Practically significant method comparison protocols for machine learning in small molecule drug discovery.

Jeremy R. Ash[1†], Cas Wognum[2, 3*†], Raquel Rodríguez-Pérez[4], Matteo Aldeghi[5], Alan C. Cheng[6], Djork-Arné Clevert[7], Ola Engkvist[8, 9], Cheng Fang[10], Daniel J. Price[11], Jacqueline M. Hughes-Oliver[12], W. Patrick Walters[13]

[1]Johnson & Johnson Innovative Medicine, Spring House, PA, USA.
[2]Valence Labs, Montréal, Québec, Canada.
[3]Recursion Pharmaceuticals, Salt Lake City, UT, USA.
[4]Novartis Pharma AG, Basel, Switzerland.
[5]Bayer Research and Innovation Center, Cambridge, MA, USA.
[6]Merck & Co., Inc., South San Francisco, CA, USA.
[7]Pfizer Research and Development, Berlin, Berlin, Germany.
[8]Molecular AI, Discovery Sciences AstraZeneca R&D, Gothenburg, Mölndal, Sweden.
[9]Department of Computer Science and Engineering, Chalmers University of Technology, Gothenburg, Mölndal, Sweden.
[10]Blueprint Medicines Corporation, Cambridge, MA, USA.
[11]Nimbus Therapeutics, Boston, MA, USA.
[12]Department of Statistics, North Carolina State University, Raleigh, NC, USA.
[13]Relay Therapeutics, Cambridge, MA, USA.

*Corresponding author(s). E-mail(s): cas@valencelabs.com;
†These authors contributed equally to this work.

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 and open-source software tools, lay a foundation for robust ML benchmarking and thus the development of more impactful methods.

Keywords: Machine Learning, Drug Discovery, Benchmarks, Datasets, Method Comparison, Statistical Significance, Cross-Validation, Performance Metrics

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 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.

When deploying a new model in industry or when publishing a new approach in the scientific literature, we employ method comparison protocols. In industry, established methods, which have shown robustness over time or for which mature technology is available for deployment, might be preferred. Reliable results are essential to justify the investment in deploying a new type of model. Furthermore, scientists who use ML models to inform their decision-making are typically not the ones who have developed the models. To build trust among such interdisciplinary teams, it is important that performance during testing accurately represents the performance once deployed in real drug discovery programs. When proposing a new method in the scientific literature, it is important to contextualize the results by comparing its performance to a simple baseline and the current state of art. Hence, in both cases, appropriate statistical tests and performance metrics are needed to identify robust improvements [1].

These circumstances highlight the need for statistically rigorous method comparison protocols and domain-appropriate techniques. The stochasticity in modeling methods necessitates the comparison of 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), imbalanced, and noisy datasets that are publicly available in drug discovery. 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. For ML-based property modeling, most ML 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 [2]. 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* [3]. While this is thus not specific to ML-based science, researchers were also unable to replicate a large fraction of research from the ML community [4]. 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 [5]. 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 [6].

McDermott et al. [7] 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 dataset 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). [8–11]

While some researchers in ML recognize the importance of statistical replicability [7], there is still substantial room for improvement. In fields such as computer vision and natural language processing, where fit-for-purpose datasets 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 datasets, an in-depth statistical analysis may also be computationally infeasible. In contrast, datasets 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 [12]. We try to address this gap by providing clear guidelines, annotated examples, and by integrating the suggested techniques in open-source software to simplify the adoption of best practices

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.

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 cookbook and, in practice, each case scenario will likely require its own unique considerations. Based on the characteristics of the dataset or project's goal, deviations from this workflow are reasonable. Transparency is key in the absence of a perfect solution for every scenario.

In the rest of this section, 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 dataset 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.

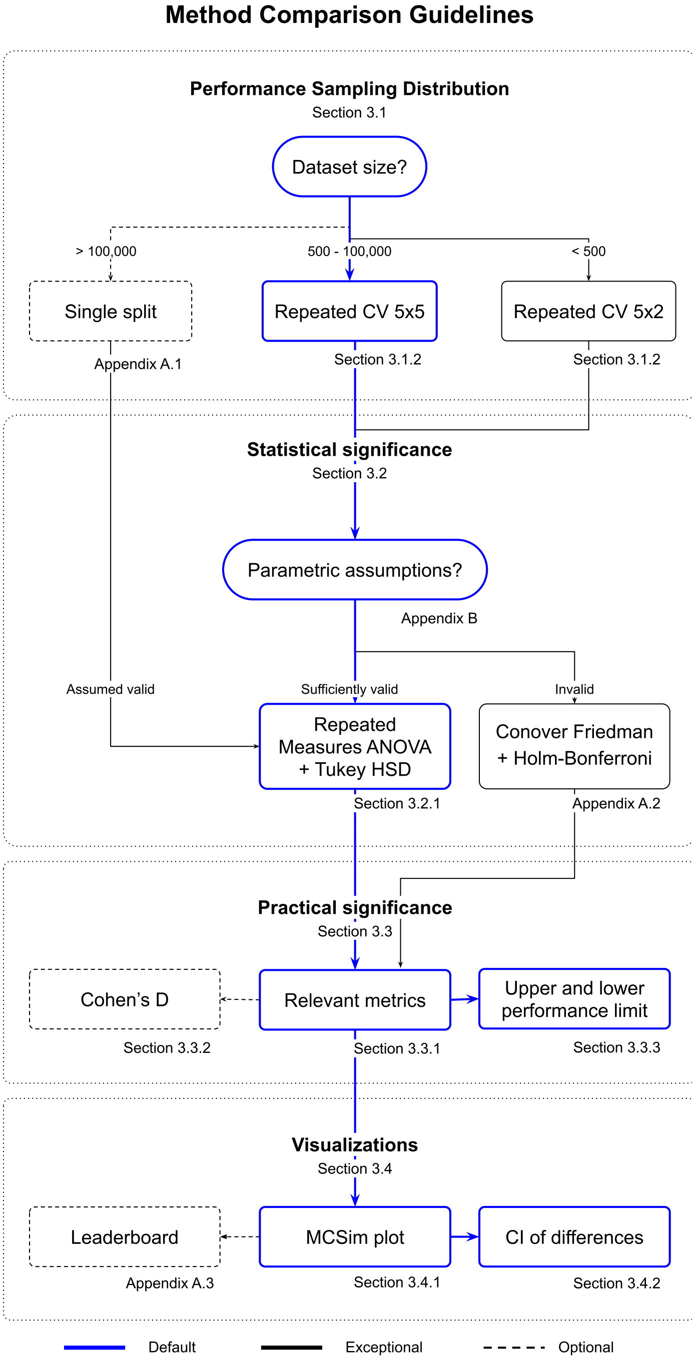


Fig. 1: The method comparison guidelines presented in this work are summarized by this decision tree. The path through the decision tree shown in blue should apply to most use cases, but solutions for exceptional cases are presented as well.

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 datasets.

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:

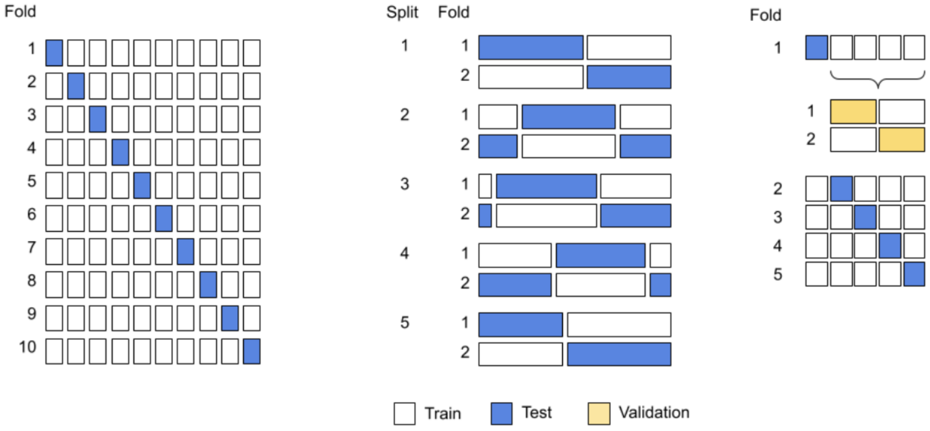
Guidelines 1 (Performance sampling distribution)**.** *We recommend using a 5x5 repeated cross-validation procedure to sample the performance distribution. This procedure suits typical dataset sizes used in small molecule property modeling (e.g., 500 - 100,000). The training set can be further split into a training and validation set if needed.*

In the exceptional case of a dataset having fewer than 500 or more than 100,000 molecules, we provide additional guidance in Appendix 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 dataset (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 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 dataset. 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 dataset, and there exists no perfect solution that will work in every case. What works best depends on the specific dataset



(a) Vanilla CV (b) Repeated CV (5x2) (c) Nested CV

Fig. 2: Visualization of different cross-validation resampling techniques.

and modeling objective. New techniques to sample performance distributions are also actively being researched [13].

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 to result in strong dependency between samples and are generally not recommended [13].

Dieterrich proposed a 5x2 repeated CV to address some of these concerns (see Figure 2b). 5x2 CV splits the dataset five times, with two folds each time. Having only two folds reduces the dependence across CV folds within a split because the training sets do not overlap. Repeated splitting does introduce dependence across splits as training and test sets overlap between replicates. However, such overlap is less substantial than what would be observed when getting the same number of samples with vanilla 10-fold CV.

Even though Deitterich found that 5x2 repeated CV struck the right balance, his paper was based on simulations with datasets of only 300 observations. For modern data set sizes, the 5x2 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. [13], derived 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 Appendix B), we show that for representatively sized datasets, 5x5 repeated CV (i.e. 5 replicates of 5 fold CV) provides a reasonable approximation and a more stable and accurate variance estimate than the commonly recommended Deitterich's 5x2 and McNemar procedures. This experiment leads us to suggest the use of 5x5 repeated CV in our guidelines for improved statistical testing.

3.1.3 Cross-validation with advanced splits

When evaluating a method, it is critical to avoid a model simply "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 dataset and which split is best depends on the application. One can split a dataset randomly, based on temporal information (e.g. compound synthesis or measurement dates), or to minimize the structural overlap between train and test. In this last case, the splitting procedure can be based on chemical scaffolds or similarity clustering. We aim to provide guidance on measuring generalization through data splitting 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 dataset can be partitioned into non-overlapping, roughly equally-sized groups. It is essential to check that folds do not significantly overlap across replicates and that target distributions stay reasonably similar. It is recommended to visually inspect these constraints (see Appendix 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 iteration of 5x5 repeated CV, we recommend performing a single 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). Because many CV replicates

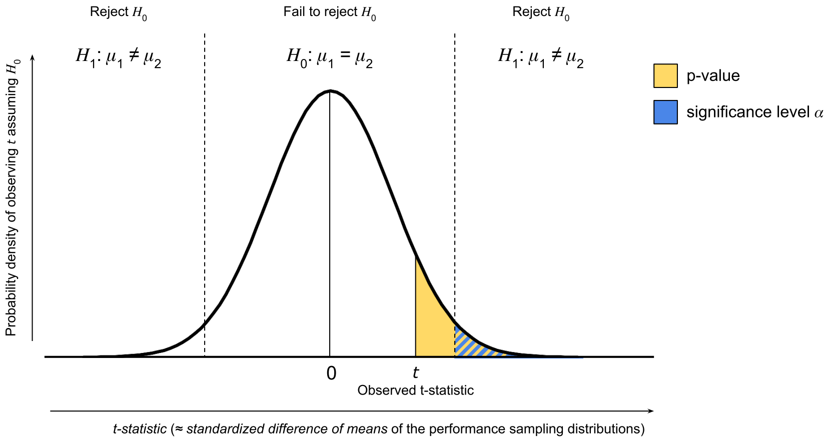

Reject H_0
Fail to reject H_0
Reject H_0
H_1: $\mu_1 \neq \mu_2$
H_0: $\mu_1 = \mu_2$
H_1: $\mu_1 \neq \mu_2$
p-value
significance level α
Probability density of observing t assuming H_0
0
t
Observed t-statistic
t-statistic (≈ *standardized difference of means* of the performance sampling distributions)

Fig. 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.

are already performed by 5x5 repeated CV, this will collect a sufficient number of samples for method comparison.

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 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 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.*

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

3.2.1 Statistical tests

Statistical tests for differences between distributions can be broadly separated into parametric and non-parametric tests. Parametric tests make stronger assumptions about the distributions under comparison (e.g. normality, see also Appendix C), compared to non-parametric tests. One common misconception is that non-parametric tests do not make assumptions. Even though non-parametric 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 non-parametric 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 non-parametric 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 non-parametric 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 non-parametric 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. Non-parametric tests can also be substantially less powerful than parametric tests if the distributional assumptions of the parametric tests are met. See Appendix C.2 for more details on the advantages and disadvantages of parametric and non-parametric 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 TukeyHSD alone.

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 Appendix C.1 for details). If the assumptions of the parametric test are strongly violated, then we recommend a non-parametric test workflow that will also be suitable for method comparisons (Appendix 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 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. [14] for a review. The Bonferroni correction [15] is a simple approach that is commonly used. However this correction is known to have low statistical power when the number of comparisons is large.

The recommended 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 Appendix A.2 for guidance on multiple testing for a large number of method comparisons (> 10).

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 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. Appendix 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

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

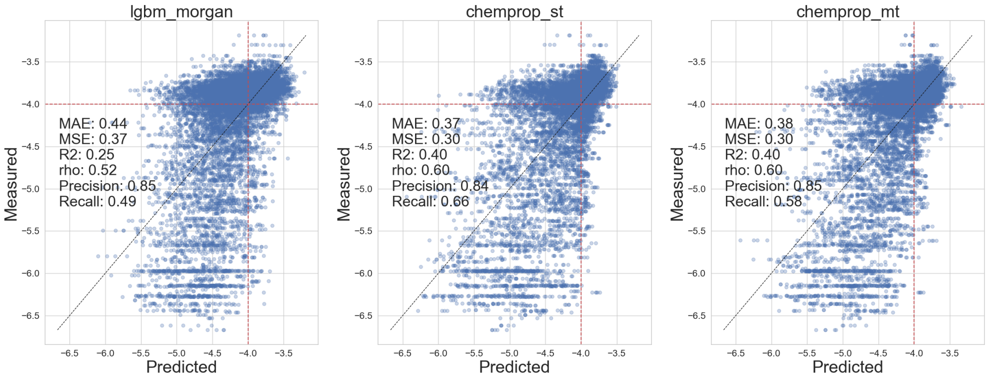


Fig. 4: An example using *post hoc* classification for a regression model to investigate practical significance with precision and recall.

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 [16, 17].

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 AppendixD.2 for details).

Figure 4 shows a comparison of three machine learning models, ChemProp Multitask [18] (`chemprop_mt`), ChemProp Single Task (`chemprop_st`), and Light Gradient Boosting Machines (`lightGBM`) on the same dataset. In drug discovery, we typically screen early for molecules 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 is > 100 µM. After training three regression models for solubility, statistically significant differences in MAE, MSE, and R2 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 uM 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 µM, `lightGBM` would thus reject more molecules with good solubility. As we will later show in Figure 8 and Table 1, the estimated improvement in recall of `chemprop_st` over `lightGBM` is .17 (.15, .19), meaning `chemprop_st` would identify 17% more molecules with good solubility. This would likely have a real practical impact on drug discovery programs.

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 Mean Absolute Error (MAE). Although this metric is not the only metric that should be used for method development (see Appendix 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.

Dynamic Range

Both correlation and error metrics are influenced by the dynamic range of the data being modeled. Achieving a high correlation on datasets with a broader range of experimental values is generally easier, whereas datasets 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 dataset [19] in the MoleculeNet [20] benchmark. This dataset reports the log of the aqueous solubility (LogS) for 2,173 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 R2 for LogP vs LogS for the full 13-log range of the Delaney solubility dataset, we achieve a respectable Pearson r2 of 0.68. However, if we only consider values in the 1 µM to 1 mM (log solubility -6 to -3) range typically observed in drug discovery projects, the R2 value drops to a less impressive 0.33. Figure 5 illustrates this issue by showing the full range of the Delaney dataset, with a more realistic dynamic range between the red lines.

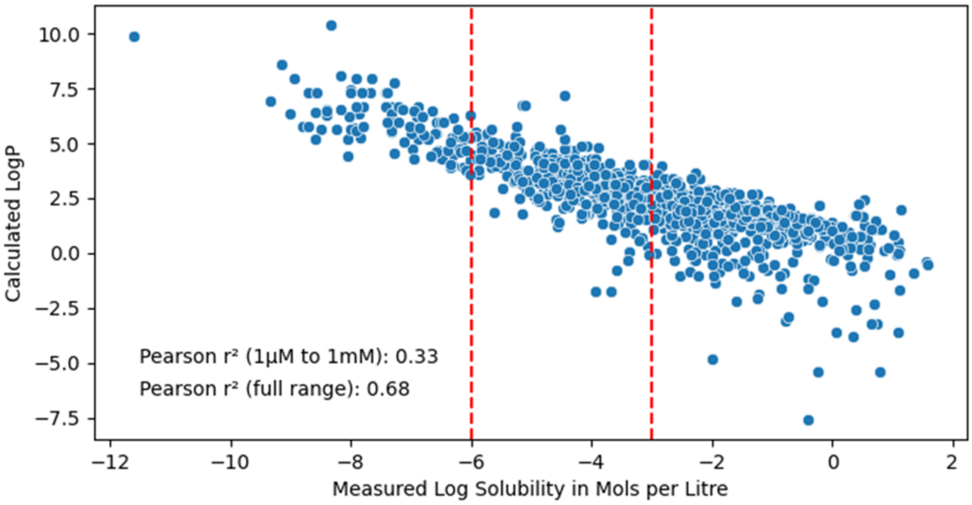


Fig. 5: Examining the impact of dynamic range on correlation. If the entirety of this dataset, which spans 13 logs of dynamic range, is considered, there is a high correlation between measured and estimated values. However, the correlation is much lower if the more realistic 3-log range between the red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

Class imbalance

Classification metrics can be misleading in cases where classification datasets are highly imbalanced, as is common in small molecule drug discovery. In this case, using metrics that account for this imbalance is important (see Appendix D.2).

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 ≥ 0.8, implying a small, medium, or large effect size, respectively [21]. Statisticians often

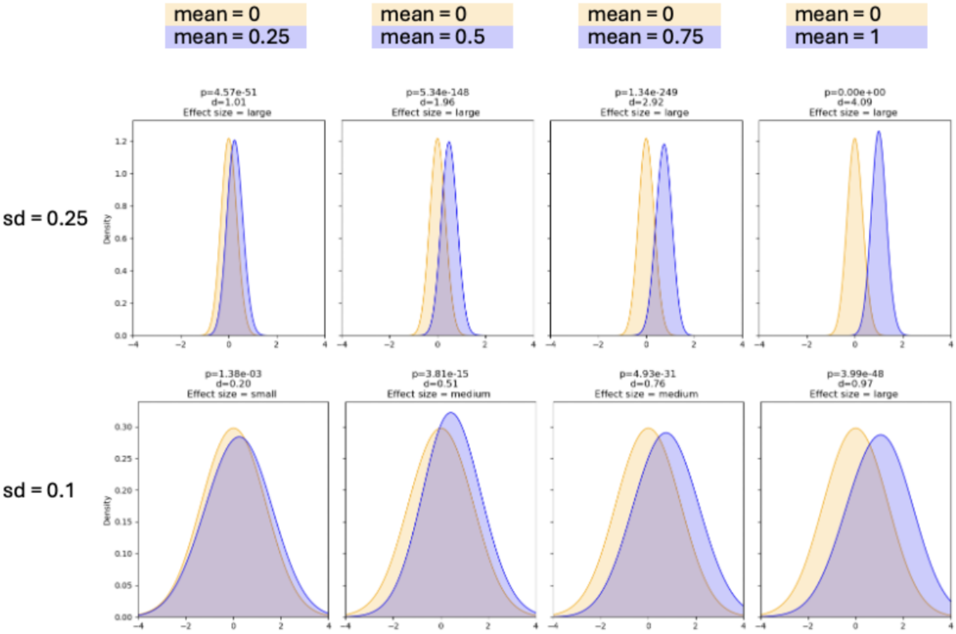


Fig. 6: An illustration of effect size. In all of the subplots, the two distributions show a statistically significant difference. For each column, the mean of the same-colored distributions is equal. However, because the distributions in the top row have a lower variance, the effect size of these comparisons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

advise using these cutoffs as a last resort when there is insufficient understanding of whether a difference is meaningful from domain knowledge [22].

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.

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.

Upper limit: Experimental variability

If the experimental variability of the underlying assay is known, it can be used to estimate the maximum expected performance [23]. For example, the noise in activity biochemical assays measuring half-maximal inhibitory concentration (IC50) is commonly estimated to be 0.3 log units (i.e. 2-fold). If the MAE of an IC50 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 [24].

In the special case of correlation metrics for regression models, Brown et al. [25] outlined a procedure for a dataset 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 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 [26], performance per chemical series [27], or uncertainty estimation [28]) 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 an extension of the sign plot, which includes statistical significance and effect size in a single plot. We recommend including an additional plot in the supplementary material that conveys the confidence intervals of the effect size of the pairwise comparisons.*

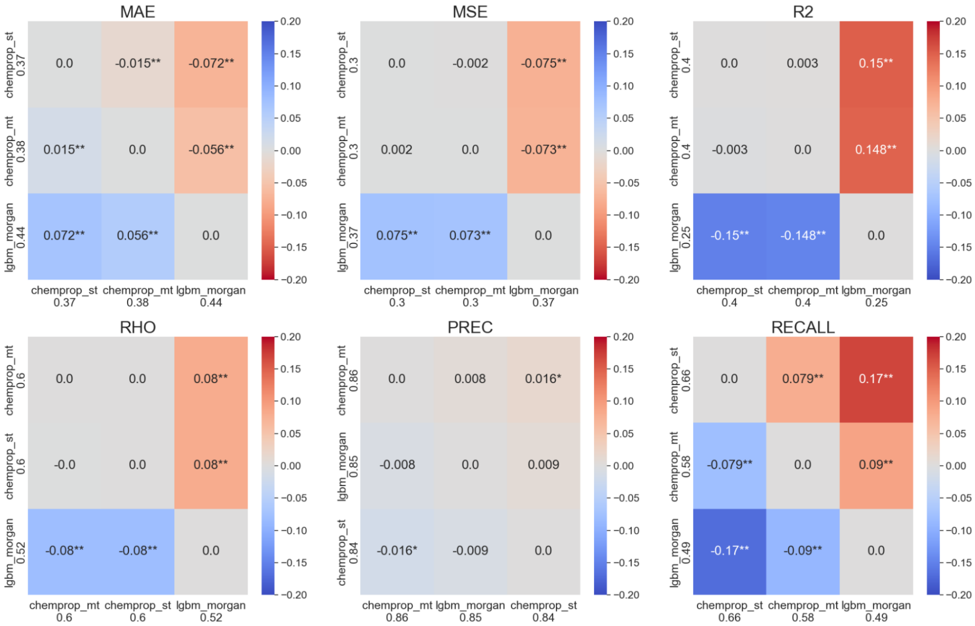


Fig. 7: An example of the Multiple Comparisons Similarity (MCSim) plot. Color is used to convey the effect size, whereas star annotations are used to convey statistical significance. The effect size reported is the difference in average performance between methods. A numeric difference is shown in the cells, but this can be suppressed if a large number of comparisons is performed.

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

3.4.1 The Multiple Comparisons Similarity plot

The first plot is an extension to the sign plot provided by the scikit-posthocs Python package. 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 (* $p < .05$, ** $p < .01$, *** $p < .001$). 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.

3.4.2 Confidence intervals of the difference in mean performance

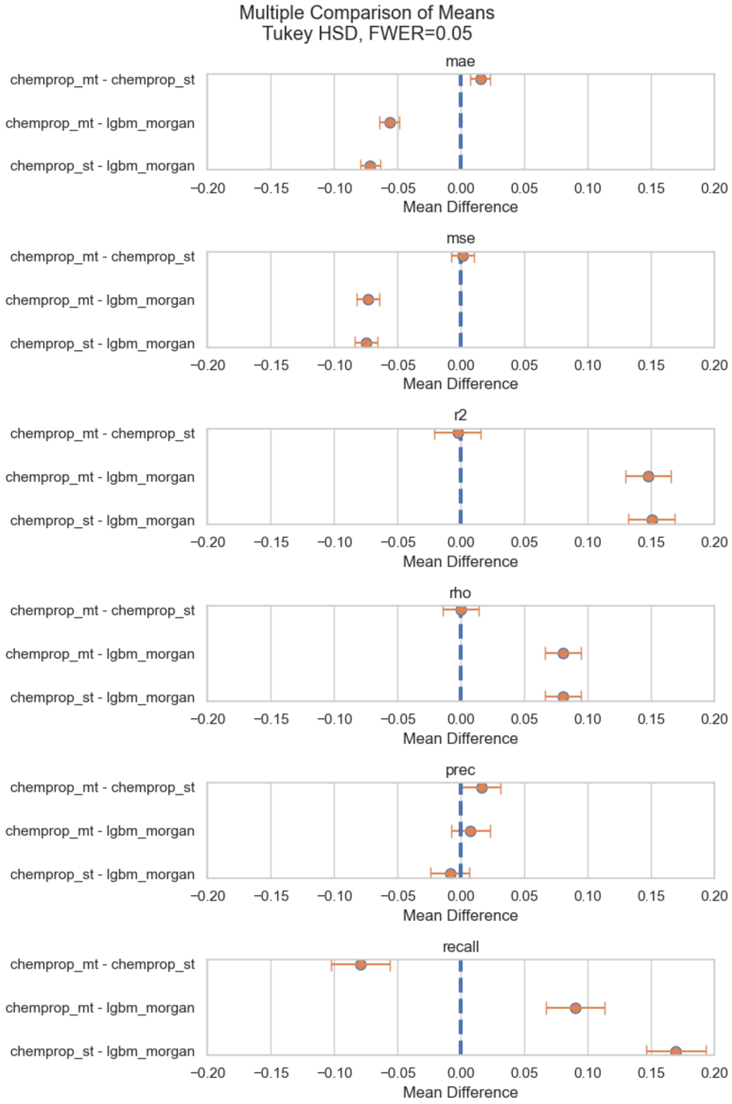


Fig. 8: Confidence Intervals (CI) of the difference in mean performance between methods, presented as a plot. Intervals that do not cross the zero line imply statistical significance.

	MAE	MSE	R2	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)

Table 1: Confidence Intervals (CI) of the difference in mean performance between methods, presented as a table.

Package	Language	Description
scikit-posthocs [30]	Python	Implements Multiple Pairwise Comparisons Tests in Python
scikit-learn	Python	This well-known machine learning library for Python has a mature cross-validation API.
pingouin [31]	Python	Implements various statistical methods in Python.
chemmodlab [32]	R	A Cheminformatics Modeling Laboratory for Fitting and Assessing Machine Learning Models

Table 2: An overview of useful open-source software.

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 non-parametric workflow.

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 supplementary materials as either a plot (see Figure 8) 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 avoid data dredging [29].

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.

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 datasets, 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 5x5 repeated cross-validation procedure to sample the performance distribution. This procedure suits typical dataset sizes used in small molecule property modeling (e.g., 500 - 100,000). The training set can be further split into a training and validation set if needed.
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.
4. We recommend an extension of the sign plot, which includes statistical significance and effect size in a single plot. We recommend including an additional plot in the supplementary material that conveys the confidence intervals of the effect size of the pairwise comparisons.
5. Statistical testing is not a cookbook 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 dataset curation and measuring generalization (e.g. through data splitting methods).

Appendix

A Exceptional Cases

A.1 Performance Sampling Distributions

We provide recommendations for the following exceptional cases. If the dataset is small (< 500), we recommend performing 5x2 repeated CV. This is supported by Deiterrich's original simulation experiments [33]. If the dataset is very large (e.g., $> 100,000$), such that repeated CV is no longer computationally tractable, then statistical testing is unlikely necessary, as even small differences between methods are likely statistically significant.

There have been statistical tests proposed for large data sets where only one split and model fit are required (e.g., McNemar's test for classification models) [34]. We do not recommend these tests because they do not assess the variability of performance across multiple data splits, potentially resulting in an elevated false positive rate.

A.2 Statistical Testing

If the Tukey HSD assumptions are egregiously violated, a non-parametric test should be used. The supplementary notebooks provide an example of the non-parametric testing workflow. We recommend using the Conover-Friedman test for pairwise comparisons and the Holm-Bonferroni correction for multiple testing.

If a large number of tests are being performed (e.g., all pairwise comparisons between > 10 methods), the Tukey HSD and Conover-Friedman test will have low power to detect significant differences. In this case, it may be preferable to perform a multiplicity adjustment designed for a large number of tests. The Benjamini-Hochberg correction [35] is recommended in these settings. Compared to Bonferroni or Tukey HSD, which control the familywise error rate (FWER), the Benjamini-Hochberg (BH) procedure has greater statistical power, especially when dealing with a large number of comparisons. While FWER control methods aim to limit the probability that at least one false positive occurs in a set of tests, the BH procedure focuses on controlling the false discovery rate, the expected proportion of false positives among the rejected hypotheses. This makes the BH procedure particularly useful for a large number of tests, where maintaining a balance between Type I error control and statistical power is key.

A.3 Leaderboards

For leaderboards, we recommend selecting a primary performance measure and ranking methods according to this metric. Statistical tests may also be performed according to the metric, and results are presented as a compact letter display [36].

In the Table 3 MSE was selected as the primary metric for demonstration purposes. A significant difference was not found between chemprop_st and chemprop_mt so a letter “a” is assigned to both methods. lightgbm was significantly worse than both methods so it is assigned a letter “b”.

CLD	Method	MAE	MSE	R2	Rho	Recall	Precision
a	chemprop_st	0.37	0.30	0.40	0.60	0.66	0.84
a	chemprop_mt	0.38	0.30	0.40	0.60	0.58	0.86
b	lgbm_morgan	0.44	0.37	0.25	0.52	0.49	0.85

Table 3: Example of leaderboard with compact letter display.

B Cross-Validation Experiment

We provide an experiment in the supplementary that compares several approximation methods against the estimate from Bates et al. [13] We use a representative dataset for small molecule property modeling with 2,000 observations. We show that 5x5 repeated CV will provide a more stable and accurate variance estimate than the commonly recommended Deitterich’s 5x2 and McNemar procedures. We also extend the Bates et al. R package to support a variety of classification metrics relevant to imbalanced datasets. Our results were reproduced across performance metrics. Since variance is the crucial parameter to estimate for method comparison statistical testing, these results imply that 5x5 repeated CV will result in a more performant test. Also, the increased number of samples collected will lead to a more powerful test and tighter confidence intervals for the difference between methods. Furthermore, the increased number of CV folds results in more training data being used in each replicate than Deitterich’s 5x2, and this will produce more accurate performance measures overall.

C Statistical Testing Details

C.1 Examining the Parametric Assumptions

We recommend always checking the parametric assumptions of the tests, but following Guidelines 1, assumptions should be reasonably met in most applications in small molecule property modeling. In such cases, we recommend the repeated measures ANOVA followed by Tukey HSD test for pairwise comparison of methods.

These tests make three assumptions on the distribution of performance metrics across CV splits, not the distribution of individual errors. In order of importance, the assumptions of ANOVA and *post hoc* Tukey HSD are:

1. **Independence**: The samples (i.e., performance metrics across CV splits) should be independent. Information about one sample should not allow one to estimate the value for another sample. There is no straightforward way to test for this, which is why the usage of appropriate sampling mechanisms (such as 5x5 repeated CV) is critical to ensure the samples are sufficiently independent.

2. **Homogeneity of Variances**: The variance of the performance sampling distributions for each of the models is approximately equal. A typical rule of thumb is that the ratio of the largest and smallest variance should not be larger than 3. However, in the case of equally sized groups, much larger variance ratios are tolerated and these can be as high as 9 [37]. Since the same number of CV iterations are performed for each model in 5x5 repeated CV, the groups are equally sized. This assumption will rarely be violated.
3. **Normality**: The performance metric distributions are assumed to be approximately normal for each model. ANOVA and Tukey HSD will be robust to moderate normality violations for the 25 samples collected by 5x5 repeated CV. Note that only approximate normality of the data generating *population distribution* is assumed. It can be difficult to examine this when sample sizes are small. In these cases, an argument is often made on the conceptual level for approximate normality. This argument may be made for performance metrics since they are typically the sum of many variables (e.g., the sum of individual errors) and will be approximately normal due to the Central Limit Theorem (CLT) [38]. Moreover, even if normality may not be directly justifiable for the population distribution of performance metrics, the CLT will typically rescue the normality assumption; this is because of the averaging that is performed at the heart of ANOVA and Tukey HSD testing. A question may be raised as to whether a sample size of 25 is large enough for the benefits of the CLT to be realized; we argue this will be sufficient in most cases. Since 25 samples are collected from each distribution, one is able to check for strong violations of normality. The best way of doing this is by visualizing the distribution (e.g., quantile-quantile or QQ plots of ANOVA residuals). See the supplementary notebooks for an example. We recommend against using tests for normality as they may have low statistical power, or may be even more sensitive to their associated assumptions than the low level of sensitivity of the ANOVA and Tukey HSD procedures to the assumption of normality.

C.2 Parametric vs. Non-Parametric Tests

Despite the increased flexibility of non-parametric tests, we recommend the parametric workflow for method comparison whenever the assumptions are met. Parametric tests are easier to understand and interpret as well as more statistically powerful (i.e., require fewer samples).

Tukey HSD is more interpretable because it tests for differences in distributions using the mean, an easily described summary measure of the data. On the contrary, non-parametric tests, such as the Wilcoxon signed-rank test, are typically based on rank and do not necessarily test for a difference in distribution in an easily described summary measure of the data [39]. Non-parametric tests are often considered a test for a difference in the median, but this interpretation is only valid if the shapes of the two distributions being compared are the same. This makes it more difficult to tell whether a statistically significant difference is meaningful. For example, two methods may show the same median in their performance distribution but differ in skewness. This skewness difference could be statistically significant according to a non-parametric test, but it is unlikely meaningful in practice if the medians are equivalent.

It is common for researchers to assume a non-parametric test is necessary because their data are not normally distributed. For the comparison of performance metrics, the normality assumption will often be reasonably satisfied (see Appendix C.1).

Generally, non-parametric methods are focused on hypothesis testing rather than the estimation of an interpretable effect size. Since effect size estimation is a goal in our method comparison guidelines, Tukey HSD is recommended.

D Performance Metrics

When evaluating ML models, it is imperative to understand and report relevant performance metrics. Over the past century, statisticians and other computational scientists have developed many valuable metrics for regression and classification models. This section reviews several of these metrics and provides recommendations to ensure accurate and meaningful model evaluations.

D.1 Evaluating Models as Regressors

Ideally, regression models will accurately estimate a property in its original units. This will enable chemists to understand the magnitude of differences between compounds, which is ideal for many use cases such as drug design. Accurate regressor performance also provides confidence that models can be used as components of in-vivo models or multi-parameter optimization scores.

Two main categories of metrics are commonly used when evaluating regression models: correlation metrics and error metrics. Correlation metrics such as Pearson's r and the coefficient of determination (R^2) quantify the strength and direction of the relationship between estimated and true values.

Error metrics such as the mean absolute error (MAE) or root mean squared error (RMSE) are calculated as the mean difference between the estimated and true values.

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|T_i - P_i|$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(T_i - P_i)^2}$$

For the MAE, the median may be used instead of the mean for a comparable measure that is less sensitive to large outliers. But note that the median only provides certainty that half of errors are below a threshold. For stronger control, a metric like the percentile absolute error (PAE) will guarantee most model errors are within a certain bound. For example if the 90PAE is equal to 3-fold, then 90% of the data has less than 3-fold error.

While MAE, RMSE, and PAE can often provide an intuitive perception of model performance, they pose some potential difficulties. Like the correlation metrics, error metrics can be impacted by the dynamic range of the data. Datasets with a smaller dynamic range can produce unrealistically small values for error metrics, even in the case of poorly performing models.

When evaluating regression models, we recommend reporting the following.

- At least one correlation metric (Pearson r, R^2)
- At least one error metric (MAE, RMSE, PAE)

All metrics above can be easily calculated using the sckit-learn or scipy Python libraries.

D.2 Evaluating Models as Binary Classifiers

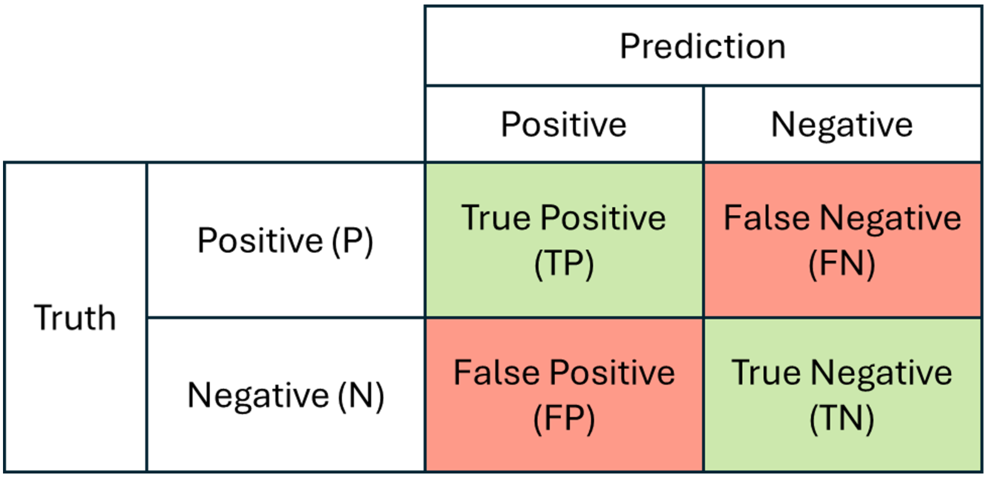
This section is relevant to both classification models and regression models evaluated as binary classifiers. Often if a regression model has suboptimal performance as a regressor, it can still provide value as a classifier.

Classification models often estimate the probability of positive class membership. Classification is performed by applying a threshold to the estimated probability. Regression models can be converted to a classifier by applying a threshold to their property estimations. See Section 3.3.1 for a description of how thresholding can be useful when assessing the decisional impact of the performance differences between regression models.

Once classification is performed, most methods for evaluating models are derived from the confusion matrix shown below. The confusion matrix quantifies the number of negative and positive classifications that agree and disagree with the true values.

Subsequently, the number of true positive (TP), false positive (FP), true negative (TN), and false negative (FN) classifications can be used to calculate several metrics.

In small molecule property modeling, classification datasets are often highly imbalanced and contain a small number of positive examples and a large number of negative examples. Some metrics can be misleading in these cases. Consider **accuracy** (ACC), one of the most commonly used classification metrics. In the equation below, TP and TN are as above, and P and N are the number of positive and negative examples.



		Prediction	
		Positive	Negative
Truth	Positive (P)	True Positive (TP)	False Negative (FN)
	Negative (N)	False Positive (FP)	True Negative (TN)

Fig. S1: The confusion matrix is used to calculate a range of classification metrics.

$$ACC = \frac{TP + TN}{P + N}$$

If we have a dataset with 95 positive examples and 5 negative examples where all the examples are classified as negative, our accuracy is (0+95)/(5+95)=0.95. While the model is accurate, it has no practical value. When imbalanced datasets are used, a more appropriate metric like **Cohen's** κ should be used.

$$\kappa = \frac{2 * (TP * TN - FN * FP)}{(TP + FP) * (FP + TN) * (TP + FN) * (FN + TN)}$$

The previous example, which had an accuracy of 0.95, had a Cohen's κ of 0.0.

Another classification metric, **Matthew's Correlation Coefficient (MCC)**, reframes classification in a form that may be more accessible to those familiar with regression metrics. This method, also known as the phi coefficient, is often considered the classification analog of Pearson's r. One of the strengths of MCC is its ability to balance positive and negative classifications and handle imbalanced datasets.

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

It is important to note that Cohen's κ and MCC place equal importance on estimating the positive and negative class. While this provides a useful summary of performance across both classes, for many applications in property modeling we only care about estimation of the positive class. Two performance metrics typically used in this context are **precision** and **recall**.

$$Precision = \frac{TP}{TP + FP}$$
$$Recall = \frac{TP}{TP + FN}$$

Precision indicates the fraction of true examples retrieved. If we apply a threshold to a solubility classification model's estimated probabilities and classify 100 compounds as good solubility, precision quantifies the fraction of compounds selected that are true positives. If 80 compounds selected are TP and 20 are FP, we have a precision of 80/(80+20) = 0.8. Recall is the fraction of TP compounds selected. If in the same example, there were actually 500 positives in the entire set, then we have 80 TP and 420 positive molecules misclassified as negative (FN). We can calculate the recall as 80/(80 + 420) = 0.16.

For imbalanced data, it is often informative to report precision relative to the prevalence of the positive class, which is the fraction of true positives in the data. Even if precision appears to be suboptimal it may provide substantial enrichment over the baseline prevalence if positives are rare. Common approaches to this are to take the ratio (precision / prevalence) which is called the **enrichment factor** [40], or to take the difference (precision - prevalence) for a more interpretable measure of lift. Consider the case where precision is 50% and the positive prevalence is 10%. While this precision may seem suboptimal, the enrichment factor is 5 fold, which is a substantial improvement over baseline.

In assessing the performance of classification models, we frequently aim to evaluate the balance between accurately identifying true positives and avoiding false positives. One effective approach is to plot the true positive rate against the false positive rate at various thresholds, thereby creating a Receiver Operating Characteristic (ROC) curve. The area under this curve, known as the Area Under the ROC (AUROC), serves as a valuable metric for evaluating classifier performance.

Because the ROC assesses performance over a range of thresholds, it is possible that much of the curve will evaluate thresholds that would not be used in practice. This is particularly the case for highly imbalanced datasets where estimation of the positive class is most important. In these cases only the extreme left of the curve is of interest and the AUROC has limited utility. [41]

PRAUC is more appropriate for this type of imbalanced data. [17] PRAUC plots precision vs. recall and calculates the Area Under the Precision-Recall Curve. This curve plots precision vs. recall at various thresholds. Figure S2 shows examples of ROC and PR curves calculated using the scikit-learn Python library.

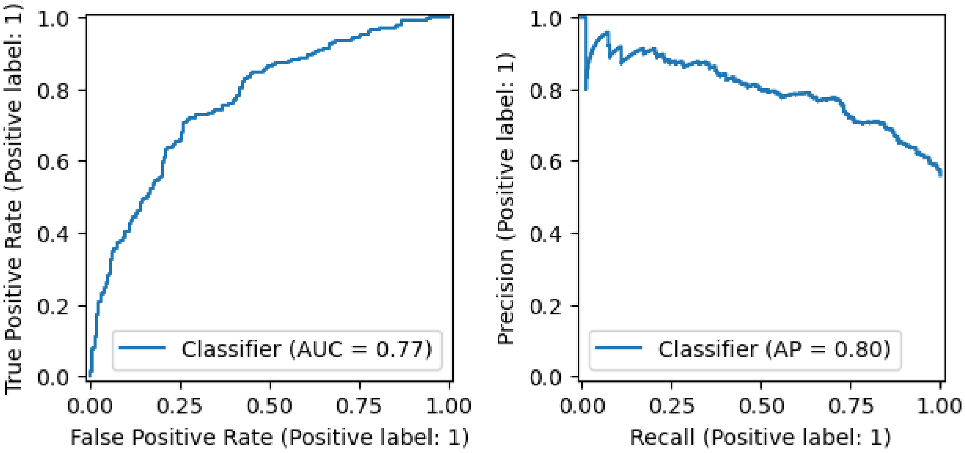


Fig. S2: A ROC curve (left) and PR curve (right).

AUC metrics evaluate performance over a range of thresholds. While these metrics are useful summary measures, it can be difficult to determine when a difference in area is practically significant. To assess the decisional impact in a more interpretable way, it is useful to pick a representative threshold for the use case and report metrics specific to that threshold. This is equivalent to picking one point along the ROC and PR curve. We provide an example of how the decisional impact might be assessed in the context of the example in Section 3.3.1.

When comparing classification models, we recommend reporting the following metrics:

- Matthews Correlation Coefficient
- ROCAUC and PRAUC
- Decisional impact metrics relevant to the application

All the metrics above can be easily calculated using the scikit-learn or scipy Python libraries.

D.3 Evaluating Models as Ranking Algorithms

In drug design, accurate ranking ordering is often desired to provide additional granularity beyond classification. Ranking can be used to prioritize compounds with the most optimal properties, providing chemists with additional guidance on what compounds to make first, and a better understanding of the relationship between compounds.

Both regression and classification models can be used to rank compounds. For classification models, compounds can be ranked by the estimated probability of the positive class, which orders compounds by confidence.

Spearman's ρ and Kendall's τ, evaluate the accuracy of the rank ordering generated by a predictive model. While these metrics are often reported for the entire ranked list, in drug discovery settings it is often most important to accurately rank compounds at the top of the list. Computing the ranking metric at a top fraction according to prediction model score (e.g., top 10%) may be more relevant than a global ranking measure.

When comparing ranking algorithms, we recommend reporting the following metrics. Spearman's ρ or Kendall's τ.

All the metrics above can be easily calculated using the scikit-learn or scipy Python libraries.

D.4 Choosing the Modes for Model Evaluation

For regression models we recommend evaluating models as regressors and as ranking algorithms. If regression performance is suboptimal and a typical target threshold used in practice is known, then we recommend reporting classification metrics as well.

Classification models should be evaluated as classifiers and if additional decisional granularity is desired (e.g., models will be used for drug design) we recommend reporting ranking metrics as well.

E Fold Diagnostic Plots

When using advanced splitting methods, visualizations like Figure S3 can help ensure minimal overlap between folds and reasonable target distribution per fold. Roughly equal fold sizes improve statistical power by reducing performance variability due to fold size.

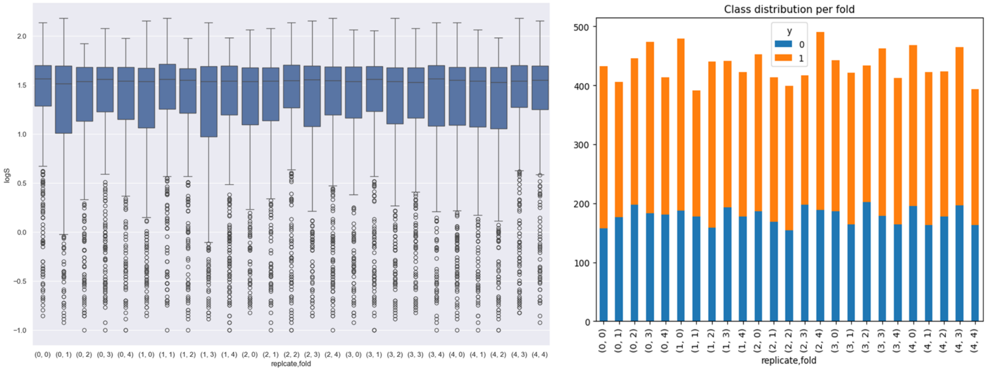


Fig. S3: Target distributions per fold should be approximately equal. For regression (left), the dynamic ranges should be similar. For classification (right), the class distribution should be approximately equal.

References

[1] Wognum, C. *et al.* A call for an industry-led initiative to critically assess machine learning for real-world drug discovery. *Nature Machine Intelligence* **6**, 1120–1121 (2024). URL https://doi.org/10.1038/s42256-024-00911-w.

[2] Kapoor, S. *et al.* Reforms: Consensus-based recommendations for machine-learning-based science. *Science Advances* **10**, eadk3452 (2024). URL https://www.science.org/doi/abs/10.1126/sciadv.adk3452.

[3] Baker, M. 1,500 scientists lift the lid on reproducibility. *Nature* **533**, 452–454 (2016). URL https://doi.org/10.1038/533452a.

[4] Kapoor, S. & Narayanan, A. Leakage and the reproducibility crisis in machine-learning-based science. *Patterns* **4** (2023). URL https://doi.org/10.1016/j.patter.2023.100804.

[5] *Reproducibility and replicability in science* (National Academies Press, 2019).

[6] Beam, A. L., Manrai, A. K. & Ghassemi, M. Challenges to the reproducibility of machine learning models in health care. *Jama* **323**, 305–306 (2020).

[7] McDermott, M. B. A. *et al.* Reproducibility in machine learning for health research: Still a ways to go. *Science Translational Medicine* **13**, eabb1655 (2021). URL https://www.science.org/doi/abs/10.1126/scitranslmed.abb1655.

[8] Musgrave, K., Belongie, S. & Lim, S.-N. Vedaldi, A., Bischof, H., Brox, T. & Frahm, J.-M. (eds) *A metric learning reality check.* (eds Vedaldi, A., Bischof, H., Brox, T. & Frahm, J.-M.) *Computer Vision – ECCV 2020*, 681–699 (Springer International Publishing, Cham, 2020).

[9] Melis, G., Dyer, C. & Blunsom, P. On the state of the art of evaluation in neural language models. *CoRR* **abs/1707.05589** (2017). URL http://arxiv.org/abs/1707.05589.

[10] Lucic, M., Kurach, K., Michalski, M., Gelly, S. & Bousquet, O. Are GANs Created Equal? A Large-Scale Study. *Advances in neural information processing systems* **31** (2018).

[11] Recht, B., Roelofs, R., Schmidt, L. & Shankar, V. Chaudhuri, K. & Salakhutdinov, R. (eds) *Do ImageNet classifiers generalize to ImageNet?* (eds Chaudhuri, K. & Salakhutdinov, R.) *Proceedings of the 36th International Conference on Machine Learning*, Vol. 97 of *Proceedings of Machine Learning Research*, 5389–5400 (PMLR, 2019). URL https://proceedings.mlr.press/v97/recht19a.html.

[12] Diaba-Nuhoho, P. & Amponsah-Offeh, M. Reproducibility and research integrity: the role of scientists and institutions. *BMC Research Notes* **14**, 451 (2021). URL https://doi.org/10.1186/s13104-021-05875-3.

[13] Bates, S., Hastie, T. & Tibshirani, R. Cross-validation: What does it estimate and how well does it do it? *Journal of the American Statistical Association* **119**, 1434–1445 (2023). URL http://dx.doi.org/10.1080/01621459.2023.2197686.

[14] Chen, S.-Y., Feng, Z. & Yi, X. A general introduction to adjustment for multiple comparisons. *Journal of Thoracic Disease* **9** (2017). URL https://jtd.amegroups.org/article/view/13609.

[15] Dunn, O. J. Multiple comparisons among means. *Journal of the American statistical association* **56**, 52–64 (1961).

[16] Maier-Hein, L. *et al.* Metrics reloaded: recommendations for image analysis validation. *Nature methods* **21**, 195–212 (2024).

[17] Murphy, K. P. *Probabilistic machine learning: an introduction* (MIT press, 2022).

[18] Heid, E. *et al.* Chemprop: a machine learning package for chemical property prediction. *Journal of Chemical Information and Modeling* **64**, 9–17 (2023).

[19] Delaney, J. S. Esol: estimating aqueous solubility directly from molecular structure. *Journal of chemical information and computer sciences* **44**, 1000–1005 (2004).

[20] Wu, Z. *et al.* Moleculenet: a benchmark for molecular machine learning. *Chemical science* **9**, 513–530 (2018).

[21] Cohen, J. *Statistical power analysis for the behavioral sciences* (routledge, 2013).

[22] Ellis, P. D. *The essential guide to effect sizes: Statistical power, meta-analysis, and the interpretation of research results* (Cambridge university press, 2010).

[23] Fluetsch, A., Trunzer, M., Gerebtzoff, G. & Rodríguez-Pérez, R. Deep learning models compared to experimental variability for the prediction of cyp3a4 time-dependent inhibition. *Chemical research in toxicology* **37**, 549–560 (2024).

[24] Kramer, C., Kalliokoski, T., Gedeck, P. & Vulpetti, A. The experimental uncertainty of heterogeneous public ki data. *Journal of medicinal chemistry* **55**, 5165–5173 (2012).

[25] Brown, S. P., Muchmore, S. W. & Hajduk, P. J. Healthy skepticism: assessing realistic model performance. *Drug discovery today* **14**, 420–427 (2009).

[26] Van Tilborg, D., Alenicheva, A. & Grisoni, F. Exposing the limitations of molecular machine learning with activity cliffs. *Journal of chemical information and modeling* **62**, 5938–5951 (2022).

[27] Bio, I. Get with the program - inductive bio blog (2024). URL https://www.inductive.bio/blog/building-better-benchmarks-for-adme-optimization. Accessed: 2024-10-08.

[28] Lanini, J., Huynh, M. T. D., Scebba, G., Schneider, N. & Rodríguez-Pérez, R. Unique: A framework for uncertainty quantification benchmarking. *ChemRxiv* (2024).

[29] Smith, G. D. & Ebrahim, S. Data dredging, bias, or confounding: They can all get you into the bmj and the friday papers (2002).

[30] Terpilowski, M. A. scikit-posthocs: Pairwise multiple comparison tests in python. *Journal of Open Source Software* **4**, 1169 (2019).

[31] Vallat, R. Pingouin: statistics in python. *J. Open Source Softw.* **3**, 1026 (2018).

[32] Ash, J. R. & Hughes-Oliver, J. M. chemmodlab: a cheminformatics modeling laboratory r package for fitting and assessing machine learning models. *Journal of Cheminformatics* **10**, 1–20 (2018).

[33] Dietterich, T. G. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation* **10**, 1895–1923 (1998).

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

[35] Benjamini, Y. & Hochberg, Y. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)* **57**, 289–300 (1995).

[36] Gramm, J. *et al.* Algorithms for compact letter displays: Comparison and evaluation. *Computational Statistics & Data Analysis* **52**, 725–736 (2007).

[37] Blanca, M. J., Alarcón, R., Arnau, J., Bono, R. & Bendayan, R. Effect of variance ratio on anova robustness: Might 1.5 be the limit? *Behavior Research Methods* **50**, 937–962 (2018).

[38] Kwak, S. G. & Kim, J. H. Central limit theorem: the cornerstone of modern statistics. *Korean journal of anesthesiology* **70**, 144–156 (2017).

[39] Lumley, T., Diehr, P., Emerson, S. & Chen, L. The importance of the normality assumption in large public health data sets. *Annual review of public health* **23**, 151–169 (2002).

[40] Rodríguez-Pérez, R., Trunzer, M., Schneider, N., Faller, B. & Gerebtzoff, G. Multispecies machine learning predictions of in vitro intrinsic clearance with uncertainty quantification analyses. *Molecular Pharmaceutics* **20**, 383–394 (2022).

[41] Saito, T. & Rehmsmeier, M. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PloS one* **10**, e0118432 (2015).