REVEALING DATA LEAKAGE IN PROTEIN INTERACTION BENCHMARKS

Anton Bushuiev[1*] **Roman Bushuiev**[1,4*] **Jiri Sedlar**[1] **Tomas Pluskal**[4]
Jiri Damborsky[2,3] **Stanislav Mazurenko**[2,3] **Josef Sivic**[1]
[1]Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University
[2]Loschmidt Laboratories, Department of Experimental Biology and RECETOX, Masaryk University
[3]International Clinical Research Center, St. Anne's University Hospital Brno
[4]Institute of Organic Chemistry and Biochemistry of the Czech Academy of Sciences

ABSTRACT

In recent years, there has been remarkable progress in machine learning for protein–protein interactions. However, prior work has predominantly focused on improving learning algorithms, with less attention paid to evaluation strategies and data preparation. Here, we demonstrate that further development of machine learning methods may be hindered by the quality of existing train-test splits. Specifically, we find that commonly used splitting strategies for protein complexes, based on protein sequence or metadata similarity, introduce major data leakage. This may result in overoptimistic evaluation of generalization, as well as unfair benchmarking of the models, biased towards assessing their overfitting capacity rather than practical utility. To overcome the data leakage, we recommend constructing data splits based on 3D structural similarity of protein–protein interfaces and suggest corresponding algorithms. We believe that addressing the data leakage problem is critical for further progress in this research area.

1 INTRODUCTION

Proteins rarely act alone (Berggård et al., 2007), with *protein–protein interactions* (PPIs) playing a critical role in various biological processes, such as cell signaling, metabolism, and gene regulation (Alberts et al., 2015). These interactions occur when two or more proteins bind together, forming a *protein complex*, and can be either transient or persistent. Over the past decade, there has been a significant advancement in machine learning approaches to better understand protein–protein interactions and to guide wet-lab experiments (Geng et al., 2019), with tremendous progress in recent years (Rogers et al., 2023).

The vast majority of protein–protein interactions remain undiscovered. For instance, in the well-studied model organism *Escherichia coli*, only about half of the estimated interactions have been experimentally identified, and a mere 9% have known structures (Green et al., 2021). Therefore, biological experts are interested in using machine learning models to make predictions about their PPI case studies of interest (Bennett et al., 2023). It means that in a practical setting, a machine learning model is expected to generalize to protein interactions most likely not observed in the training data. Therefore, testing machine learning models on protein complexes distinct from training ones is crucial for practically-relevant evaluation.

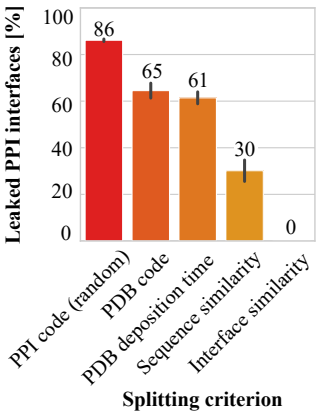

Figure 1: **Data leakage in protein–protein interaction splits.** Bars show the average percentage of test examples having a nearly identical training example for $90\%/10\%$ splits of 50,000 protein–protein interactions from the Protein Data Bank, with standard deviations (error bars) across 5 random samples. Near duplicates are identified using the iDist algorithm.

[*]These authors contributed equally.

In this work, we reveal that typical data splits of protein complexes do not satisfy this condition, suffering from significant *data leakage*, i.e., the models are tested on interactions that are nearly identical to those used during training. Specifically, we find that the conventional splitting techniques, which are often based on metadata and sequence similarity, are not sufficient for creating benchmarks to effectively assess generalization beyond training data (Figure 1; Section 3). We further discuss several recent successful examples of data splits (Section 4) and propose recommendations for splitting PPIs based on *interface* similarity, directly comparing the 3D structures of interacting regions (Section 5).

2 Related work

The problem of data leakage has been recently identified and rectified in some other domains such as protein–ligand interactions (Buttenschoen et al., 2024; Li et al., 2023) or predicting protein properties (Cheng et al., 2023; Li et al., 2020). Nevertheless, a corresponding analysis for protein interactions is missing. Next, we review the main relevant protein–protein interaction problems.

Protein interface prediction. To understand the interactions between two protein structures, machine learning can be employed to predict the pairs of interacting amino acids from each protein. This task, known as protein interface prediction (PIP), has been addressed by various machine learning methods (Morehead et al., 2023; Williams et al., 2023; Du et al., 2023; Morehead et al., 2022; Townshend et al., 2019; Fout et al., 2017). In this paper, we argue that further advancements in this area might be hindered by the low quality of sequence-based data splits, which are commonly used as benchmarks.

Protein docking. Another challenge is to directly predict the 3D structure of a complex formed by two interacting protein structures. Following initial promising results by Ganea et al. (2021), a range of machine learning algorithms have been developed for this problem (Yu et al., 2024; Sverrisson et al., 2023; Du et al., 2023; Chu et al., 2023; Ketata et al., 2023; Williams et al., 2023). We find that most research in this area uses leaking benchmarks based on sequence similarity, inherited from the domain of protein interface prediction. To the best of our knowledge, only a single recent study (Sverrisson et al., 2023) rethinks data splits to focus on the similarity of protein interface structures.

Protein binder design. Once the structure of a protein complex is known, machine learning can be used to design the binding interface, specifically to predict amino acid mutations that enhance binding strength. This task has been tackled by machine learning for over a decade (Geng et al., 2019). Initial research used completely random splits of mutations across PPIs, leading to overestimated performance (Tsishyn et al., 2024). Here, we show that most recent studies (Liu et al., 2023; Luo et al., 2023; Yue et al., 2023; Shan et al., 2022; Liu et al., 2021) still typically employ benchmarks with high data leakage, based on PDB code splitting. In contrast, Bushuiev et al. (2024) utilize data splits based on interface similarity. In this workshop paper, we use the iDist method from Bushuiev et al. (2024) for large-scale PPI comparison, applying it to comprehensively assess data leakage across various tasks.

Other tasks. Several other tasks related to protein–protein interactions are beyond the scope of this study, particularly those that do not require the structures of protein complexes as inputs. These tasks include, for example, protein complex folding (Evans et al., 2021) and problems related to protein–protein networks (Bernett et al., 2023; Tang et al., 2023).

3 Problems of existing data splits for protein complexes

In this section, we examine the issue of data leakage prevalent in most of the recently utilized data splits of protein complexes. To quantify the leakage, we calculate the ratio of test PPIs that have structural near duplicates in the training set. For detecting near duplicates, we use the iDist algorithm (Bushuiev et al., 2024), which enables large-scale structural similarity search for protein–protein interfaces, i.e., directly comparing interacting regions in 3D.

The iDist algorithm efficiently approximates iAlign (Gao & Skolnick, 2010b) – the adaptation of traditional TM-align to protein interactions (Zhang & Skolnick, 2005). iDist performs distance-

weighted message passing across interface amino acids, aggregating their patterns into real-valued vectors. Each PPI is then represented by a single vector aggregating its structural features over the whole interface. Near-duplicate interfaces are then identified based on thresholding Euclidean distance between their representing vectors. As shown in Bushuiev et al. (2024), this efficient, alignment-free approximation of the traditional iAlign structural alignment is also accurate. iDist identifies near duplicates found by iAlign with 99% precision and 97% recall.

Next, using iDist we show that splitting PPIs based on metadata (Section 3.1) or sequence similarity (Section 3.2) is insufficient for effectively measuring generalization beyond training interactions. Specifically, we find that these proxy similarity measures result in high ratio of leaked PPIs, as directly measured by the structural similarity of interacting interface regions.

3.1 SPLITTING BY METADATA IS NOT ENOUGH

The Protein Data Bank (PDB) is a central resource for protein structures (Berman et al., 2000). With over 200,000 entries, each containing an arbitrary number of interacting proteins, it is common to compare these entries based on their metadata.

The basic data splitting strategy is to split PPIs randomly, ensuring that the same interactions from the same entries do not appear more than once across data folds (Luo et al., 2023; Shan et al., 2022; Liu et al., 2021). Technically, this is achieved by splitting PPI-specific identifiers (e.g., `3BTD_I_E` for an interaction between protein chains `E` and `I` from the `3BTD` entry in PDB, see Figure 2). A more cautious method involves splitting PPIs based on the PDB entry codes from which they originate (e.g., `3BTD`). This ensures that same interactions from a single symmetric complex (e.g., three nearly identical PPIs from `1UIJ` in Figure 4) remain within the same data fold (Morehead et al., 2022). Another alternative is to divide the entries according to their submission date to the PDB, utilizing the most recent entries for testing. Although, to the best of our knowledge, this method has not been applied to PPI data, it is a frequent practice in other domains working with PDB data (Corso et al., 2022), often combined with other similarity criteria (Evans et al., 2021; Jumper et al., 2021).

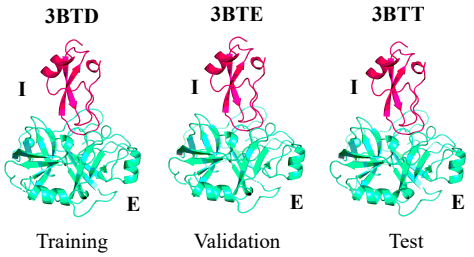

Figure 2: **Splitting by PDB codes causes data leakage in benchmarks for PPI design.** The figure shows three protein complexes taken from SKEMPI v2.0, a standard dataset of annotated PPI mutations. Different chains in the entries are color-coded and labeled with their respective codes. In total, the dataset contains 10 such near-duplicate interactions (PDB codes `3BTD`, `3BTE`, `3BTT`, `3BTM`, `3BTQ`, `3BTW`, `3BTH`, `3BTF`, `3BTG`, `2FTL`), representing single-point mutants of the same interaction between a serine protease and its inhibitor (Krowarsch et al., 1999). Recent machine learning research in protein–protein interactions employed PDB-code splitting, resulting in near-duplicate entries, similar to those shown in this figure, scattered across train-validation-test folds.

Here, we demonstrate that each of the three aforementioned methods alone leads to significant data leakage due to the extensive redundancy within the Protein Data Bank (Burra et al., 2009), as well as the extreme repetitiveness of protein–protein interfaces (Figure 4; Gao & Skolnick (2010b); Garma et al. (2012)). To assess the data leakage statistically, we first create five random samples of PPIs from PDB, each containing roughly 50,000 interactions. Then, we split each sample according to the aforementioned practices in the $90\%/10\%$ ratio to simulate the train-test splits. Subsequently, we quantify the number of test entries with at least one near-duplicate in the training set, as detected by the iDist algorithm. The results are shown in Figure 1.

We find that splits based on PPI codes, on average, lead to 86% data leakage, which is expected due to the high redundancy in PDB. Splits based on PDB codes improve the situation, yet still lead to 65% data leakage. Further refinement of the splits considering the deposition time of the entries leads to a slight improvement, reducing leaks to 61%. For more details on the experiment, please refer to Appendix A. In summary, all examined metadata-based splits result in a majority of test PPIs being leaked from the training data, indicating that these naive approaches should not be utilized in practical scenarios.

Finally, we validate our findings by examining data leakage in a recent split (Luo et al., 2023) of SKEMPI v2.0 (Jankauskaitė et al., 2019), a standard benchmark dataset in the field of protein binder design. This approach divides all 343 PPIs derived from the PDB into three sets for three-fold cross-validation and testing based on PPI codes (e.g., 3BTD_I_E; see Figure 2). Using iDist, we identify test PPIs that are similar to those in the training or validation sets in all three evaluation scenarios. On average, iDist detects 56% (57%, 57%, and 54%) of the test PPIs as leaked from the training data (i.e., with a near-duplicate interface structure in the training data), despite the small size of the dataset.

3.2 SPLITTING BY SEQUENCE SIMILARITY IS NOT ENOUGH

Compared to the field of protein structure, the area of learning from protein sequences is more established, offering advanced tools for sequence comparison and splitting without data leakage (Suzek et al., 2007). Specifically, the MMseqs2 algorithm (Steinegger & Söding, 2018; 2017) enables ultra-fast large-scale search for similar proteins, and led to the creation of the clustered UniRef database (The UniProt Consortium, 2023).

Drawing inspiration from this, the machine learning community has employed sequence similarity methods to split protein complexes, ensuring that the protein sequences in test complexes do not have close homologs in the training data (Williams et al., 2023; Du et al., 2023; Dauparas et al., 2022; Morehead et al., 2022; Townshend et al., 2019). Here, we show that even sequence-based splitting of protein complexes can result in a substantial rate of leaked test interactions, i.e., interactions in the test set that have *structurally* near-duplicate interactions in the training data, as measured by interface similarity.

Previously, it was demonstrated that the DIPS dataset (Townshend et al., 2019), when split by protein families to separate similar proteins (Ganea et al., 2021), suffers from approximately 53% structural data leakage (Bushuiev et al., 2024). Figure 3 visualizes an example of the leak.

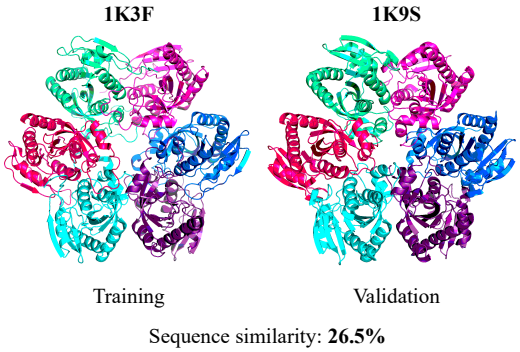

Figure 3: **Splitting by sequence similarity introduces data leakage in benchmarks for protein docking and interface prediction.** The figure shows two phosphorylase homooligomers, taken from DIPS, a standard dataset for training and validating machine learning models. The complex to the left (PDB code 1K3F), as well as the complex to the right (1K9S), is composed of five identical proteins (highlighted with colors). Nevertheless, the proteins across the entries have very low sequence similarity (26.5%). Despite the sequences in the complexes being different, the secondary structure of the chains, the topology of the interactions, as well as the 3D structure and the amino acids at the interfaces are highly similar across the entries (iDist < 0.04, the near-duplicate threshold; iAlign's p-value $< 10^{-6}$). Recent machine learning research for protein docking and interface prediction employed data splitting based on sequence similarity, resulting in data leakage.

In this study, we investigate a significantly more strict sequence-based splitting strategy, based on sequence alignment. Following Section 3.1, we assess data leakage using five PPI samples, each containing around 50,000 interactions. First, we employ the MMseqs2 algorithm (Steinegger & Söding, 2017) to cluster the protein sequences involved in all PPIs so that sequences sharing at least 30% sequence identity are in the same clusters. The MMseqs2 algorithm enables large-scale clustering of protein sequences by reducing sequence alignment only to the pairs that have common fragments, which are identified through a rapid search across the fragment database. After clustering all the sequences, we form groups of PPIs as connected components within a graph, where nodes represent proteins and edges are given either by protein–protein interactions in the dataset or sharing the same MMseqs2 cluster. This leads to 5,600 groups of interactions on average, which we subsequently split using the $90\%/10\%$ ratio.

We find that this sequence-based splitting approach yields a substantial improvement in structural data leakage compared to metadata-based splits, with a leakage rate of 30% (Figure 1). Additionally, we perform the same experiment using Linclust (Steinegger & Söding, 2018), the MMsesq2 mode

enabling even faster clustering, which may be suitable for larger PPI sets available in future. As expected, using Linclust results in a worse average leakage ratio of 37%. Overall, although a certain level of test data similarity to the training set is acceptable, sequence-based splitting may still result in nearly identical interfaces and topologies distributed across different folds (Figure 3).

4 BEST PRACTICES FOR DATA SPLITTING OF PROTEIN COMPLEXES

In this section, we review the recent works addressing the data leakage issue, highlighting the best practices toward high-quality splits and making a step towards evaluation of generalization to new unseen interaction modes, with distinct interfaces.

4.1 SPLITTING BY INTERFACE SIMILARITY IS RECOMMENDED

To overcome the limitations of sequence-based protein comparison, van Kempen et al. (2023) introduced Foldseek—a fast method to compare and cluster structures of single proteins. Foldseek converts consecutive structural patterns of the protein backbone and its contacts into a sequence in the alphabet of structural features and then uses MMseqs2 (Steinegger & Söding, 2017) for fast sequence-based search. Some of the most recent works employed Foldseek to compare protein complexes. Specifically, Sverrisson et al. (2023) and Morehead et al. (2023) used the method to map residues between two complexes and count the number of matches at the interaction interface regions. While this approach scales well to large PPI data, it may be biased towards comparing whole complexes rather than interacting regions, as Foldseek is designed to match complete protein folds, based on sequential information of the protein backbones. This may be a significant limitation when comparing protein–protein interfaces since protein pairs with different folds may establish similar interfaces (Sen & Madhusudhan, 2022; Mirabello & Wallner, 2018).

Gainza et al. (2023) overcame this limitation by directly aligning interacting interface regions using the traditional TM-align method (Zhang & Skolnick, 2005), applied pairwise to all interfaces, which was computationally feasible for their smaller dataset. Finally, Bushuiev et al. (2024) developed iDist—a scalable approximation of TM-align-based structural alignment methods. This method enabled to construct PPIRef (Bushuiev et al., 2024), a non-redundant dataset of protein–protein interactions mined from the whole PDB. The iDist algorithm directly compares protein–protein interfaces on a large scale, and can be used to create non-leaking data splits of large PPI sets. Alternatively, when the dataset size does not exceed several thousand interactions, non-leaking splits may also be achieved by employing pairwise comparison of all interfaces using traditional alignment-based tools such as aforementioned TM-align or its adaption to PPIs, iAlign (Gao & Skolnick, 2010a). Overall, directly comparing PPI interfaces addresses the issues of metadata- and sequence-based splitting, and the emerging fast methods enable large-scale preparation of non-leaking splits.

4.2 HUMAN EXPERTISE IS HIGHLY-BENEFICIAL

It is important to note that some datasets have a natural schema of splitting given by their construction or manual analysis by the authors. For instance, SKEMPI v2.0 (Jankauskaitė et al., 2019), a standard dataset for PPI mutations, was semi-automatically categorized into distinct groups (such as antibody–antigen or protease–inhibitor interactions) specifically for machine learning purposes. However, this grouping was initially overlooked by the machine learning community in favor of simpler splits based on metadata similarity (Bushuiev et al., 2024; Tsishyn et al., 2024). We validate that splitting SKEMPI v2.0 based on the domain expertise leads to higher-quality splits. Specifically, when using the train-test splits from Bushuiev et al. (2024) based on the described domain-expertise grouping, iDist identifies 0% leakage compared to 56% leakage with a PPI code-based split (Luo et al., 2023) discussed in Section 3.1.

Another example is BM5 (Pang et al., 2017), a dataset commonly used for training and validating models for scoring PPI docking poses (Xu & Bonvin, 2024; Réau et al., 2023; Geng et al., 2020). Although BM5 comprises nearly six million structures, these are synthetic redocked poses of 232 non-redundant complexes, easy to split properly without large-scale analysis. Here, we validate that even the commonly-employed random 10-fold cross-validation split of the poses on the level of originating complexes (Xu & Bonvin, 2024; Réau et al., 2023; Renaud et al., 2021) has sufficient quality. Specifically, iDist only detects a single leak ($< 1\%$), where one of the interfaces consists

of four disjoint parts similar to the other interface. Ignoring the details of the dataset construction in favor of a random split of the six million poses would instead result in near-duplicate redocked structures of same complexes scattered across folds with near-100% leakage.

5 RECOMMENDATIONS

In this work, we have highlighted the insufficiency of splitting protein complexes based on metadata or protein sequence similarity for effective evaluation, due to the high structural redundancy of the datasets derived from the Protein Data Bank (Berman et al., 2000). We believe that the abundant issue of data leakage across different addressed tasks may hinder the further advancements in the field. Therefore, we propose the following recommendations for data splitting of protein–protein interactions:

1. **Use interface similarity as the standard criterion for splitting protein interactions**. Directly dividing complexes based on interface similarity addresses the limitations of traditional metadata- and sequence-based methods, yet it has been rarely utilized. Until recently, large-scale comparison of PPIs was infeasible due to the high computational demands of structural alignment. However, recent advancements have led to the development of efficient approximation methods (see Section 4.1).
2. **Thoroughly review the information provided by dataset authors**. Domain experts often provide information beyond the basic inputs and outputs for machine learning models, which can be crucial for developing evaluation strategies that align with practical needs. Neglecting this vital information can trap machine learning research into repeatedly using flawed, leak-prone benchmarks for years (see Section 4.2).
3. **Quantify and report data leakage when there is no control over train-test splits**. When training from large data or using pre-trained models, the data splitting strategy may become irrelevant or given beforehand. In such scenarios, we recommend employing interface similarity to evaluate and report the extent of overlap between training and test examples. This approach is a standard practice in other machine learning-driven fields, such as computer vision (Radford et al., 2021, Appendix C) and natural language processing (Brown et al., 2020, Appendix C).

ACKNOWLEDGMENTS

This work was supported by the Ministry of Education, Youth and Sports of the Czech Republic through projects e-INFRA CZ [ID:90254], ELIXIR [LM2023055], CETOCOEN Excellence CZ.02.1.01/0.0/0.0/17_043/0009632, and ESFRI RECETOX RI [LM2023069]. This work was also supported by the European Union (ERC project FRONTIER no. 101097822) and the CETOCOEN EXCELLENCE Teaming project supported from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 857560. This work was also supported by the Czech Science Foundation (GA CR) through grant 21-11563M and through the European Union's Horizon 2020 research and innovation programme under Marie Skłodowska-Curie grant agreement No. 891397. Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

REFERENCES

Bruce Alberts, Dennis Bray, Karen Hopkin, Alexander D Johnson, Julian Lewis, Martin Raff, Keith Roberts, and Peter Walter. *Essential Cell Biology*. Garland Science, 2015. URL https://wwnorton.com/books/9781324033356.

Nathaniel R Bennett, Brian Coventry, Inna Goreshnik, Buwei Huang, Aza Allen, Dionne Vafeados, Ying Po Peng, Justas Dauparas, Minkyung Baek, Lance Stewart, et al. Improving de novo protein binder design with deep learning. *Nature Communications*, 14(1):2625, 2023. doi: 10.1038/s41467-023-38328-5. URL https://doi.org/10.1038/s41467-023-38328-5.

Tord Berggård, Sara Linse, and Peter James. Methods for the detection and analysis of protein–protein interactions. *Proteomics*, 7(16):2833–2842, 2007. doi: 10.1002/pmic.200700131. URL https://pubmed.ncbi.nlm.nih.gov/17640003/.

Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The Protein Data Bank. *Nucleic Acids Research*, 28(1): 235–242, 2000. doi: 10.1093/nar/28.1.235. URL https://doi.org/10.1093/nar/28.1.235.

Judith Bernett, David B Blumenthal, and Markus List. Cracking the black box of deep sequence-based protein-protein interaction prediction. *bioRxiv*, pp. 2023–01, 2023. doi: 10.1093/bib/bbae076. URL https://doi.org/10.1093/bib/bbae076.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. doi: 10.48550/arXiv.2005.14165. URL https://doi.org/10.48550/arXiv.2005.14165.

Prasad V Burra, Ying Zhang, Adam Godzik, and Boguslaw Stec. Global distribution of conformational states derived from redundant models in the pdb points to non-uniqueness of the protein structure. *Proceedings of the National Academy of Sciences*, 106(26):10505–10510, 2009. doi: 10.1073/pnas.0812152106. URL https://pubmed.ncbi.nlm.nih.gov/19553204/.

Anton Bushuiev, Roman Bushuiev, Petr Kouba, Anatolii Filkin, Marketa Gabrielova, Michal Gabriel, Jiri Sedlar, Tomas Pluskal, Jiri Damborsky, Stanislav Mazurenko, and Josef Sivic. Learning to design protein-protein interactions with enhanced generalization. *The Twelfth International Conference on Learning Representations*, 2024. doi: 10.48550/arXiv.2310.18515. URL https://doi.org/10.48550/arXiv.2310.18515.

Martin Buttenschoen, Garrett M Morris, and Charlotte M Deane. PoseBusters: AI-based docking methods fail to generate physically valid poses or generalise to novel sequences. *Chemical Science*, 2024. doi: 10.48550/arXiv.2308.05777. URL https://doi.org/10.48550/arXiv.2308.05777.

Brad Chapman and Jeffrey Chang. Biopython: Python tools for computational biology. *ACM Sigbio Newsletter*, 20(2):15–19, 2000. doi: 10.1145/360262.360268. URL https://doi.org/10.1145/360262.360268.

Jun Cheng, Guido Novati, Joshua Pan, Clare Bycroft, Akvilė Žemgulytė, Taylor Applebaum, Alexander Pritzel, Lai Hong Wong, Michal Zielinski, Tobias Sargeant, et al. Accurate proteome-wide missense variant effect prediction with AlphaMissense. *Science*, 381(6664):eadg7492, 2023. doi: 10.1126/science.adg7492. URL https://doi.org/10.1126/science.adg7492.

Lee-Shin Chu, Jeffrey A Ruffolo, Ameya Harmalkar, and Jeffrey J Gray. Flexible protein-protein docking with a multi-track iterative transformer. *Protein Science*, pp. e4862, 2023. doi: 10.1101/2023.06.29.547134. URL https://pubmed.ncbi.nlm.nih.gov/37425754/.

Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. DiffDock: Diffusion steps, twists, and turns for molecular docking. *arXiv preprint arXiv:2210.01776*, 2022. doi: 10.48550/arXiv.2210.01776. URL https://doi.org/10.48550/arXiv.2210.01776.

Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J Ragotte, Lukas F Milles, Basile IM Wicky, Alexis Courbet, Rob J de Haas, Neville Bethel, et al. Robust deep learning–based protein sequence design using ProteinMPNN. *Science*, 378(6615):49–56, 2022. doi: 10.1126/science.add2187. URL https://www.science.org/doi/10.1126/science.add2187.

Hao Du, Yuchen Ren, Yan Lu, He Huang, Yating Liu, Zhendong Mao, Xinqi Gong, and Wanli Ouyang. Split and merge proxy: pre-training protein-protein contact prediction by mining rich information from monomer data, 2023. URL https://openreview.net/pdf?id=o8fqVVKN3H.

Richard Evans, Michael O'Neill, Alexander Pritzel, Natasha Antropova, Andrew Senior, Tim Green, Augustin Žídek, Russ Bates, Sam Blackwell, Jason Yim, et al. Protein complex prediction with AlphaFold-Multimer. *biorxiv*, pp. 2021–10, 2021. doi: 10.1101/2021.10.04.463034. URL https://doi.org/10.1101/2021.10.04.463034.

Alex Fout, Jonathon Byrd, Basir Shariat, and Asa Ben-Hur. Protein interface prediction using graph convolutional networks. *Advances in Neural Information Processing Systems*, 30, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/f507783927f2ec2737ba40afbd17efb5-Paper.pdf.

Pablo Gainza, Sarah Wehrle, Alexandra Van Hall-Beauvais, Anthony Marchand, Andreas Scheck, Zander Harteveld, Stephen Buckley, Dongchun Ni, Shuguang Tan, Freyr Sverrisson, et al. De novo design of protein interactions with learned surface fingerprints. *Nature*, pp. 1–9, 2023. doi: 10.1038/s41586-023-05993-x. URL https://doi.org/10.1038/s41586-023-05993-x.

Octavian-Eugen Ganea, Xinyuan Huang, Charlotte Bunne, Yatao Bian, Regina Barzilay, Tommi Jaakkola, and Andreas Krause. Independent SE(3)-equivariant models for end-to-end rigid protein docking. *arXiv preprint arXiv:2111.07786*, 2021. doi: 10.48550/arXiv.2111.07786. URL https://doi.org/10.48550/arXiv.2111.07786.

Mu Gao and Jeffrey Skolnick. iAlign: a method for the structural comparison of protein–protein interfaces. *Bioinformatics*, 26(18):2259–2265, 2010a. doi: 10.1093/bioinformatics/btq404. URL https://doi.org/10.1093/bioinformatics/btq404.

Mu Gao and Jeffrey Skolnick. Structural space of protein–protein interfaces is degenerate, close to complete, and highly connected. *Proceedings of the National Academy of Sciences*, 107(52): 22517–22522, 2010b. doi: 10.1073/pnas.1012820107. URL https://doi.org/10.1073/pnas.1012820107.

Leonardo Garma, Srayanta Mukherjee, Pralay Mitra, and Yang Zhang. How many protein-protein interactions types exist in nature? *PLoS One*, 7(6):e38913, 2012. doi: 10.1371/journal.pone.0038913. URL https://doi.org/10.1371/journal.pone.0038913.

Cunliang Geng, Li C Xue, Jorge Roel-Touris, and Alexandre MJJ Bonvin. Finding the $\Delta\Delta$G spot: Are predictors of binding affinity changes upon mutations in protein–protein interactions ready for it? *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 9(5):e1410, 2019. doi: 10.1002/wcms.1410. URL https://api.semanticscholar.org/CorpusID:91262027.

Cunliang Geng, Yong Jung, Nicolas Renaud, Vasant Honavar, Alexandre MJJ Bonvin, and Li C Xue. iScore: a novel graph kernel-based function for scoring protein–protein docking models. *Bioinformatics*, 36(1):112–121, 2020. doi: 10.1093/bioinformatics/btz496. URL https://doi.org/10.1093/bioinformatics/btz496.

Anna G Green, Hadeer Elhabashy, Kelly P Brock, Rohan Maddamsetti, Oliver Kohlbacher, and Debora S Marks. Large-scale discovery of protein interactions at residue resolution using co-evolution calculated from genomic sequences. *Nature Communications*, 12(1):1396, 2021. doi: 10.1038/s41467-021-21636-z. URL https://doi.org/10.1038/s41467-021-21636-z.

Justina Jankauskaitė, Brian Jiménez-García, Justas Dapkūnas, Juan Fernández-Recio, and Iain H Moal. SKEMPI 2.0: an updated benchmark of changes in protein–protein binding energy, kinetics and thermodynamics upon mutation. *Bioinformatics*, 35(3):462–469, 2019. doi: 10.1093/bioinformatics/bty635. URL https://pubmed.ncbi.nlm.nih.gov/30020414/.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, 2021. doi: 10.1038/s41586-021-03819-2. URL https://doi.org/10.1038/s41586-021-03819-2.

Mohamed Amine Ketata, Cedrik Laue, Ruslan Mammadov, Hannes Stärk, Menghua Wu, Gabriele Corso, Céline Marquet, Regina Barzilay, and Tommi S Jaakkola. DiffDock-PP: Rigid protein-protein docking with diffusion models. *arXiv preprint arXiv:2304.03889*, 2023. doi: 10.48550/arXiv.2304.03889. URL https://doi.org/10.48550/arXiv.2304.03889.

Tzu-Ping Ko, Joseph D Ng, and Alexander McPherson. The three-dimensional structure of canavalin from jack bean (canavalia ensiformis). *Plant Physiology*, 101(3):729–744, 1993. doi: 10.1104/pp.101.3.729. URL https://pubmed.ncbi.nlm.nih.gov/8310056/.

Daniel Krowarsch, Michal Dadlez, Olga Buczek, Izabela Krokoszynska, Arne O Smalas, and Jacek Otlewski. Interscaffolding additivity: binding of P1 variants of bovine pancreatic trypsin inhibitor to four serine proteases. *Journal of Molecular Biology*, 289(1):175–186, 1999. doi: 10.1006/jmbi.1999.2757. URL https://pubmed.ncbi.nlm.nih.gov/10339415/.

Bian Li, Yucheng T Yang, John A Capra, and Mark B Gerstein. Predicting changes in protein thermodynamic stability upon point mutation with deep 3d convolutional neural networks. *PLoS computational biology*, 16(11):e1008291, 2020. doi: 10.1371/journal.pcbi.1008291. URL https://doi.org/10.1371/journal.pcbi.1008291.

Jie Li, Xingyi Guan, Oufan Zhang, Kunyang Sun, Yingze Wang, Dorian Bagni, and Teresa Head-Gordon. Leak proof PDBBind: A reorganized dataset of protein-ligand complexes for more generalizable binding affinity prediction. *ArXiv*, 2023. doi: 10.48550/arXiv.2308.09639. URL https://doi.org/10.48550/arXiv.2308.09639.

Shiwei Liu, Tian Zhu, Milong Ren, Chungong Yu, Dongbo Bu, and Haicang Zhang. Predicting mutational effects on protein-protein binding via a side-chain diffusion probabilistic model. *arXiv preprint arXiv:2310.19849*, 2023. doi: 10.48550/arXiv.2310.19849. URL https://doi.org/10.48550/arXiv.2310.19849.

Xianggen Liu, Yunan Luo, Pengyong Li, Sen Song, and Jian Peng. Deep geometric representations for modeling effects of mutations on protein-protein binding affinity. *PLoS computational biology*, 17(8):e1009284, 2021. doi: 10.1371/journal.pcbi.1009284. URL https://doi.org/10.1371/journal.pcbi.1009284.

Shitong Luo, Yufeng Su, Zuofan Wu, Chenpeng Su, Jian Peng, and Jianzhu Ma. Rotamer density estimator is an unsupervised learner of the effect of mutations on protein-protein interaction. In *The Eleventh International Conference on Learning Representations*, 2023. doi: 10.1101/2023.02.28.530137. URL https://openreview.net/forum?id=_X9Yl1K2mD.

Claudio Mirabello and Björn Wallner. Topology independent structural matching discovers novel templates for protein interfaces. *Bioinformatics*, 34(17):i787–i794, 2018. doi: 10.1093/bioinformatics/bty587. URL https://doi.org/10.1093/bioinformatics/bty587.

Alex Morehead, Chen Chen, and Jianlin Cheng. Geometric transformers for protein interface contact prediction. In *International Conference on Learning Representations*, 2022. doi: 10.48550/arXiv.2110.02423. URL https://doi.org/10.48550/arXiv.2110.02423.

Alex Morehead, Chen Chen, Ada Sedova, and Jianlin Cheng. DIPS-Plus: The enhanced database of interacting protein structures for interface prediction. *Scientific Data*, 10(1):509, 2023. doi: 10.48550/arXiv.2106.04362. URL https://doi.org/10.48550/arXiv.2106.04362.

Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Jingfang Xu, and Xueqi Cheng. DeepRank: A new deep architecture for relevance ranking in information retrieval. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 257–266, 2017. doi: 10.48550/arXiv.1710.05649. URL https://doi.org/10.48550/arXiv.1710.05649.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021. doi: 10.48550/arXiv.2103.00020. URL https://doi.org/10.48550/arXiv.2103.00020.

Manon Réau, Nicolas Renaud, Li C Xue, and Alexandre MJJ Bonvin. DeepRank-GNN: a graph neural network framework to learn patterns in protein–protein interfaces. *Bioinformatics*, 39(1): btac759, 2023. doi: 10.1093/bioinformatics/btac759. URL https://doi.org/10.1093/bioinformatics/btac759.

Nicolas Renaud, Cunliang Geng, Sonja Georgievska, Francesco Ambrosetti, Lars Ridder, Dario F Marzella, Manon F Réau, Alexandre MJJ Bonvin, and Li C Xue. Deeprank: a deep learning framework for data mining 3d protein-protein interfaces. *Nature communications*, 12(1): 7068, 2021. doi: 10.1038/s41467-021-27396-0. URL https://doi.org/10.1038/s41467-021-27396-0.

Julia R Rogers, Gergő Nikolényi, and Mohammed AlQuraishi. Growing ecosystem of deep learning methods for modeling protein–protein interactions. *Protein Engineering, Design and Selection*, 36:gzad023, 2023. doi: 10.1093/protein/gzad023. URL https://doi.org/10.1093/protein/gzad023.

Neeladri Sen and Mallur S Madhusudhan. A structural database of chain–chain and domain–domain interfaces of proteins. *Protein Science*, 31(9):e4406, 2022. doi: 10.1002/pro.4406. URL https://doi.org/10.1002/pro.4406.

Sisi Shan, Shitong Luo, Ziqing Yang, Junxian Hong, Yufeng Su, Fan Ding, Lili Fu, Chenyu Li, Peng Chen, Jianzhu Ma, et al. Deep learning guided optimization of human antibody against SARS-CoV-2 variants with broad neutralization. *Proceedings of the National Academy of Sciences*, 119(11):e2122954119, 2022. doi: 10.1073/pnas.2122954119. URL https://doi.org/10.1073/pnas.2122954119.

Martin Steinegger and Johannes Söding. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature Biotechnology*, 35(11):1026–1028, 2017. doi: 10.1038/nbt.3988. URL https://doi.org/10.1038/nbt.3988.

Martin Steinegger and Johannes Söding. Clustering huge protein sequence sets in linear time. *Nature communications*, 9(1):2542, 2018. doi: 10.1038/s41467-018-04964-5. URL https://doi.org/10.1038/s41467-018-04964-5.

Baris E Suzek, Hongzhan Huang, Peter McGarvey, Raja Mazumder, and Cathy H Wu. UniRef: comprehensive and non-redundant uniprot reference clusters. *Bioinformatics*, 23(10):1282–1288, 2007. doi: 10.1093/bioinformatics/btm098. URL https://pubmed.ncbi.nlm.nih.gov/17379688/.

Freyr Sverrisson, Mehmet Akdel, Dylan Abramson, Jean Feydy, Alexander Goncearenco, Yusuf Adeshina, Daniel Kovtun, Céline Marquet, Xuejin Zhang, David Baugher, et al. DiffMaSIF: Surface-based protein-protein docking with diffusion models. In *Machine Learning in Structural Biology workshop at NeurIPS 2023*, 2023. URL https://hal.science/hal-04360638.

Tao Tang, Xiaocai Zhang, Yuansheng Liu, Hui Peng, Binshuang Zheng, Yanlin Yin, and Xiangxiang Zeng. Machine learning on protein–protein interaction prediction: models, challenges and trends. *Briefings in Bioinformatics*, 24(2):bbad076, 2023. doi: 10.1093/bib/bbad076. URL https://doi.org/10.1093/bib/bbad076.

The UniProt Consortium. UniProt: the universal protein knowledgebase in 2023. *Nucleic Acids Research*, 51(D1):D523–D531, 2023. doi: 10.1093/nar/gkac1052. URL https://doi.org/10.1093/nar/gkac1052.

Raphael Townshend, Rishi Bedi, Patricia Suriana, and Ron Dror. End-to-end learning on 3D protein structure for interface prediction. *Advances in Neural Information Processing Systems*, 32, 2019. doi: 10.48550/arXiv.1807.01297. URL https://doi.org/10.48550/arXiv.1807.01297.

Matsvei Tsishyn, Fabrizio Pucci, and Marianne Rooman. Quantification of biases in predictions of protein–protein binding affinity changes upon mutations. *Briefings in Bioinformatics*, 25 (1):bbad491, 2024. doi: 10.1093/bib/bbad491. URL https://doi.org/10.1093/bib/bbad491.

Michel van Kempen, Stephanie S Kim, Charlotte Tumescheit, Milot Mirdita, Jeongjae Lee, Cameron LM Gilchrist, Johannes Söding, and Martin Steinegger. Fast and accurate protein structure search with Foldseek. *Nature Biotechnology*, pp. 1–4, 2023. doi: 10.1038/s41587-023-01773-0. URL https://doi.org/10.1038/s41587-023-01773-0.

Nathan P Williams, Carlos HM Rodrigues, Jia Truong, David B Ascher, and Jessica K Holien. DockNet: high-throughput protein–protein interface contact prediction. *Bioinformatics*, 39(1): btac797, 2023. doi: 10.1093/bioinformatics/btac797. URL https://pubmed.ncbi.nlm.nih.gov/36484688/.

Xiaotong Xu and Alexandre MJJ Bonvin. DeepRank-GNN-esm: A graph neural network for scoring protein-protein models using protein language model. *Bioinformatics Advances*, 4 (1):vbad191, 2024. doi: 10.1093/bioadv/vbad191. URL https://doi.org/10.1093/bioadv/vbad191.

Ziyang Yu, Wenbing Huang, and Yang Liu. Rigid protein-protein docking via equivariant elliptic-paraboloid interface prediction. *arXiv preprint arXiv:2401.08986*, 2024. doi: 10.48550/arXiv.2401.08986. URL https://doi.org/10.48550/arXiv.2401.08986.

Yang Yue, Shu Li, Lingling Wang, Huanxiang Liu, Henry HY Tong, and Shan He. MpbPPI: a multi-task pre-training-based equivariant approach for the prediction of the effect of amino acid mutations on protein–protein interactions. *Briefings in Bioinformatics*, 24(5):bbad310, 2023. doi: 10.1093/bib/bbad310. URL https://doi.org/10.1093/bib/bbad310.

Chengxin Zhang, Morgan Shine, Anna Marie Pyle, and Yang Zhang. US-align: universal structure alignments of proteins, nucleic acids, and macromolecular complexes. *Nature methods*, 19(9): 1109–1115, 2022. doi: 10.1038/s41592-022-01585-1. URL https://doi.org/10.1038/s41592-022-01585-1.

Yang Zhang and Jeffrey Skolnick. TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Research*, 33(7):2302–2309, 2005. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1084323/.

A METHODS

In this section, we provide technical details of the methods used to quantify similarities between protein–protein interactions. Appendix A.1 describes the implementation details of the methods, and Appendix A.2 provides details on randomized train-test splitting experiments (see Figure 1). The source code[1] for the experiments is available as a part of the PPIRef repository for working with 3D structures of protein–protein interactions.

A.1 COMPARING PROTEIN–PROTEIN INTERACTIONS

Sequence similarity. For illustrative examples, we measure sequence similarity by calculating sequence identity. Specifically, to calculate sequence similarity between two complexes, we take the maximum sequence identity upon the pairwise alignment between all proteins from one complex and all from the other. To align sequences, we follow Buttenschoen et al. (2024) and use `PairwiseAligner` from Biopython[2] (Chapman & Chang, 2000) with an open gap penalty of -11 and an extension gap penalty of -1, as well as the BLOSUM62 substitution matrix. We always normalize sequence identities by dividing by the length of the shorter of the two sequences.

The described pairwise sequence similarity with Biopython, using dynamic programming, is too slow for data splitting experiments with thousands of PPIs. Therefore, to split interactions according to their sequence similarity (see Figure 1), we use the MMseqs2 clustering algorithm[3] (Steinegger & Söding, 2017). Specifically, we run `mmseqs easy-cluster` with the standard parameters and `--min-seq-id 0.3` to ensure strict clustering where two proteins with at least 30% sequence identity fall under the same cluster. As discussed in Section 3.2, we also experiment with Linclust (Steinegger & Söding, 2018). We run the algorithm using `mmseqs easy-linclust` with the default parameters.

Interface similarity. To measure interface similarity across experiments, we use iDist (Bushuiev et al., 2024), a scalable and accurate approximation of iAlign (Gao & Skolnick, 2010a) and US-align (Zhang et al., 2022), two methods based on conventional TM-align (Zhang & Skolnick, 2005) for protein structures. We use the iDist implementation from the PPIRef package[4]. We always extract PPI interfaces based on the 6Å contacts between heavy atoms (using the `PPIExtractor` class). Subsequently, we consider PPIs as near duplicates if their iDist distance (`IDist.compare`) is lower than 0.04, as reported in Bushuiev et al. (2024).

To detect near duplicate PPI interfaces on a large scale for estimating data leakage, we follow the same methodology in a scalable implementation. Specifically, we first embed all interfaces in parallel (`IDist.embed_parallel`) and then query the test interfaces against the training interfaces (`IDist.query`). We consider a test interface leaked if it has at least one near-duplicate hit in the training data.

A.2 ESTIMATING DATA LEAKAGE

To assess the quality of individual methods for train-test splitting, we perform randomized large-scale experiments. In the first step, we extract all 6Å heavy-atom interfaces from PDB as of January 27, 2024, using the aforementioned PPIRef package. We filter out improper interfaces using the standard criteria for buried surface area and structure quality (Townshend et al., 2019; Bushuiev et al., 2024). This leads to 349,685 interfaces from 99,401 PDB entries.

In the next step, we create 5 subsamples of PPIs by randomly drawing 15,000 PDB codes from the data and selecting all corresponding interactions. This results in 5 sets of roughly 50,000 interactions each. Then, we split each subsample with four different methods discussed in Section 3 in the approximate 90%/10% ratio (20 splits in total), and report the means and standard deviations of their leakage (Figure 1).

[1] https://github.com/anton-bushuiev/PPIRef/blob/main/notebooks/revealing_data_leakage_iclr_gem_2024.ipynb

[2] https://github.com/biopython/biopython

[3] https://github.com/soedinglab/mmseqs2

[4] https://github.com/anton-bushuiev/PPIRef

Specifically, for splitting based on PDB (e.g., `1BUI`) and PPI codes (e.g., `1BUI_A_C`), we simply split the codes in the 90%/10% ratio. For splitting based on PDB deposition time, we sort all the PDB codes according to their deposition time and select the most recent 10% for testing. For splitting based on MMseqs2 clusters, we divide PPIs into groups such that the groups do not share sequences from the same clusters, and split the groups in the 90%/10% ratio.

B SUPPLEMENTARY FIGURES

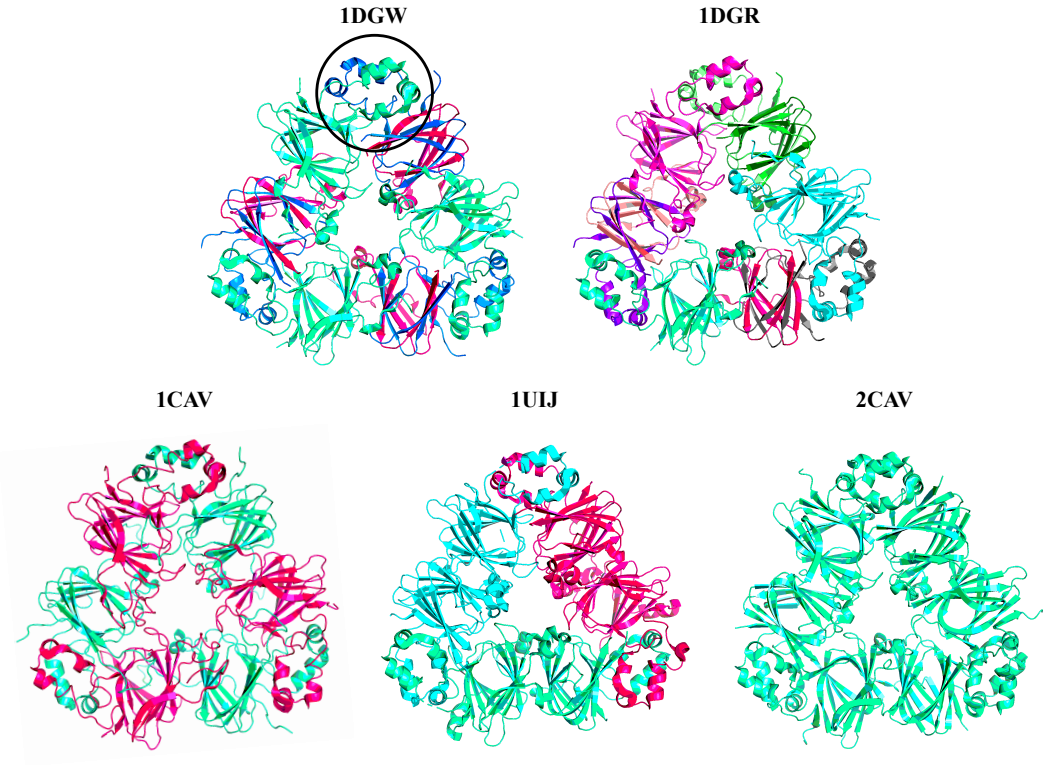

Figure 4: **Structural redundancy challenges data splitting of protein interactions.** The figure shows five different PDB entries representing the 3D structure of the canavalin trimer (Ko et al., 1993), with protein chains in different colors. The high level of structural redundancy observed in protein–protein interactions, both within individual entries and across multiple entries, necessitates careful data splitting strategies (for example, the interaction highlighted within the circle is represented in the figure fifteen times, occurring three times in each structure). However, inconsistent metadata and the modular nature of protein structures make naive approaches, such as splitting based on PDB codes or sequence similarity, prone to fail, distributing the same interactions across training-validation-test folds.