
# Practically Significant Method Comparison Protocols for Machine Learning in Small Molecule Drug Discovery

JCIM JOURNAL OF CHEMICAL INFORMATION AND MODELING

Editors' Choice

pubs.acs.org/jcim

Perspective

# Practically Significant Method Comparison Protocols for Machine Learning in Small Molecule Drug Discovery

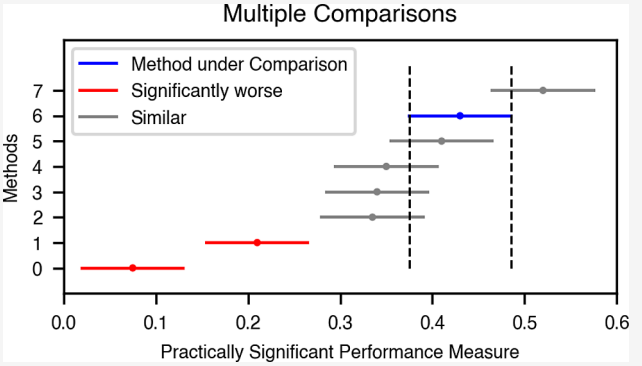
Jeremy R. Ash,[III] Cas Wognum,*[,III] Raquel Rodríguez-Pérez, Matteo Aldeghi, Alan C. Cheng, Djork-Arné Clevert, Ola Engkvist, Cheng Fang, Daniel J. Price, Jacqueline M. Hughes-Oliver, and W. Patrick Walters

Cite This: *J. Chem. Inf. Model.* 2025, 65, 9398−9411

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

**ABSTRACT:** Machine Learning (ML) methods that relate molecular structure to properties are frequently proposed as in silico surrogates for expensive or time-consuming experiments. In small molecule drug discovery, such methods inform high-stakes decisions like compound synthesis and in vivo studies. This application lies at the intersection of multiple scientific disciplines. When comparing new ML methods to baseline or state-of-the-art approaches, statistically rigorous method comparison protocols and domain-appropriate performance metrics are essential to ensure replicability and ultimately the adoption of ML in small molecule drug discovery. This paper proposes a set of guidelines to incentivize rigorous and domain-appropriate techniques for method comparison tailored to small molecule property modeling. These guidelines, accompanied by annotated examples using open-source software tools, lay a foundation for robust ML benchmarking and thus the development of more impactful methods.






## 1. INTRODUCTION

In drug discovery, expensive and time-consuming experiments are used to profile molecules and gain insights into their therapeutic potential. Such experimental assays are typically organized in a cascade, where subsequent experiments test fewer molecules at a higher cost per molecule. As in silico surrogates to such experiments, both regression and classification Machine Learning (ML) models can be trained to estimate molecular properties (i.e., experimental results) from chemical structure. Such models could inform drug design and prioritize experiments by scoring a set of candidate molecules. These ML models thus inform high-stakes decisions and help drug discovery research progress more quickly and efficiently. Hence, it is important that models provide reliable forecasting of experimental results.

In this paper we will define a method as a procedure for creating a predictive model given training data, and a model as the output of said procedure. When deploying a new model in industry or publishing a new method in the scientific literature, rigorous method comparison is essential. In an industrial setting, replacing established methods requires a significant investment that must be justified by reliable results; this reliability is demonstrated through rigorous statistical comparisons between the proposed and existing models. Furthermore, since the scientists using these models are often not their developers, building trust within interdisciplinary teams requires that the chosen performance metrics accurately represent performance once deployed in real drug discovery programs. Similarly, in the scientific literature, a new method's contribution is contextualized by comparing its performance against both simple baselines and the current state-of-the-art to justify follow-up research. Therefore, in both industry and academia, appropriate statistical tests and performance metrics are critical tools needed to identify robust improvements.[1]

These circumstances highlight the need for statistically rigorous method comparison protocols and domain-appropriate techniques. Because there is inherent stochasticity in the data used to train models and in the modeling methods themselves, it is necessary to compare populations of models different methods generate (e.g., through cross-validation). Furthermore, appropriate statistical methods should be used to compare performance distributions and determine whether the differences could be attributed to random chance. Similarly performing methods can produce seemingly large differences, especially with the classically smaller (i.e., $\leq 10^4$ samples),

**Received:** July 9, 2025
**Revised:** August 21, 2025
**Accepted:** August 21, 2025
**Published:** September 11, 2025

imbalanced, and noisy data sets that are publicly available in drug discovery. In these contexts, large observed performance differences can show high sensitivity to small changes in the data, such as the addition or removal of a few data points. To account for this, tests to establish the statistical significance of differences are common in many other fields, such as engineering and clinical medicine. However, this practice has been largely absent from ML-based cheminformatics literature.[2−4] For ML-based property modeling, most benchmark studies simply report mean performance values over a series of replicates, disregarding that distributions are being compared.

Furthermore, despite the importance of hypothesis testing, establishing that there is a statistically significant difference does not directly imply practical significance. In molecular property modeling, statistically significant differences in performance distributions might not translate to key decisional impact for drug discovery, such as what compounds to synthesize. Method comparison protocols should, therefore, also analyze the effect size and use performance metrics that better translate to decisional impact.

Proposing statistically rigorous and domain-appropriate method comparison protocols for small molecule drug discovery is an inherently difficult task due to its multidisciplinary nature. Lacking such protocols or methodological guidelines risks a disconnect between perceived progress and real-world impact, slowing the adoption of ML methods in small molecule drug discovery.

In this work, we first establish the importance of statistical testing in Section 2. We then present a set of beginner-friendly guidelines for method comparison in Section 3, tailored to small molecule property modeling applications. In Section 4, we present annotated code examples to accompany these guidelines. The code examples use open-source software to demonstrate each step. We cover several key aspects, including cross-validation techniques, post hoc tests, multiple comparisons, visualizations, and effect size. All of the code can be found on Github. Finally, in Section 5, we summarize the method comparison protocol and suggest future research directions.

## 2. MOTIVATION: REPLICABILITY CRISIS IN ML-BASED SCIENCE

As in any other scientific discipline, in ML-based drug discovery experiments are carried out to improve our understanding of the system under study. These experiments add to a shared body of knowledge that new research can then build upon. Therefore, the adherence to good scientific principles to obtain reliable and replicable insights from experiments is key.[5] Otherwise, research directions might be pursued based on fragile assumptions.

In a recent survey, the majority of researchers in the broader scientific community indicated that they have failed to replicate others' or even their own published results, which led 90% of them to proclaim a replicability crisis.[6] While this is thus not specific to ML-based science, researchers were also unable to replicate a large fraction of investigations from the ML community.[7] If a method is claimed to be superior to the current state of the art on a benchmark, then we expect this result to be replicable by other ML scientists or on similar benchmarks, but this is frequently not the case.

It is important to differentiate the terms replicability and reproducibility. Authors at times use the terms interchangeably, but in many fields (e.g., statistics, computational biology) there are distinct meanings. We follow the convention of the National Academies of Sciences, Engineering, and Medicine.[8] We define replicability to mean the ability of an independent group to recreate results on a new data set collected under the same conditions. This is a stronger condition than reproducibility which is the ability for an independent group to recreate results if given access to the same code and data. While researchers often focus on reproducibility in ML research, replicability is the ultimate goal.[9]

McDermott et al.[10] identify three main components of replicability:

- Technical Replicability: Can results be replicated under technically identical conditions?
- Statistical Replicability: Can results be replicated under statistically identical conditions?
- Conceptual Replicability: Can results be replicated under conceptually identical conditions?

Technical replicability refers to the ability to replicate results using the code and data shared by authors. Conceptual replicability refers to ability to replicate results under conditions that match the conceptual description of the study. For example, results should be able to be replicated when methods are applied to a new data set generated under the same conditions.

In this work we focus on statistical replicability. Statistical replicability is demonstrated when the same results are observed across experiments performed under equivalent conditions. To draw a parallel with wet lab experiments, statistical replicability is often established by performing several replicates of the same experiment (i.e., same day, instrument, conditions). In ML research, statistical replicability can be assessed using a single data set with approaches like data resampling. Considering statistical replicability is important because it can eliminate results that are confidently not reproducible, which has the potential to substantially reduce the number of false positives (i.e., overly optimistic results).[11−14]

While some researchers in ML recognize the importance of statistical replicability,[10] there is still substantial room for improvement. In fields such as computer vision and natural language processing, where fit-for-purpose data sets with millions of observations are available, a statistical replicability assessment is less critical because even small differences are likely statistically significant. With such extremely large data sets, an in-depth statistical analysis may also be computationally infeasible. In contrast, data sets in small molecule property modeling tend to be expensive to generate. They are substantially smaller than in these other ML fields and tend to be highly heterogeneous, imbalanced, and noisy. All of these factors increase the expected variability in performance metrics one will see when carrying out several random data splits, making statistical replicability analysis essential.

There are many reasons that contribute to the gap between the perceived importance of statistical replicability and the usage of appropriate statistical methods in research papers. Few user-friendly tools exist for these analyses, and the statistical knowledge required to perform them is often a barrier. Beyond these more technical reasons, researchers and research institutions also play a role, e.g. replicability and robust statistical analyses could be incentivized more.[15] We try to address this gap by providing clear guidelines, annotated examples, and by implementing the suggested techniques using open-source software to simplify the adoption of best practices.



https://doi.org/10.1021/acs.jcim.5c01609
J. Chem. Inf. Model. 2025, 65, 9398−9411

## 3. METHOD COMPARISON GUIDELINES

In this section, we will review best practices for method comparison and translate these to a set of guidelines specific to small molecule property modeling for drug discovery. Figure 1 summarizes these guidelines and serves as a visual table of contents to easily navigate this paper.

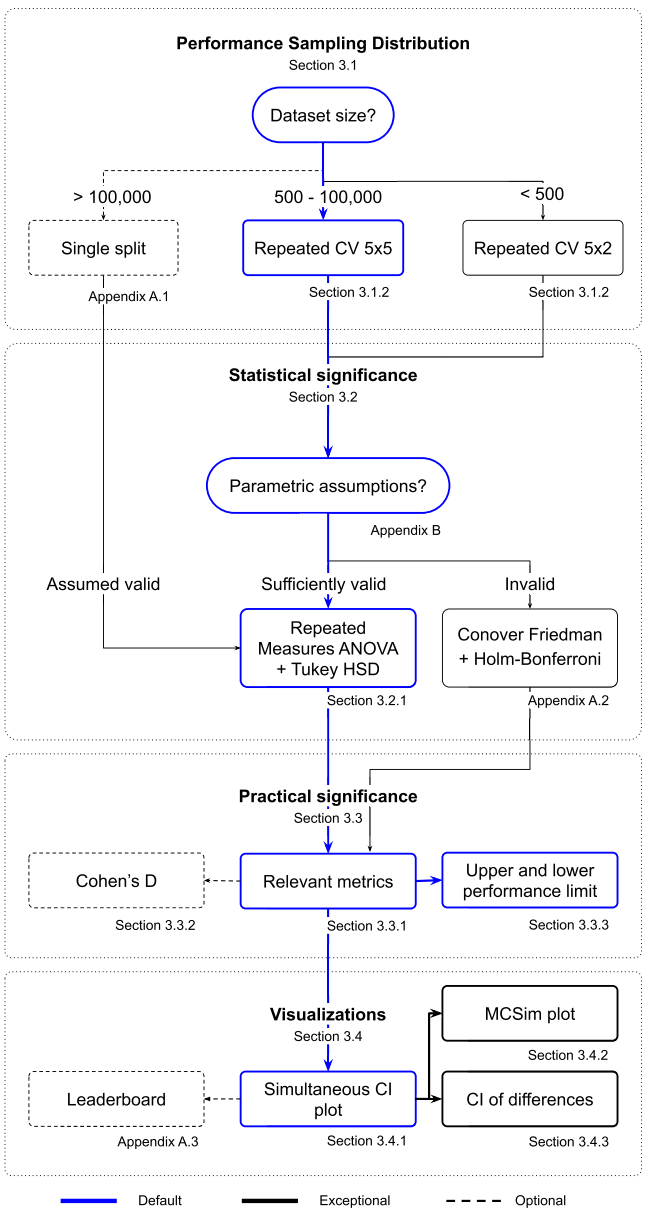



**Figure 1.** Method comparison guidelines presented in this work are summarized by this decision tree. The path through the decision tree shown in blue should apply to most use cases, but solutions for exceptional cases are presented as well.

Throughout this section, we will recommend ways to examine a model's performance and the assumptions behind each proposed technique. Please keep in mind, however, that these guidelines are not a recipe to blindly follow and, in practice, each case scenario will likely require its own unique considerations. Based on the characteristics of the data set or project's goal, deviations from this workflow are reasonable. Transparency is key in the absence of a perfect solution for every scenario.

We will discuss different techniques for sampling the performance distribution in Section 3.1. Then, in Section 3.2, we will discuss different statistical tests that can be used to compare the performance sampling distributions. In Section 3.3, we will explain the importance of domain-appropriate performance metrics in achieving practical significance. Finally, Section 3.4 will discuss how to present the results of these tests.

**3.1. Performance Sampling Distribution.** New methods are often benchmarked against control baselines and state-of-the-art methods to contextualize performance. This type of comparison is typically done using retrospective benchmarks for the sake of practicality, where a data set is split in training and test sets. The more representative the test set is of the downstream application, the better one can prospectively assess the performance of a model.

To avoid biasing the results, a test set should ideally be used only once. In practice, however, many modeling attempts (e.g., different methods or model architectures) are typically made. While this goes against best practices, the scientific community relies on static test sets because the cost of data generation limits the availability and accessibility of newly generated data. When all methods are repeatedly evaluated on a single test set, it is common to find differences by chance that are dependent on the particular split of the data. In these cases, different splits will likely result in different conclusions. Method comparison should therefore not be performed on a single split of the data.

Using only a single split of data is akin to running a bench experiment with only a single replicate, something that is usually not acceptable in science. To properly account for stochasticity, a method comparison protocol should run replicates and compare the performance distributions of the populations of models the different methods produce. This allows the identification of robust improvements that are expected to generalize to similar data sets.

There are different mechanisms to accurately estimate a method's performance distribution based on a finite number of random samples from this distribution, also known as performance sampling distribution. We recommend the following data resampling mechanism:

**Guidelines 1** (performance sampling distribution) we recommend using a 5 × 5 repeated cross-validation procedure to sample the performance distribution. This procedure suits typical data set sizes used in small molecule property modeling (e.g., 500−100,000), and generates 25 sufficiently independent samples, meeting the sample size requirements for statistical testing. The training set can be further split into a training and a validation set if needed. Care should be taken to consider how the choice of data-splitting approach might systematically overestimate or underestimate model performance.

In the exceptional case of a data set having fewer than 500 or more than 100,000 molecules, we provide additional guidance in Section A.1.

*3.1.1. Sampling Mechanisms.* We can use two different mechanisms to sample the distribution: introducing variance in the model's parameters (e.g., different random seeds or initializations in a neural network), or resampling the data set (e.g., different data splits). It is good practice to use both sampling mechanisms jointly. Since introducing variance in the model's parameters is trivial, this work focuses on data resampling techniques. Our goal with these sampling mechanisms is to reduce the dependence between samples collected


https://doi.org/10.1021/acs.jcim.5c01609
J. Chem. Inf. Model. 2025, 65, 9398−9411

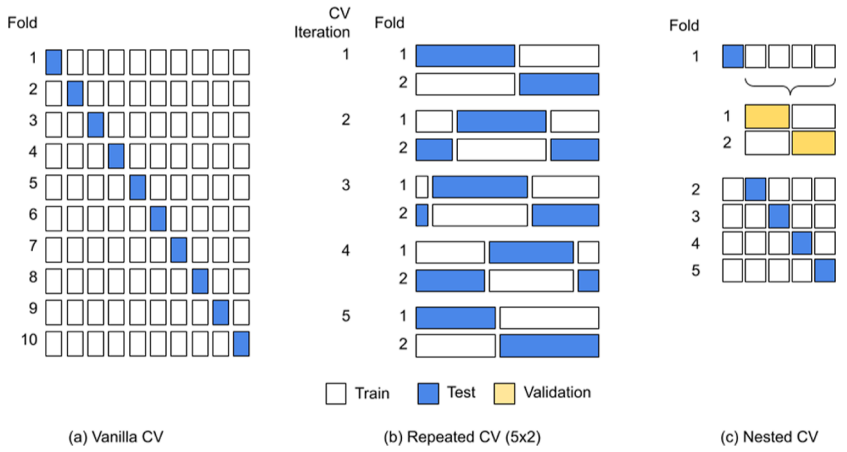



**Figure 2.** Visualization of different cross-validation resampling techniques: (a) vanilla cross-validation, (b) repeated cross-validation, and (c) nested cross-validation.

and obtain an accurate estimate of variance in performance, and we will thus focus our guidelines on data splitting techniques.

*Cross-validation* (CV, see Figure 2) is a popular method for resampling a data set. It is worth noting, however, that CV is not a single approach. CV refers to a set of different techniques by which one can resample (or split) a data set, and there exists no perfect solution that will work in every case. What works best depends on the specific data set and modeling objective. New techniques to sample performance distributions are also actively being researched.[16]

As illustrated in Figure 2, we adopt the following naming conventions: "fold" refers to a data partition (i.e., each square in Figure 2); "split" refers to a type of data split (e.g., random, scaffold-based, similarity-based, temporal, etc.); "CV iteration" refers to an iteration of a CV evaluation procedure (i.e., each row and subrow in Figure 2).

*3.1.2. Different Cross-Validation Techniques.* In vanilla CV, the data is split into n disjoint sets (or folds), with one fold used as the test set and the remaining folds used for training. When comparing methods, the same data split (i.e., using the same random seed) is typically performed, offering a more direct head-to-head comparison that usually results in increased precision. Figure 2a illustrates this with 10 folds. This raises the question of how many folds to use. With many folds, the different training sets overlap substantially, creating strong dependence between the samples. This underestimates variance, violates the assumptions of statistical tests, and results in elevated false positive rates (see Section 3.3 for a review of statistical testing). With few folds, the statistical tests will be underpowered (i.e., have low statistical power) due to the small sample size of the performance sampling distribution. Commonly used alternatives to CV like bootstrapping and repeated random splits of the data have also been shown in simulation to result in strong dependency between samples and are generally not recommended.[16,17] Notably, Dietterich[17] performed simulations showing that statistical tests using repeated random sampling or vanilla CV have an unacceptably high type I error rate (i.e., false positive rate).

Dietterich proposed a 5 × 2 repeated CV to address these concerns (see Figure 2b). Five × 2 CV splits the data set five times, with two folds each time. Having only two folds reduces the dependence across folds within a CV iteration because the training sets do not overlap. Repeated splitting does introduce dependence across CV iterations as training and test sets overlap between iterations. However, such overlap is less substantial than what would be observed when getting the same number of samples with vanilla 10-fold CV or other commonly used procedures.[17]

Even though Deitterich found that 5 × 2 repeated CV struck the right balance, his paper was based on simulations with data sets of only 300 observations. For modern data set sizes, the 5 × 2 settings result in an underpowered test as well as poor performance estimates because 2 fold CV is used. This was addressed in a recent paper by Bates et al.,[16] where the authors derive a nested CV procedure (see Figure 2c) more accurate than vanilla CV and other sampling methods. Unfortunately, this procedure is too computationally expensive for most small molecule property modeling applications and the procedure also limits the performance metrics one can use.

Although the nested CV procedure by Bates et al. is computationally expensive, other CV procedures can be evaluated against their method. Through an experiment (see Section B), we show that for representatively sized data sets, 5 × 5 repeated CV (i.e., 5 iterations of 5 fold CV) provides a reasonable approximation and a more stable and accurate variance estimate than the commonly recommended Deitterich's 5 × 2 and McNemar procedures. This experiment leads us to suggest the use of 5 × 5 repeated CV in our guidelines for improved statistical testing.

Note that in statistical testing procedure that we propose, no aggregation across CV folds is performed when computing performance metrics. This means 5 × 5 repeated CV generates 25 samples from the performance sampling distribution. The 25 samples are sufficiently independent to meet the minimum sample size required by the statistical testing procedure (see Section C).

While we provide specific CV recommendations for defined data set sizes ranges (<500, 500−100,000, >100,000) as general guidance, we acknowledge that data set sizes are a continuum. For instance, at the higher end of the 500−100,000 range, more folds will be tolerated with sufficient independence of samples. This would result in more samples for more accurate performance estimates and higher-powered statistical tests. If different fold sizes are used, we recommend ensuring at least 25 samples are collected in total, as this is the minimum sample size required for our recommended testing procedure.

*3.1.3. Cross-Validation with Advanced Splits.* When evaluating a method, it is critical to avoid a model simply



https://doi.org/10.1021/acs.jcim.5c01609
J. Chem. Inf. Model. 2025, 65, 9398−9411

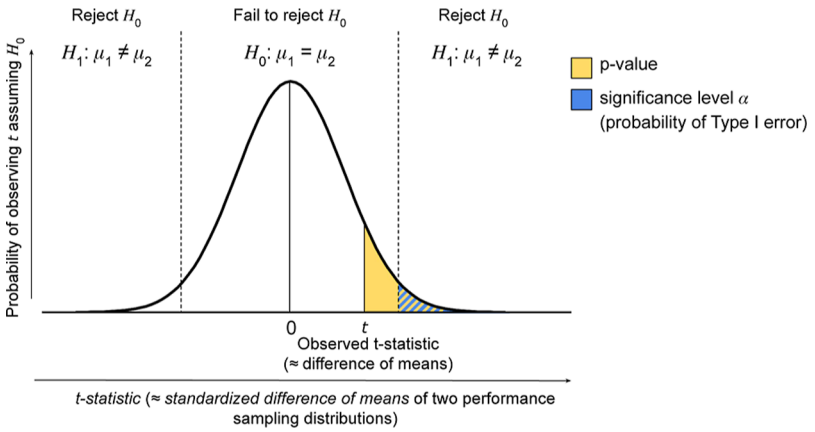



**Figure 3.** Visualization of a paired *t*-test for difference in performance between two methods. Intuitively, the *t*-test estimates the probability of observing a test statistic as extreme or more extreme assuming both samples come from the same distribution. The test statistic measures how closely the observed distribution matches the distribution assumed by the null hypothesis. The assumed distribution under the null is shown above along with the observed test statistic and the estimated *p*-value (in yellow). In this specific example, since the *p*-value is higher than the chosen significance level (in blue), this test would fail to reject the null hypothesis. Tukey HSD is an extension of the *t*-test to the scenario where there are more than two models and all pairwise comparisons are performed.

"memorizing" the training data, known as overfitting. To assess the ability of a model to generalize, the similarity between training and test sets should accurately reflect the downstream application. There are many ways to split a data set and which split is best depends on the application.[18−21] One can split a data set randomly, based on temporal information (e.g., compound synthesis or measurement dates), or to minimize the structural overlap between training and test sets. In this last case, the splitting procedure can be based on chemical scaffolds or similarity clustering. An inappropriate choice of splitting method can lead to systematically under-estimating or overestimating model performance.[22,23] We aim to provide guidance on measuring generalization and model validation in future work.

Within the context of this work, it is worth noting that CV is compatible with these more advanced splitting methods as long as the data set can be partitioned into nonoverlapping, roughly equally sized groups. It is essential to check that folds do not significantly overlap across CV iterations and that target distributions stay reasonably similar. It is recommended to visually inspect these constraints (see Section E).

*3.1.4. Cross-Validation with Hyperparameter Optimization.* Besides assessing generalization with a hold-out test set that is not used during method development and selection, there are also cases where one might want to use a second evaluation set during method development, such as with hyperparameter optimization. In such cases, nested CV is commonly recommended to split the data into three subsets: training, validation, and test. However, this substantially increases the total number of iterations (i.e., the number of models to train). For each CV iteration within 5 × 5 repeated CV, we recommend performing a split of the training set into training and validation for hyperparameter optimization. This is comparable to performing one iteration of the inner loop of nested CV (see Figure 2c), and results in 25 distinct validation sets, one per CV sample. Five × 5 repeated CV collects 25 samples from the performance sampling distribution, which is already a sufficient number of samples for statistical testing, so a full nested CV is unnecessary.

**3.2. Statistical Significance.** After collecting the performance sampling distributions for each of our methods, an appropriate technique for comparing these distributions should be selected.

Since finite samples of a distribution are being compared, we cannot unequivocally state that the two sampled distributions are different. However, we can hypothesize that the two populations of samples come from distributions having the same mean value and compute a *p*-value for testing that null hypothesis (see Figure 3). The *p*-value estimates the probability of observing the test statistic at least as extreme under the null hypothesis. If that probability is lower than a chosen significance level, we reject the hypothesis and conclude that there is a statistically significant difference between the two distributions.

The false positive rate (or type I error rate) of a test is the probability of falsely rejecting the null hypothesis, i.e. falsely concluding that there is a difference between the performance distributions of two methods while there is not. If the assumptions of the test are met, then the false positive rate will be less than the significance level. The significance level is set by the researcher based on the amount of confidence that is needed in the conclusion. A commonly used level is 0.05, which should provide reasonable control of false positive rate for methods comparisons after correction for multiple comparisons (see Section 3.2.2).

The type II error rate of a test is the probability of failing to detect a difference between performance distributions when one exists. Statistical power is equal to 1—type II error rate, and is the probability that a true difference in distributions will be detected by the test.

An optimal statistical test will have (1) a false positive rate at the level advertised and (2) high statistical power. Condition 1 should be met first for method comparison. When we claim statistical significance this gives researchers confidence that a real difference in performance exists between methods. We want to be confident that we are not giving people an inflated sense of certainty. If the assumptions of the statistical test are violated, then the test may have false positive rate higher than advertised or low statistical power. This is why it is important to understand and examine the assumptions of a test, as explained in the next section.

There are various tests for statistically significant differences, which differ in the assumptions they make on the sampling



https://doi.org/10.1021/acs.jcim.5c01609
J. Chem. Inf. Model. 2025, 65, 9398−9411

distributions under comparison. We recommend the following test:

**Guidelines 2** (statistical testing) we recommend repeated measures ANOVA (analysis of variance) with the post hoc Tukey's HSD (honestly significant difference) test for pairwise comparisons between models. We recommend always checking the parametric assumptions of the tests, but if you follow **Guidelines 1**, these assumptions should be reasonably met in most applications in small molecule property modeling (see Section C.1 for details).

In the exceptional case in which the parametric assumptions are not met, we provide additional guidance in Section A.2.

*3.2.1. Statistical Tests.* Statistical tests for differences between distributions can be broadly separated into parametric and nonparametric tests. Parametric tests make stronger assumptions about the distributions under comparison (e.g., normality, see also Section C), compared to nonparametric tests. One common misconception is that nonparametric tests do not make assumptions. Even though nonparametric tests have weaker distributional assumptions, they do still make assumptions and these are often harder to understand and examine than parametric tests. The most important assumption made by both parametric and nonparametric tests is that samples are independent, which means that an appropriate CV protocol (see Section 3.1) that minimizes the dependence between samples is necessary for both tests.

It is common for researchers to use a nonparametric test because they make fewer assumptions. However, researchers are often unaware of the disadvantages of these tests. For method comparisons, the most important is that nonparametric tests typically focus on hypothesis testing and less on estimation of an interpretable effect size. While it is possible to estimate effect size and confidence intervals with nonparametric methods it is typically not straightforward. Because our method comparison workflow focuses on estimating effect size in addition to hypothesis testing, a parametric test with an interpretable associated effect size (e.g., the difference in means) is preferred. Nonparametric tests can also be substantially less powerful than parametric tests if the distributional assumptions of the parametric tests are met. See Section C.2 for more details on the advantages and disadvantages of parametric and nonparametric tests.

We recommend the following parametric testing workflow: repeated measures ANOVA followed by the Tukey HSD test. During the repeated CV procedure, competing methods are being fit to the same splits of data. To appropriately account for this dependency, we perform repeated measures ANOVA, and then provide the sum of squared errors output to the Tukey HSD procedure. This results in a test with higher statistical power than Tukey HSD alone. Note that if only two comparisons are performed, this procedure simplifies to a paired (repeated measures) *t*-test.

The parametric workflow compares the means and is known to be highly robust to moderate violations of the underlying assumptions. This is particularly true in the context of a method comparison protocol (see Section C.1 for details). If the assumptions of the parametric test are strongly violated, then we recommend a nonparametric test workflow that will also be suitable for method comparisons (Section A.2). We provide an example of examining the parametric testing assumptions in the supplementary notebooks.

*3.2.2. Pairwise Comparisons and Corrections for Multiple Testing.* We typically compare more than two methods in ML benchmarks and are interested in all pairwise comparisons. This results in a large number of tests. When we perform many comparisons simultaneously, the probability of falsely rejecting the null hypothesis increases. For example, say we picked a significance level of 0.05. In other words, in 5% of tests, we expect to conclude that there is a statistically significant difference between distributions while there, in reality, is no such difference. If we run this test $N$ times, the expected number of falsely rejected null hypotheses linearly increases with $N$. For $N = 100$, we would thus expect to falsely reject 5 null hypotheses. The process of finding false positives when many comparisons are performed is referred to as "p-hacking" (or more specifically, "data-dredging"[24]) and is a source of prevalent false positives and publication bias across the sciences, see Ioannidis[25] but also Jager et al.[26] and the surrounding debate.[27] See also Head et al.[28] for importance of attention to effect size to avoid p-hacking, which is a focus of this paper.

The number of pairwise comparisons N in turn grows combinatorially with the number of methods under comparison, because of which multiple testing can quickly become problematic. There are several techniques to correct for this, see Chen et al.[29] for a review. The Bonferroni correction[30] is a simple approach that is commonly used. However, this correction is known to be overly conservative, meaning that it has low statistical power, when the number of comparisons is large.

We recommend the Tukey HSD test, which is specifically designed for pairwise comparisons and incorporates a correction for multiple testing. Compared to other multiple testing correction procedures like Bonferroni, it has good statistical power for all pairwise comparisons. It ensures that the family wise error rate (FWER), which is the probability that at least one false positive occurs in a set of tests, is less than a given significance level (e.g., 0.05), regardless of the number of tests performed. See Section A.2 for guidance on multiple testing for a large number of method comparisons (>10).

When multiple metrics are being considered in an evaluation, the following procedure will appropriately correct for multiple testing across metrics.

(1) Perform repeated measures ANOVA for each metric.
(2) Perform Bonferroni correction on the ANOVA significance level. Divide the significance level by the number of metrics to obtain an adjusted significance level.
(3) For all metrics with a significant ANOVA test, perform a Tukey HSD posthoc test.

A simple Bonferroni correction for the ANOVA tests is appropriate because the number of metrics evaluated is often small.

In the cases where only the comparisons of one method against all others are of interest, then less tests need to be corrected for than all pairwise comparisons as in Tukey HSD. However, since Tukey HSD has good statistical power compared to other procedures, we recommend this for general use. This gives researchers a "license to fish", providing protection against any pairwise comparison that might be performed.

**3.3. Practical Significance.** With statistical significance, we establish that there is a difference between means, but we can not yet conclude the magnitude of that difference. The Tukey HSD procedures, however, not only provide us with statistical significance (i.e., an assessment that the means of the distributions under comparisons are the same) but also with


https://doi.org/10.1021/acs.jcim.5c01609
J. Chem. Inf. Model. 2025, 65, 9398−9411

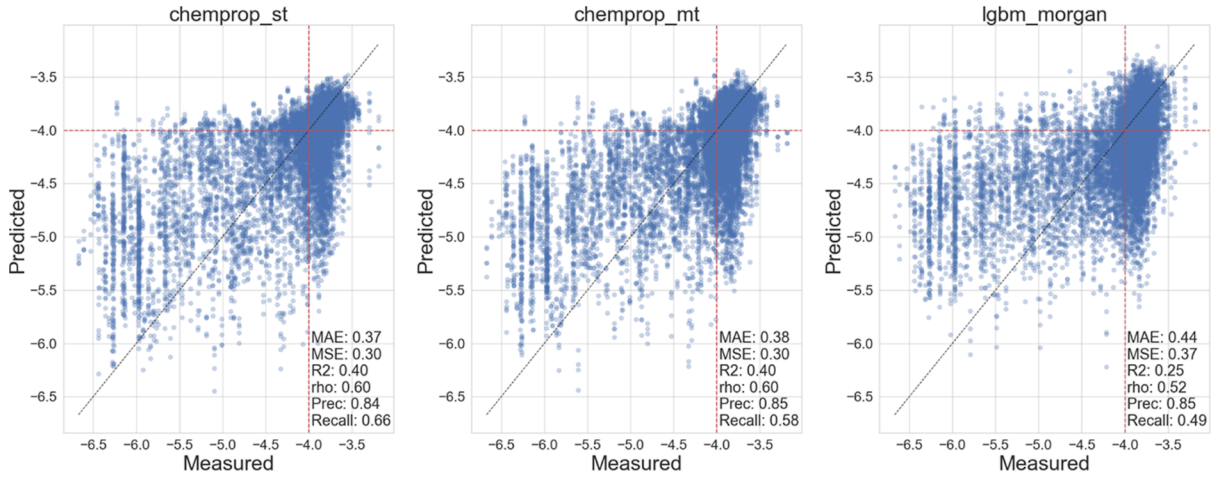



**Figure 4.** An example using post hoc classification for a regression model to investigate practical significance with precision and recall. Measured vs Predicted logS (solubility from Fang et al.). Reported are mean absolute error (MAE), mean squared error (MSE), coefficient of determination ($R^2$), Spearman's rank correlation coefficient (rho), precision (Prec) and recall. From left to right, we show the results for ChemProp Single Task (`chemprop_st`), ChemProp Multitask[31] (`chemprop_mt`), and Light Gradient Boosting Machines (`lgbm_morgan`).

effect size (i.e., the magnitude of the difference in mean between two distributions).

However, this raises the question whether any given effect size is also practically significant. Practical significance is established when there is a large enough difference between methods to be meaningful in practice. In small molecule property modeling, this boils down to whether a new method impacts a drug discovery scientist's decision-making regarding which experiments to prioritize. To measure practical significance, we need to use relevant, contextualized performance metrics that are informed by our downstream application. We recommend the following:

**Guidelines 3** (practical significance) when reporting a significant difference between methods, also provide an explanation of how the result is practically significant. Use metrics that are motivated by the downstream application and contextualize results by estimating the lower and upper performance limits.

Over the past century, statisticians have developed many valuable metrics for evaluating the performance of regression and classification models. Section D reviews several of these metrics and provides recommendations to ensure accurate and meaningful model evaluations from a statistical point of view. The rest of this section specifically describes different ways to measure impact in small molecule drug discovery.

*3.3.1. Relevant Performance Metrics. 3.3.1.1. Decisional Impact.* A typical application of a property model is to inform two key decisions: (1) deciding what compounds to make and (2) deciding what compounds not to make. When prioritizing a set of molecules, drug discovery scientists typically classify each of the properties of interest in two or three bins (or categories), e.g. "soluble" and "insoluble", to inform their decision-making. To measure the real-world utility of small molecule property models, one can thus investigate whether a model can help decide which molecules to make or not to make by using these bins.

When deciding what compounds to make, a filter is often applied to a large set of compounds by applying a threshold to a property estimation. We would like to be confident that everything left after filtering will have a good property value when measured. We would also like the set to be as large as possible because this provides chemists with more diversity for design. One approach to achieve this task is to select a minimum acceptable precision (e.g., 75%), and then select the threshold with the maximum recall subject to this constraint. The model with the best performance will have the largest recall. This typically referred to as recall@precision in the ML literature.[32,33]

Another decision is what compounds to not make. In this context we would like to eliminate a large number of bad compounds while eliminating as few positives as possible. One approach is to select a minimum acceptable recall for the positive class. For example, we may require 90% recall, so that no more than 10% of true positives are thrown out. We then select the threshold with the maximum true negative rate (TNR) subject to this constraint. The model with the best performance will thus have the largest TNR@recall.

This can also be done in a regression setting by using post hoc classification (see section Section D.2 for details).

Figure 4 shows a comparison of three machine learning models, ChemProp Multitask[31] (`chemprop_mt`), ChemProp Single Task (`chemprop_st`), and Light Gradient Boosting Machines (`lgbm_morgan`) on the same data set. The data set, provided by Fang et al.,[34] contains 2173 compounds with aqueous solubility determined using an assay routinely employed in drug discovery. In drug discovery, we typically screen early for compounds with good aqueous solubility, as that property often translates to solubility in intestinal fluid for oral drugs, as well as solubility in intravenous formulations for when not orally administered. A typical threshold for good solubility for oral drugs is >100 $\mu$M.

After training three regression models for solubility, statistically significant differences in mean absolute error (MAE), mean squared error (MSE), and coefficient of determination ($R^2$) are found between `lightGBM` and the two ChemProp models. To assess whether such difference was large enough to be meaningful, post hoc classification with a 100 $\mu$M threshold was carried out. Precision is essentially equivalent across methods but recall is substantially lower for `lightGBM`. If one used these models as a compound filter at 100 $\mu$M, `lightGBM` would reject more compounds with good solubility. To see this, note that there are substantially more compounds in the bottom right quadrant for `lightGBM` than


https://doi.org/10.1021/acs.jcim.5c01609
J. Chem. Inf. Model. 2025, 65, 9398−9411

**Table 1. Confidence Intervals (CI) of the Difference in Mean Performance Between Methods, Presented as a Table**

| | MAE | MSE | $R^2$ | Rho | precision | recall |
|---|---|---|---|---|---|---|
| `chemprop_mt—chemprop_st` | 0.02 (0.01, 0.02) | 0.00 (−0.01, 0.01) | 0.00 (−0.02, 0.02) | 0.00 (−0.01, 0.01) | 0.02 (0.00, 0.03) | −0.08 (−0.10, −0.06) |
| `chemprop_mt—lightGBM` | −0.06 (−0.06, −0.05) | −0.07 (−0.08, −0.06) | 0.15 (0.13, 0.17) | 0.08 (0.07, 0.09) | 0.01 (−0.01, 0.02) | 0.09 (0.07, 0.11) |
| `chemprop_st—lightGBM` | −0.07 (−0.08, −0.06) | −0.07 (−0.08, −0.07) | 0.15 (0.13, 0.17) | 0.08 (0.07, 0.09) | −0.01 (−0.02, 0.01) | 0.17 (0.15, 0.19) |

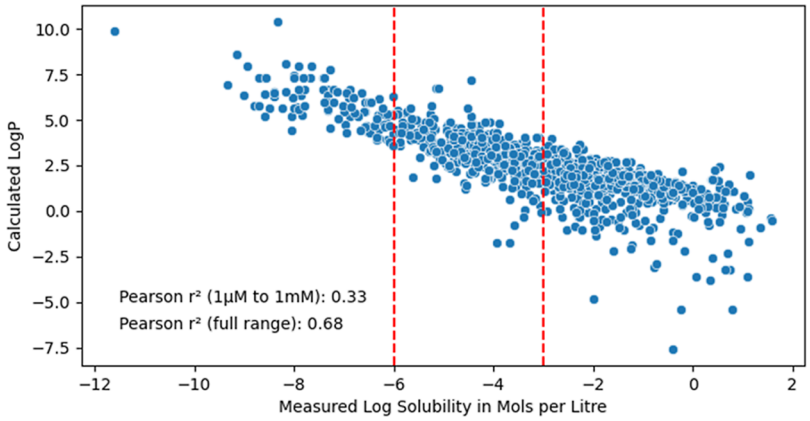



**Figure 5.** Examining the impact of dynamic range on correlation. If the entirety of this data set, which spans 13 log units of dynamic range, is considered, there is a high correlation between measured and estimated values. However, the correlation is much lower if the more realistic 3-log range between the red lines is considered.

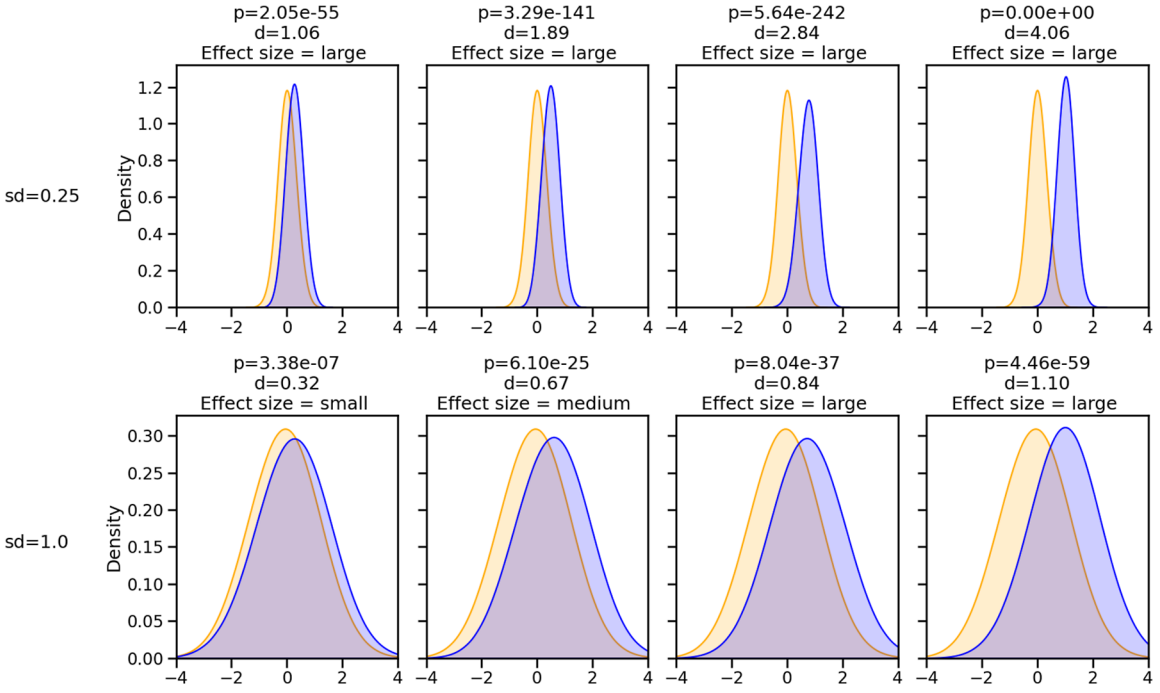



**Figure 6.** An illustration of effect size. In all of the subplots, the two distributions show a statistically significant difference. For each column, the mean of the same-colored distributions is equal. However, because the distributions in the top row have a lower variance, the effect size of these comparisons is higher than for the distributions in the bottom row.

there are for other methods. As we will later show in Figure 9 and Table 1, the estimated improvement in recall of `chemprop_st` over `lightGBM` is 0.17 (0.15, 0.19), meaning `chemprop_st` would identify 17% more molecules with good solubility. This would likely have a real practical impact on drug discovery programs.

*3.3.1.2. Interpretability.* Domain experts who use an ML model in a real drug discovery program need context on which differences are impactful. For those with a limited statistical background, statistical measures can be hard to interpret. To facilitate interdisciplinary communication, it can therefore be helpful to report the MAE. Although this metric is not the only metric that should be used for method development (see



https://doi.org/10.1021/acs.jcim.5c01609
J. Chem. Inf. Model. 2025, 65, 9398−9411

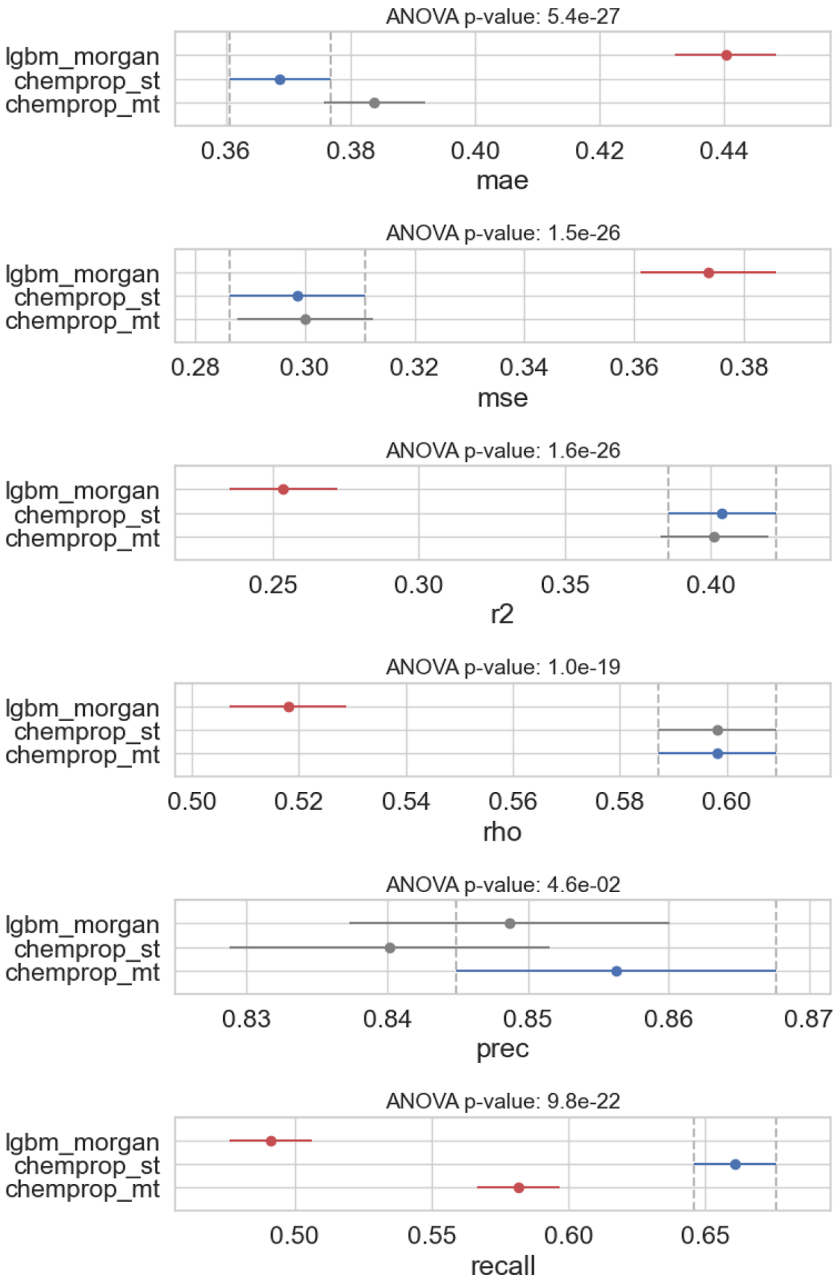



**Figure 7.** An example of the simultaneous confidence interval plot.

Section D), it is important to report because the unit of MAE is the same as the property being modeled. MAE is often used in log scale by medicinal chemists or pharmacologists to indicate fold differences between observed and measured values (where a MAE of 0.3 log units would correspond to 2-fold error). Thus, average fold errors or percentage of errors within 2- or 3-fold are often reported to facilitate discussions within drug discovery teams.

*3.3.1.3. Dynamic Range.* Both correlation and error metrics are influenced by the dynamic range of the data being modeled. Achieving a high correlation on data sets with a broader range of experimental values is generally easier, whereas data sets with a smaller dynamic range can produce unrealistically small values for error metrics. This can lead to deceptive conclusions.

For instance, consider the Delaney solubility data set[35] in the MoleculeNet[36] benchmark. This data set reports the log of the aqueous solubility (LogS) for 2173 compounds. The LogS values span more than 13 logs, significantly larger than the 3−4 log dynamic range typically encountered in drug discovery. Consider a simple model that uses a calculated octanol−water partition coefficient (LogP) to estimate LogS. If we calculate the coefficient of determination ($r^2$) for LogP vs LogS for the full 13-log range of the Delaney solubility data set, we achieve a respectable $r^2$ of 0.68. However, if we only consider values in the 1 $\mu$M to 1 mM (log solubility −6 to −3) range typically observed in drug discovery projects, the $r^2$ value drops to a less impressive 0.33. Figure 5 illustrates this issue by showing the full range of the Delaney data set, with a more realistic dynamic range between the red lines.

*3.3.1.4. Class Imbalance.* Classification metrics can be misleading in cases where classification data sets are highly imbalanced, as is common in small molecule drug discovery. In this case, using metrics that account for this imbalance is important (see Section D.2).



https://doi.org/10.1021/acs.jcim.5c01609
J. Chem. Inf. Model. 2025, 65, 9398−9411

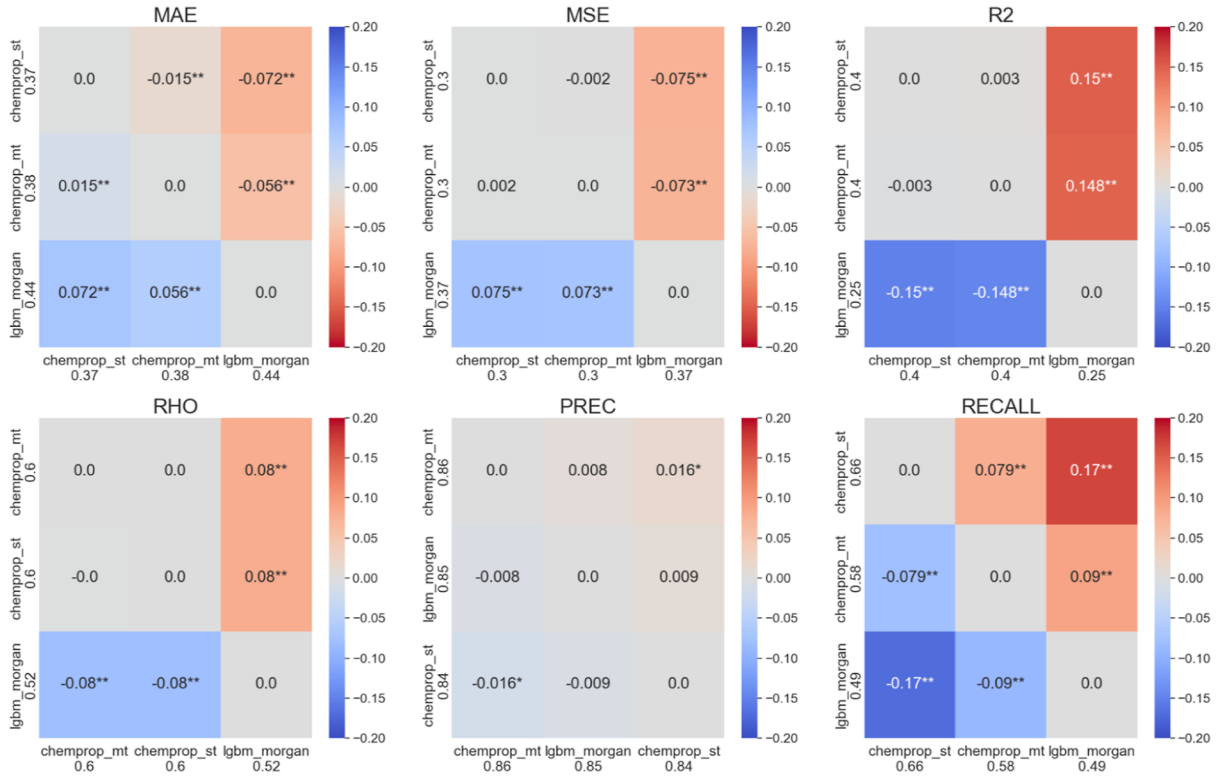



**Figure 8.** An example of the multiple comparisons similarity (MCSim) plot. Color is used to convey the effect size, whereas star annotations are used to convey statistical significance. The effect size reported is the difference in average performance between methods. A numeric difference is shown in the cells, but this can be suppressed if a large number of comparisons is performed.

*3.3.2. Cohen's D.* In Section 3.3.1, we covered some ways of measuring performance of a ML model in the context of small molecule drug discovery. Often researchers understand whether a performance difference is large enough to be practically significant, and in these contexts a simple difference in means is recommended as an interpretable effect size. However, providing meaningful context to a difference is sometimes problematic, and in these cases Cohen's D can be a useful measure of effect size. Cohen's D standardizes the difference in means by the pooled standard deviation. This results in a unitless measure of difference in distribution which considers the variance of both distributions (Figure 6).

$$d = \frac{\mu_1 - \mu_2}{\sqrt{\frac{\sigma_1^2 + \sigma_2^2}{2}}}$$

Commonly used cutoffs for interpreting Cohen's $d$ are $d \geq 0.2$, $d \geq 0.5$, and $d \geq 0.8$, implying a small, medium, or large effect size, respectively.[37] Statisticians often advise using these cutoffs as a last resort when there is insufficient understanding of whether a difference is meaningful from domain knowledge.[38]

*3.3.3. Lower and Upper Performance Limits.* As discussed in Section 3.3.1, performance metrics can be misleading depending on the underlying distribution being modeled. Furthermore, the endpoints we are estimating are subject to experimental noise, which implies a maximum expected model performance. To address these concerns and help improve the interpretability of the performance metrics, it is important to contextualize results with both a lower and upper limit for the performance.

*3.3.3.1. Lower Limit: Null Models.* Null models consistently assign the majority class for a classification task, or the mean (or median) of the training set for a regression task. If the performance metrics for a model are close to those of the null model, one should question the results.

*3.3.3.2. Upper Limit: Experimental Variability.* If the experimental variability of the underlying assay is known, it can be used to estimate the maximum expected performance.[39] Experimental variability is often referred to as aleatoric uncertainty, or data uncertainty. For example, the noise in activity biochemical assays measuring half-maximal inhibitory concentration ($IC_{50}$) is commonly estimated to be 0.3 log units (i.e., 2-fold).[40] If the MAE of an $IC_{50}$ model is less than 0.3, one should question the results. In a case where the experimental variability is not known, it is common to assume experimental variability of 2- or 3-fold, depending on the dynamic range and nature of the data.[41]

In the special case of correlation metrics for regression models, Brown et al.[42] outlined a procedure for a data set $X$ with $N$ values and an experimental fold error $A$.

For 1000 trials:

(1) Generate $N$ normally distributed random variables $R$ with a mean of 0 and a standard deviation of $\log_{10}(A)$.

(2) Add $R$ and $X$ to create a new vector $RX$

(3) Calculate the correlation between $X$ and $RX$

The mean of the correlations over the 1000 trials calculated above typically provides a reasonable estimate of the upper limit of achievable correlation. If the observed correlation exceeds this value, the benchmark result should be questioned.

*3.3.4. Holistic Evaluation.* A single performance measure is unlikely to capture real-world utility. Instead, practitioners typically rely on a holistic view that evaluates performance along multiple dimensions to inform the usage of a ML model in a real-world context, which can span various applications. We



https://doi.org/10.1021/acs.jcim.5c01609
J. Chem. Inf. Model. 2025, 65, 9398−9411

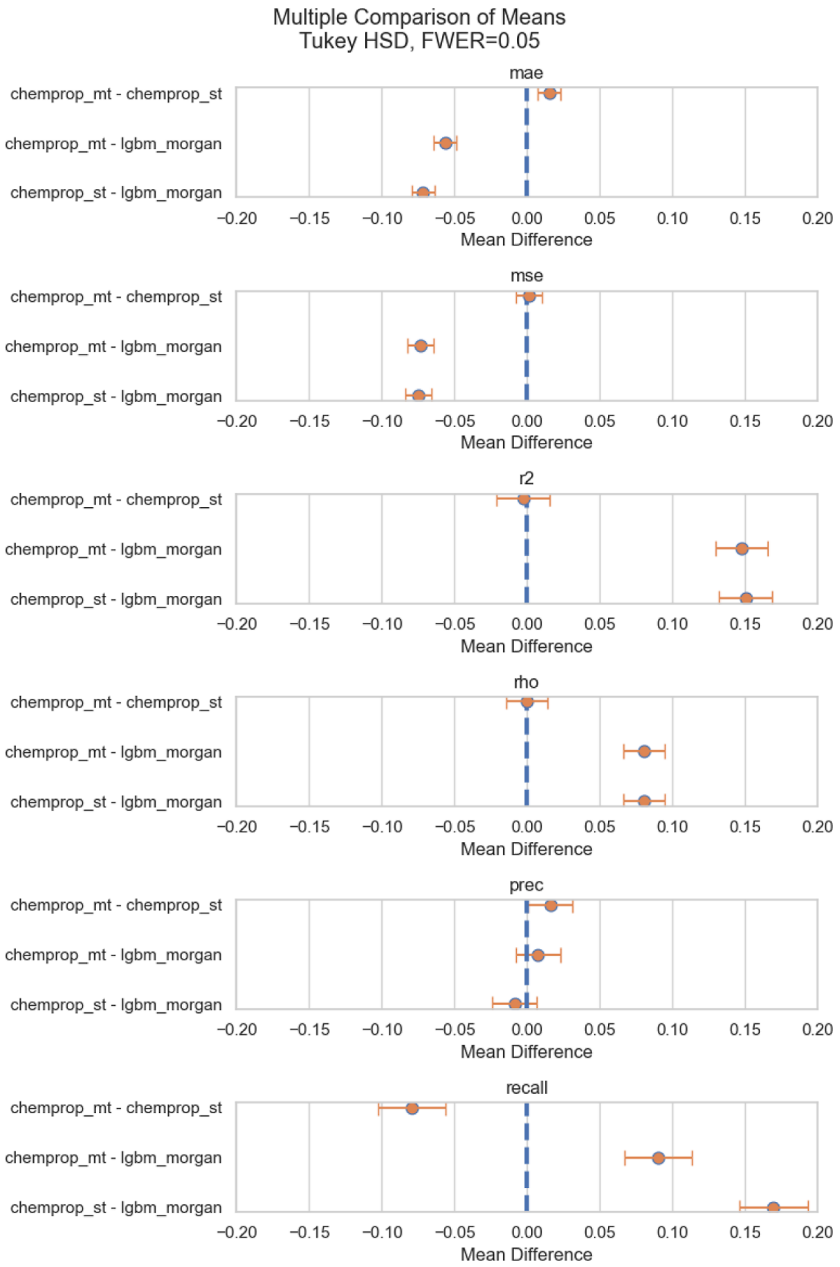



**Figure 9.** Confidence Intervals (CI) of the difference in mean performance between methods, presented as a plot. Intervals that do not cross the zero line imply statistical significance.

therefore recommend at least reporting multiple performance measures. Furthermore, a thorough investigation of the capabilities and limitations of a ML method (e.g., performance on activity cliffs,[43] performance per chemical series,[44] or uncertainty estimation[45]) significantly increases its scientific and real-world utility.

**3.4. Presenting the Results.** Using statistical tests produces information beyond a performance metric table. Typical methods for presenting the results, such as leaderboards, are unsuitable for presenting this information. We therefore provide guidance on appropriate visualizations:

**Guidelines 4** (presenting the results) we recommend a plot to visualize the results of the pairwise comparisons, such as the simultaneous confidence interval plot or the multiple comparisons similarity plot. For regression models, we recommend including additional scatter plots in the Supporting Information that shows the models' predictions versus the ground-truth labels. For classification models, we similarly recommend including the confusion matrices.

In the exceptional case where the reader requires a leaderboard, we provide additional guidance in Section A.3.

*3.4.1. The Simultaneous Confidence Interval Plot.* The first plot we recommend is the simultaneous confidence intervals plot provided by the statsmodels Python package[46] (see Figure 7). In these plots, the method with the best performance metric is displayed in blue. Dashed vertical lines surround the confidence intervals for the best method. Methods equivalent to the best model are represented in gray. Methods that show statistically significant differences from the best model are indicated in red. All confidence intervals have been adjusted for multiple comparisons such that any pair of intervals that are



https://doi.org/10.1021/acs.jcim.5c01609
J. Chem. Inf. Model. 2025, 65, 9398−9411

nonoverlapping implies a statistically significant difference according to the Tukey HSD testing procedure.

For this visualization, advantages are that it is easy to construct, it concisely represents comparisons to the best performing method, and it provides a confidence interval of the mean performance of each method. A disadvantage is that the effect size that is considered practically significant is not shown explicitly on the plot, though context on what differences are considered practically significant can be provided in the text of the paper.

*3.4.2. The Multiple Comparisons Similarity Plot.* Another option for visualizing results is an extension to the sign plot provided by the scikit-posthocs Python package[47] (see Figure 8). The original sign plot showed a heatmap of all pairwise p-values. Our extension, which we call the multiple comparisons similarity (MCSim) plot, uses color to convey effect size instead of $p$-values since practical significance is more important than statistical significance in the context of a method comparison protocol.

To simplify the interpretation of the plot, the MCSim plot sorts the methods in the rows and columns by their average performance, which are also annotated in the margins. The top left block of methods without statistically significant differences are thus the plausible top performers. Cells are colored by the difference in average performance between methods. Each cell in the heatmap also has a star annotation to indicate the level of significance ($*p < 0.05$, $**p < 0.01$, $***p < 0.001$). The color range is determined by the user and should be set to be large enough to cover a range of practically significant differences. The ranges will differ by metric, so different color scales are necessary for each plot.

An advantage of this visualization is that it explicitly represents the effect sizes that are considered practically significant. A disadvantage is that the simultaneous presentation of effect size and statistical significance makes these plots more complex to construct and interpret.

*3.4.3. Confidence Intervals of the Difference in Mean Performance.* Another option is a plot or table showing confidence intervals of the difference in mean performance. The results of the Tukey HSD test can be used to construct confidence intervals for the differences between methods. These confidence intervals allow us to understand the uncertainty associated with the differences reported. A point estimate for the difference between methods may appear substantial, but if the associated confidence interval is large, then the result is less convincing. While confidence intervals can be easily calculated for parametric methods, they are not straightforward to obtain with the nonparametric workflow, though a range measure such as the interquartile range may be reported.

As the number of pairwise comparisons is often large (i.e., the number of comparisons grows combinatorially with the number of methods under comparison), the relationships between methods will be difficult to visualize in a single plot, especially if multiple metrics are used. We therefore recommend providing these results in the Supporting Information as either a plot (see Figure 9) or tabular form (see Table 1). Alternatively, practitioners may find it optimal only to show a few differences of interest, such as comparing a new method to a set of baselines. However, it is important to apply the Tukey HSD to all comparisons that were examined originally to properly correct for multiple comparisons and avoid p-hacking (see Section 3.2.2).

## 4. ANNOTATED EXAMPLES

To simplify the adoption of the guidelines we presented in this work, all guidelines presented throughout this paper are accompanied by a set of annotated examples that use open-source software to implement the proposed method comparison protocol. These annotated examples provide an easy to use template to incorporate these guidelines in your own research. These annotated examples can be found at https://github.com/polaris-hub/polaris-method-comparison. An overview of Open-Source Software that can be used to implement these guidelines can be found in Table 2.

**Table 2. An Overview of Useful Open-Source Software**

| package | language | description |
|---|---|---|
| `scikit-posthocs`[47] | Python | implements multiple pairwise comparisons tests in python |
| `scikit-learn`[48] | Python | this well-known machine learning library for Python has a mature cross-validation API |
| `pingouin`[49] | Python | implements various statistical methods in Python |
| `statsmodels`[46] | Python | implements various statistical methods in Python |
| `chemmodlab`[50] | R | a cheminformatics modeling laboratory for fitting and assessing machine learning models |

## 5. CONCLUSION

ML-based research is facing a replicability crisis. These issues are further amplified in small molecule property modeling due to the high-stakes applications, the heterogeneous, imbalanced, and noisy data sets, and the interdisciplinary teams. It is essential that statistically robust and domain-appropriate method comparison protocols are employed to close the gap between perceived progress and real-world impact.

In this work, we proposed beginner-friendly guidelines for method comparison protocols in small molecule property modeling. We simplified the adoption of these guidelines with annotated examples that use open-source software. These guidelines are

(1) We recommend using a 5 × 5 repeated cross-validation procedure to sample the performance distribution. This procedure suits typical data set sizes used in small molecule property modeling (e.g., 500−100,000), and generates 25 sufficiently independent samples, meeting the sample size requireements for statistical testing. The training set can be further split into a training and validation set if needed. Care should be taken to consider how the choice of data-splitting approach might systematically overestimate or underestimate model performance.

(2) We recommend the Tukey HSD test for pairwise comparisons between models. We recommend always checking the parametric assumptions of the Tukey HSD test, but if you follow Guidelines 1, these assumptions should be reasonably met in most applications in small molecule property modeling.

(3) When reporting a significant difference between methods, also provide an explanation of how the result is practically significant. Use metrics that are motivated by the downstream application and contextualize results by estimating the lower and upper performance limits.



https://doi.org/10.1021/acs.jcim.5c01609
J. Chem. Inf. Model. 2025, 65, 9398−9411

(4) We recommend a plot to visualize the results of the pairwise comparisons, such as the simultaneous confidence interval plot or the multiple comparisons similarity plot. For regression models, we recommend including additional scatter plots in the Supporting Information that shows the models' predictions versus the ground-truth labels. For classification models, we similarly recommend including the confusion matrices.

(5) Statistical testing is not a recipe to blindly follow and there are valid reasons to deviate from the above guidelines. Transparency is key in the absence of a perfect solution for every scenario.

In future work, we aim to tackle other important aspects of benchmarking ML models in small molecule property modeling, such as data set curation and measuring generalization (e.g., through data splitting methods).

## ■ ASSOCIATED CONTENT

### Data Availability Statement

All the data underlying this study stem from freely available, public data sources. The software that can be used to reproduce our results is open-source and publicly available on Github at https://github.com/polaris-hub/polaris-method-comparison.

### Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jcim.5c01609.

Additional best practices for exceptional cases, a detailed description of our cross-validation experiment, background information on statistical testing, as well as on performance metrics, and details on a supporting visualization (PDF)

## ■ AUTHOR INFORMATION


### Corresponding Author

**Cas Wognum** − *Valence Laboratories, Montréal, Québec H2S 3G6, Canada; Recursion Pharmaceuticals, Salt Lake City, Utah 84101, United States;* orcid.org/0009-0006-2742-4817; Email: cas@valencelabs.com

### Authors

**Jeremy R. Ash** − *Johnson & Johnson Innovative Medicine, Spring House, Pennsylvania 19477, United States*

**Raquel Rodríguez-Pérez** − *Novartis Pharma AG, Basel CH-4056, Switzerland;* orcid.org/0000-0002-2992-3402

**Matteo Aldeghi** − *Bayer Research and Innovation Center, Cambridge, Massachusetts 02142, United States*

**Alan C. Cheng** − *Merck & Co., Inc., South San Francisco, California 94080, United States;* orcid.org/0000-0003-3645-172X

**Djork-Arné Clevert** − *Pfizer Research and Development, Berlin 10117, Germany*

**Ola Engkvist** − *Department of Computer Science and Engineering, Chalmers University of Technology & University of Gothenburg, Gothenburg, Mölndal 412 58, Sweden; Molecular AI, Discovery Sciences AstraZeneca R&D, Gothenburg, Mölndal 431 83, Sweden;* orcid.org/0000-0003-4970-6461

**Cheng Fang** − *Blueprint Medicines Corporation, Cambridge, Massachusetts 02139, United States;* orcid.org/0000-0002-9767-2043

**Daniel J. Price** − *Nimbus Therapeutics, Boston, Massachusetts 02210, United States*

**Jacqueline M. Hughes-Oliver** − *Department of Statistics, North Carolina State University, Raleigh, North Carolina 27607, United States*

**W. Patrick Walters** − *Relay Therapeutics, Cambridge, Massachusetts 02139, United States;* orcid.org/0000-0003-2860-7958

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jcim.5c01609


### Author Contributions

‖‖Jeremy Ash and Cas Wognum contributed equally to this work, and jointly wrote the initial draft, wrote the accompanying code, and created the visualizations. All authors contributed to the conceptualization of this study, notably the best practices for method comparison. All authors reviewed and edited the initial draft.

### Notes

The authors declare the following competing financial interest(s): Except for Professor Hughes-Oliver, all authors were employed by for-profit companies during the writing of this article. While there may be financial or non-financial interests related to their employer, the authors affirm their commitment to scientific integrity. The article is presented objectively, and steps were taken to minimize any potential influence from their employment. The corresponding author is available for further inquiries.

## ■ ACKNOWLEDGMENTS

We would like to acknowledge the support of various community members in refining this paper. Initially shared as a preprint in November of 2024, the authors called for the support from the broader scientific community to improve the proposed best practices through an open-feedback period hosted on Github; this version incorporates this valuable community feedback. Beyond the authors' employment by for-profit companies, this research received no financial support by additional funding sources.

## ■ REFERENCES


(1) Wognum, C.; Ash, J. R.; Aldeghi, M.; Rodríguez-Pérez, R.; Fang, C.; Cheng, A. C.; Price, D. J.; Clevert, D.-A.; Engkvist, O.; Walters, W. P. A call for an industry-led initiative to critically assess machine learning for real-world drug discovery. *Nature Machine Intelligence* **2024**, *6*, 1120−1121.

(2) Demšar, J. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research* **2006**, *7*, 1−30.

(3) Raschka, S. Model evaluation, model selection, and algorithm selection in machine learning. *arXiv* **2018**, arXiv:1811.12808.

(4) Rainio, O.; Teuho, J.; Klén, R. Evaluation metrics and statistical tests for machine learning. *Sci. Rep.* **2024**, *14*, 6086.

(5) Kapoor, S.; Cantrell, E. M.; Peng, K.; Pham, T. H.; Bail, C. A.; Gundersen, O. E.; Hofman, J. M.; Hullman, J.; Lones, M. A.; Malik, M. M.; et al. REFORMS: Consensus-based Recommendations for Machine-learning-based Science. *Sci. Adv.* **2024**, *10*, No. eadk3452.

(6) Baker, M. 1,500 scientists lift the lid on reproducibility. *Nature* **2016**, *533*, 452−454.

(7) Kapoor, S.; Narayanan, A. Leakage and the reproducibility crisis in machine-learning-based science. *Patterns* **2023**, *4*, 100804.

(8) Committee on Reproducibility and Replicability in Science *Reproducibility and Replicability in Science*; National Academies Press, 2019.

(9) Beam, A. L.; Manrai, A. K.; Ghassemi, M. Challenges to the reproducibility of machine learning models in health care. *JAMA* **2020**, *323*, 305−306.

(10) McDermott, M. B. A.; Wang, S.; Marinsek, N.; Ranganath, R.; Foschini, L.; Ghassemi, M. Reproducibility in machine learning for health research: Still a ways to go. *Sci. Transl. Med.* **2021**, *13*, No. eabb1655.
(11) Musgrave, K.; Belongie, S.; Lim, S.-N. A Metric Learning Reality Check. In *Computer Vision − ECCV 2020*: Cham, 2020; pp 681−699.
(12) Melis, G.; Dyer, C.; Blunsom, P. *On the State of the Art of Evaluation in Neural Language Models*. CoRR 2017.
(13) Lucic, M.; Kurach, K.; Michalski, M.; Gelly, S.; Bousquet, O. Are GANs Created Equal? A Large-Scale Study. In *32nd Conference on Neural Information Processing Systems (NeurIPS 2018)*, 2018.
(14) Recht, B.; Roelofs, R.; Schmidt, L.; Shankar, V. Do ImageNet Classifiers Generalize to ImageNet?. In *Proceedings of the 36th International Conference on Machine Learning*, 2019, pp 5389−5400.
(15) Diaba-Nuhoho, P.; Amponsah-Offeh, M. Reproducibility and research integrity: the role of scientists and institutions. *BMC Research Notes* **2021**, *14*, 451.
(16) Bates, S.; Hastie, T.; Tibshirani, R. Cross-Validation: What Does It Estimate and How Well Does It Do It? *J. Am. Stat. Assoc.* **2024**, *119*, 1434−1445.
(17) Dietterich, T. G. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation* **1998**, *10*, 1895−1923.
(18) Tossou, P.; Wognum, C.; Craig, M.; Mary, H.; Noutahi, E. Real-World Molecular Out-Of-Distribution: Specification and Investigation. *J. Chem. Inf. Model.* **2024**, *64*, 697−711.
(19) Landrum, G. A.; Beckers, M.; Lanini, J.; Schneider, N.; Stiefl, N.; Riniker, S. SIMPD: an algorithm for generating simulated time splits for validating machine learning approaches. *J. Cheminf.* **2023**, *15*, 119.
(20) Ektefaie, Y.; Shen, A.; Bykova, D.; Marin, M. G.; Zitnik, M.; Farhat, M. Evaluating generalizability of artificial intelligence models for molecular datasets. *Nature Machine Intelligence* **2024**, *6*, 1512−1524.
(21) Steshin, S. Lo-hi: Practical ml drug discovery benchmark. In *Advances in Neural Information Processing Systems*, 2023, pp 64526−64554.
(22) Sheridan, R. P.; Feuston, B. P.; Maiorov, V. N.; Kearsley, S. K. Similarity to molecules in the training set is a good discriminator for prediction accuracy in QSAR. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 1912−1928.
(23) Sheridan, R. P. Time-split cross-validation as a method for estimating the goodness of prospective prediction. *J. Chem. Inf. Model.* **2013**, *53*, 783−790.
(24) Smith, G. D.; Ebrahim, S. Data dredging, bias, or confounding: They can all get you into the BMJ and the Friday papers. *BMJ.* **2002**, *325*, 1437−1438.
(25) Ioannidis, J. P. Why most published research findings are false. *PLOS Medicine* **2005**, *2*, No. e124.
(26) Jager, L. R.; Leek, J. T. An estimate of the science-wise false discovery rate and application to the top medical literature. *Biostatistics* **2014**, *15*, 1−12.
(27) Benjamini, Y.; Hechtlinger, Y. Discussion: an estimate of the science-wise false discovery rate and applications to top medical journals by Jager and Leek. *Biostatistics* **2014**, *15*, 13−16.
(28) Head, M. L.; Holman, L.; Lanfear, R.; Kahn, A. T.; Jennions, M. D. The extent and consequences of p-hacking in science. *PLOS Biology* **2015**, *13*, No. e1002106.
(29) Chen, S.-Y.; Feng, Z.; Yi, X. A general introduction to adjustment for multiple comparisons. *Journal of Thoracic Disease* **2017**, *9*, 1725−1729.
(30) Dunn, O. J. Multiple comparisons among means. *J. Am. Stat. Assoc.* **1961**, *56*, 52−64.
(31) Heid, E.; Greenman, K. P.; Chung, Y.; Li, S.-C.; Graff, D. E.; Vermeire, F. H.; Wu, H.; Green, W. H.; McGill, C. J. Chemprop: a machine learning package for chemical property prediction. *J. Chem. Inf. Model.* **2024**, *64*, 9−17.
(32) Maier-Hein, L.; Reinke, A.; Godau, P.; Tizabi, M. D.; Buettner, F.; Christodoulou, E.; Glocker, B.; Isensee, F.; Kleesiek, J.; Kozubek, M.; et al. Metrics reloaded: recommendations for image analysis validation. *Nat. Methods* **2024**, *21*, 195−212.
(33) Murphy, K. P. *Probabilistic Machine Learning: An Introduction*; MIT press, 2022.
(34) Fang, C.; Wang, Y.; Grater, R.; Kapadnis, S.; Black, C.; Trapa, P.; Sciabola, S. Prospective validation of machine learning algorithms for absorption, distribution, metabolism, and excretion prediction: An industrial perspective. *J. Chem. Inf. Model.* **2023**, *63*, 3263−3274.
(35) Delaney, J. S. ESOL: estimating aqueous solubility directly from molecular structure. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 1000−1005.
(36) Wu, Z.; Ramsundar, B.; Feinberg, E. N.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: a benchmark for molecular machine learning. *Chem. Sci.* **2018**, *9*, 513−530.
(37) Cohen, J. *Statistical Power Analysis for the Behavioral Sciences*; Routledge, 2013.
(38) Ellis, P. D. *The Essential Guide to Effect Sizes: Statistical Power, Meta-Analysis, and the Interpretation of Research Results*; Cambridge University Press, 2010.
(39) Fluetsch, A.; Trunzer, M.; Gerebtzoff, G.; Rodríguez-Pérez, R. Deep Learning Models Compared to Experimental Variability for the Prediction of CYP3A4 Time-Dependent Inhibition. *Chem. Res. Toxicol.* **2024**, *37*, 549−560.
(40) Landrum, G. A.; Riniker, S. Combining IC50 or Ki Values from Different Sources Is a Source of Significant Noise. *J. Chem. Inf. Model.* **2024**, *64*, 1560−1567.
(41) Kramer, C.; Kalliokoski, T.; Gedeck, P.; Vulpetti, A. The experimental uncertainty of heterogeneous public Ki data. *J. Med. Chem.* **2012**, *55*, 5165−5173.
(42) Brown, S. P.; Muchmore, S. W.; Hajduk, P. J. Healthy skepticism: assessing realistic model performance. *Drug Discovery Today* **2009**, *14*, 420−427.
(43) Van Tilborg, D.; Alenicheva, A.; Grisoni, F. Exposing the limitations of molecular machine learning with activity cliffs. *J. Chem. Inf. Model.* **2022**, *62*, 5938−5951.
(44) Bio, I. Get with the program - Inductive Bio Blog. 2024; https://www.inductive.bio/blog/building-better-benchmarks-for-adme-optimization, (accessed Oct 8, 2024).
(45) Lanini, J.; Huynh, M. T. D.; Scebba, G.; Schneider, N.; Rodriguez-Perez, R. UNIQUE: A framework for uncertainty quantification benchmarking. *J. Chem. Inf. Model.* **2024**, *64*, 8379−8386.
(46) Seabold, S.; Perktold, J. statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*, 2010, pp 92−96..
(47) Terpilowski, M. A. scikit-posthocs: Pairwise multiple comparison tests in Python. *J. Open Source Softw.* **2019**, *4*, 1169.
(48) Pedregosa, F.; et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **2011**, *12*, 2825−2830.
(49) Vallat, R. Pingouin: statistics in Python. *J. Open Source Softw.* **2018**, *3*, 1026.
(50) Ash, J. R.; Hughes-Oliver, J. M. chemmodlab: a cheminformatics modeling laboratory R package for fitting and assessing machine learning models. *J. Cheminf.* **2018**, *10*, 57.


https://doi.org/10.1021/acs.jcim.5c01609
J. Chem. Inf. Model. 2025, 65, 9398−9411