A *Lo-Hi* benchmark

Lo-Hi is a practical ML drug discovery benchmark, comprising two tasks: Hit Identification (Hi) and Lead Optimization (Lo). Hi corresponds to a binary classification problem, wherein the goal is to identify novel hits that differ significantly from the training dataset [10, 11, 16, 27, 28]. This is why there are no molecules in the test set with ECFP4 Tanimoto similarity exceeding 0.4 to the training set. Models are compared using the PR AUC metric.

Lo is a ranking problem that pertains to optimizing molecules or guiding molecular generative models. The test set consists of clusters of similar molecules that are largely dissimilar from the training set, except for one molecule representing a known hit. The task involves ranking the activity of the molecules within clusters, hence we use mean intercluster Spearman correlation to evaluate models. To ensure that the variation in intracluster activity stems from actual differences in activity rather than random noise, we selected clusters demonstrating high variation, as detailed in Appendix B and C.

The datasets each consist of three folds. We advise using the first fold for hyperparameter selection, and then applying these hyperparameters across all folds.

Datasets are released under the MIT license. Authors bear all responsibility in case of violation of rights. Datasets are small .csv files, that is why we are going to keep them in the public GitHub repository. Reviewers can find datasets in `data` folder.

In this section, we provide further information regarding the datasets and preprocessing steps. The size and diversity of the original datasets are displayed in Table 3.

Table 3: Original datasets

Dataset	Size	#Circles [69] (0.5)	Active fraction
DRD2 (Ki)	8482	837	0.731
HIV	41127	19222	0.035
KDR (IC50)	8826	791	0.643
Sol	2173	1763	0.216
KCNH2 (IC50)	11159	2128	NA

A.1 Data preprocessing

We began by canonicalizing all SMILES using RDKit 2022.9.5.

For `DRD2-Hi`, `DRD2-Lo`, `KDR-Hi`, `KDR-Lo` and `KCNH2-Lo` we utilized data from the ChEMBL30 [74] database. We collected data points that measured Ki (for DRD2) and IC50 (for KCNH2 and KDR) with `confidence_score` $\geq$ 6. We selected those for which `standard_units` were in "nM". We converted `standard_value` to logarithmic scale, also known as pChembl(`https://chembl.gitbook.io/chembl-interface-documentation/frequently-asked-questions/chembl-data-questions#what-is-pchembl`).

For binary `DRD2-Hi` and `KDR-Hi` we binarized the data such that log activity values greater than 6 (which is < 10 muM) were designated as 1, and all others as 0. We removed any ambiguous data points (e.g. with `standard_relation` of "<" and an activity value more than 10 muM, because those could not be binarized reliably). Following this, we selected data points with identical SMILES, discarding any with differing binarized activities.

For the continuous `DRD2-Lo`, `KDR-Lo` and `KCNH2-Lo` datasets, we selected data points that had `standard_relation` of '=' and a log activity value greater than 5 but less than 9. We selected data points with identical SMILES, discarding any with activity differences greater than 1.0. For the remaining data, we took the median of each group.

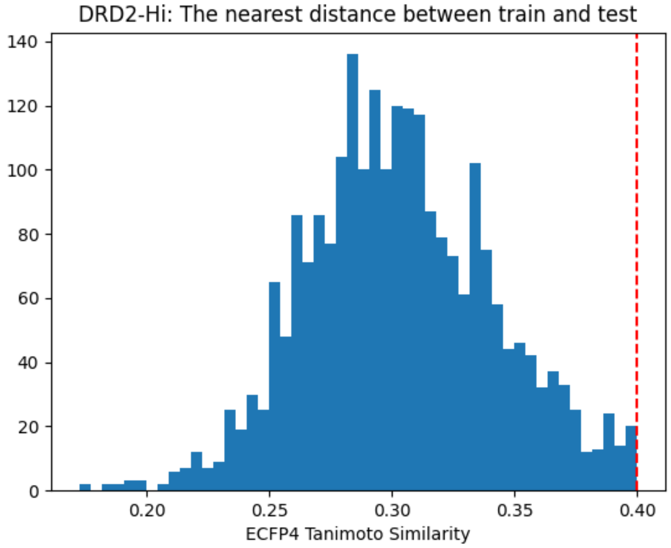


Figure 4: DRD2-Hi: Fold 1

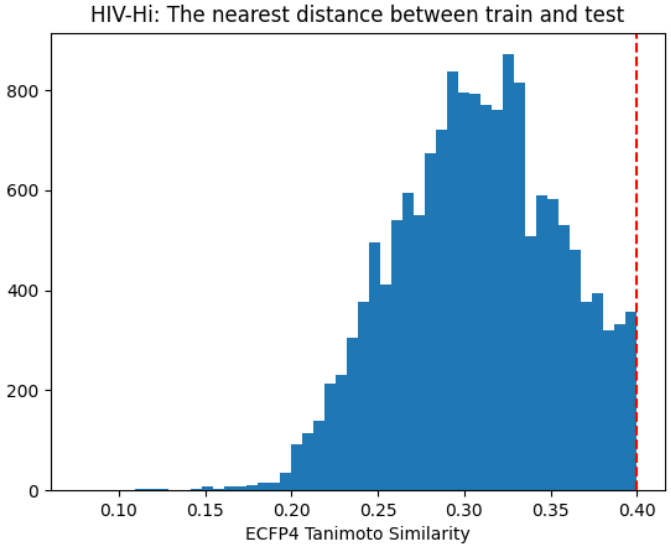


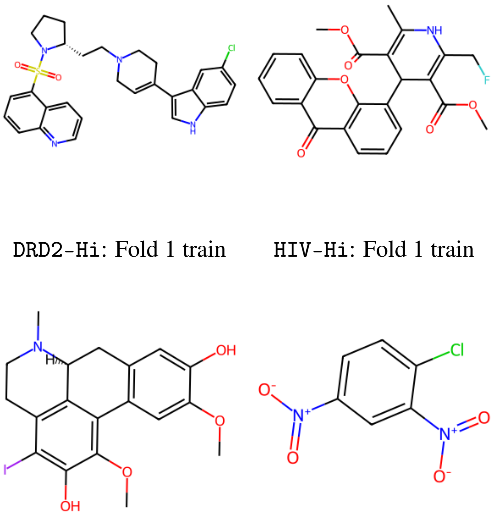
Figure 5: HIV-Hi: Fold 1

Table 4: Hi folds

Dataset	Train 1	Test 1	Train 2	Test 2	Train 3	Test 3
DRD2-Hi	2385	1190	2381	1194	2384	1191
HIV-Hi	15696	7847	15695	7848	15695	7848
KDR-Hi	500	3116	500	3125	500	2285
Sol-Hi	1442	721	1442	721	1442	721

Table 5: Lo folds

Dataset	Train 1	Test 1	Train 2	Test 2	Train 3	Test 3
DRD2-Lo	2206	267	2128	267	2257	262
KCNH2-Lo	3313	406	3313	406	3313	406
KDR-Lo	500	437	500	520	500	417



DRD2-Hi: Fold 1 train HIV-Hi: Fold 1 train

DRD2-Hi: Fold 1 test HIV-Hi: Fold 1 test

Figure 6: The most similar pairs of molecules between train and test.

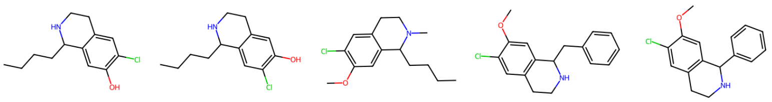


Figure 7: Example of Lo cluster in DRD2-Lo

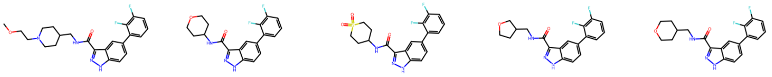


Figure 8: Example of Lo cluster in KCNH2-Lo

B Lo dataset is not just noise

Experimental data inherently contain noise. Consequently, selecting similar molecules may result in clusters that possess such a small variation that it could be solely attributable to experimental noise, thereby invalidating the Lo task. This potential issue underlines the importance of ascertaining that the clusters exhibit a significant signal to ensure the validity of the task.

As reported [87], the standard deviation for the same ligand-protein pair's pIC50 is $\sigma_{pIC50} \approx 0.20$ when measured in the same laboratory, and $\sigma_{pIC50} \approx 0.68$ in the ChEMBL database. In similar work [88] standard deviation for ChEMBL pKi was found to be $\sigma_{pKi} \approx 0.56$. Therefore, based on these findings, we opted to select only those clusters that displayed a standard deviation exceeding 0.70 for pIC50 and more than 0.60 for pKi. These selection criteria enhance the confidence in the validity of the Lo task by prioritizing clusters with significant intracluster variation.

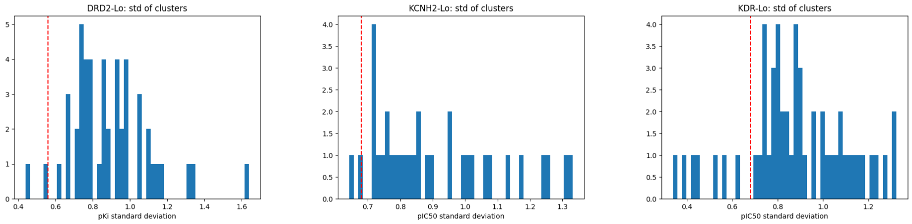


Figure 9: Within cluster variability is higher than noise standard deviation.

C Lo algorithm

The Python implementation can be found in `code/splits.py`.

Algorithm 1 Get Lo Split

Input: List of molecular SMILES S, similarity threshold t, minimum cluster size m, maximum number of clusters M, activity values V, standard deviation threshold std_t

Output: List of SMILES clusters C, list of remaining training SMILES $train\_S$

```
procedure GETLOSPLIT($S$, $t$, $m$, $M$, $V$, $std_t$)
    $C, train\_S \leftarrow$ SELECTDISTINCTCLUSTERS$(S, t, m, M, V, std_t)$
    for each $cluster$ in $C$ do
        Move central molecule from $cluster$ to $train\_S$
    end for
    return $C$, $train\_S$
end procedure
```

Algorithm 2 Select Distinct Clusters

Input: List of molecular SMILES S, similarity threshold t, minimum cluster size m, maximum number of clusters M, activity values V, standard deviation threshold std_t

Output: List of SMILES clusters C, list of the rest training SMILES $train\_S$

```
function SELECTDISTINCTCLUSTERS($S$, $t$, $m$, $M$, $V$, $std_t$)
    $train\_S \leftarrow S$
    Initialize list $C$ as empty
    while length of $C < M$ do
        Compute fingerprints $F$ from SMILES in $train\_S$
        Compute total number of neighbors $N$ for each fingerprint in $F$
        Compute $STD$ standard deviation of $V$ of neighbors for each fingerprint in $F$
        Set $central\_idx$ to None
        Set $least\_neighbors$ to max($N$)
        for each $idx$ in 0..$|train\_S|$ do          ▷ Find the smallest cluster that meets criteria
            if $N[idx] > m$ and $STD[idx] > std_t$ and $N[idx] < least\_neighbors$ then
                $central\_idx \leftarrow idx$
                $least\_neighbors \leftarrow N[idx]$
            end if
        end for
        if $central\_idx$ is None then          ▷ Exit if there are no more clusters that meet criteria
            break
        end if
        Add $central\_idx$ molecule and its neighbors to list of clusters $C$
        Remove the cluster and its neighbors from $train\_S$
    end while
    return $C$, $train\_S$
end function
```

D Additional benchmarks analysis

Distribution of Tanimoto Similarity between the nearest molecules between train and test.

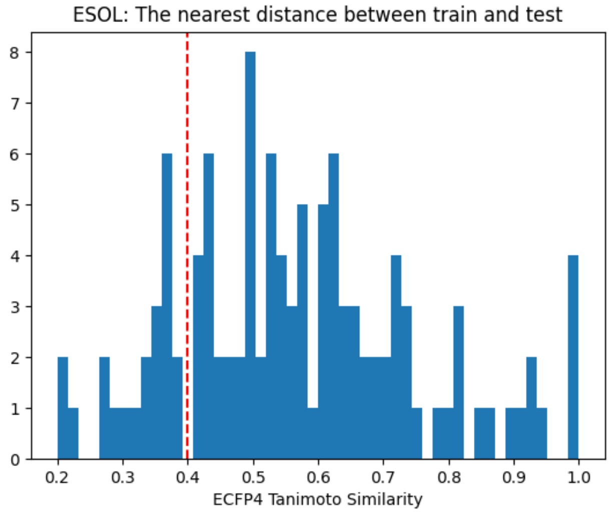


Figure 10: ESOL

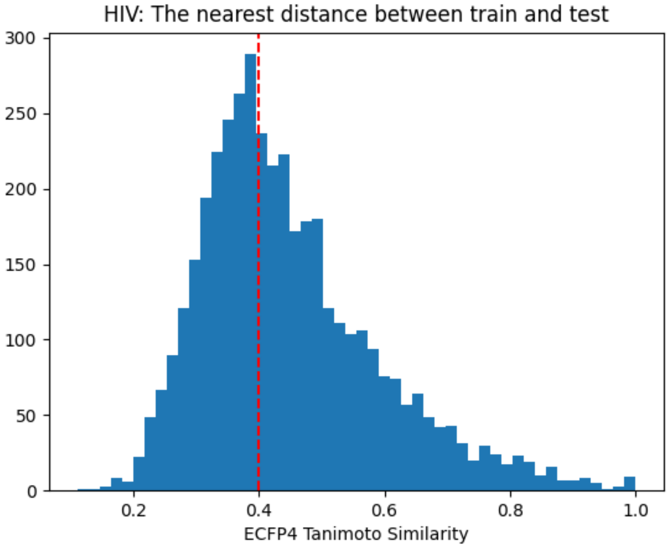


Figure 11: HIV

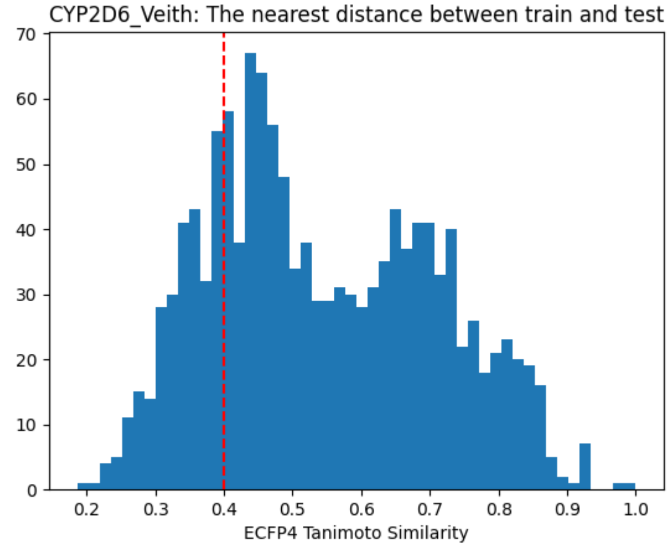


Figure 12: TDC

We additionally analyzed other ligand-based MoleculeNet datasets.

Table 6: Fraction of test molecules in various MoleculeNet datasets with a Tanimoto similarity >0.4 to the train set using ECFP4 fingerprints.

Dataset	Fraction of Test Molecules Similar to Train Set
QM7	0.93
QM8	0.98
QM9	0.99
FreeSolv	0.8
Lipophilicity	0.67
PCBA	>0.93
MUV	0.96
BACE	0.77
Tox21	0.52
SIDER	0.48

E Graph coarsening algorithm

The Python implementation can be found in `code/min_vertex_k_cut.py`. We are planning to release it as a pip package.

Algorithm 3 Calculate Neighbors

```
Input: Graph $G = (V, E)$, similarity threshold $\theta$
Output: List of tuples $n\_neighbors$
function calculateNeighbors($G$, $\theta$)
    Initialize list $n\_neighbors$ as empty
    for each node $v$ in $V$ do
        Initialize $total\_neighbors$ as 0
        for each edge $e$ incident on node $v$ do
            if $e$['similarity'] $> \theta$ then
                $total\_neighbors \leftarrow total\_neighbors + 1$
            end if
        end for
        Append $(total\_neighbors,$ index of $v)$ to $n\_neighbors$
    end for
    return $n\_neighbors$
end function
```

Algorithm 4 Cluster Nodes

```
Input: Sorted list $n\_neighbors$, Graph $G = (V, E)$, similarity threshold $\theta$
Output: Cluster assignment $node\_to\_cluster$, number of clusters $total\_clusters$
function CLUSTERNODES($n\_neighbors$, $G$, $\theta$)
    Initialize array $node\_to\_cluster$ of size $|V|$ as $-1$
    Initialize $total\_clusters$ as 1
    for each tuple $(count, node)$ in $n\_neighbors$ do
        if $node\_to\_cluster[node] = -1$ then
            $node\_to\_cluster[node] \leftarrow total\_clusters$          ▷ Assign new cluster
            for each edge $e$ incident on node $node$ do
                if $e$['similarity'] $> \theta$ then
                    $adjacent\_node \leftarrow e[1]$
                    if $node\_to\_cluster[adjacent\_node] = -1$ then
                        $node\_to\_cluster[adjacent\_node] \leftarrow total\_clusters$
                    end if
                end if
            end for
            $total\_clusters \leftarrow total\_clusters + 1$
        end if
    end for
    return $node\_to\_cluster$, $total\_clusters$
end function
```

Algorithm 5 Build Coarse Graph

```
Input: Cluster assignment $node\_to\_cluster$, number of clusters $total\_clusters$, Graph $G = (V, E)$
Output: Coarsened Graph $G_{\text{coarse}}$
function BUILDCOARSEGRAPH($node\_to\_cluster$, $total\_clusters$, $G$)
    Compute $clusters\_size$, count of each unique element in $node\_to\_cluster$
    Initialize $G_{\text{coarse}}$ as an empty graph
    for $cluster$ in 0 to $total\_clusters - 1$ do          ▷ Add nodes
        Add node $cluster$ with weight $clusters\_size[cluster]$ to $G_{\text{coarse}}$
    end for
    for $cluster$ in 0 to $total\_clusters - 1$ do          ▷ Add edges
        Initialize $connected\_clusters$ as an empty set
        Get nodes of $cluster$ as $this\_cluster\_indices$ where $node\_to\_cluster$ equals $cluster$
        for each $node$ in $this\_cluster\_indices$ do
            for each edge $e$ incident on node $node$ do
                Add $node\_to\_cluster[e[1]]$ to $connected\_clusters$
            end for
        end for
        for each $connected\_cluster$ in $connected\_clusters$ do
            Add edge from $cluster$ to $connected\_cluster$ in $G_{\text{coarse}}$
        end for
    end for
    return $G_{\text{coarse}}$
end function
```

Algorithm 6 Main Procedure

```
Input: Graph $G = (V, E)$, similarity threshold $\theta$
Output: Coarsened graph $G_{\text{coarse}}$
procedure COARSEGRAPH($G$, $\theta$)
    $n\_neighbors \leftarrow$ CALCULATENEIGHBORS$(G, \theta)$
    Sort $n\_neighbors$ in descending order of first element of each tuple
    $node\_to\_cluster, total\_clusters \leftarrow$ CLUSTERNODES$(n\_neighbors, G, \theta)$
    $G_{\text{coarse}} \leftarrow$ BUILDCOARSEGRAPH$(node\_to\_cluster, total\_clusters, G)$
    return $G_{\text{coarse}}$
end procedure
```

F Hi-split predicts virtual screening hit rate better than scaffold split

For effective virtual screening, predicting experimental outcomes prior to experimentation is paramount. In this study, we compare the predictive performance of the novel Hi-split approach with the traditional scaffold split method under a Hit Identification scenario. Following existing literature [10, 11, 16, 27, 28], we simulate testing on novel molecules with an ECFP4 Tanimoto similarity of ≤ 0.4 to the training set. The dataset is partitioned using both splitting methods to form separate training and validation sets for hyperparameter selection. Hyperparameter search is performed for gradient boosting on ECFP4 fingerprints, identified as the most efficient Hi model that facilitates quick training.

After selecting the optimal hyperparameters, performance metrics are computed on the validation set. Subsequently, the model is retrained on the combined training and validation sets, and performance metrics for the hold-out test set are calculated to simulate the application of a trained model in virtual screening. The results are summarized in Table 7.

Table 7: Hi-split vs scaffold split

Dataset	Validation	Test
`DRD2-Hi` (Hi split)	0.603	**0.677**
`DRD2-Hi` (Scaffold split)	0.872	0.663
`HIV-Hi` (Hi split)	0.069	**0.084**
`HIV-Hi` (Scaffold split)	0.189	0.078

The Hi-split method demonstrates superior predictive performance for virtual screening hit rate compared to the scaffold split method, which is over-optimistic in the Hit Identification scenario. It also improved the test evaluation metric, although the difference is not substantial. The improved performance of the Hi-split may be attributed to the selection of more regularized models.

G Novelty consensus analysis

We have reproduced the work presented in [43] using binary ECFP4 fingerprints, as calculated by RDKit version 2022.9.5. The results can be found in Figure 13. For this particular work, we selected 0.40 as the novelty threshold.

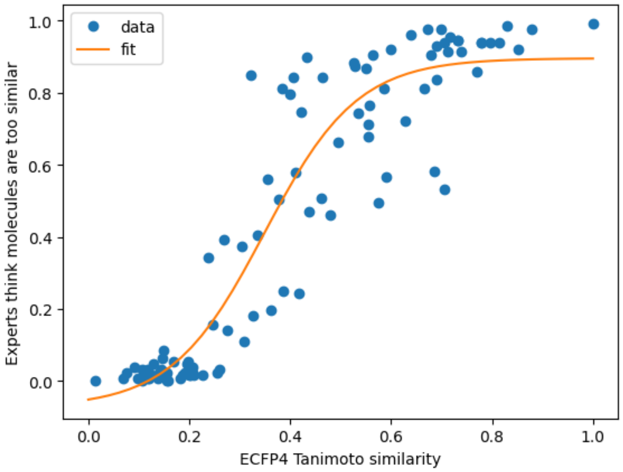


Figure 13: Sigmoid fit to [43] data

H Hyperparameter optimization

We used random or grid search to optimize hyperparameters for all models except for the Graphormer, which was too slow for meticulous hyperparameter search. Here we provide optimization parameters and additional commentary on the training.

We utilized a single NVIDIA RTX 2070 SUPER with CUDA 11.7 and calculated binary 1024 ECFP4 and MACCS fingerprints using RDKit 2022.9.5.

H.1 Dummy baseline

Always predicts the same constant value.

H.2 KNN

We used `scipy.spatial.distance.jaccard` as the distance metric, as it outperformed the standard Euclidian distance in our use case. We used grid search with all combinations of parameters. For ECFP4 it was:

```
params = {
    'n_neighbors': [3, 5, 7, 10],
    'weights': ['uniform', 'distance'],
}
```

and for MACCS:

```
params = {
    'n_neighbors': [3, 5, 7, 10, 12, 15],
    'weights': ['uniform', 'distance'],
}
```

H.3 Gradient Boosting

We used 30 iterations of random search with these parameters:

```
params = {
    'n_estimators': [10, 50, 100, 150, 200, 250, 500],
    'learning_rate': [0.01, 0.1, 0.3, 0.5, 0.7, 1.0],
    'subsample': [0.4, 0.7, 0.9, 1.0],
    'min_samples_split': [2, 3, 5, 7],
    'min_samples_leaf': [1, 3, 5],
    'max_depth': [2, 3, 4],
    'max_features': [None, 'sqrt']
}
```

H.4 SVM

We used grid search with these parameters:

```
params = {
    'C': [0.1, 0.5, 1.0, 2.0, 5.0],
}
```

H.5 MLP

We implemented a feed-forward neural network using Pytorch 2.0.0+cu117 and Pytorch Lightning 2.0.2. It consisted of several feed-forward layers with optional dropout layers. We used early stopping to prevent overfitting with patience 20 for the Hi tasks, and 10 for the Lo tasks. We used learning rate 0.01. We used batch size 32. We conducted 30 iterations of random search. For ECFP4 we used these parameters:

```
param_dict = {
    ’layers’: [
        [1024, 32, 32],
        [1024, 16, 16],
        [1024, 32],
        [1024, 8, 4],
        [1024, 4]
    ],
    ’dropout’: [0.0, 0.0, 0.2, 0.4, 0.6],
    ’l2’: [0.0, 0.0, 0.001, 0.005, 0.01],
}
```

For MACCS we used these parameters:

```
param_dict = {
    ’layers’: [
        [167, 32, 32],
        [167, 16, 16],
        [167, 32],
        [167, 8, 4],
        [167, 4]
    ],
    ’dropout’: [0.0, 0.0, 0.2, 0.4, 0.6],
    ’l2’: [0.0, 0.0, 0.001, 0.005, 0.01],
}
```

After the selection of the best hyperparameters, we selected a fixed number of the training epochs using early stopping. We used the same number of epochs for all the folds.

H.6 Chemprop

We used Chemprop 1.5.2 with rdkit features. We found the evaluation metrics to be a little better with them, but it was SOTA for Hi even without them:

```
’--features_generator rdkit_2d_normalized’,
’--no_features_scaling’,
```

We used 20 iterations of random search with these parameters:

```
param_dict = {
    ’--depth’: [’3’, ’4’, ’5’, ’6’],
    ’--dropout’: [’0.0’, ’0.2’, ’0.3’, ’0.5’, ’0.7’],
    ’--ffn_hidden_size’: [’600’, ’1200’, ’2400’, ’3600’],
    ’--ffn_num_layers’: [’1’, ’2’, ’3’],
    ’--hidden_size’: [’600’, ’1200’, ’2400’, ’3600’]
}
```

We selected the number of epochs using only the first fold. After the hyperparameters were selected, we trained the model and did not expose it to the test data. The full command for training Chemprop for `HIV-Hi` dataset:

```
chemprop_train --data_path data/hi/hiv/train_1.csv --dataset_type classification \
--save_dir checkpoints/hi/hiv/ \
--config_path configs/hiv_hi \
--separate_val_path data/hi/hiv/train_1.csv \
--separate_test_path data/hi/hiv/train_1.csv \
--metric ’prc-auc’ \
--epochs 40 \
--features_generator rdkit_2d_normalized \
--no_features_scaling
```

For the `DRD-Hi` the best hyperparameters were:

```
{
"depth": 6,
"dropout": 0.0,
"ffn_hidden_size": 2400,
"ffn_num_layers": 1,
"hidden_size": 2400
}
```

For the `HIV-Hi` the best hyperparameters were:

```
{
"depth": 6,
"dropout": 0.2,
"ffn_hidden_size": 3600,
"ffn_num_layers": 2,
"hidden_size": 3600
}
```

H.7 Graphormer

We used Graphormer with the last commit 77f436db46fb9013121289db670d1a763f264153. We applied two fixes, that we found in issues `https://github.com/microsoft/Graphormer/issues/158#issuecomment-1500311589` and `https://github.com/microsoft/Graphormer/issues/130#issuecomment-1207316808` that solved our problems. However, we set up an in-house Graphormer some time ago and currently, it cannot be reinstalled from scratch due to multiple broken dependencies.

We modified code to calculate and track PR AUC metrics, to add our datasets, and to evaluate trained models. We manually optimized the hyperparameters over approximately 10 iterations. We found Graphormer to be inferior to Chemprop, which is consistent with our previous experience with different datasets.

We faced numerous technical difficulties in executing and modifying Graphormer [80, 81] due to improper dependency pinning by the authors. We found the training to be slow, which limited our ability to optimize hyperparameters. Because of technical difficulties, we decided not to test it for the Lo task.

H.8 `HIV-Hi` balance

`HIV-Hi` is a highly unbalanced binary classification problem with only 3% of positive examples. Due to this imbalance, we experimented with weighted options of classical ML algorithms and manually resampled positive examples for neural networks.

I Spearman distribution

The test set of the Lo datasets is composed of molecular clusters. To evaluate the models, the Spearman correlation coefficient is calculated within each cluster, comparing the actual activity values to the predicted ones. The final Lo metric is the average of the Spearman coefficients across all clusters.

In the following, we present a histogram of Spearman coefficients for the best models across various datasets. Note that the `KDR-Lo` dataset is more challenging than both `DRD2-Lo` and `KCNH2-Lo`.

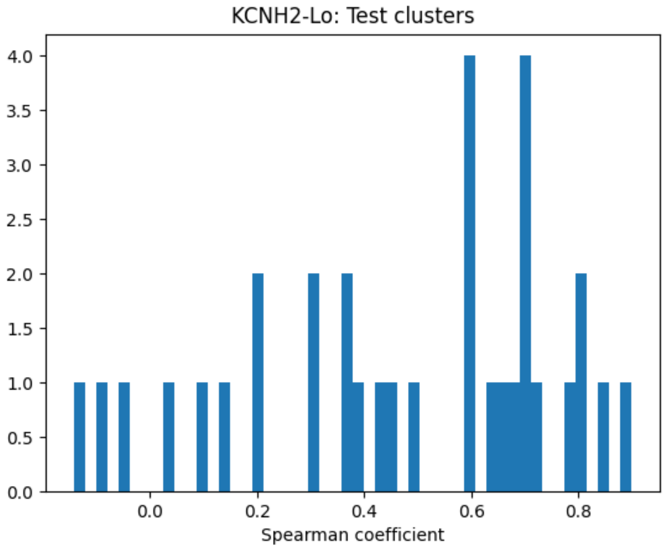


Figure 14: `KCNH2-Lo` Spearman coefficient distribution for SVM-ECFP4

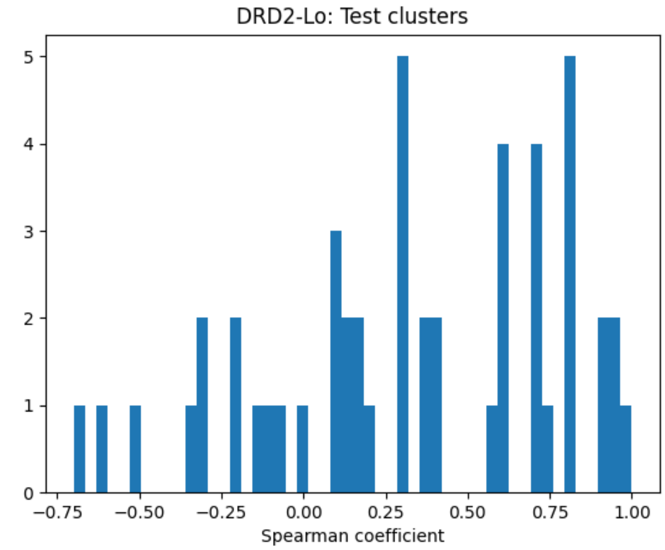


Figure 15: `DRD2-Lo` Spearman coefficient distribution for SVM-ECFP4

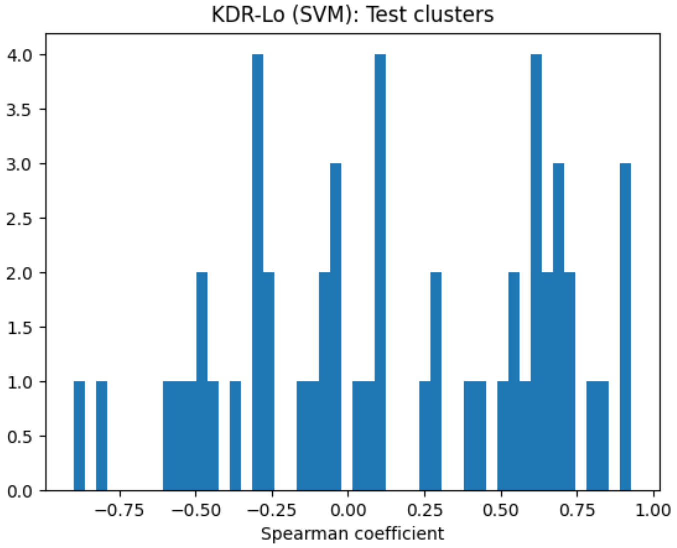


Figure 16: KDR-Lo Spearman coefficient distribution for SVM-ECFP4

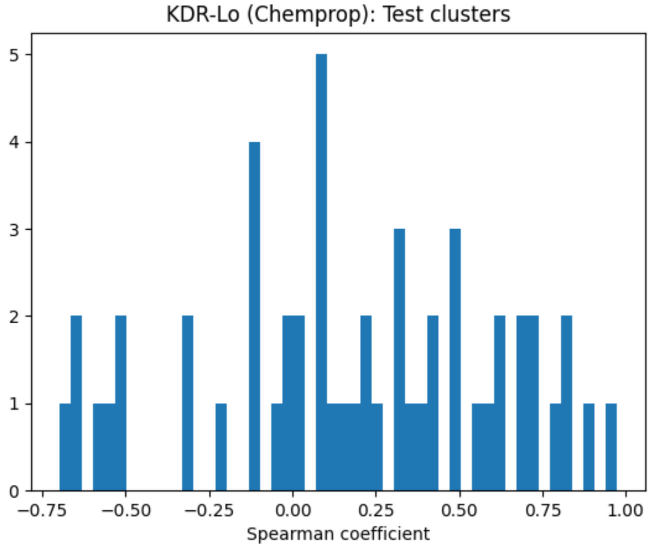


Figure 17: KDR-Lo Spearman coefficient distribution for Chemprop

References

[1] Shu-Feng Zhou and Wei-Zhu Zhong. Drug design and discovery: principles and applications, 2017.

[2] Jing Lin, Diana C Sahakian, SM De Morais, Jinghai J Xu, Robert J Polzer, and Steven M Winter. The role of absorption, distribution, metabolism, excretion and toxicity in drug discovery. *Current topics in medicinal chemistry*, 3(10):1125–1154, 2003.

[3] Sabina Podlewska and Rafał Kafel. Metstabon—online platform for metabolic stability predictions. *International journal of molecular sciences*, 19(4):1040, 2018.

[4] Mohsen Sharifi and Taravat Ghafourian. Estimation of biliary excretion of foreign compounds using properties of molecular structure. *The AAPS journal*, 16:65–78, 2014.

[5] Patrizia Crivori, Gabriele Cruciani, Pierre-Alain Carrupt, and Bernard Testa. Predicting blood-brain barrier permeation from three-dimensional molecular structure. *Journal of medicinal chemistry*, 43(11):2204–2216, 2000.

[6] Qiang Tang, Fulei Nie, Qi Zhao, and Wei Chen. A merged molecular representation deep learning method for blood–brain barrier permeability prediction. *Briefings in Bioinformatics*, 23(5), 2022.

[7] Raimund Mannhold and Han Van de Waterbeemd. Substructure and whole molecule approaches for calculating log p. *Journal of Computer-Aided Molecular Design*, 15:337–354, 2001.

[8] Ashok K Sharma, Gopal N Srivastava, Ankita Roy, and Vineet K Sharma. Toxim: a toxicity prediction tool for small molecules developed using machine learning and chemoinformatics approaches. *Frontiers in pharmacology*, 8:880, 2017.

[9] Rakesh Kanji, Abhinav Sharma, and Ganesh Bagler. Phenotypic side effects prediction by optimizing correlation with chemical and target profiles of drugs. *Molecular BioSystems*, 11 (11):2900–2906, 2015.

[10] Brian J Bender, Stefan Gahbauer, Andreas Luttens, Jiankun Lyu, Chase M Webb, Reed M Stein, Elissa A Fink, Trent E Balius, Jens Carlsson, John J Irwin, et al. A practical guide to large-scale docking. *Nature protocols*, 16(10):4799–4832, 2021.

[11] Reed M Stein, Hye Jin Kang, John D McCorvy, Grant C Glatfelter, Anthony J Jones, Tao Che, Samuel Slocum, Xi-Ping Huang, Olena Savych, Yurii S Moroz, et al. Virtual discovery of melatonin receptor ligands to modulate circadian rhythms. *Nature*, 579(7800):609–614, 2020.

[12] Ralf Mueller, Alice L Rodriguez, Eric S Dawson, Mariusz Butkiewicz, Thuy T Nguyen, Stephen Oleszkiewicz, Annalen Bleckmann, C David Weaver, Craig W Lindsley, P Jeffrey Conn, et al. Identification of metabotropic glutamate receptor subtype 5 potentiators using virtual high-throughput screening. *ACS chemical neuroscience*, 1(4):288–305, 2010.

[13] Liying Zhang, Denis Fourches, Alexander Sedykh, Hao Zhu, Alexander Golbraikh, Sean Ekins, Julie Clark, Michele C Connelly, Martina Sigal, Dena Hodges, et al. Discovery of novel antimalarial compounds enabled by qsar-based virtual screening. *Journal of chemical information and modeling*, 53(2):475–492, 2013.

[14] Bruno J Neves, Rafael F Dantas, Mario R Senger, Cleber C Melo-Filho, Walter CG Valente, Ana CM de Almeida, João M Rezende-Neto, Elid FC Lima, Ross Paveley, Nicholas Furnham, et al. Discovery of new anti-schistosomal hits by integration of qsar-based virtual screening and high content screening. *Journal of medicinal chemistry*, 59(15):7075–7088, 2016.

[15] Georges E Janssens, Xin-Xuan Lin, Lluís Millan-Ariño, Alan Kavšek, Ilke Sen, Renée I Seinstra, Nicholas Stroustrup, Ellen AA Nollen, and Christian G Riedel. Transcriptomics-based screening identifies pharmacological inhibition of hsp90 as a means to defer aging. *Cell Reports*, 27(2): 467–480, 2019.

[16] Jonathan M Stokes, Kevin Yang, Kyle Swanson, Wengong Jin, Andres Cubillos-Ruiz, Nina M Donghia, Craig R MacNair, Shawn French, Lindsey A Carfrae, Zohar Bloom-Ackermann, et al. A deep learning approach to antibiotic discovery. *Cell*, 180(4):688–702, 2020.

[17] Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 9(2):513–530, 2018.

[18] Kexin Huang, Tianfan Fu, Wenhao Gao, Yue Zhao, Yusuf Roohani, Jure Leskovec, Connor W Coley, Cao Xiao, Jimeng Sun, and Marinka Zitnik. Therapeutics data commons: Machine learning datasets and tasks for drug discovery and development. *arXiv preprint arXiv:2102.09548*, 2021.

[19] Arash Keshavarzi Arshadi, Milad Salem, Arash Firouzbakht, and Jiann Shiun Yuan. Moldata, a molecular benchmark for disease and target based machine learning. *Journal of Cheminformatics*, 14(1):1–18, 2022.

[20] Yuanfeng Ji, Lu Zhang, Jiaxiang Wu, Bingzhe Wu, Long-Kai Huang, Tingyang Xu, Yu Rong, Lanqing Li, Jie Ren, Ding Xue, et al. Drugood: Out-of-distribution (ood) dataset curator and benchmark for ai-aided drug discovery–a focus on affinity prediction problems with noise annotations. *arXiv preprint arXiv:2201.09637*, 2022.

[21] Shurui Gui, Xiner Li, Limei Wang, and Shuiwang Ji. Good: A graph out-of-distribution benchmark. *arXiv preprint arXiv:2206.08452*, 2022.

[22] Ziqiao Zhang, Bangyi Zhao, Ailin Xie, Yatao Bian, and Shuigeng Zhou. Activity cliff prediction: Dataset and benchmark. *arXiv preprint arXiv:2302.07541*, 2023.

[23] Derek van Tilborg, Alisa Alenicheva, and Francesca Grisoni. Exposing the limitations of molecular machine learning with activity cliffs. *Journal of Chemical Information and Modeling*, 62(23):5938–5951, 2022.

[24] György M Keserű and Gergely M Makara. Hit discovery and hit-to-lead approaches. *Drug discovery today*, 11(15-16):741–748, 2006.

[25] Asher Mullard. How much do phase iii trials cost? *Nature Reviews. Drug Discovery*, 17(11): 777–777, 2018.

[26] Linda Martin, Melissa Hutchens, Conrad Hawkins, and Alaina Radnov. How much do clinical trials cost. *Nat Rev Drug Discov*, 16(6):381–382, 2017.

[27] Jiankun Lyu, Sheng Wang, Trent E Balius, Isha Singh, Anat Levit, Yurii S Moroz, Matthew J O'Meara, Tao Che, Enkhjargal Algaa, Kateryna Tolmachova, et al. Ultra-large library docking for discovering new chemotypes. *Nature*, 566(7743):224–229, 2019.

[28] Felix Wong, Satotaka Omori, Nina M Donghia, Erica J Zheng, and James J Collins. Discovering small-molecule senolytics with deep neural networks. *Nature Aging*, pages 1–17, 2023.

[29] Gary Liu, Denise B Catacutan, Khushi Rathod, Kyle Swanson, Wengong Jin, Jody C Mohammed, Anush Chiappino-Pepe, Saad A Syed, Meghan Fragis, Kenneth Rachwalski, et al. Deep learning-guided discovery of an antibiotic targeting acinetobacter baumannii. *Nature Chemical Biology*, pages 1–9, 2023.

[30] RD Brown and YC Martin. An evaluation of structural descriptors and clustering methods for use in diversity selection. *SAR and QSAR in Environmental Research*, 8(1-2):23–39, 1998.

[31] Bie Verbist, Günter Klambauer, Liesbet Vervoort, Willem Talloen, Ziv Shkedy, Olivier Thas, Andreas Bender, Hinrich WH Göhlmann, Sepp Hochreiter, QSTAR Consortium, et al. Using transcriptomics to guide lead optimization in drug discovery projects: Lessons learned from the qstar project. *Drug discovery today*, 20(5):505–513, 2015.

[32] George Papadatos, Anthony WJ Cooper, Visakan Kadirkamanathan, Simon JF Macdonald, Iain M McLay, Stephen D Pickett, John M Pritchard, Peter Willett, and Valerie J Gillet. Analysis of neighborhood behavior in lead optimization and array design. *Journal of chemical information and modeling*, 49(2):195–208, 2009.

[33] Nathan Brown, Marco Fiscato, Marwin HS Segler, and Alain C Vaucher. Guacamol: benchmarking models for de novo molecular design. *Journal of chemical information and modeling*, 59(3):1096–1108, 2019.

[34] Tianfan Fu, Cao Xiao, Xinhao Li, Lucas M Glass, and Jimeng Sun. Mimosa: Multi-constraint molecule sampling for molecule optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 125–133, 2021.

[35] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. In *International conference on machine learning*, pages 2323–2332. PMLR, 2018.

[36] Maxime Langevin, Hervé Minoux, Maximilien Levesque, and Marc Bianciotto. Scaffold-constrained molecular generation. *Journal of Chemical Information and Modeling*, 60(12): 5637–5646, 2020.

[37] William J Godinez, Eric J Ma, Alexander T Chao, Luying Pei, Peter Skewes-Cox, Stephen M Canham, Jeremy L Jenkins, Joseph M Young, Eric J Martin, and W Armand Guiguemde. Design of potent antimalarials with generative chemistry. *Nature Machine Intelligence*, 4(2):180–186, 2022.

[38] Wenhao Gao, Tianfan Fu, Jimeng Sun, and Connor Coley. Sample efficiency matters: a benchmark for practical molecular optimization. *Advances in Neural Information Processing Systems*, 35:21342–21357, 2022.

[39] David E Patterson, Richard D Cramer, Allan M Ferguson, Robert D Clark, and Laurence E Weinberger. Neighborhood behavior: a useful concept for validation of "molecular diversity" descriptors. *Journal of medicinal chemistry*, 39(16):3049–3059, 1996.

[40] Yvonne C Martin, James L Kofron, and Linda M Traphagen. Do structurally similar molecules have similar biological activity? *Journal of medicinal chemistry*, 45(19):4350–4358, 2002.

[41] Gerald Maggiora, Martin Vogt, Dagmar Stumpfe, and Jurgen Bajorath. Molecular similarity in medicinal chemistry: miniperspective. *Journal of medicinal chemistry*, 57(8):3186–3204, 2014.

[42] Alex Zhavoronkov and Alán Aspuru-Guzik. Reply to 'assessing the impact of generative ai on medicinal chemistry'. *Nature Biotechnology*, 38(2):146–146, 2020.

[43] Pedro Franco, Nuria Porta, John D Holliday, and Peter Willett. The use of 2d fingerprint methods to support the assessment of structural similarity in orphan drug legislation. *Journal of cheminformatics*, 6:1–10, 2014.

[44] David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010.

[45] John D Holliday, Naomie Salim, Martin Whittle, and Peter Willett. Analysis and display of the size dependence of chemical similarity coefficients. *Journal of chemical information and computer sciences*, 43(3):819–828, 2003.

[46] Wei Lu, Qifeng Wu, Jixian Zhang, Jiahua Rao, Chengtao Li, and Shuangjia Zheng. Tankbind: Trigonometry-aware neural networks for drug-protein binding structure prediction. *bioRxiv*, pages 2022–06, 2022.

[47] Seul Lee, Jaehyeong Jo, and Sung Ju Hwang. Exploring chemical space with score-based out-of-distribution generation. In *International Conference on Machine Learning*, pages 18872–18892. PMLR, 2023.

[48] Hyeoncheol Cho and Insung S Choi. Enhanced deep-learning prediction of molecular properties via augmentation of bond topology. *ChemMedChem*, 14(17):1604–1609, 2019.

[49] Yoonho Jeong, Jihoo Kim, Yeji Kim, and Insung S Choi. Development of a chemically intuitive filter for chemical graph convolutional network. *Bulletin of the Korean Chemical Society*, 43(7): 934–936, 2022.

[50] Benson Chen, Regina Barzilay, and Tommi Jaakkola. Path-augmented graph transformer network. *arXiv preprint arXiv:1905.12712*, 2019.

[51] Prateeth Nayak, Andrew Silberfarb, Ran Chen, Tulay Muezzinoglu, and John Byrnes. Transformer based molecule encoding for property prediction. *arXiv preprint arXiv:2011.03518*, 2020.

[52] Ross Irwin, Spyridon Dimitriadis, Jiazhen He, and Esben Jannik Bjerrum. Chemformer: a pre-trained transformer for computational chemistry. *Machine Learning: Science and Technology*, 3(1):015022, 2022.

[53] Yao Zhang et al. Bayesian semi-supervised learning for uncertainty-calibrated prediction of molecular properties and active learning. *Chemical science*, 10(35):8154–8163, 2019.

[54] Guy W Bemis and Mark A Murcko. The properties of known drugs. 1. molecular frameworks. *Journal of medicinal chemistry*, 39(15):2887–2893, 1996.

[55] Bihter Das, Mucahit Kutsal, and Resul Das. Effective prediction of drug–target interaction on hiv using deep graph neural networks. *Chemometrics and Intelligent Laboratory Systems*, 230: 104676, 2022.

[56] Garrett B Goh, Charles M Siegel, Abhinav Vishnu, and Nathan O Hodas. Chemnet: A transferable and generalizable deep neural network for small-molecule property prediction. Technical report, Pacific Northwest National Lab.(PNNL), Richland, WA (United States), 2017.

[57] Shion Honda, Shoi Shi, and Hiroki R Ueda. Smiles transformer: Pre-trained molecular fingerprint for low data drug discovery. *arXiv preprint arXiv:1911.04738*, 2019.

[58] Chao Shang, Qinqing Liu, Ko-Shin Chen, Jiangwen Sun, Jin Lu, Jinfeng Yi, and Jinbo Bi. Edge attention-based multi-relational graph convolutional networks. *arXiv preprint arXiv: 1802.04944*, 2018.

[59] Karim Abbasi, Antti Poso, Jahanbakhsh Ghasemi, Massoud Amanlou, and Ali Masoudi-Nejad. Deep transferable compound representation across domains and tasks for low data drug discovery. *Journal of chemical information and modeling*, 59(11):4528–4539, 2019.

[60] Dilyana Dimova and Jürgen Bajorath. Advances in activity cliff research. *Molecular informatics*, 35(5):181–191, 2016.

[61] Denis Cornaz, Fabio Furini, Mathieu Lacroix, Enrico Malaguti, A Ridha Mahjoub, and Sébastien Martin. Mathematical formulations for the balanced vertex k-separator problem. In *2014 International Conference on Control, Decision and Information Technologies (CoDIT)*, pages 176–181. IEEE, 2014.

[62] Egon Balas and Cid C de Souza. The vertex separator problem: a polyhedral investigation. *Mathematical Programming*, 103(3):583–608, 2005.

[63] Stephan Schwartz. An overview of graph covering and partitioning. *Discrete Mathematics*, 345 (8):112884, 2022.

[64] Denis Cornaz, Fabio Furini, Mathieu Lacroix, Enrico Malaguti, A Ridha Mahjoub, and Sébastien Martin. The vertex k-cut problem. *Discrete Optimization*, 31:8–28, 2019.

[65] Paolo Paronuzzi. Models and algorithms for decomposition problems. 2020.

[66] Fabio Furini, Ivana Ljubić, Enrico Malaguti, and Paolo Paronuzzi. On integer and bilevel formulations for the k-vertex cut problem. *Mathematical Programming Computation*, 12: 133–164, 2020.

[67] Yangming Zhou, Gezi Wang, and MengChu Zhou. Detecting k-vertex cuts in sparse networks via a fast local search approach. *IEEE Transactions on Computational Social Systems*, 2023.

[68] Darko Butina. Unsupervised data base clustering based on daylight's fingerprint and tanimoto similarity: A fast and automated way to cluster small and large data sets. *Journal of Chemical Information and Computer Sciences*, 39(4):747–750, 1999.

[69] Yutong Xie, Ziqiao Xu, Jiaqi Ma, and Qiaozhu Mei. How much space has been explored? measuring the chemical space covered by databases and machine-generated molecules. In *The Eleventh International Conference on Learning Representations*.

[70] Thelma Beatriz González-Castro, Yazmin Hernandez-Diaz, Isela Esther Juárez-Rojop, María Lilia López-Narváez, Carlos Alfonso Tovilla-Zárate, Alma Genis-Mendoza, and Mariela Alpuin-Reyes. The role of c957t, taqi and ser311cys polymorphisms of the drd2 gene in schizophrenia: systematic review and meta-analysis. *Behavioral and Brain Functions*, 12(1): 1–14, 2016.

[71] Ritushree Kukreti, Sudipta Tripathi, Pallav Bhatnagar, Simone Gupta, Chitra Chauhan, Shobhana Kubendran, YC Janardhan Reddy, Sanjeev Jain, and Samir K Brahmachari. Association of drd2 gene variant with schizophrenia. *Neuroscience letters*, 392(1-2):68–71, 2006.

[72] V McGuire, SK Van Den Eeden, CM Tanner, F Kamel, DM Umbach, K Marder, R Mayeux, B Ritz, GW Ross, H Petrovitch, et al. Association of drd2 and drd3 polymorphisms with parkinson's disease in a multiethnic consortium. *Journal of the neurological sciences*, 307(1-2): 22–29, 2011.

[73] Dongjun Dai, Yunliang Wang, Lingyan Wang, Jinfeng Li, Qingqing Ma, Jianmin Tao, Xingyu Zhou, Hanlin Zhou, Yi Jiang, Guanghui Pan, et al. Polymorphisms of drd2 and drd3 genes and parkinson's disease: A meta-analysis. *Biomedical reports*, 2(2):275–281, 2014.

[74] David Mendez, Anna Gaulton, A Patrícia Bento, Jon Chambers, Marleen De Veij, Eloy Félix, María Paula Magariños, Juan F Mosquera, Prudence Mutowo, Michał Nowotka, et al. Chembl: towards direct deposition of bioassay data. *Nucleic acids research*, 47(D1):D930–D940, 2019.

[75] Siddharth J Modi and Vithal M Kulkarni. Vascular endothelial growth factor receptor (vegfr-2)/kdr inhibitors: medicinal chemistry perspective. *Medicine in Drug Discovery*, 2:100009, 2019.

[76] Cheng Fang, Ye Wang, Richard Grater, Sudarshan Kapadnis, Cheryl Black, Patrick Trapa, and Simone Sciabola. Prospective validation of machine learning algorithms for absorption, distribution, metabolism, and excretion prediction: An industrial perspective. *Journal of Chemical Information and Modeling*, 2023.

[77] Takaya Saito and Marc Rehmsmeier. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PloS one*, 10(3):e0118432, 2015.

[78] Kelly Rae Chi. Revolution dawning in cardiotoxicity testing. *Nature reviews. Drug discovery*, 12(8):565, 2013.

[79] Kevin Yang, Kyle Swanson, Wengong Jin, Connor Coley, Philipp Eiden, Hua Gao, Angel Guzman-Perez, Timothy Hopper, Brian Kelley, Miriam Mathea, et al. Analyzing learned molecular representations for property prediction. *Journal of chemical information and modeling*, 59 (8):3370–3388, 2019.

[80] Yu Shi, Shuxin Zheng, Guolin Ke, Yifei Shen, Jiacheng You, Jiyan He, Shengjie Luo, Chang Liu, Di He, and Tie-Yan Liu. Benchmarking graphormer on large-scale molecular modeling datasets. *arXiv preprint arXiv:2203.04810*, 2022. URL `https://arxiv.org/abs/2203.04810`.

[81] Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. Do transformers really perform badly for graph representation? In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021. URL `https://openreview.net/forum?id=OeWooOxFwDa`.

[82] Joelle Pineau, Philippe Vincent-Lamarre, Koustuv Sinha, Vincent Larivière, Alina Beygelzimer, Florence d'Alché Buc, Emily Fox, and Hugo Larochelle. Improving reproducibility in machine learning research (a report from the neurips 2019 reproducibility program). *The Journal of Machine Learning Research*, 22(1):7459–7478, 2021.

[83] Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are gans created equal? a large-scale study. *Advances in neural information processing systems*, 31, 2018.

[84] Gábor Melis, Chris Dyer, and Phil Blunsom. On the state of the art of evaluation in neural language models. *arXiv preprint arXiv:1707.05589*, 2017.

[85] Megan Stanley, John F Bronskill, Krzysztof Maziarz, Hubert Misztela, Jessica Lanini, Marwin Segler, Nadine Schneider, and Marc Brockschmidt. Fs-mol: A few-shot learning dataset of molecules. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

[86] Zaixi Zhang and Qi Liu. Learning subpocket prototypes for generalizable structure-based drug design. *arXiv preprint arXiv:2305.13997*, 2023.

[87] Tuomo Kalliokoski, Christian Kramer, Anna Vulpetti, and Peter Gedeck. Comparability of mixed ic50 data–a statistical analysis. *PloS one*, 8(4):e61007, 2013.

[88] Christian Kramer, Tuomo Kalliokoski, Peter Gedeck, and Anna Vulpetti. The experimental uncertainty of heterogeneous public k i data. *Journal of medicinal chemistry*, 55(11):5165–5173, 2012.