\|x - NN_k^{train}(x)\|$. The local density is inversely related to the distance. The higher the local density is, the lower the distance is to the neighbors. The score of the applicability domain using local density [39] is:

$$\rho_L(x) = \frac{\frac{1}{k}\sum_{i=1}^{k}\|x - NN_i^{train}(x)\|}{\frac{1}{k^2}\sum_{i=1}^{k}\sum_{j=1}^{k}\left\|NN_i^{train}(x) - NN_j^{train}\left(NN_i^{train}(x)\right)\right\|}. \tag{4}$$

The denominator is the average distance from the nearest neighbors $NN^{train}(x)$ to their nearest neighbors in the training set $NN^{train}\left(NN^{train}(x)\right)$.

3.3. Residual estimation with an I/O kernel (RIO)

Residual estimation with an I/O kernel (RIO) is a Gaussian processes (GP) method designed to be applied on top of any trained NN model [23]. RIO quantifies the uncertainty in the point-predictions of NN models without retraining or modifying any component of them.

Let $\hat{y}_i$ be the point prediction of a NN model given $\mathbf{x}_i$. RIO models the residuals between observed outcomes y and NN predictions $\hat{y}$ using GP with a composite kernel. The residuals $\{r_i\}_{i=1}^{n}$ between observed outcomes and NN predictions on the training dataset $\mathcal{D}$ is

$$r_i = y_i - \hat{y}_i, \text{ for } i = 1, 2, \ldots, n. \quad (5)$$

Let $\mathbf{r}$ be the vector of all residuals and $\hat{\mathbf{y}}$ be the vector of all NN predictions. RIO trains a GP with a composite kernel assuming $\mathbf{r} \sim \mathcal{N}(0, \mathbf{K}_\theta((\mathbf{X}, \hat{\mathbf{y}}), (\mathbf{X}, \hat{\mathbf{y}})) + \sigma_n^2 \mathbf{I})$, where $\mathcal{N}$ denotes a multivariate Gaussian distribution with mean 0 and covariance matrix $\mathbf{K}_\theta((\mathbf{X}, \hat{\mathbf{y}}), (\mathbf{X}, \hat{\mathbf{y}})) + \sigma_n^2 \mathbf{I})$, and σ_n^2 is the noise variance of observations. The composite kernel $\mathbf{K}_\theta((\mathbf{X}, \hat{\mathbf{y}}), (\mathbf{X}, \hat{\mathbf{y}}))$ is an $n \times n$ covariance matrix at all pairs of training points whose elements are:

$$k_\theta((\mathbf{x}_i, \hat{y}_i), (\mathbf{x}_j, \hat{y}_j)) = k_{\theta_{\text{in}}}(\mathbf{x}_i, \mathbf{x}_j) + k_{\theta_{\text{out}}}(\hat{y}_i, \hat{y}_j), \text{for } i, j = 1, 2, \ldots, n. \quad (6)$$

This kernel design enables the evaluation on both the novelty of a data point and the prediction residual. The composite kernel is the sum of two kernels, one for the NN input data and the other for the NN output data. RIO applies the composite kernel as the covariance function of GP. The GP optimizes the hyperparameters of the covariance function, θin and θout, by maximizing the log marginal likelihood $\log p(\mathbf{r} \mid \mathbf{X}, \hat{\mathbf{y}})$.

4. Experimental design

We randomly select 9000 drug compounds available from the ChEMBL database [40,41], a large, open-access bioactivity data source. The drug compounds are presented in the SMILES (simplified molecular-input line-entry system) strings, which describes the structure of the chemical compounds in the dataset. We use two featurization schemes to map the SMILES strings into model input features. We first compute the chemical descriptors using Molecular Operating Environment (MOE) software [42]. Many MOE descriptors were strongly correlated with each other due to the fact that they scaled with molecular size, as measured by the total number of atoms, denoted as a_count [43]. We replace all descriptors, d, having Pearson correlation r (d, a_count) > 0.5 with the computed d/a_count to remove the dependency. We further eliminate descriptors that are duplicated, have constant values or are linear functions of other descriptors. The final set of the MOE descriptors contain 306 features. The other featurization scheme that we also compute is the extended connectivity fingerprint (ECFP4). We use RDKit [44] to generate the ECFP4 bit vectors with length 1024.

We utilize the molecular mechanics generalized Born surface area (MM/GBSA) continuum solvent approach to rescore the binding affinities of the chemical compounds [45,46]. Note that there are other approaches used in computational drug discovery [25]. The computed MM/GBSA scores are not from experimental measurements but rather estimate binding affinity between a compound and a target protein. The target protein used for this study is the main spike protein for SARS-CoV-2. Applying them as the responses of the NN models reduces their aleatoric uncertainty. Since the aleatoric uncertainty is irreducible to the model, making it more controllable could help observing other uncertainty sources. We have also observed that the MOE descriptors are highly correlated to the MM/GBSA scores in the randomly selected 9000 drug compounds. Among the total 306 MOE features there are 22 features having correlation larger than 0.5 as shown in Table 1.

To better understand how the uncertainty quantification methods work for the protein-ligand binding prediction in the drug discovery tasks, we make use of the known correlation between the MOE descriptors and the MM/GBSA scores to design the experiments. We visualize the correlation in the dataset using the t-distributed stochastic neighbor embedding (t-SNE). We project the top 22 correlated MOE features employing ten principle components before computing the two coordinates of the t-SNE, according to the suggested usage of the method [47]. Fig.1 displays the two-dimensional t-SNE scatter plot of the dataset.

It appears that the dataset consists of five clusters. The color on the data points represents the MM/GBSA scores where a more negative value indicates stronger binding. Cluster 2, 3 and 4 marked on the plot contain data points with large negative MM/GBSA scores, while cluster 1 and 3 contain more high value MM/GBSA scores. Fig. 2 shows the distribution of the MM/GBSA scores for each cluster.

We split the dataset into training set and test set based on the clusters. The numbers of the data points in the clusters are 2132, 2455, 3252, 1154 and 7, respectively, from cluster 1 to cluster 5. We exclude cluster 5 due to its small size. We randomly select 1000 data points in each cluster as the training set, another 100 data points as the validation set and the rest as the test set. We apply the same splits to both the MOE features and the ECFP features, creating eight splittings of training, validation and test sets (from four clusters and two featurization methods). We include all of the 306 MOE features in the dataset as the 22 high correlation features discussed earlier are for visualization purposes only.

We train the Neural Network models using a data-driven modeling pipeline, AMPL, developed by our group at the ATOM Consortium [26]. The underlying models were implemented with the DeepChem package [48]. The architecture of the neural networks consists of one, two or three fully connected hidden layers. Each layer contains varying numbers of rectified linear unit (ReLU) nodes. During training and evaluation we randomly drop out 30% of nodes to avoid overfitting. To optimize the performance of neural networks, we run hyperparameter searches, varying the numbers of hidden layers, numbers of nodes per layer, and learning rates. We evaluate an average of 1500 regression models for each training set and select the best model for further analysis on quantifying the point-prediction uncertainties. We select the best model parameters based on validation set performance following standard machine learning practices. We evaluate the performance of regression models using the coefficient of determination (R^2)[49] defined as

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}$$

where y_i is the actual MM/GBSA score for a compound i, $\bar{y}$ is the average value of the actual scores and $\hat{y}_i$ is the predicted value of y_i. An R^2 close to 1 implies an almost perfect case when modeled values nearly match the observed values. An R^2 equal to 0 implies a baseline model, which always predicts the mean. A negative R^2 indicates that a model has worse predictions than the baseline.

5. Experimental results

5.1. Model performance

The best regression model after hyperparameter tuning, measured by R^2 scores, is kept for each cluster and each type of chemical representation (MOE or ECFP). Tables 2 and 3 collect the separated test set performances of the best model trained on each cluster. Recall that we take 1000 randomly selected data points from each cluster for model training. We use the rest of the data points as the test set in the cluster to show how each model perform on single clusters.

We found that the models trained on MOE descriptors perform better than the ones trained on ECFP. The clusters used for the training and test sets are created from the most correlated MOE descriptors. Since the clusters are not explicitly separated by their ECFP distances, there would be more similarity between the training and test sets in ECFP. It is surprising that the ECFP models do more poorly in the cross cluster prediction task.

The results also indicate that the NN models perform the best in their own clusters, except on the model trained on cluster 3 with MOE descriptors. Data points in cluster 3 cover a wider range of the MM/GBSA scores. Interestingly the model trained on cluster 3 using MOE descriptors performed the best on cluster 1 with R^2 at 0.685, while performed the second best on the test set from its own cluster with R^2 at 0.580. We also observed that the models trained on cluster 2 and cluster 4 do not perform well on predicting data points in other single clusters. Overall, most models have worst performance on their test set, and best performance on their training set. This aligns with what we expect when designing the splits of the data. Chemical compounds outside a cluster tend to be more different from those in the cluster. It is a challenge to the model trained on a cluster to test its performance outside the cluster. Next, we investigate how the selected UQ methods capture such challenges.

5.2. Uncertainty evaluation

Since we use the t-SNE plot to generate the clusters where distances in the original data space are not preserved in the 2D feature space. The clusters in the plot may be shown to appear in non-clustered data.

Values generated by the AD method using empirical distance distribution (AD-DD) measuring the dissimilarities to the training set can help realize the actual distances. Figs. 3, 5, 7 and 9 display the uncertainty estimations of the point predictions using the four uncertainty quantification methods for the models trained on cluster 1,2,3 and 4, respectively. The AD-DD uncertainty values are significantly lower in the training clusters than in the rest of the chemical compounds, especially in cluster 1,2 and 3. When using the cluster 4 as the training set, AD-DD method does not give as distinctly smaller values for the training data. It appears that chemical compounds described by MOE descriptors in cluster 4 are close to data points in cluster 2 and 3, indicating that the cluster covers a wide range of chemical compounds. Furthermore, data points represented by ECFP features in cluster 4 spread out from each other, suggesting non-clustered data.

Recall that distance-based novelty detection needs a threshold to decide whether a new compound is actually novel [37]. If we use a 95 % quantile of the k-nearest neighbor within the training set distance distribution as a threshold, the corresponding z-score is around 1.6 in a standard normal distribution. We can use the value 1.6 for AD-DD values as a threshold to identify novel test compounds. We can see that this threshold distinguishes the training cluster 1,2 and 3 from their corresponding test set clusters.

The AD method using local density (AD-LD) highlights more data points with larger values than the AD-DD method. The distinctiveness between the training and the test sets using AD-LD is less significant than AD-DD. Only the models trained on cluster 1 with MOE descriptors gives significantly lower AD-LD values on the training cluster than the rest of chemical compounds. Moreover, the models trained with ECFP features yield lower ranges of AD-LD values, which indicates that data points in ECFP features spread out and may not form definite clusters.

Models using different compound representation may have opposite uncertainty values from MC-dropout. The model uncertainties (evaluated by MC-dropout) that appear high in one compound representations may be low in other compound representations. The third rows of the figures display the UQ values from MC-dropout used to present the model uncertainties. For the models trained on cluster 1 and cluster 3, MC-dropout gives opposite degrees of uncertainties when using different compound representations. As shown in Figs. 3 and 7 the MC-dropout method gives high uncertainty values with MOE descriptors while giving low uncertainty values with ECFP features for most predictions. To investigate which uncertainty values to trust, we compare the UQ values with the prediction errors in the Figs. 4 and 8, which display the actual MM/GBSA scores versus the predicted, colored by the UQ values. The distance from a data point to the diagonal indicate its prediction error. We found that the UQ values using MC-dropout on the MOE descriptors tend to be high on the prediction values that have data points far away from the diagonal and low on the predictions that are closer to the diagonal, indicating that MC-dropout on the MOE descriptors can reflect possible prediction errors. We also know that the models trained on the ECFP features do not perform well on the predictions. We may conclude that the UQ values from MC-dropout are more reliable on the MOE descriptors than on the ECFP features possibly due to better trained models with MOE descriptors. Furthermore, MC-dropout does not necessarily give lower uncertainties in the training than in the test set. There are high MC-dropout uncertainty values in the training clusters where the other three methods tend to have low uncertainty values (Figs. 7 and 8).

RIO method gives near constant uncertainty values to the majority of the chemical compounds, especially on the models trained with ECFP features. There are only a few data points that have relative high RIO uncertainty values. In general RIO picked up part of high uncertainties similar to AD-DD when the models are trained with MOE descriptors. Exceptionally, for the model trained on cluster 4 with MOE descriptors RIO marks particularly high uncertainty values on several chemical compounds; while all other methods mark similar high uncertainty values but they are less distinct than what RIO gives as shown in Figs. 9 and 10.

5.3. Uncertainties v.s. prediction errors

When using NN model predictions to direct experimental design, unanticipated imprecision wastes valuable time and resources for drug discovery applications [50]. We examine whether the UQ methods are able to identify the prediction errors when using NN models for new chemical compounds. We plot the actual MM/GBSA scores versus the predicted values of the chemical compounds from the test sets. We color their uncertainty values from the four evaluation methods for the models trained on cluster 1, 2, 3 and 4 in Figs. 4, 6, 8 and 10, respectively. The diagonal line on the plots represents the values where the predicted is equal to the actual MM/GBSA score. Data points that are far away from the diagonal have large prediction errors. We examine the results for each cluster in the following.

The majority of the chemical compounds in cluster 1 have MM/GBSA scores ranging from around −15 to −35, indicating weaker binding. It is reasonable that larger prediction errors fall on the stronger binding side, ranging from −30 to −40 as shown in Fig. 4. This range is near the tail of the training set. Predicted values that fall outside the training set can be less reliable. The AD-LD, MC-dropout and RIO uncertainty values are high for those less reliable predictions for the model trained with MOE descriptors, while only the AD-DD and AD-LD values are high for ECFP. For the model trained on cluster 1 with ECFP the model uncertainty from MC-dropout appears opposite to the desired values, rendering low values on the high prediction error ranges and high values on the low prediction error ranges. For the same model the RIO method gives relatively high evaluations only to a few predicted values at the two ends of the predicting range, around −20 and −35.

Cluster 2 has strong binding free energy with the MM/GBSA scores approximately ranging from −25 to −55 as the majority. For the model trained on cluster 2 with MOE descriptors the AD-DD and AD-LD methods give relatively high values on the chemical compounds that also have large prediction errors as shown in Fig. 6. For the same model with the MOE descriptors, the model uncertainties from MC-dropout are moderately high on the lower half of the prediction values, while the RIO uncertainties are distinctly higher at the points far away from the diagonal. The four uncertainty evaluation methods do not capture any prediction errors for the model trained on cluster 2 with the ECFP features, whose training set performance was poor with R^2 at 0.412. The model predicts strong binding for all data points, and has a small range of prediction values, from −32 to −44.

The chemical compounds in cluster 3 and cluster 4 cover a wider range of MM/GBSA scores than cluster 1 and cluster 2. Between the two clusters, cluster 3 contains more high value MM/GBSA scores (meaning weaker binding) while cluster 4 contains more low value MM/GBSA scores (meaning stronger binding). The models trained on cluster 3 have the best performance on the test set prediction. This also reflects on the actual versus predicted plots in Fig. 8 where most data points are along the diagonal. AD-DD, AD-LD and RIO give high values at the two ends of the predicting range for the model trained with MOE descriptors. RIO uncertainty gives distinct large uncertainty values at the lowest predicted MM/GBSA scores and only a few at the highest, reflecting the uncertainty due to the lack of low MM/GBSA scores in the training set. For the models trained on cluster 3 model uncertainties from MC-dropout are high at the lower half of the predicted range with MOE

descriptors, but high at the highest predicted values with ECFP. The model trained with ECFP again does not perform well and has a narrow range of predicted values, ranging from −30 to −43. The AD-DD and AD-LD values are spread out while the RIO uncertainties are close to a constant for the model trained with ECFP.

The model trained on cluster 4 with MOE descriptors predicts high MM/GBSA scores with large errors. All four methods successfully indicate these high prediction errors. Particularly the RIO method gives distinctively high uncertainties on the predictions larger than −20 where the large errors occur. Similar to the model trained on cluster 3, the model trained with ECFP does not perform well and has a narrow range of prediction values, ranging from −30 to −43. The AD-DD and AD-LD values are spread out. The model uncertainties are high at the high predicted values. The RIO uncertainties are close to a constant for most chemical compounds except the lowest and the highest ends of the predicted values.

Although RIO gives almost all predictions a near constant uncertainty estimation, we observed that these constants reflect possible prediction errors. Fig. 11 display the box plots for the test set uncertainties and prediction errors. Fig. 11(C) and Fig. 11(B) contain the estimated mean and standard deviation of the prediction errors from the Gaussian Processes used in RIO. On the x-axis are the trained models where the first column C1 indicates the model trained on cluster 1. The corresponding data points in the box plot are from C1's test set, composed of cluster 2,3 and 4. The UQ value that RIO estimates for a point prediction is the standard deviation of the prediction errors. RIO gives high uncertainty values to most of their test set predictions except C1 with MOE descriptors, reflecting the possibility of large test set prediction failures.

Remarkably by ordering the uncertainty values and examining the test set performance, we found that the magnitude of the uncertainties does not necessarily reflect the level of prediction errors. We sort the uncertainty values in a decreasing order in the test sets for each of the UQ methods. We compute the test set coefficient of determination (also known as the R^2 score) and remove the top 5% largest uncertainty values in the test set, iteratively, until no samples are left. These computed values form the performance curves shown in Fig. 12. If the magnitude of the uncertainty values reflect the level of the prediction errors, the test set performance will increase as the top 5% largest uncertainty values are removed from the test set. The model trained on cluster 4 with MOE descriptors is the only one where all four UQ methods are able to reflect the level of prediction errors. These UQ methods evaluating other NN models barely echo the level of prediction errors.

6. Conclusion

In this article, we investigate selected uncertainty quantification (UQ) methods that provides estimates of different sources of predictive uncertainty for neural networks (NN) aimed at protein-ligand binding prediction. The applicability domain (AD) methods capture distributional uncertainties. MC-dropout estimates model uncertainties. RIO perceives NN behavior by estimating the NN prediction residuals. The selected methods can be directly applied to any standard NN without having to modify the model formulation or training procedure, making them more accessible to analysts.

To test how these UQ methods work on NN model predictions for the drug discovery tasks, we carry out a series of carefully designed experiments. We randomly select drug compounds in the form of SMILES strings from public available database. We rescore the binding affinities of the chemical compounds by employing the molecular mechanics generalized Born surface area (MM/GBSA) continuum solvent approach. Since the MM/GBSA scores are calculated and not from the experimental measurements, it reduces the otherwise irreducible aleatoric uncertainty in the prediction. We use two featurization schemes, the MOE descriptors and the ECFP4 bit vectors, as model features. We design the splits of the training and test set using our prior knowledge on the correlation between MOE descriptors and the MM/GBSA scores. The visualization algorithm, t-distributed stochastic neighbor embedding (t-SNE), gives four visible clusters on the subset of MOE features that are highly correlated to the MM/GBSA scores. The clusters contain different distributions of the MM/GBSA scores. We take each cluster as a training set and the rest of the chemical compounds as the corresponding test set. We apply hyperparameter search to build optimal NN models. Finally, the selected UQ methods provide uncertainty values for the point predictions made by the trained NN models.

Our experiments show that the selected UQ methods describe different sources of uncertainty for point predictions of NN models. The AD method using distance distribution (AD-DD) retain the distance scale of the original data space, which the projected 2-dimensional t-SNE space has lost. Our results show that the AD-DD method can highlight novel chemical compounds, which gives the data in the other clusters higher uncertainties. It can also identify non-clustered data in the t-SNE plot, such as the cluster 4. When using cluster 4 as the training cluster, AD-DD gives high uncertainty values to several data points in the cluster, indicating the chemical compounds are very different from each other. The AD method using local density (AD-LD) verifies the clarity of the training set clusters.

MC-dropouts points to the sensitivity of the training parameters to the final predictions. MC-dropout can give opposite levels of uncertainty estimations when using different compound representations (i.e. MOE or ECFP). Our results show that MC-dropout is more reliable with MOE descriptors than with ECFP, possibly due to the better trained models with MOE descriptors. MC-dropout can reflect possible prediction errors on some of the predictions made by the models trained with MOE descriptors on cluster 1 and cluster 3. However, MC-dropout provide high uncertainty values to the predictions in the training clusters where the other methods tend to give low values.

RIO method gives near constant uncertainty values to the majority of chemical compounds. We observed that these constants reflect possible prediction errors made by poorly performing models. When the model is not performing well, the RIO gives high near-constant uncertainty values to most predictions. Only in the case where the chemical compounds are very different from the applicable domain and at the same time the model causes large prediction errors in the test set, RIO gives high uncertainty values to the data points that are distinctly larger than the near-constant uncertainty values.

Moreover, we observed that not all UQ methods can capture high prediction errors invariably. Only for cluster 4 with MOE descriptors do all four models capture the high

prediction errors. Furthermore, the magnitude of the uncertainties does not clearly reflect the level of prediction errors. Again, only the models trained on cluster 4 with MOE descriptors form increasing performance curves when the top 5% largest uncertainty values are removed iteratively. Generally, the AD methods can only detect the novelty of the chemical compounds without any knowledge from the NN model. MC-dropout reflects the prediction variance due to changes in training parameters. RIO gives extra high uncertainties to extreme predicted values. These scenario do not always come with high prediction errors.

The differences in the uncertainty estimations made by the selected UQ methods provide more insight into the behavior of NN models for protein-ligand binding prediction. We suggest to model the different types of predictive uncertainty separately. Knowing the assorted types of uncertainty in a point prediction may assist practitioners in taking an appropriate UQ aware selection strategy.

Acknowledgement

LLNL-JRNL-839676. This work represents a multi-institutional effort and is supported by the Accelerating Therapeutics for Opportunities in Medicine (ATOM) Consortium under CRADA TC02349.9. Funding sources include the following: Lawrence Livermore National Laboratory internal funds; the National Nuclear Security Administration; DTRA under Award HDTRA1036045 and federal funds from the National Cancer Institute, National Institutes of Health, and the Department of Health and Human Services, Leidos Biomedical Research Contract No. 75N91019D00024, Task Order 75N91019F00134. This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory [Contract No. DE-AC52-07NA27344].

References

[1]. Cohen P, Protein kinases - the major drug targets of the twenty-first century? Nat. Rev. Drug Discov 1 (2002) 309–315. [PubMed: 12120282]

[2]. Noble MEM, Endicott JA, Johnson LN, Protein kinase inhibitors: insights into drug design from structure, Science 303 (5665) (2004) 1800–1805, 10.1126/science.1095920 [PubMed: 15031492]

[3]. Stevenson GA, Jones D, Kim H, Bennett WFD, Bennion BJ, Borucki M, Bourguet F, Epstein A, Franco M, Harmon B, He S, Katz MP, Kirshner D, Lao V, Lau EY, Lo J, McLoughlin K, Mosesso R, Murugesh DK, Negrete OA, Saada EA, Segelke B, Stefan M, Torres MW, Weilhammer D, Wong S, Yang Y, Zemla A, Zhang X, Zhu F, Lightstone FC, Allen JE, High-throughput virtual screening of small molecule inhibitors for sars-cov-2 protein targets with deep fusion models, in: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, ser. SC '21.New York, NY, USA: Association for Computing Machinery, 2021. 〈10.1145/3458817.3476193〉.

[4]. Vamathevan J, Clark D, Czodrowski P, Dunham I, Ferran E, Lee G, Li B, Madabhushi A, Shah P, Spitzer M, Zhao S, Applications of machine learning in drug discovery and development, Na. Rev. Drug Discov 18 (6) (2019) 463–477 ⟨10.1038/s41573-019-0024-5⟩

[5]. Jiménez-Luna J, Grisoni F, Schneider G, Drug discovery with explainable artificial intelligence, Nat. Mach. Intell 2 (10) (2020) 573–584 〈10.1038/442256-020-00236-4〉.

[6]. Ching T, Himmelstein DS, Beaulieu-Jones BK, Kalinin AA, Do BT, Way GP, Ferrero E, Agapow P-M, Zietz M, Hoffman MM, Xie W, Rosen GL, Lengerich BJ, Israeli J, Lanchantin J, Woloszynek S, Carpenter AE, Shrikumar A, Xu J, Cofer EM, Lavender CA, Turaga SC, Alexandari AM, Lu Z, Harris DJ, DeCaprio D, Qi Y, Kundaje A, Peng Y, Wiley LK, Segler MHS, Boca SM, Swamidass SJ, Huang A, Gitter A, Greene CS, Opportunities and obstacles for deep learning in biology and medicine, J. R. Soc. Interface 15 (141) (2018) 20170387 〈https://pubmed.ncbi.nlm.nih.gov/29618526〉. [PubMed: 29618526]

[7]. Hie B, Bryson BD, Berger B, Leveraging uncertainty in machine learning accelerates biological discovery and design, Cell Systems 11 (5) (2020) 461–477. [PubMed: 33065027]

[8]. Mervin LH, Johansson S, Semenova E, Giblin KA, Engkvist O, Uncertainty quantification in drug design. Drug Discov. Today 26 (2) (2020) 474–489.

[9]. Yang K, Swanson K, Jin W, Coley C, Eiden P, Gao H, Guzman-Perez A, Hopper T, Kelley B, Mathea M, Palmer A, Settels V, Jaakkola T, Jensen K, Barzilay R, Analyzing learned molecular representations for property prediction, J. Chem. Inform. Model 59 (8) (2019) 3370–3388, 10.1021/acs.jcim.9b00237

[10]. Mervin LH, Cao Q, Barrett IP, Firth MA, Murray D, McWilliams L, Haddrick M, Wigglesworth M, Engkvist O, Bender A, Understanding cytotoxicity and cytostaticity in a high-throughput screening collection, ACS Chem. Biol 11 (11) (2016) 3007–3023 〈10.1021/acschembio.6b0053〉. [PubMed: 27571164]

[11]. Bosc N, Atkinson F, Felix E, Gaulton A, Hersey A, Leach AR, Large scale comparison of qsar and conformal prediction methods and their applications in drug discovery, J.Cheminform 11 (1) (2019) 4 〈10.1186/s13321-018-0325-4〉 [PubMed: 30631996]

[12]. Rodríguez-Pérez R, Vogt M, Bajorath J, Influence of varying training set composition and size on support vector machine-based prediction of active compounds, *J*. Chem. Inform. Model 57 (4) (2017) 710–716 〈10.1021/acs.jcim.7b00088〉.

[13]. Abdar M, Pourpanah F, Hussain S, Rezazadegan D, Liu L, Ghavamzadeh M, Fieguth P, Cao X, Khosravi A, Acharya UR, Makarenkov V, Nahavandi S, A review of uncertainty quantification in deep learning: techniques, applications and challenges, Inform. Fusion 76 (2021) 243–297 〈https://www.sciencedirect.com/science/article/pii/S1566253521001081〉.

[14]. Fox CR, Ülkümen G, Distinguishing two dimensions of uncertainty, Perspect. Think. Judg. Decision Mak (2011) 21–35.

[15]. Kononenko I, Bayesian neural networks, Biol. Cybernet 61 (5) (1989) 361–370 〈10.1007/BF00200801〉.

[16]. MacKay DJC, A practical bayesian framework for backpropagation networks, Neural Comput. 4 (3) (1992) 448–472 〈10.1162/neco.1992.4.3448〉.

[17]. Bishop C, Neural Networks for Pattern Recognition, Oxford University Press, USA, 1995.

[18]. Gal Y, Ghahramani Z, Dropout as a Bayesian approximation: Representing model uncertainty in deep learning, in: Proceedings of the Thirty Third International Conference on, ser. Proceedings of Research Balcan MF and Weinberger KQ, Eds., PMLR, 20-22 Jun 2016, New York, New York, USA, 1050–1059, 48. 〈https://proceedings.mlr.press/v48/gal16.html〉

[19]. Lakshminarayanan B, Pritzel A, Blundell C, Simple and scalable predictive uncertainty estimation using deep ensembles, in: Proceedings of the Thirty First International Conference on Neural Information Processing Systems, ser. NIPS'17, Curran Associates Inc., Red Hook, NY, USA, 2017, pp. 6405–6416.

[20]. Du H, Barut E, Jin F, Uncertainty quantification in cnn through the bootstrap of convex neural networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, 35, 13, 12 078–12 085, May 2021. 〈https://ojs.aaai.org/index.php/AAAI/article/view/17434〉

[21]. Malinin A, Gales M, Predictive uncertainty estimation via prior networks, in: Systems S, Bengio H, Wallach H, Larochelle K, Grauman N, Garnett Cesa-Bianchi, R. (Eds.), Advances in Neural Information Processing, 31 Curran Associates, Inc, 2018, 〈https://proceedings.neurips.cc/paper/2018/file/3ea2db50e62ceefceaf70a9d9a56a6f4-Paper.pdf〉.

[22]. Wang D, Yu J, Chen L, Li X, Jiang H, Chen K, Zheng M, Luo X, A hybrid framework for improving uncertainty quantification in deep learning-based qsar regression modeling, J.Cheminform 13 (1) (2021) 69〈10.1186/s13321-021-00551-x〉. [PubMed: 34544485]

[23]. Qiu X, Meyerson E, Miikkulainen R, Quantifying point-prediction uncertainty in neural networks via residual estimation with an I/O kernel, in International Conference on Learning Representations, 2020. 〈https://openreview.net/forum?id=rkxNh1Stvr〉

[24]. Rasmussen CE, Williams CKI, Gaussian Processes for Machine Learning, MIT Press, 2006.

[25]. Schaduangrat N, Lampa S, Simeon S, Gleeson MP, Spjuth O, Nantasenamat C, Towards reproducible computational drug discovery, J. Cheminform 12 (1) (2020) 9〈10.1186/s13321-020-0408-x〉. [PubMed: 33430992]

[26]. Minnich AJ, McLoughlin K, Tse M, Deng J, Weber A, Murad N, Madej BD, Ramsundar B, Rush T, Calad-Thomson S, Brase J, Allen JE, Ampl: a data-driven modeling pipeline for drug discovery, J. Chem. Inform. Model 60 (4) (2020) 1955–1968 〈10.1021/acs.jcim.9b01053〉 .

[27]. Scalia G, Grambow CA, Pernici B, Li Y-P, Green WH, Evaluating scalable uncertainty estimation methods for deep learning-based molecular property prediction, J. Chem. Inform. Model 60 (6) (2020) 2697–2717 〈10.1021/acs.jcim.9b00975〉 .

[28]. Hinton GE, van Camp D, Keeping the neural networks simple by minimizing the description length of the weights, in: Proceedings of the Sixth Annual Conference on Computational Learning Theory, ser. COLT ' 93dl, Association for Computing Machinery, New York, NY, USA 1993, 5–13. 〈10.1145/168304.168306〉

[29]. Neal RM, Bayesian Learning for Neural Networks, Springer-Verlag, Berlin, Heidelberg, 1996.

[30]. Wilson AG, Hu Z, Salakhutdinov R, Xing EP, Deep kernel learning, in Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, ser. Proceedings of Machine Learning Research Gretton A and Robert CC, Eds., 51.Cadiz, Spain: PMLR, 09-11 May 2016, 370–378. 〈https://proceedings.mlr.press/v51/wilson16.html〉 .

[31]. Tang Y-H, de Jong WA, Prediction of atomization energy using graph kernel and active learning, J. Chem. Phys 150 (4) (2019) 044107 〈10.1063/1.5078640〉 . [PubMed: 30709286]

[32]. Han K, Lakshminarayanan B, Liu JZ, Reliable graph neural networks for drug discovery under distributional shift, in: Proceedings of the NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications, 2021. 〈https:/openreview.net/forum?id=311QRRkfrep〉

[33]. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res 15 (56) (2014) 1929–1958 〈http://jmlr.org/papers/v15/srivastava14a.html〉 .

[34]. Graves A, Practical variational inference for neural networks, in: Proceedings of the Twenty Fourth International Conference on Neural Information Processing Systems, ser. NIPS'11.Red Hook, Curran Associates Inc. ,NY, USA, 2011, 2348–2356.

[35]. Kingma DP, Salimans T, Welling M, Variational dropout and the local reparameterization trick, in: Proceedings of the Twenty Eighth International Conference on Neural Information Processing Systems - Volume 2, ser. NIPS'15.Cambridge, MA, USA: MIT Press, 2015, 2575–2583.

[36]. Louizos C, Welling M, Structured and efficient variational deep learning with matrix gaussian posteriors, in: Proceedings of the Thirty Third International Conference on International Conference on Machine Learning, 48, ser. ICML'16. JMLR.org, 2016, 1708–1716.

[37]. Mathea M, Klingspohn W, Baumann K, Chemoinformatic classification methods and their applicability domain, Mol. Inform 35 (2016), pp. 160–180 〈 https://onlinelibrary.wiley.com/doi/abs/10.1002/minf.201501019〉 . [PubMed: 27492083]

[38]. Knorr EM ,Ng RT, A unified notion of outliers: Properties and computation, in: Proceedings of the Third International Conference on Knowledge Discovery and Data Mining, ser. KDD'97.AAAI Press, 1997, 219–222.

[39]. Tax D, Duin R, Outlier detection using classifier instability, in: Advances in Pattern Recognition, Lecture notes in Computer Science, Springer, 1998, pp. 593–601.

[40]. Mendez D, Gaulton A, Bento AP, Chambers J, De Veij M, Félix E, Magariños M, Mosquera J, Mutowo P, Nowotka M, Gordillo-Marañón M, Hunter F, Junco L, Mugumbate G, Rodriguez-Lopez M, Atkinson F, Bosc N, Radoux C, Segura-Cabrera A, Hersey A, Leach A, ChEMBL: towards direct deposition of bioassay data, Nucl. Acids Res 47 (D1) (2019) D930–D940 〈10.1093/nar/gky1075〉 . [PubMed: 30398643]

[41]. Davies M, Nowotka M, Papadatos G, Dedman N, Gaulton A, Atkinson F, Bellis L, Overington JP. Chembl web services: streamlining access to drug discovery data and utilities, Nucl. Acids Res 43 (W1) (2015) W612–W620 〈https://pubmed.ncbi.nlm.nih.gov/25883136〉 . [PubMed: 25883136]

[42]. Chemical ComputingGroup ULC, Molecular Operating Environment (MOE), Montreal, Canada, 2022.

[43]. McLoughlin KS, Jeong CG, Sweitzer TD, Minnich AJ, Tse MJ, Bennion BJ, Allen JE, Calad-Thomson S, Rush TS, Brase JM, Machine learning models to predict inhibition of the bile salt export pump, J. Chem. Inform. Model 61 (2) (2021) 587–602 〈10.1021/acs.jcim.0c00950〉 .

[44]. Landrum G, RDKit: Open-source Cheminformatics. 〈http://www.rdkit.org〉 .

[45]. Massova I, Kollman PA, Combined molecular mechanical and continuum solvent approach (MM-PBSA/GBSA) to predict ligand binding, Perspect. Drug Discov. Des 18 (1) (2000) 113–135 〈10.1023/A:1008763014207〉 .

[46]. Mongan J, Simmerling C, McCammon JA, Case DA, Onufriev A, Generalized born model with a simple, robust molecular volume correction, J. Chem. Theory Comput 3 (1) (2007) 156–169 〈10.1021/ct600085e〉 . [PubMed: 21072141]

[47]. van der Maaten L, Hinton GE, Visualizing high-dimensional data using t-sne, J. Mach. Learn. Res 9 (2008) 2579–2605.

[48]. Ramsundar B, Eastman P, Walters P, Pande V, Leswing K, Wu Z, Deep Learning for the Life Sciences, O'Reilly Media, 2019.

[49]. Barrett JP, The coefficient of determination-some limitations, Am Stat. 28 (1) (1974) 19–20 ([Online]. Available), 〈10.1080/00031305.1974.10479056〉 .

[50]. Hirschfeld L, Swanson K, Yang K, Barzilay R, Coley CW, Uncertainty quantification using neural networks for molecular property prediction, J Chem. Inform. Model 60 (8) (2020) 3770–3780 〈10.1021/acs.jcim.0c00502〉 .

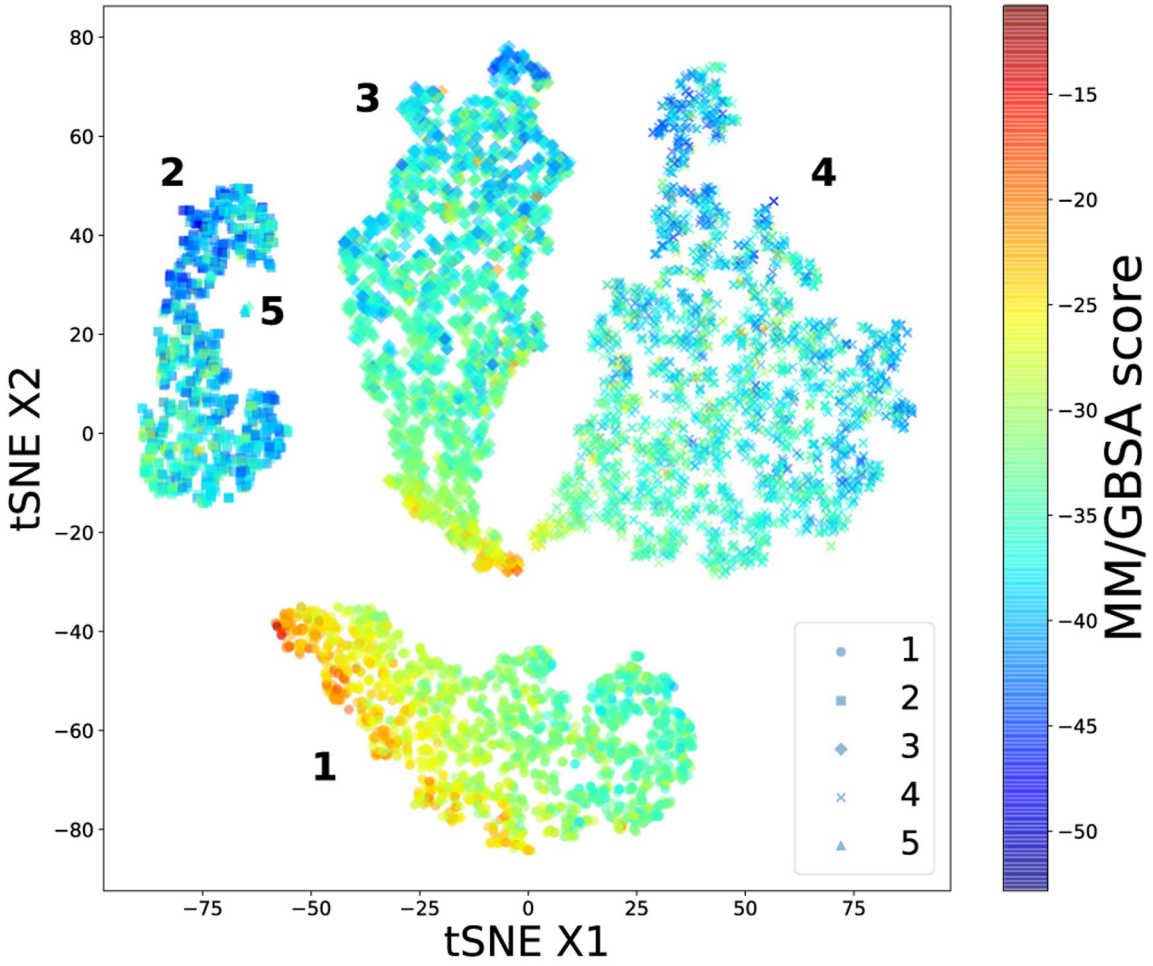

Fig. 1.
Clusters using the two tSNE coordinates.

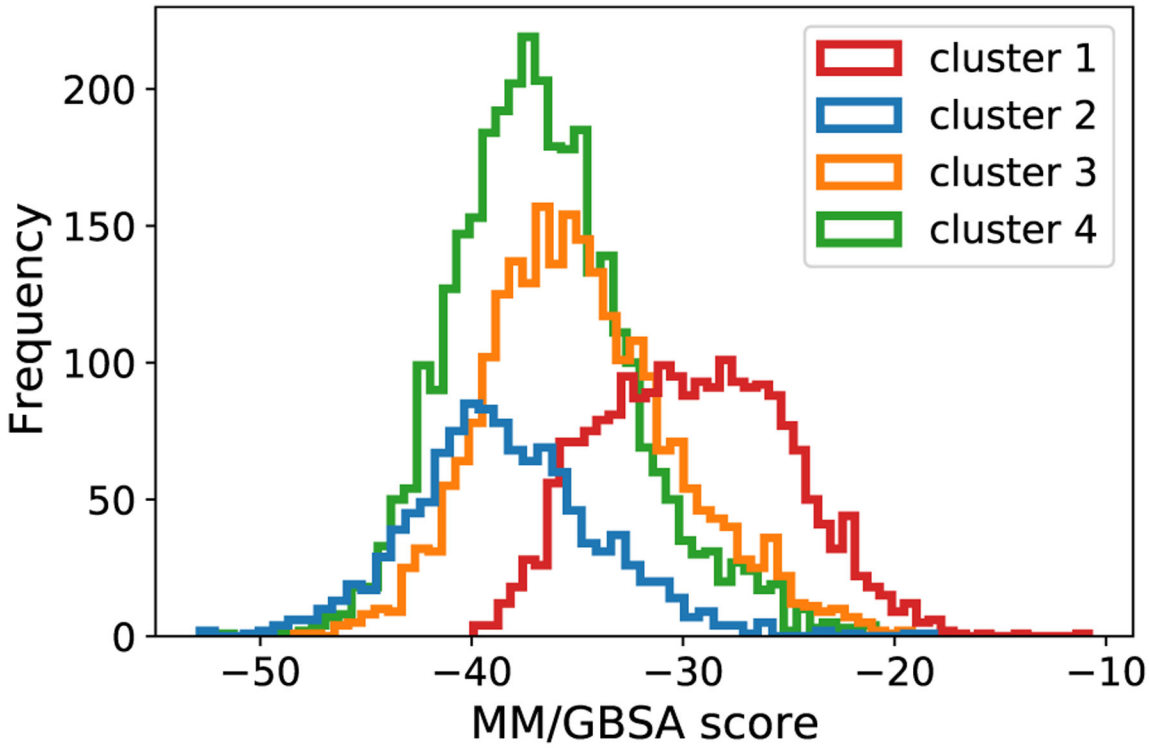

Fig. 2.
MM/GBSA score distribution in each cluster.

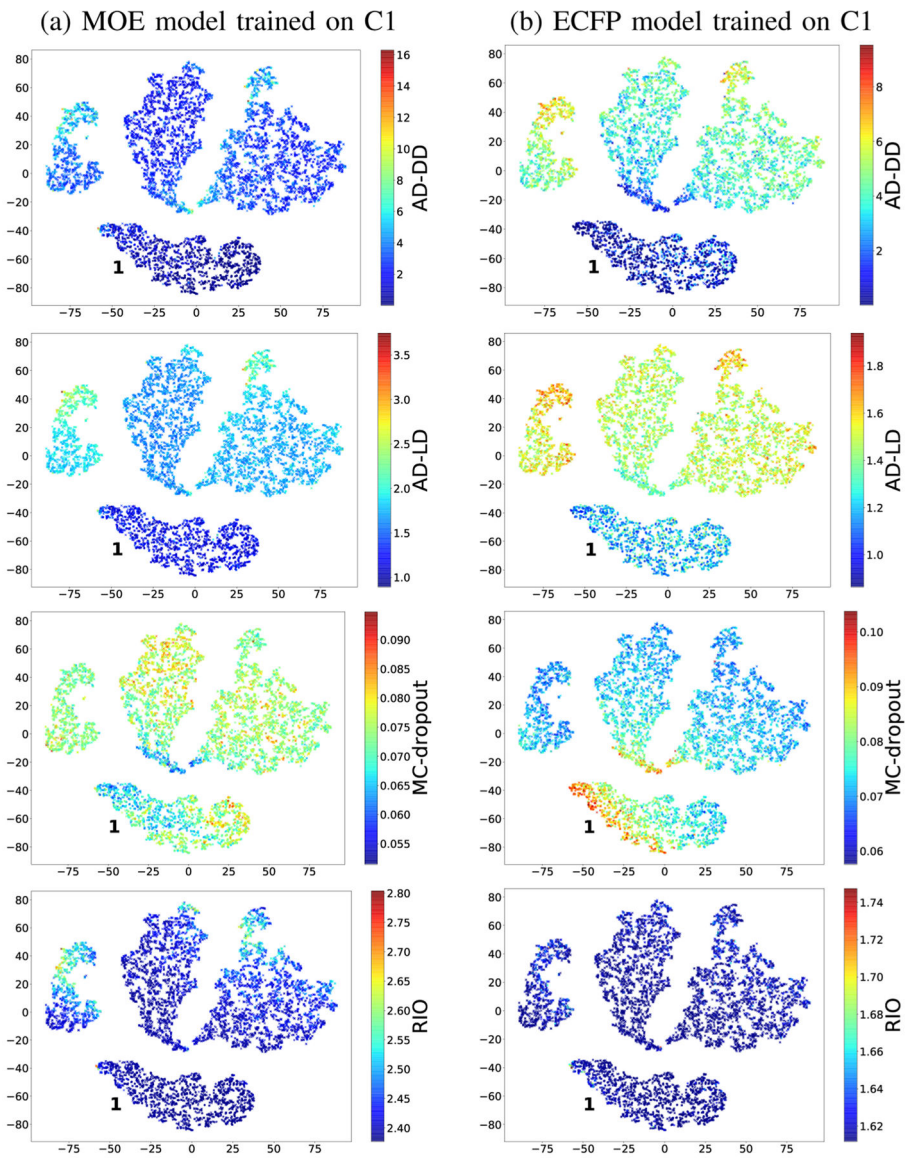

Fig. 3.

Uncertainty values for the models trained on cluster 1 (C1). The bold number on the plot indicates the training set cluster. The x and y axes are the two coordinates from tSNE using top correlated MOE features.

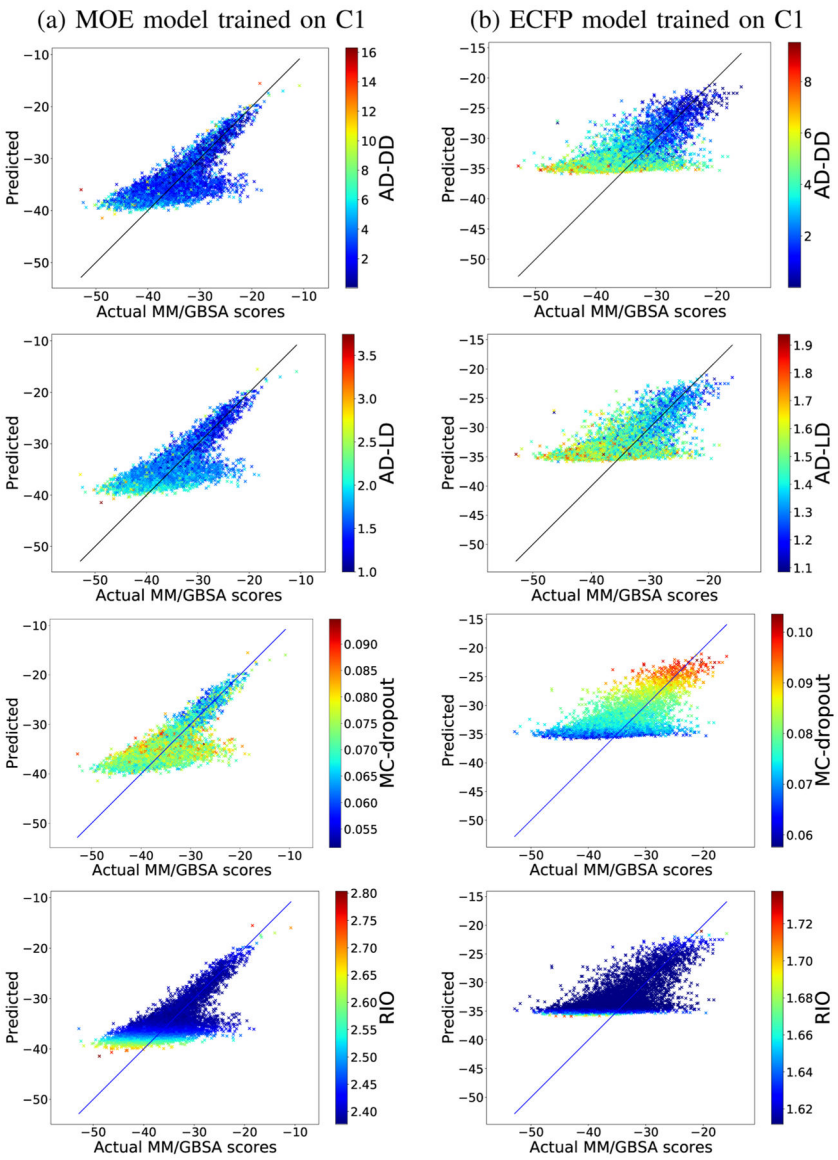

Fig. 4.

Test set actual MM/GBSA scores versus predicted plot from the model trained on cluster 1 (C1). The diagonal represents the values where the predicted is equal to the actual MM/GBSA score.

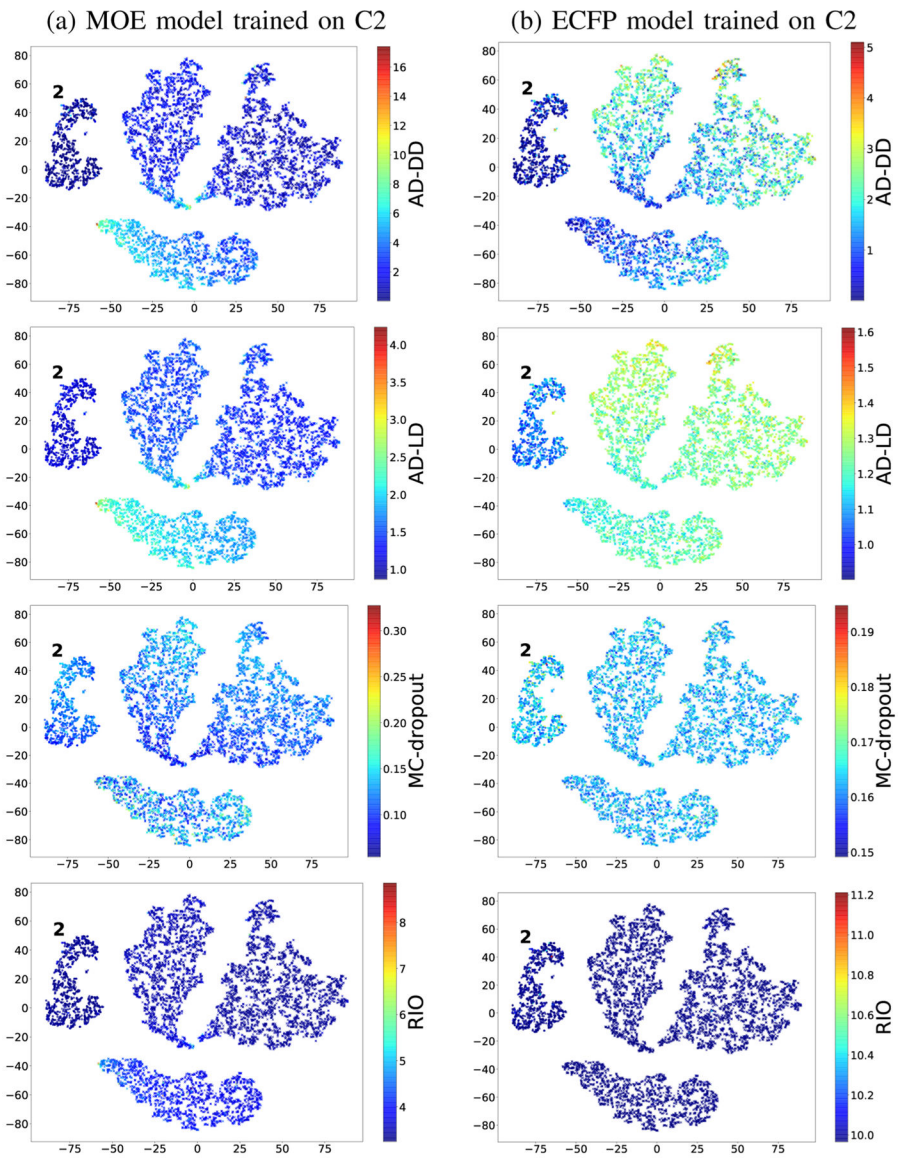

Fig. 5.
Uncertainty values for the model trained on cluster 2 (C2).

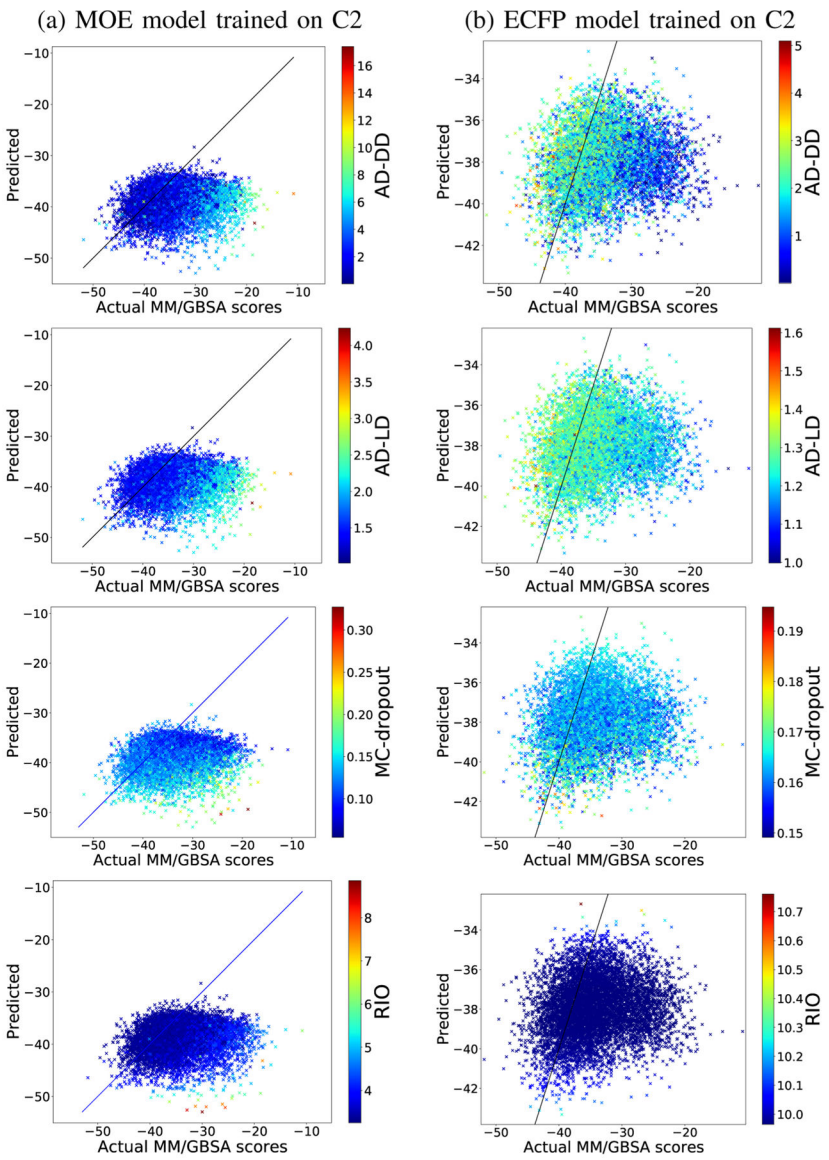

Fig. 6.

Test set actual MM/GBSA scores versus predicted plot from the model trained on cluster 2 (C2).

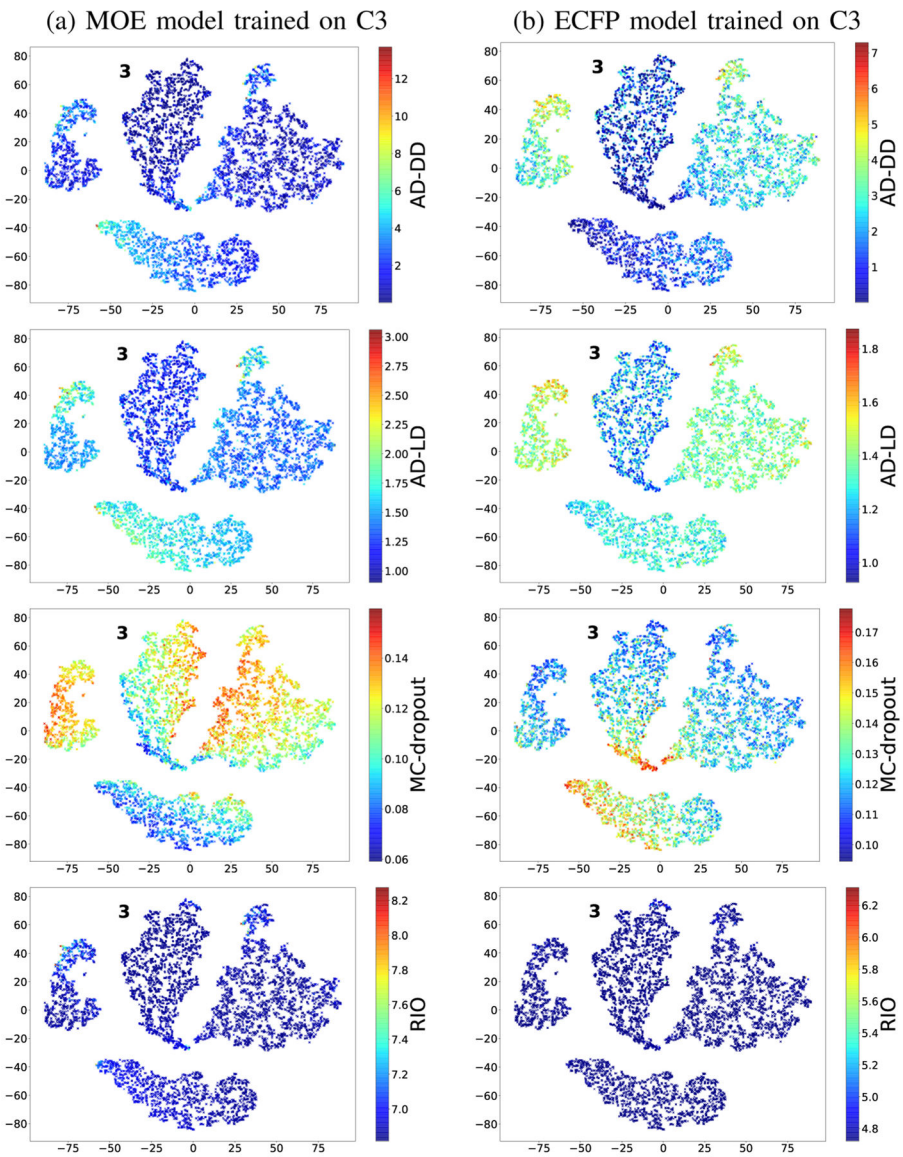

Fig. 7.
Uncertainty values for the model trained on cluster 3 (C3).

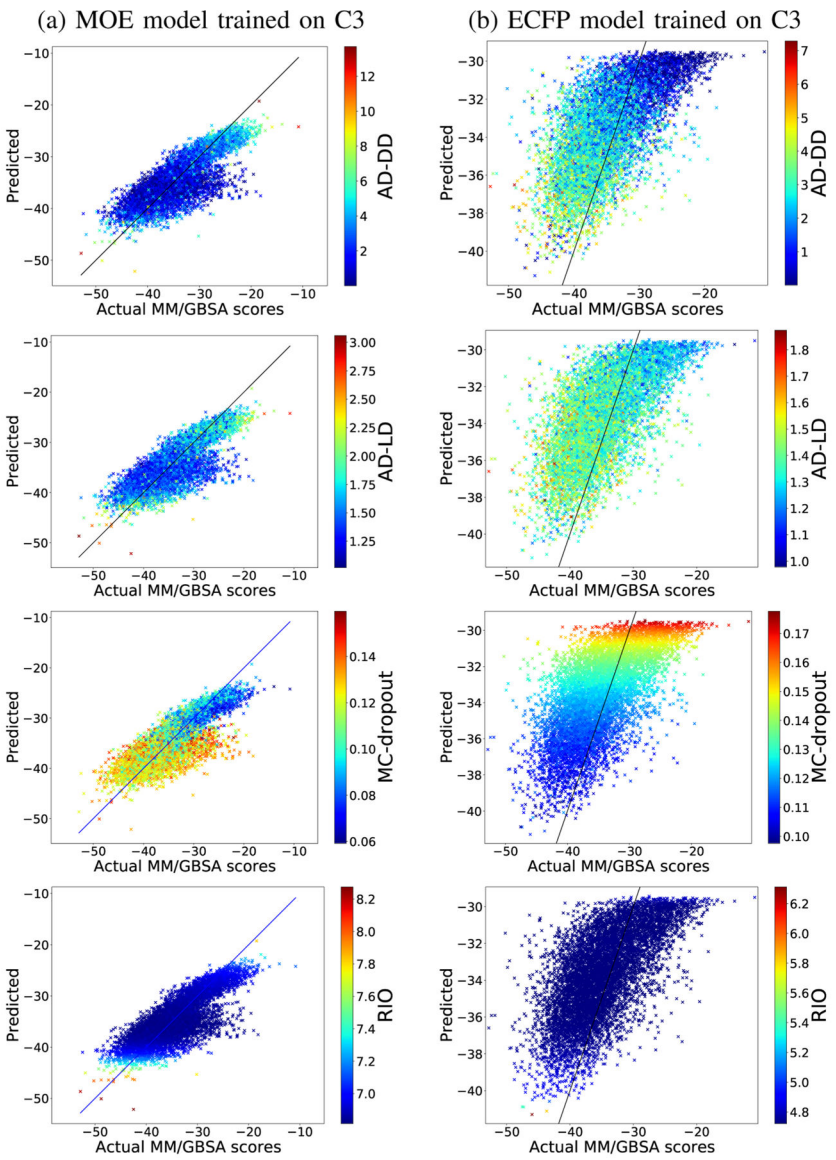

Fig. 8.

Test set actual MM/GBSA scores versus predicted plot from the model trained on cluster 3 (C3).

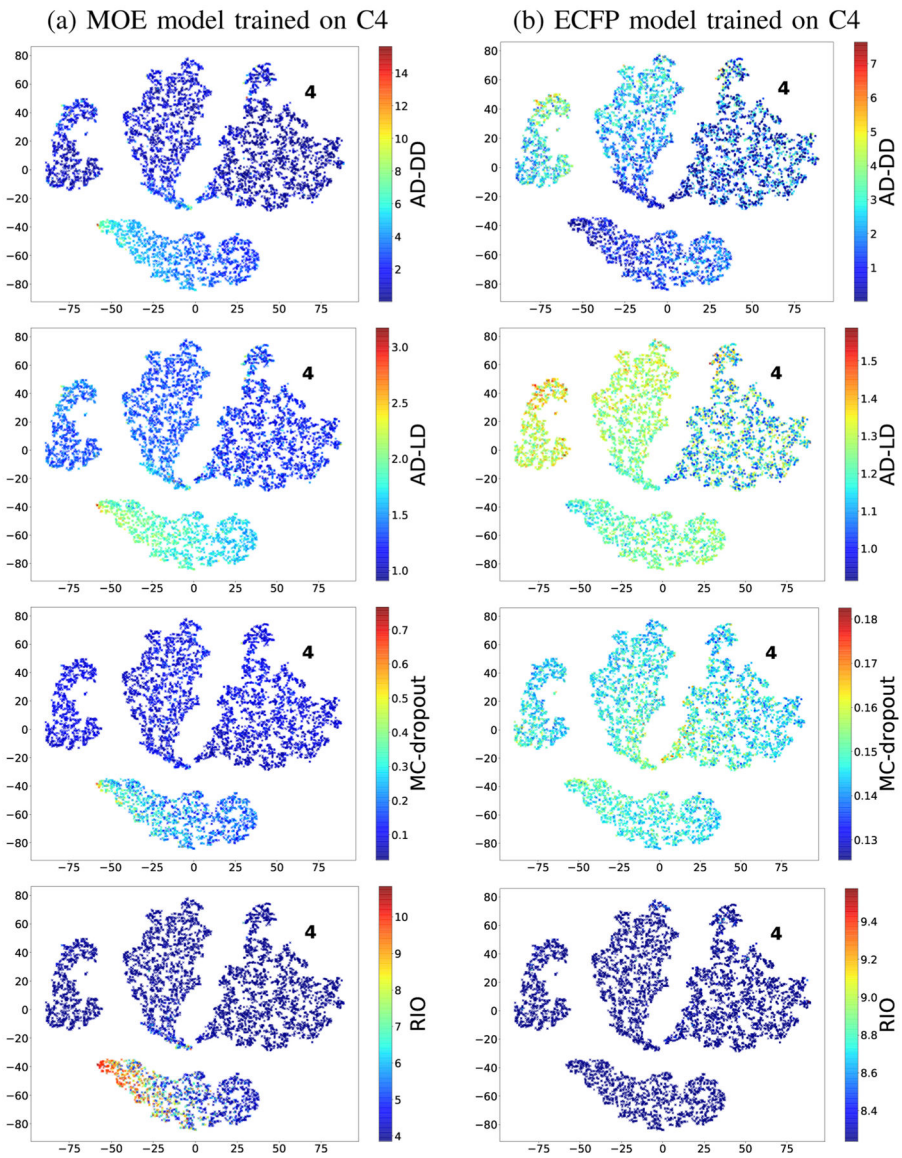

Fig. 9.
Uncertainty values for the model trained on cluster 4 (C4).

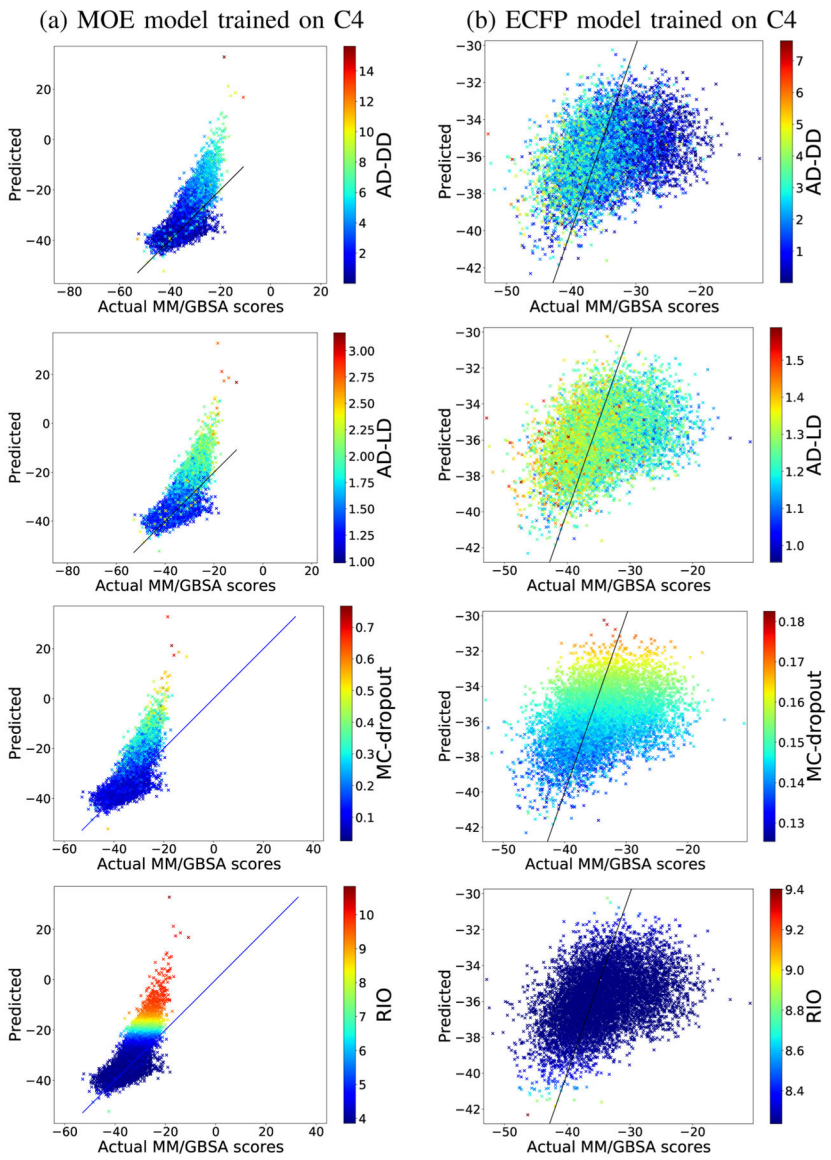

Fig. 10.

Test set actual MM/GBSA scores versus predicted plot from the model trained on cluster 4 (C4).

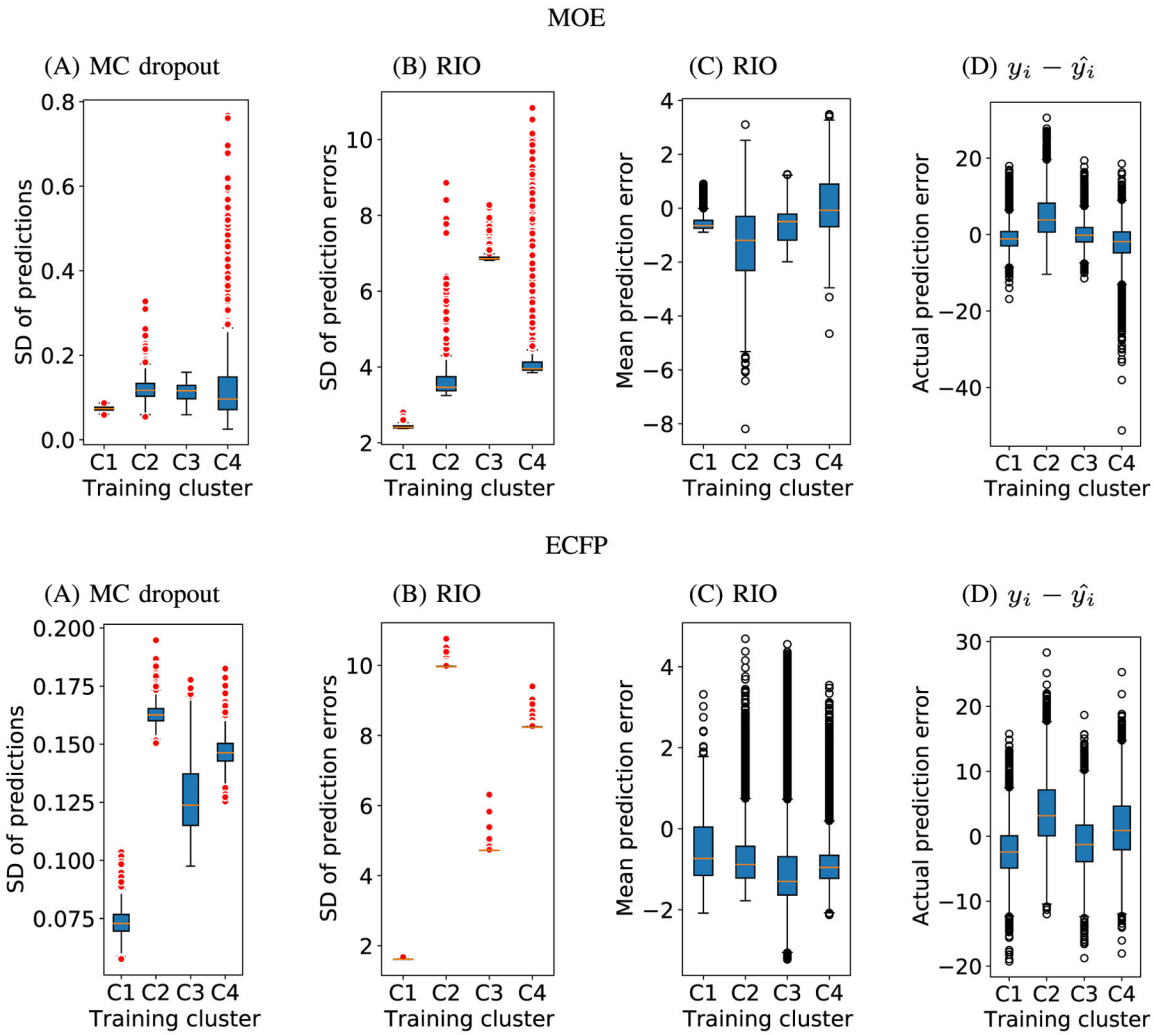

Fig. 11.

Box plots for the test set uncertainties and prediction errors from the models trained with MOE features and ECFP features.The first column C1 indicates the model trained on cluster 1 and the corresponding data points in the box plot are from the test set, composed of cluster 2, 3 and 4. (A) MC dropout uses the standard deviation of the predictions as its uncertainty estimation for the NN point prediction. For all the test sets MC dropout gives the uncertainty values in a narrow range. (B) RIO provides the standard deviation of the prediction errors (the residuals) to the NN point prediction. (C) RIO also estimates the mean of the prediction errors to the NN point prediction. (D) The actual prediction error is the actual MM/GBSA score subtracted by the NN predicted value. The actual prediction errors are close to zero with some outliers that have large errors. Although RIO estimates most mean prediction errors close to zero, it gives high standard deviations to the predictions made by the models trained on cluster 2, 3 and 4, indicating high uncertainties in some data points.

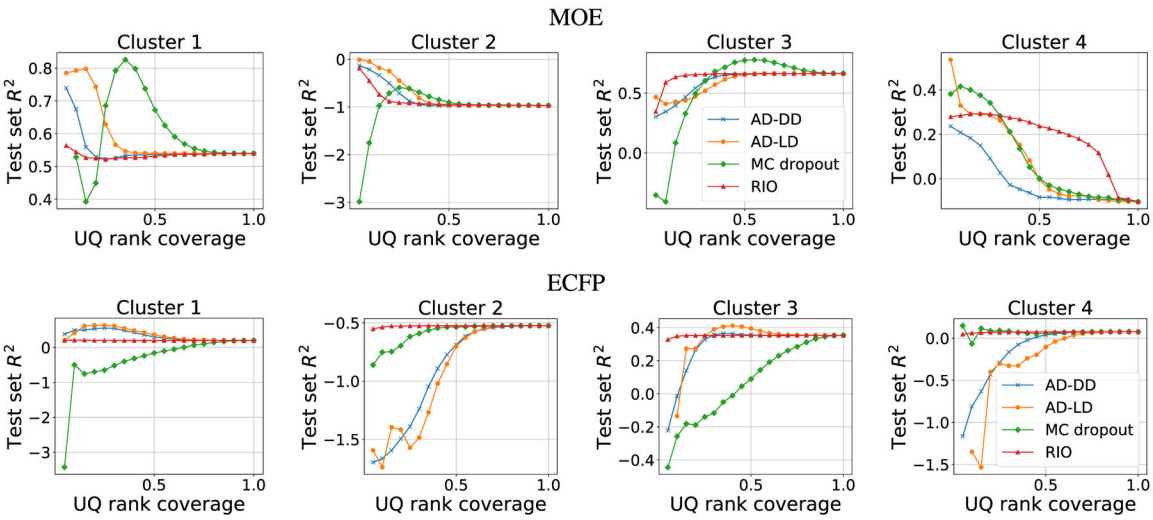

Fig. 12.

Ranked UQ values versus the prediction performance on the test sets from the models trained with MOE and ECFP features.

Table 1

For visualization we select the subset of MOE features whose values are highly correlated to the MM/GBSA scores. There are 22 features having their Pearson correlation coefficient larger than 0.5 or lower than −0.5.

Rank	MOE feature	Correlation
1	BCUT_SMR_0_per_atom	−0.738
2	BCUT_SLOGP_0_per_atom	−0.735
3	GCUT_SMR_3_per_atom	0.729
4	GCUT_SLOGP_3_per_atom	0.725
5	BCUT_SMR_3_per_atom	0.722
6	VAdjEq_per_atom	0.721
7	a_count	−0.694
8	vsurf_R_per_atom	0.688
9	vsurf_CW1_per_atom	0.686
10	PEOE_RPC-_per_atom	0.682
11	RPC-_per_atom	0.674
12	opr_leadlike_per_atom	0.668
13	vsurf_G_per_atom	0.664
14	balabanJ_per_atom	0.662
15	weinerPath	−0.658
16	VAdjMa_per_atom	0.644
17	VDistMa_per_atom	0.621
18	VDistEq_per_atom	0.568
19	ast_violation	−0.552
20	ast_fraglike	0.538
21	std_dim1	−0.528
22	pmi_per_atom	−0.506

Table 2

Model performance in R^2 on each cluster using MOE descriptors.

MOE	Test set			
Training set	**Cluster 1**	**Cluster 2**	**Cluster 3**	**Cluster 4**
Cluster 1	**0.841**	0.088	0.485	0.237
Cluster 2	−5.898	**0.721**	−0.652	−0.029
Cluster 3	**0.685**	0.256	0.580	0.363
Cluster 4	−3.332	0.182	0.323	**0.455**

Table 3

Model performance in R^2 on each cluster using ECFP features.

ECFP	Test set			
Training set	**Cluster 1**	**Cluster 2**	**Cluster 3**	**Cluster 4**
Cluster 1	**0.767**	−0.834	0.222	−0.317
Cluster 2	−3.917	**0.399**	−0.453	0.046
Cluster 3	0.036	−0.497	**0.455**	−0.054
Cluster 4	−1.885	−0.053	0.159	**0.276**