Docking and Machine Learning approaches to explore new scaffolds for molecules of therapeutic interest

Philippe Pinel

PSL UNIVERSITÉ PARIS

THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PSL

Préparée à Mines Paris-PSL

Docking and Machine Learning approaches to explore new scaffolds for molecules of therapeutic interest

Approches de docking et de Machine Learning pour l'exploration de nouveaux « scaffolds » lors de la recherche de molécules d'intérêt thérapeutique.

Soutenue par
Philippe Pinel
Le 9 Juillet 2024

École doctorale n°621
Ingénierie des Systèmes, Matériaux, Mécaniques, Énergétique

Spécialité
Bio-informatique

Composition du jury :

Arnaud Blondel Senior Scientist Institut Pasteur	*Rapporteur*
Catherine Etchebest Professeure Université Paris Cité	*Rapportrice*
Dragos Horvath Directeur de recherche CNRS	*Examinateur*
Bogdan Iorga Directeur de recherche CNRS	*Examinateur*
Olivier Lequin Professeur Sorbonne Université	*Examinateur*
Véronique Stoven Professeure Mines Paris-PSL	*Directrice de thèse*

MINES PARIS | PSL

Acknowledgement

A l'heure où j'écris les dernières lignes de ce manuscrit, je ne puis m'empêcher de remercier toutes les personnes ayant participé au succès de cette longue et périeuse aventure.

Tout d'abord, je tiens à remercier le jury d'avoir assisté à la soutenance, en particulier Arnaud Blondel et Catherine Etchebest qui ont étudié ce manuscrit de fond en comble. Merci à Dragos Horvath, Bogdan Iorga et Paul Fogel pour leurs échanges durant la soutenance, et à Olivier Lequin pour avoir présidé le jury.

Je souhaite également remercier Yann Gaston-Mathé, Quentin Perron et Nicolas Do Huu qui ont permis une collaboration fluide entre Iktos et l'Université PSL.

Cette thèse n'aurait pu être une réussite si je n'avais pas été si bien encadré. Pour cela, je tiens à exprimer ma gratitude envers ma directrice de thèse, Véronique Stoven. Je ne me souviens pas d'un seul instant où nous n'avons pas été sur la même longueur d'onde. Son expertise, son exigence et sa franchise m'ont permis de gagner en rigueur et ont abouti à un travail dont je peux être fier. Je tiens également à remercier chaleureusement Brice Hoffmann qui m'a encadré côté Iktos. Sa disponibilité, sa bienveillance, mais aussi ses remarques expertes toujours pertinentes ont sans aucun doute contribué à la réussite de ce projet. Si cette collaboration a été un succès, c'est aussi grâce à la fluidité de nos échanges avec Véronique.

Outre l'apport professionnel, la thèse est également l'occasion de rencontrer des "compagnons de galère". A tous mes compatriotes du CBIO, je vous remercie pour votre soutien ! En particulier Chloé, Florian, Vincent et Thomas W qui ont réussi à créer un environment de travail si agréable, Gwenn pour m'avoir financé mes repas, Matthieu pour tous nos cafés qu'on aurait pu assimiler à des séances de psy, Maguette pour son incroyable rapidité à déjeuner, Victor pour ta superbe moustache, Arthur pour ce magnifique template de thèse, Tristan pour m'avoir fait découvrir Mark Shadow, Tom pour avoir revolutionné la "spatial transcriptomics" telle que nous la connaissions, Thomas B pour avoir commencé sa thèse en tant que professeur et manager, Marvin pour m'avoir appris à prononcer "protein", mais aussi à Alice, GK, Guillaume, Jérémy, Julian, Julie, Katia, Paul et Simon pour tous ces moments nous faisant oublier la pression qui reposait sur nos têtes.

Au cours de cette thèse Cifre, j'ai eu l'opportunité d'échanger avec de nombreuses personnes au sein d'Iktos, qui m'ont épaulé depuis le premier jour. Je souhaiterais en particulier remercier Nicolas Devaux pour m'avoir tant appris, Guillaume, Maud et Maoussi pour tous nos restaurants à 5 km du bureau le midi, Ennys pour tous nos "pétages de câble", Dara pour son extrême sympathie, Pierre pour nos batailles de poste de travail, mais aussi Anna, Arthur, Harold, Juan et JB pour tout leur soutien et leur aide.

Parce qu'une thèse ne se joue pas qu'au bureau, mais aussi à l'extérieur, je voudrais

remercier tous mes amis pour avoir été présents durant ces dures années, ma famille qui, bien que n'ayant pas trop compris ce sujet, a été si fière de moi, et ma belle-famille pour leur grand intérêt. Enfin, ces remerciements ne pourraient être complets sans remercier Eléonore, qui m'a tant soutenu tout au long de la thèse, et sans qui cette aventure eut été bien plus triste.

Résumé

La découverte de médicaments, de l'identification de candidats jusqu'au développement clinique, implique parfois de résoudre des problèmes de 'scaffold hopping', dans le but d'optimiser l'activité biologique, la sélectivité, les propriétés ADME, ou de réduire les préoccupations toxicologiques des molécules. Ils consistent à identifier des molécules actives dont les modes de liaison sont similaires mais dont les structures chimiques sont différentes de celles des actifs connus. Le 'large-step scaffold hopping', qui correspond au degré le plus élevé de différence structurelle avec la molécule initiale, nécessite l'aide de méthodes calculatoires. Le docking est considéré comme la méthode de choix pour l'identification de telles molécules isofonctionnelles. Cependant, la structure de la protéine peut ne pas être adaptée au docking en raison d'une faible résolution, voire être inconnue. Dans de tels cas, les approches 'ligand-based' sont prometteuses mais souvent insuffisantes car basées sur des descripteurs moléculaires n'ayant pas été spécifiquement développés pour le 'large-step scaffold hopping'. La résolution de ces problèmes se résume à l'identification de descripteurs correspondant à une représentation de l'espace chimique dans laquelle deux molécules qui sont des cas de 'scaffold hopping' sont similaires, bien qu'elles soient dissemblables dans l'espace représenté par les descripteurs basés principalement sur la structure chimique. Afin d'évaluer la capacité des descripteurs à les résoudre, nous avons constitué un ensemble de cas de 'scaffold hopping' de haute qualité comprenant des paires de molécules actives pour une variété de protéines. Nous avons ensuite proposé une stratégie pour évaluer la pertinence des descripteurs pour résoudre ces problèmes, correspondant à des cas réels où une molécule active est connue, et la seconde active est recherchée parmi un ensemble de molécules leurres choisies de manière à éviter les biais statistiques. Nous avons ainsi illustré les limites des descripteurs classiques 2D et 3D. Par conséquent, nous proposons l'Interaction Fingerprints Profile (IFPP), une représentation moléculaire qui capture les modes de liaison des molécules via des dockings sur un panel de protéines diverses. L'évaluation de cette représentation sur le benchmark démontre son intérêt pour l'identification de molécules isofonctionnelles. Cependant, son calcul coûteux limite sa mise à l'échelle pour le criblage de bibliothèques moléculaires très larges. Nous avons remedié à cela en tirant parti du Metric Learning, qui permet une estimation rapide des similarités des IFPP des molécules, fournissant ainsi une stratégie de pré-criblage efficace applicable à de larges bibliothèques. Nos résultats suggèrent que l'IFPP est un outil intéressant et complémentaire aux méthodes existantes afin de résoudre le 'scaffold hopping'.

Mots clés : Docking, Interactions moléculaires, Machine Learning, Représentation moléculaire, Scaffold hopping

Abstract

The challenges of drug discovery from hit identification to clinical development sometimes involves addressing scaffold hopping issues, in order to optimise molecular biological activity or ADME properties, improve selectivity or mitigate toxicology concerns of a drug candidate. They consist in identifying active molecules of similar binding modes but of different chemical structures to that of known active molecules. Large-step scaffold hopping, which corresponds to the highest degree of structural dissimilarity with the original hit, cannot be easily solved without the aid of computational methods. Docking is usually viewed as the method of choice for identification of such isofunctional molecules. However, the structure of the protein may not be suitable for docking because of a low resolution, or may even be unknown. In such cases, ligand-based approaches offer promise but are often inadequate to handle large-step scaffold hopping, because they are based on molecular descriptors that were not specifically developed for it. Solving those problems boils down to the identification of molecular descriptors corresponding to an embedding of the chemical space in which two molecules that are examples of large-step scaffold hopping cases are similar (i.e. close), although they are dissimilar (i.e. far) in the space embedded by molecular descriptors based principally on the chemical structure. To evaluate molecular descriptors to solve this particular challenging task, we built a high quality dataset of scaffold hopping examples comprising pairs of active molecules and including a variety of protein targets. We then proposed a strategy to evaluate the relevance of molecular descriptors to that problem, corresponding to real-life applications where one active molecule is known, and the second active is searched among a set of decoys chosen in a way to avoid statistical bias. We assessed how limited classical 2D and 3D descriptors are at solving these problems. Therefore, we introduced the Interaction Fingerprints Profile (IFPP), a molecular representation that captures molecules' binding modes based on docking experiments against a panel of diverse high-quality protein structures. Evaluation on the benchmark demonstrated its interest for identifying isofunctional molecules. Nevertheless, its computation is expensive, which limits its scalability for screening very large molecular libraries. We proposed to overcome this limitation by leveraging Metric Learning approaches, allowing fast estimation of molecules IFPP similarities, thus providing an efficient pre-screening strategy that is applicable to very large molecular libraries. Overall, our results suggest that IFPP provides an interesting and complementary tool alongside existing methods, in order to address challenging scaffold hopping problems effectively in drug discovery.

Keywords : Docking, Molecular interactions, Machine Learning, Molecular representation, Scaffold hopping

Contents

List of Figures

List of Tables

Glossary

3D Three-dimensional

ADMET Absorption, distribution, metabolism, excretion and toxicology

CHC Cumulative Histogram Curves

DL Deep Learning

DNN Deep Neural Networks

EF Enrichment Factor

GE Graph Embedding

GNN Graph Neural Networks

H Hydrogen

IFP Interaction Fingerprint

IFPP Interaction Fingerprints Profile

LB Ligand-based

MCS Maximum Common Substructure

ML Machine Learning

MLP Multilayer Perceptron

PDB Protein Data Bank

PK Pharmacokinetic

QED Quantitative Estimate of Drug-likeness

QSAR Quantitative Structure-Activity Relationship

RMSD Root Mean Squared Deviation

SAR Structure-activity relationship

SB Structure-based

SMARTS SMILES arbitrary target specification

SMILES Simplified Molecular-Input Line-Entry System

SVM Support Vector Machines

1
Introduction

Abstract:

Identification of novel chemical compounds with biological activity similar to a known active molecule is an important challenge in drug discovery called 'scaffold hopping'. Small-, medium-, and large-step scaffold hopping efforts may lead to increasing degrees of chemical structure novelty with respect to the parent compound. Docking is usually viewed as the method of choice for identification of isofunctional molecules, i.e. highly dissimilar molecules that share common binding modes with a protein target. However, the structure of the protein may not be suitable for docking because of a low resolution, or may even be unknown. In such cases, ligand-based approaches offer promise but are often inadequate to handle large-step scaffold hopping, because they are based on molecular descriptors usually relying on molecular structure and not dedicated to this challenging task.

Résumé:

L'identification de nouveaux composés chimiques ayant une activité biologique similaire à une molécule active connue est un défi important dans la découverte de médicaments, appelé 'scaffold hopping'. Les stratégies de 'small-step', 'medium-step', et 'large-step' scaffold hopping peuvent conduire à des degrés croissants de nouveauté de structure chimique par rapport au composé parent. Le docking est généralement considéré comme la méthode de choix pour l'identification de molécules isofonctionnelles, c'est-à-dire des molécules très différentes qui partagent des modes de liaison communs avec une cible protéique. Cependant, la structure de la protéine peut ne pas être adaptée au docking en raison d'une faible résolution, voire être inconnue. Dans de tels cas, les approches 'ligand-based' offrent des perspectives mais sont souvent insuffisantes pour gérer le large-step scaffold hopping, car elles sont basées sur des descripteurs moléculaires qui reposent généralement sur la structure moléculaire et ne sont pas dédiés à résoudre cette tâche.

Contents

This PhD thesis is devoted to solving a particularly difficult task in drug discovery: how to solve scaffold hopping cases? In other words, finding biologically active molecules against a protein target of interest, but displaying dissimilar structures with respect to known hits for this target. This task becomes even harder when the 3D structure of the protein is unknown, which is the setting chosen in the present work. In this introduction, I will start by briefly presenting the fundamentals of drug discovery, in the particular case of small molecule drugs. Then, I will define the term scaffold hopping in more details, summarily describe state-of-the-art approaches to solve these problems, and present what challenges remain. I will finally provide an overview of my thesis objectives and a summary of this manuscript to guide the reader.

1.1 Drug Discovery

Drug discovery is the search for molecular compounds of therapeutic interest to treat a specific disease. On average, this risky process usually takes around 10 years from inception to market entry, and may cost as much as several billion dollars [Hughes *et al.*(2011)]. It consists in 5 stages that we describe briefly in the following.

Target Identification. Initiating the drug discovery process involves pinpointing a protein that plays a role in a disease development, and whose functional modulation (for example, activation or inhibition) by drugs provides a clinical benefit. Available biomedical data, such as gene expression, proteomics data, or phenotypic screens are the standard procedures for identifying a protein target, which is further validated through a multi-validation approach involving for example *in vitro* tests, animal models or protein modulation in patients.

Hit Discovery. Subsequently, the focus shifts to an exhaustive exploration for small molecules capable of binding to the validated target, called "hits", marking a crucial step in the drug development pipeline. The chemical space, in which such compounds are searched, is extremely vast: it encompasses between 10^{30} and 10^{60} synthetisable molecules [Walters(2019)]. However, only a tiny portion of the chemical space has been explored *in silico*, and even fewer molecules have been isolated as natural compounds or synthesized. Large databases containing thousands to billions molecules have been gathered [Bento *et al.*(2014), Wang *et al.*(2009), Pence et Williams(2010), Zhao *et al.*(2020)] but their size is nowhere near the number of potential drug-like compounds. Although the chemical space accessible in these chemical libraries is limited, these resources still provide key starting points to search for molecules of therapeutic interest, because they usually contain molecules that are available, because they have been isolated or synthesized and characterized. More recently, advances in Deep Learning lead to the design of generative models that allow exploration of the chemical space [Elton *et al.*(2019)]. Such models are trained to learn the correct syntax of molecules, and fed with tailored rewards to navigate uncharted regions of the chemical space to search for molecules expected to display the bioactivity of interest.

In practice, strategies for finding hits fall into two categories:

- Experimental *in vitro* screening: lead by High throughput screening, providing

rapid assessment of molecules bio-activities, using assays allowing the screen of 100,000 molecules per day [Hertzberg et Pope(2000)]. Still, these assays remain expensive and impractical to screen large databases of millions of compounds. Therefore, virtual screening approaches have gained momentum in the last decades.

- Virtual screening (also called *in silico* screening), is often used prior to experimental screens. These approaches rely on computational methods that attempt to predict molecules likely to present the desired biological activity. These molecules can then be tested *in vitro*, with the goal of limiting the number of experiments to be performed to discover hit molecules.

Hit to Lead. The identified hits undergo additional validation, usually composed of confirmatory testing to ensure that the activity is reproducible, as well as biophysical testing to rule out promiscuous binding. Following hit validation, intensive structure-activity relationship (SAR) investigations are undertaken to optimize de chemical structure of the hits. Additional *in vitro* assays are performed to provide important information with regard to absorption, distribution, metabolism, excretion and toxicology (ADMET) properties, as well as physicochemical and pharmacokinetic (PK) measurements, to meet the criteria required to design a drug molecule that can be administrated to patients. The hits' selectivity is also evaluated against classical off-targets known to be responsible of deleterious side effect, such as hERG (KCNH2) for QT prolongation, α1A adrenergic receptor (ADRA1A) modulation for arrhythmia (agonists) or orthostatic hypotension (antagonists) [Sutherland *et al.*(2023)]. Those that do not meet the standard drug criteria (target potency, selectivity, ADMET and PK profiles) are discarded. The molecules that successfully passed these filters, called "lead" compounds, are further optimised as detailed in the next paragraph.

Lead Optimisation. Once a lead compound has been identified, an optimisation phase begins to improve the biological properties of the molecule: improved potency and reduced off-target activities with reasonable PK profiles. This stage usually consists in chemical modification of the lead structure by various techniques relying on the SAR.

Drug Development The drug candidates with optimal biological properties enter the final evaluation stage, consisting in pre-clinical tests, clinical tests, and the pharmacovigilance process (once on the market). In the pre-clinical step, the toxicity, the pharmacokinetics and the metabolism of the compound are evaluated on microorganisms and animals before conducting human trials in the clinical phase. The scarce chemical entities (success rate $\leqslant 10\%$ [Sun *et al.*(2022)]) validating all the pre-clinical tests and proved to provide statistically clinical beneficial effects on patients are allowed to enter the drug market, after approval of specific administrations, such as the Food and Drug Administration in the United States, "Agence nationale de sécurité du médicament" in France and the European Medicines Agency.

However, the journey from hit discovery to a clinical drug is loaded with various challenges. Out of the thousands initial candidates after compound screening, only,

if any, a dozen reaches the Drug Development phase [Sun *et al.*(2022)]. Such a low success rate can be explained by the lack of hits identified, their poor selectivity or ADME profiles, toxicity which is clinically unacceptable, or laborious, inefficient, or expensive synthesis routes, which restrains development at the industrial level. For all these reasons, a hit may not be viable, so that the exploration of the chemical space is required to identify a suitable drug candidate. Such issues can be encountered in any of the phases of the Hit Discovery to Drug Development journey.

Nevertheless, even when hits have to be discarded, they still provide fruitful information to guide the identification of novel compounds with similar biological activity against the target of interest. Novel compounds can be searched among molecules that are close to the hits, because such molecules are expected to still display a relevant activity with respect to the target. However, it is sometimes necessary to identify active molecules with highly different chemical structures. This quest is referred to as **scaffold hopping** [Schneider *et al.*(1999)], a critical obstacle often faced in Drug Discovery.

1.2 Definitions of Scaffold Hopping

Solving scaffold hopping consists in identifying novel chemotypes with a biological activity similar to that of a known active molecule. Though the concept of scaffold hopping might seem simple at first glance, this term has been used in different ways in the literature, and remains ambiguous [Hu *et al.*(2017)]. One definition involves preserving substituents (R-groups) that are engaged in interactions with the targeted protein pocket, while altering the molecule's core structure, also called scaffold [Hu *et al.*(2016)]. This typically involves the substitution of ring systems and linker fragments between rings with alternative molecular moieties. Medicinal chemists have formulated diverse approaches to pinpoint such novel scaffolds originating from a parent molecule, encompassing the exchange of carbons and heteroatoms within heterocycles, as well as the opening or closure of heterocycles. Additional contributions to the field involve the identification of entirely dissimilar molecules unrelated to the parent compound, lacking any definable common R-group or core structures. Thus, different degrees of scaffold hopping have been characterised by [Sun *et al.*(2012)] to depict how much the searched molecule needs to differ from the starting point. Three main classes of hops can be defined, as described in the following.

1.2.1 Degrees of Scaffold Hopping

Small-step Scaffold Hopping. In this case, subtle changes are made to connect fragments or substituents of a molecule, while preserving its overall scaffold. These modifications typically involve swapping of atoms, like carbon, nitrogen, oxygen and sulfur, or functional groups within heterocycles. Such a simple tactic can still improve the binding affinity if the changes are involved in interactions with the protein.

This small-step scaffold hopping strategy was applied to find Cannabinoid 1 (CB1) inhibitors [Sun *et al.*(2012)]. *Rimonabant* is an anorectic antiobesity drug targeting CB1 developed by Sanofi. However, it was unable to enter the United-States market because of safety concerns. This prompted AstraZeneca to search for novel antagonists

targeting CB1 with improved ADMET and PK profiles. Three novel hits were thus identified, with simple replacements of the core heterocycle [Boström *et al.*(2007)] as displayed in Figure 1.1.

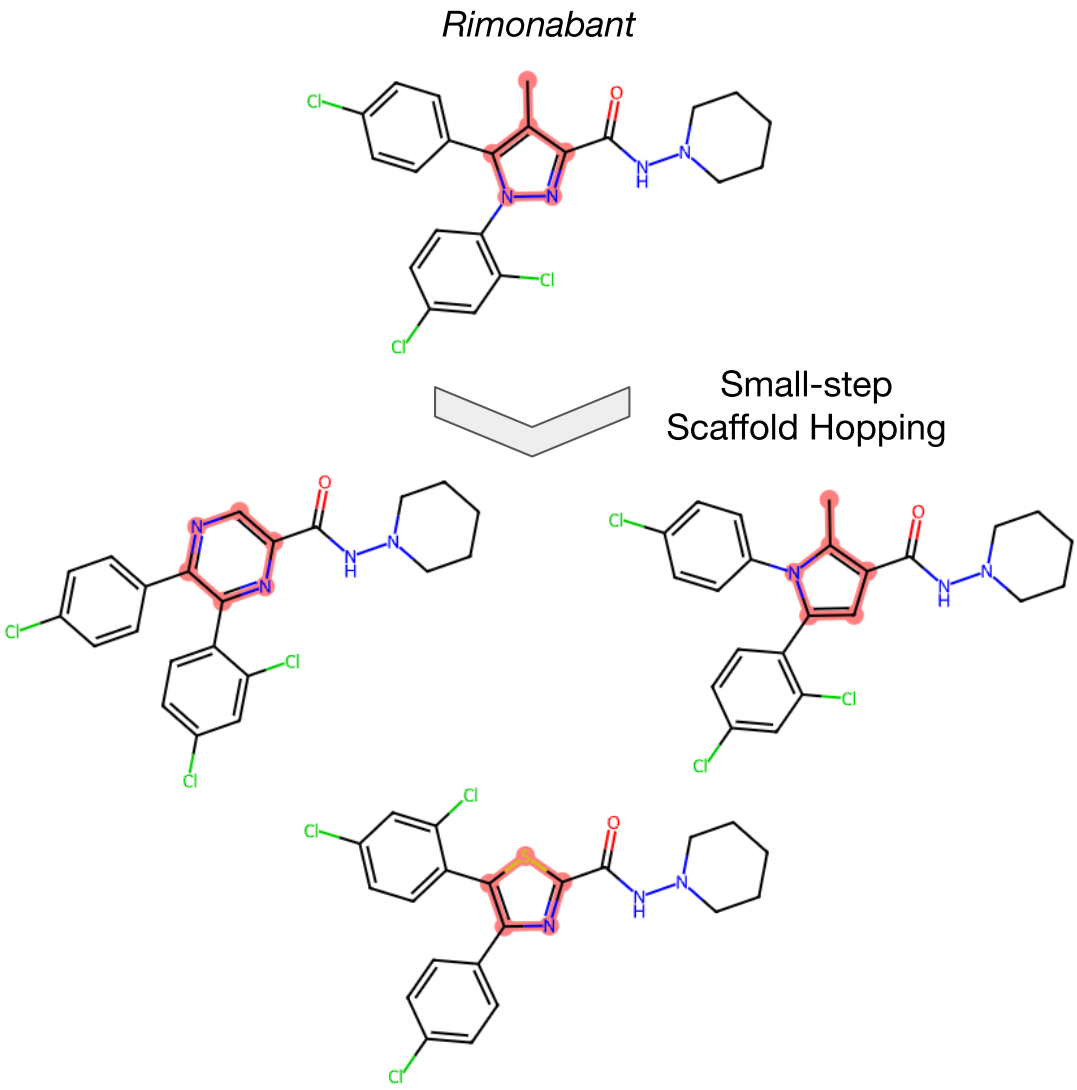


Figure 1.1: Example of small-step scaffold hopping. From *Rimonabant*, three novel inhibitors were discovered by swapping heteroatoms in the core ring highlighted in red.

Medium-step Scaffold Hopping. Ring opening and closure have an effect on the flexibility of molecules, which has a direct impact on membrane penetration and absorption [Vieth *et al.*(2004)]. This strategy provides a way to create novel scaffolds from existing molecules to overcome such limitations, though the synthetic feasibility of manipulating certain rings can sometimes be challenging.

Morphine and *tramadol* constitute a telling example of successful medium-step scaffold hopping. While *morphine* acts on the μ-opiod receptor to increase pain tolerance, it is well known for its adverse side effects like nausea or respiratory depression. This prompted for the search for a new drug, *tramadol*, which was basically obtained with

opening the fused rings of *morphine*, as showed in Figure 1.2. The flexibility thus achieved decreased both potency and side effects.

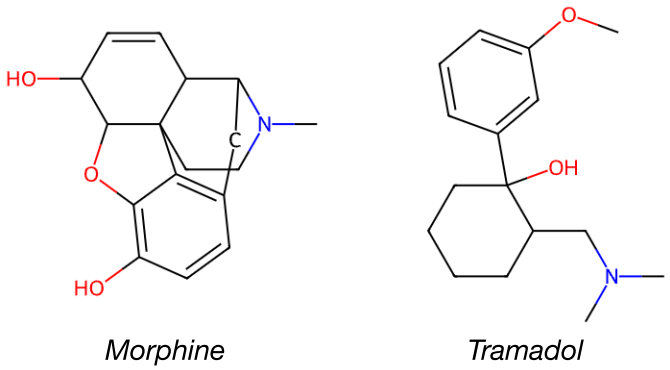


Figure 1.2: Example of medium-step scaffold hopping. By breaking six covalent bonds and three fused rings from *morphine*, a more flexible ligand, *tramadol*, was discovered.

Large-step Scaffold Hopping. Small- and medium-step strategies yield to novel compounds that still maintain a certain degree of similarity to the parent compound that is apparent when examining the chemical structures, and can be managed by a proficient medicinal chemist. Though this degree of similarity can be sufficient to overcome unacceptable ADMET and PK profiles, some cases need a larger jump in the chemical space. For example, when the scaffold of the parent molecule (and of its derivatives) is protected by a patent, it may be necessary to search for new molecules of totally different structure. This may also be required when the hit molecule presents unexpected deleterious off-targets, or when it cannot be purified or synthesised at the industrial scale. Overall, solving large-step scaffold hopping cases requires to search for a new molecule that shares very limited structure similarity with the original hit, and cannot be easily solved without the aid of computational methods.

Such methods can help the discovery of compounds with high chemical novelty compared to the parent compound, while still engaged in the same key interactions within the targeted protein pocket, in order to keep the biological activity of interest. Such pairs of molecules can be deemed as "isofunctional", and are also commonly referred to as large-step scaffold hopping cases. In the following, we will consider the concepts isofunctional molecules and large-step scaffold hopping molecules as synonyms.

Noteworthy examples of large scale scaffold hops include *Indomethacin* and *Etoricoxib*, both serving as structurally unrelated Cyclooxygenase-2 (COX-2) inhibitors [Böhm *et al.*(2004)]. Those nonsteroidal anti-inflammatory drugs display completely different chemical structures, as illustrated in Figure 1.3.

Examples of large-step scaffold hopping are rare in the literature [Sun *et al.*(2012)]. Identification of isofunctional molecules is a very challenging problem, and new efficient computational methods dedicated to solve such problems are eagerly required in the field of drug design.

The different degrees of scaffold hopping are summarized in Table 1.1 through ad-

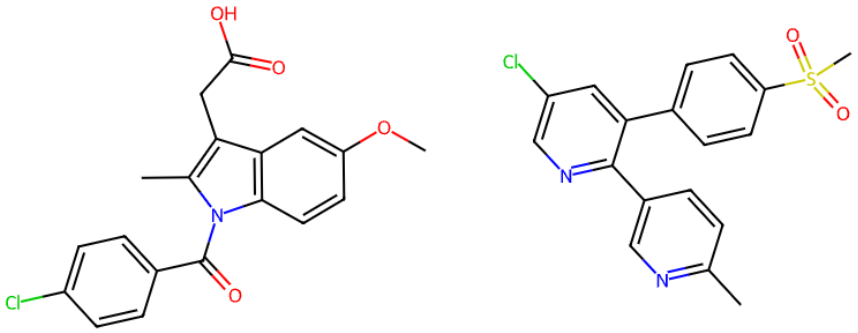


Figure 1.3: Example of large-step scaffold hopping. *Indomethacin* and *Etoricoxib* display no common substructure.

ditional examples.

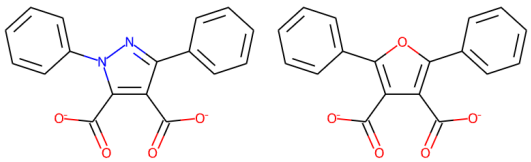
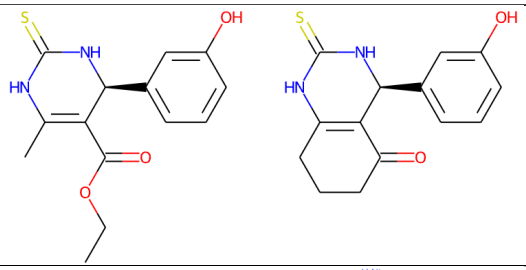
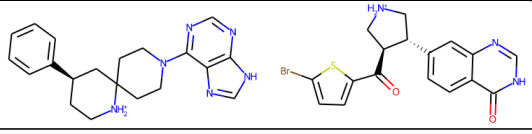
Scaffold Hopping Degree	Description	Example
Small-step	Change of atoms in heterocycles	
Medium-step	Ring opening and closure	
Large-step	Novel core structure	

Table 1.1: Summary of the three degrees of scaffold hopping.

1.2.2 Binding Modes Conservation

In the previous section, we only discussed about the degree of structural similarity between scaffold hopping molecules. However, another crucial aspect of scaffold hopping is the similarity of the binding modes shared by pairs of isofunctional molecules. Indeed, maintaining similar binding modes than the original hit compound ensures that the newly designed molecules maintain their activity with respect to the target protein.

However, very often, scaffold hopping exploration is experimentally performed based on biological assays that do not provide information about binding modes. Such experiments may identify hits with binding modes or binding location drastically different from those of the original compound (a situation we could call scaffold hopping "right for the wrong reasons"). This can be observed for example for competitive or allosteric enzyme inhibitors. In such scenario, we face a lack of functional commonalities between the initial hit and the new active molecule, and there is no biological/functional or chemical link between the initial hit to the newly identified molecule. Therefore, because there would not be any foundation to link molecules binding to different pockets, or presenting very different mechanisms of action (i.e. binding modes), in the present thesis, we only considered iso-functional molecules: molecules that share similar binding modes with the same protein pocket. This leads to exclude for example pairs of enzyme inhibitors, one binding to the active site and the other to a distinct allosteric site, or one being reversible and the other irreversible (forming a covalent bond withing the active site).

Indeed, there would be no rationale that could be exploited to discover the second molecule of such pairs, when starting from the first molecules of these pairs. I illustrate this property in Figure 1.4.

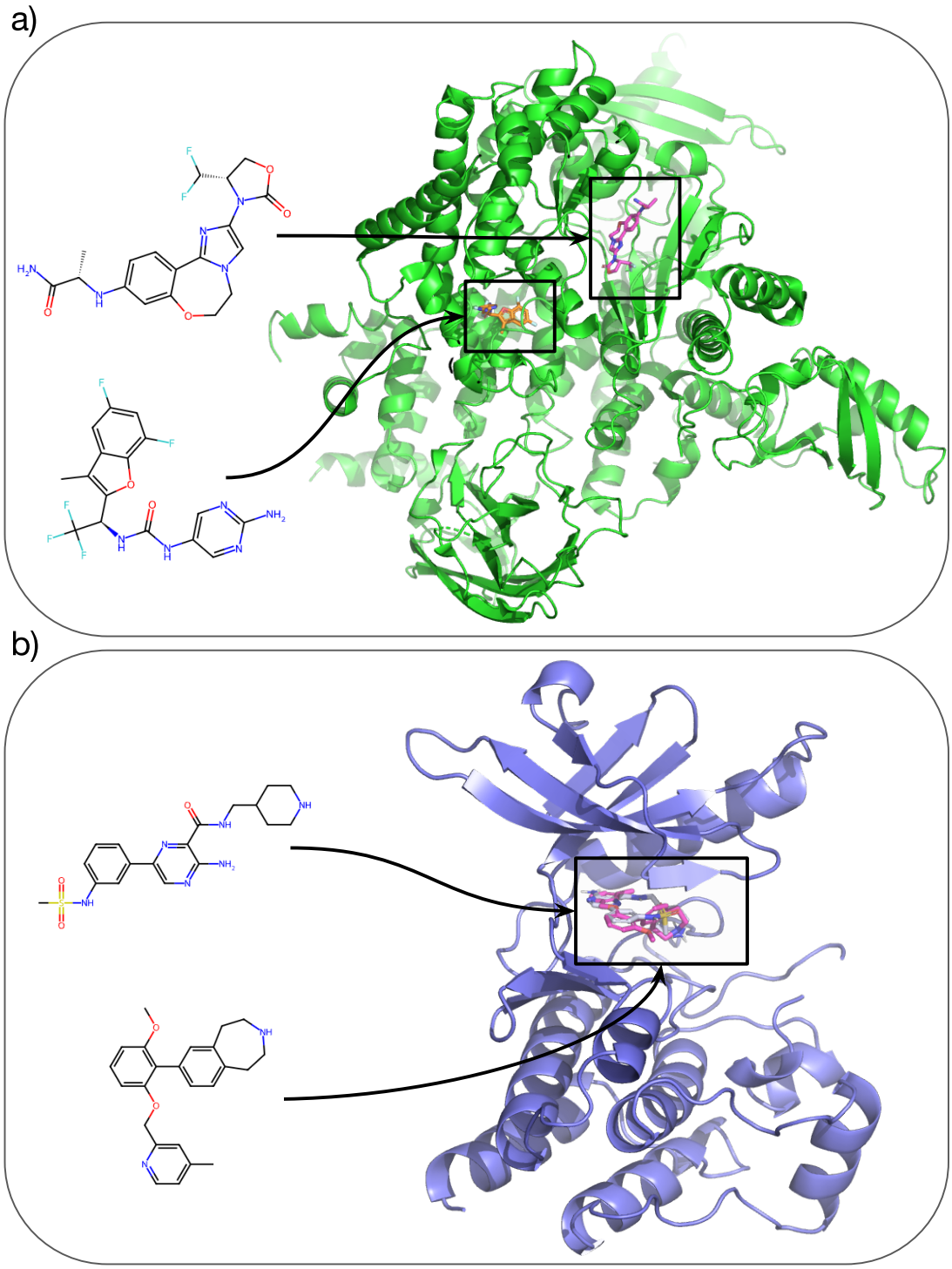


Figure 1.4: Illustration of the prominence of sharing similar binding modes for scaffold hopping. In panel a) we show dissimilar molecules targeting distant binding sites of PI3Kα. On the contrary, panel b) displays isofunctional molecules with similar binding modes against Spleen Tyrosine Kinase. Only the latter case corresponds to scaffold hopping, especially large-step scaffold hopping.

1.3 Solving Large-step Scaffold Hopping

When solving scaffold hopping, researchers aim to modify the core structure of a molecule, while preserving key interactions with the protein target. This can involve making structural changes such as replacing or modifying functional groups, altering ring systems, or even completely changing the molecular architecture while retaining essential pharmacophoric features. Due to the extensive size of the chemical space, computational methods play a crucial role in solving large-step scaffold hopping, by facilitating its exploration, predicting the biological activity of designed molecules, and guiding the selection of promising candidates for synthesis and testing.

However, the low number of successful examples of large-step scaffold hopping in the literature is a telling proof of how challenging it is, and underlines that new computational methods specifically tailored to address this category of problems need to be developed.

In principle, one can distinguish two categories of computational approaches for the discovery of isofunctional molecules: *structure-based* (SB) and *ligand-based* (LB) approaches.

1.4 Structure-based Approaches for Scaffold Hopping

As its name suggests, in the SB approach, identification of novel hits relies on the three-dimensional (3D) structure of the targeted protein. When the 3D structure of the target protein is available, docking is the standard approach for hit discovery and for solving large-step scaffold hopping. It has led to several successes documented in the literature [Pang *et al.*(2021), Kaplan *et al.*(2022)]. Indeed, the molecular mechanical equations on which it relies do not depend on the molecular structure, and should theoretically allow accurate scoring of molecules belonging to very diverse regions of the chemical space. In addition, docking facilitates rational drug design by providing insights into ligand-receptor interactions, allowing further optimization of ligand binding affinity and selectivity.

Determination of the protein 3D structure can be achieved either through experimental means such as X-ray Crystallography, Nuclear Magnetic Resonance Spectroscopy or Cryo-Electron Microscopy, or via computational methods like Homology Modelling or based on advanced Deep Learning models like AlphaFold [Jumper *et al.*(2021)].

Identification of isofunctional molecules with docking boils down to screening large molecular databases, and search for molecules that are highly dissimilar to the hit, while displaying similar interactions with the protein binding pocket. Among these molecules, those with the highest docking scores will be considered as candidate scaffold hopping molecules that should be experimentally tested. In the present thesis, docking has been used as a reference method to be compared to other approaches, but also to derive original molecular descriptors, as described in Chapter 3. Therefore, in the following, we recall the general principle of docking, and the steps than are required to implement a typical docking pipeline.

1.4.1 General Principle of Docking

Once the 3D structure of the protein target is available, *docking* is the standard approach for hit discovery, and has been applied since the 80s [Kuntz *et al.*(1982)]. Basically, its purpose is to fit a key (a molecule) into a lock (a protein binding site) using molecular mechanical equations, and scoring functions that reflect the estimated binding energy. Note that docking is not limited to predicting interactions between small molecules and a protein. Since the last decade, protein-protein docking, nucleic acid-ligand docking and nucleic acid-protein-ligand docking are now strategies handled by the different docking algorithms available in the literature (e.g. DOCK6, Vina, Gold, Glide, AutoDock). However, in the context of large-step scaffold hopping, we consider only protein-ligand docking, summarily referred to as docking in the following.

The docking process provides two outputs [Stanzione *et al.*(2021)]:

- The preferred orientation, conformation, and binding mode of a small molecule (ligand) within the binding pocket of the targeted protein, i.e. the 3D coordinates of the atoms of the molecule.

- The score of the pose obtained to assess its quality.

Pose Prediction. To predict the pose of a molecule inside a protein pocket, the docking algorithm must roam the conformational space of both the ligand and the protein. Due to the high number of degrees of freedom, exploring the search space exhaustively is not realistic. Several approaches have been developed to tackle the challenging problem of sampling conformations and rotational and translational orientations.

- Rigid docking: both the protein and the ligand are treated as rigid bodies. Thus, only the translational and rotational degrees of freedom of the ligand relative to the receptor are explored.

- Semi-flexible docking: only the flexibility of the ligand is considered, while the receptor remains rigid. Due to the ligand size, those computations are more likely to be affordable.

- Flexible docking: both the ligand and the receptor are treated as flexible. However, the conformational degrees of freedom of the latter can be limited to residue side chains.

For sampling ligand conformations, two families of methods are available. (1) Systematic methods are deterministic methods for conformation sampling. Some explore conformations by rotating all rotatable bonds in the ligand with a given interval, resulting in a huge number of combinations. Others, like the fragmentation method, divide the ligand into rigid fragments that are iteratively anchored to the protein binding site, before reconstruction of the molecule. (2) On the contrary, stochastic methods explore binding orientation and conformational space by applying changes on the ligand at random. The changes thus obtained are then accepted or rejected according to specific algorithms like genetic or Monte Carlo algorithms.

Pose Scoring. To assess the quality of the poses obtained after the sampling process, the docking algorithm predicts their binding affinity through internal scoring functions. Scoring functions can be grouped into four main classes [Li *et al.*(2019a)]:

- Physics-based functions compute the binding energy by including terms that account for various types of intermolecular interactions, such as electrostatic interactions, van der Waals forces, hydrogen bonding, and desolvation effects.

- Empirical functions also estimate the binding affinity of various energetic factors, but multiplied by coefficients determined by multiple linear regression on datasets gathering known binding affinities.

- Knowledge-based functions rely on statistical analyses of known protein-ligand complexes to derive empirical parameters that quantify the likelihood of specific interactions contributing to binding affinity. These scoring functions do not explicitly consider physical principles, but instead, leverage information from experimental data or structural databases to estimate the quality of ligand poses.

- Machine learning-based functions approximate those non-linear problems through Machine or Deep Learning algorithms trained on datasets gathering known binding affinities.

The successive steps that need to be performed in a typical docking pipeline are detailed below.

1.4.2 Typical Docking Pipeline

Docking requires a rigorous and exhaustive study of the protein as well as the ligand to calibrate the protocol. Failing to do so may result in a scenario where the poses predicted are meaningless and unrealistic. This is also why predicted poses are always analysed, either based on computational approaches or by medical chemists, to evaluate the relevance of the prediction. The docking workflow consists in six steps, described below [Stanzione *et al.*(2021)].

Protein and ligand selection. Docking relies on the 3D structure of the protein. Such structures can be found in the Protein Data Bank [Berman *et al.*(2000)] (PDB) in the `.pdb` file format. For simplification, we will employ the term "PDB" when referring to a `.pdb` file of a 3D structure. This database gathers more than 100,000 structures, but their quality differ from one another. For X-ray structures, a measure of this quality is provided by the crystal structure resolution. It quantifies the degree of order in the crystal, and to which extent the atom positions are defined. Typically, a crystal structure with a resolution below 2Å is considered of high-resolution, which translates in high confidence in the atoms locations in the structure. When trying to identify new hits for a protein using docking, an extensive prior study of all available 3D structures is needed. Only those with the best resolutions and with a clear defined ligand are considered for the following steps.

Protein and ligand preparation. Once a PDB has been chosen, a careful preparation step is performed on both the protein and the ligand. One common task is adding missing hydrogen atoms to entries in the PDB, which often lacks this information in X-Ray structures. This process is challenging due to various structural ambiguities, such as rotatable bonds affecting hydrogen atoms positions, tautomers, and protonation states of amino acids, as well as alternative water orientations and side chain flips. Additionally, protein preparation involves detecting and fixing missing bonds, assigning bond orders, and selecting atoms positions with the highest frequencies in cases of alternate locations. More complex procedures include predicting protonation states, which may play an important role in the prediction of the correct binding mode of a ligand.

Binding site identification. The binding site, also called pocket, corresponds to the area in the 3D structure into which the docking protocol is applied. It is prominent to accurately pinpoint the binding site to ensure that the molecular interactions considered during docking are significant. Ideally, the binding pocket is defined from the structure of the protein in complex with a known ligand, when available. Not only does this help calibrate the docking protocol as the binding site is clearly identified, but the knowledge of how this ligand interacts with the amino acids of the protein also provides valuable insights on the key binding mechanisms that a molecule is expected to reproduce to be considered a realistic candidate for further analysis.

In the absence such information, expert-knowledge or Deep Learning approaches like [Zhao *et al.*(2020)] can still be used to predict the location of the binding site.

Structural water molecules. Water molecules play a key role in structure-based drug design: they can improve protein-ligand binding affinity by mediating hydrogen bonds and contribute to entropic and enthalpic changes in the protein-ligand complex. Determining which water molecules should be retained in the protein structure is a mandatory step to calibrate the docking algorithm.

Calibration of the docking protocol. Docking itself, as detailed above, explores the conformational space of the ligand (and the protein in the case of flexible docking), and proposes poses that are ranked according to internal scoring functions. Like all algorithms, docking is error prone, and may produce irrelevant poses, and the docking protocol needs to be tuned to the problem of interest. Typically, re-docking experiments are performed to adjust the docking protocol. In practice, a known ligand for which a structure in complex with the targeted protein is available is redocked in the 3D structure of the protein, and the quality of the best scoring predicted poses are evaluated, with the aim that the position/conformation of the docked ligand matches those observed in the X-Ray structure of the protein-ligand complex.

A simple yet effective way to assess the quality of the pose of a known ligand is to compute the Root Mean Squared Deviation 1.1 (RMSD) of the crystallographic pose and the predicted pose of the ligand. This quantity measures how well aligned those

two conformations are.

$$\text{RMSD} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\mathbf{r}_i - \mathbf{r'}_i)^2} \tag{1.1}$$

where:

- RMSD is the root mean square deviation,
- N is the number of atoms in the molecule,
- $\mathbf{r}_i$ is the position vector of atom i in the reference structure,
- $\mathbf{r'}_i$ is the position vector of atom i in the compared structure.

If the RMSD is high (i.e. above 2 angströms), the true and predicted poses are not well aligned, because the docking protocol did not succeed in retrieving the experimental binding modes. In such cases, either the protocol needs to be adjusted to improve the predicted poses, or the PDB chosen for conducting such experiments is discarded in favor of another (if available), for which the re-docking experiment is successful.

1.4.3 Limits of Docking

When the resolution of the 3D protein structure is low, or when only apo structures (i.e. structures without bound ligands) are available, docking approaches may suffer from various limitations: it may be difficult to identify the binding pocket of interest, the apo structure may display structural rearrangements with respect to the unknown holo structure, so that the apo structure may not be reliable to perform docking. Furthermore, the 3D structure of the target may be unknown. Various approaches are available to predict the overall 3D models for proteins, including the efficient AlphaFold algorithm [Jumper *et al.*(2021)] which has revolutionized protein folding, leading to astonishing accurate 3D predictions only using the protein sequence. Nevertheless, these models may not be reliable at the level of structural details such as the orientation of side-chain or backbone loops, although these details are critical for the performance of docking approaches [Scardino *et al.*(2023)]. Docking requires precise knowledge, preparation and minimization of the protein pocket, as described previously, in order to provide accurate and meaningful predicted binding modes. Starting from a questionable protein pocket might decrease considerably the quality of docking predictions. Thus, the applicability domain of docking, though theoretically infinite in the chemical space, is limited in the protein space.

Therefore, docking is not always applicable to the scaffold hopping problem at hand, which leaves space for ligand-based approaches.

1.5 Ligand-based Methods for Scaffold Hopping

Ligand-based approaches in drug discovery refer to computational methods that analyze the chemical properties of known ligands, in order to identify new compounds with similar biological activities. These approaches do not rely on the structure of the target protein, but focus on the properties of the ligands themselves. Their general

principle is that structurally similar molecules are likely to have similar properties [Hendrickson(1991)]. Therefore, the prediction performances of ligand-based approaches strongly depend on the molecular properties that are considered, i.e. on the corresponding molecular descriptors that are considered to represent the molecules, and on the relevance of these descriptors with respect to the problem at hand.

In the following, we shortly review classical descriptors used in ligand-based approaches.

1.5.1 Molecular Representations

Various molecular representations encoding chemical properties have been designed and reported in the literature, with increasing levels of complexity: from simple 1D string-based formats such as the Simplified Molecular-Input Line-Entry System [Weininger(1988)] (SMILES), to molecular graphs [Dalby *et al.*(1992)] which are primary structural data, to feature-based formats that consist in vectors whose elements encode various molecular characteristics, and even computer-learned representations computed by neural networks [Jiang *et al.*(2021)]. We will not provide a more detailed description of the latter, because they have not been used in the present thesis, and because these representations strongly depend on the Deep Learning architecture that is used, so that they cannot be shortly reviewed.

The most commonly used molecular representations are molecular fingerprints encoding different types of descriptors. These representations have proved to help identify new hits [Dick et Cocklin(2020), Lovrics *et al.*(2019), Grisoni *et al.*(2018b), Nakano *et al.*(2021)]. They demonstrated that, even in the absence of information about the 3D structure of the binding pocket, ligand-based approaches still catch prominent information on binding and provide an interesting alternative to structure-based approaches. Figure 1.5 illustrates the information such encoded in the two prevailing molecular descriptors: the *Morgan Fingerprints* and the *3D Pharmacophore*.

Morgan Fingerprints Historically, the *Morgan Algorithm* [Morgan(1965)] was developed to solve the molecular isomorphism problem, i.e. identify cases where two molecules with different atom numberings are identical. It consists in an iterative process where numeric identifiers are assigned to each atom, from a rule encoding invariant atom information at first, then using the identifiers from the previous iteration. The iteration process stops when all atom identifiers are unique.

Recently, this algorithm lead to the design of a novel class of topological fingerprints: the Extended-connectivity fingerprints [Rogers et Hahn(2010)] (ECFPs), also called Morgan fingerprints. They consist in a binary vector encoding the different substructures present in the molecule structure. They are generated in three sequential phases analog to the Morgan Algorithm:

- An initial assignment phase in which all atoms are assigned integer identifiers,
- An iterative updating phase in which all atoms identifiers are updated to reflect the identifiers of each atom's neighbors, which is controlled by a diameter input, including identification of whether it is a structural duplicate of other features,

- A duplicate identifier removal phase in which multiple instances of identical features are reduced to a single entry in the final feature list.

Morgan fingerprints are often used to assess the 2D structural similarity between molecules, by computing their Tanimoto similarity:

$$T = \frac{c}{a + b - c} \tag{1.2}$$

where:

- T is the Tanimoto similarity coefficient,
- c is the number of common features (bits) between the two Morgan fingerprints,
- a is the number of features (bits) in the first Morgan fingerprint,
- b is the number of features (bits) in the second Morgan fingerprint.

Although these descriptors have proved to be successful in many applications in drug design, they do not appear to be suitable to the scaffold hopping problem. Indeed, Morgan fingerprints are tightly linked to the molecular graph, and therefore, searching for active molecules that are similar to a known hit based of Morgan fingerprints will not allow to identify molecules that display highly dissimilar chemical structures.

Pharmacophore descriptors A pharmacophore is a chemical feature that encodes information about whether a molecule can engage interactions with a protein. Typical pharmacophore features include hydrogen bond donors/acceptors, aromatic rings, positively or negatively charged groups, and hydrophobic regions.

Identifying the pharmacophores of a molecule and pinpointing which are involved in binding to a protein target can help design new molecules of therapeutic interest, which is the basis of pharmacophore modeling. Besides, identical pharmacophore features can represent different chemical structures. For instance, both phenyl and imidazole structures are aromating rings, thus will be encoded with identical pharmacophores. Incorporating the equivalence between such groups in ligand-based approaches when navigating the unknown chemical space is a crucial task for solving scaffold hopping, a property provided by pharmacophore approaches.

While 2D pharmacophore fingerprints have been described in the literature [McGregor et Muskal(1999)], the standard approach employs 3D pharmacophores. They rely on the generation of 3D conformers for each molecule. Several algorithms explore the conformational space of molecules and generate multiple energetically favorable conformations, like in RDKit [Landrum *et al.*(2021)]. Pharmacophore detection is then applied to these conformations in order to define the relative 3D positions of pharmacophore features present in the molecule.

Then, one possible way to compute the 3D pharmacophore similarity between two molecules is to align both conformers, and calculate the resulting Tanimoto similarity (as illustrated in the case of volume features):

$$T = \frac{V_{Overlap}}{(V_A + V_B - V_{Overlap})} \tag{1.3}$$

where:

- T is the Tanimoto similarity coefficient,
- $V_{Overlap}$ is the maximum volume overlap of the pharmacophores,
- V_A is volume of the pharmacophores of molecule A,
- V_B is volume of the pharmacophores of molecule B.

Although in principle, 3D encoding appears more relevant, because protein-ligand interactions are events that occur in the 3D space, when the active conformation of the ligand is unknown, 3D pharmacophore approaches may not reach the performances of 2D approaches [Mahé *et al.*(2006)].

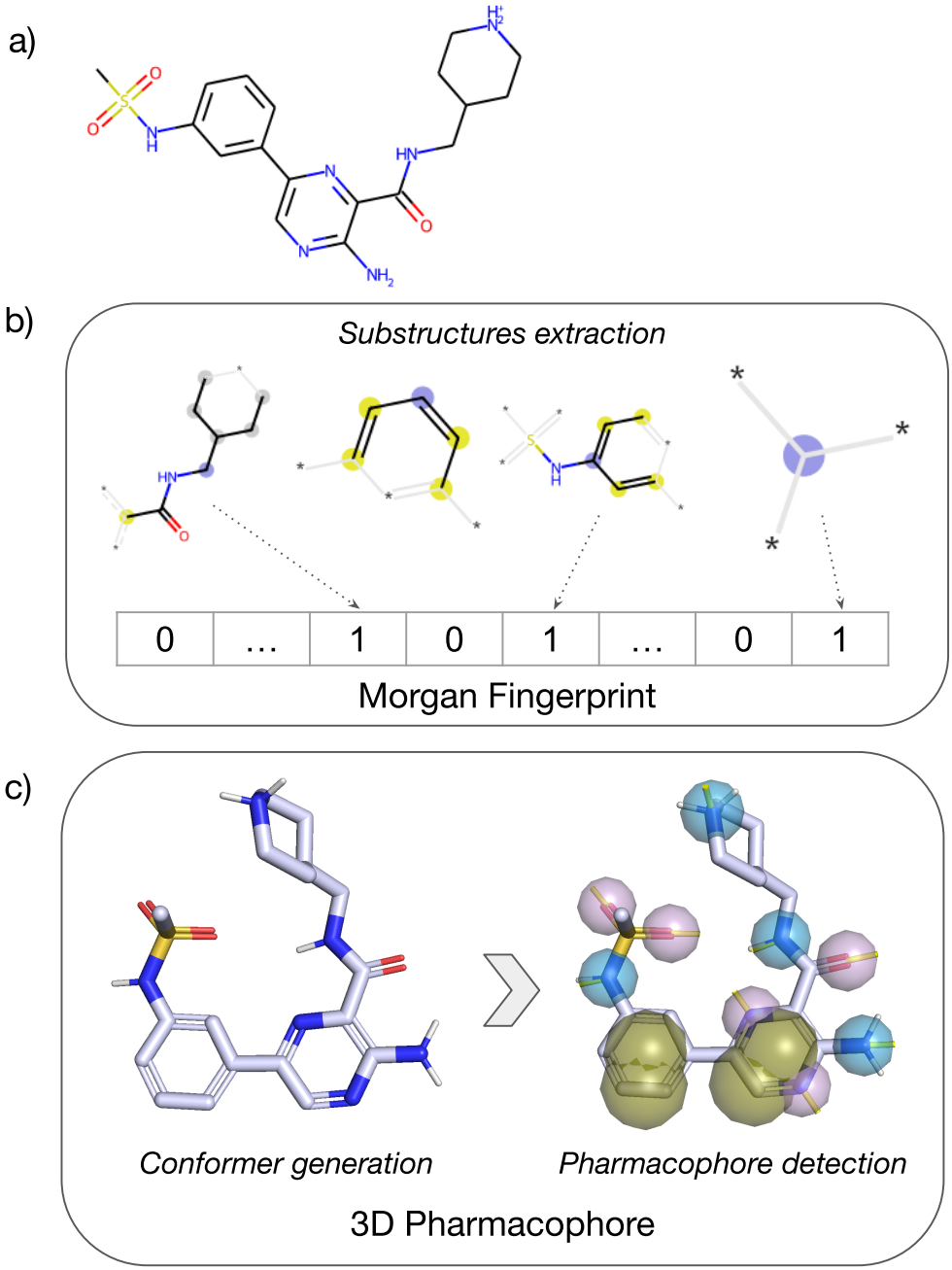


Figure 1.5: Example of two molecular representations. Substructural information of molecule in panel a) is encoded with the Morgan fingerprint b). This molecule can also be represented through its 3D pharmacophore c), which requires generation of a conformer.

1.5.2 Classical Ligand-based Algorithms

Quantitative Structure-Activity Relationship (QSAR) Models.

QSAR models correlate structural or/and physico-chemical properties of ligands, as encoded by their descriptors with their biological activities. By leveraging the relationship between these descriptors and the known ligands' activities, QSAR models aim at screening large molecular databases and predict the activity of new compounds, according to their descriptors. Globally, they rely on the idea that molecules with similar structures are expected to display similar biological activities, and in particular, bind to similar protein binding sites.

When the descriptors mainly encode the chemical structure of the molecules, such as Morgan fingerprints, QSAR approaches are not expected to be suited to identify new molecules that solve scaffold hopping problems. In fact, these approaches are more relevant at the optimisation step, when only subtle molecular modifications are searched, for example to optimise the ADME profile of a confirmed hit molecule.

As detailed above, pharmacophore descriptors capture the essential physico-chemical features that are present in a molecule and that govern protein-ligand interactions, such as hydrogen bond donors/acceptors, hydrophobic regions, aromatic rings and charged atoms. These descriptors don't depend on the chemical structure as tightly as Morgan fingerprints do, because various chemical groups can be associated to the same pharmacophores. For example, an O-H group or an N-H group can both be viewed as hydrogen donor pharmacophores.

Globally, QSAR approaches based on pharmacophore descriptors apply the idea that molecules with similar pharmacophores are expected to bind to similar protein pockets. Because different chemical groups may be represented by similar pharmacophores, in its principle, pharmacophore QSAR is expected to be a better choice to tackle the scaffold hopping problem, as illustrated in the case of [Carosati *et al.*(2007)].

Machine Learning Algorithms

Machine and Deep learning algorithms can be trained on known protein-ligand interaction databases, thus providing prediction models. They can then be used in virtual screening campaigns, to predict new ligands for a protein, and potentially to solve scaffold hopping problems. As for QSAR models, their performance for these problems is expected to strongly depend on the descriptors that are used to encode molecules.

However, one advantage of machine-learning algorithms is that they can be used in a multi-task setting, in which predictions of ligands for a given protein can be made based on all other ligands known for all other proteins. These algorithms allow to leverage much more information about any known protein-ligand interactions than QSAR models, because the latter can only take as input known ligands for the protein under study. We will not further study these approaches, and let the interested reader refer to the publication provided in Appendix A.

1.6 Formalisation of our Approach of the Scaffold Hopping Problem

As mentioned above, in the present thesis, we chose to tackle the problem of scaffold hopping specifically in the case where the 3D structure of the protein is not available, or not suitable for docking studies. In this context, LB approaches are the relevant computational methods to help solving such problems.

More precisely, we also focus on large-step scaffold hopping, the most difficult setting, where isofunctional molecule are searched. This means that one hit molecule is known, and a new molecule of dissimilar structure but that is expected to share the same binding mode within the binding pocket is searched. At this stage, this may seems to be an impossible mission where the 3D structure of the protein is unknown, but Chapter 3 presents the strategy that was used to overcome this problem.

In the previous sections, the choice of the molecular descriptors used as input of LB methods has a strong impact on their prediction performances. Currently used molecular descriptors were not specifically developed for scaffold hopping, and may not be optimal to solve these problems, even if, at this stage, pharmacophore descriptors appear to be a reasonable default choice. However, we reasoned that developing new molecular descriptors dedicated to scaffold hopping would have a stronger impact to solve these problems than fine-tuning available prediction algorithm for these problems. Indeed, the principle behind any computational method will be to implement the idea that "similar molecules will bind to similar protein pockets", and therefore, the performances will critically depend on how this "similarity" is measured, i.e. on the underlying molecular descriptors used to encode molecules.

Formally, our representation of solving large-step scaffold hopping problems boils down to the identification of molecular descriptors corresponding to an embedding of the chemical space in which two molecules that are examples of large-step scaffold hopping cases are similar (i.e. close), although they are dissimilar (i.e. far) in the space embedded by molecular descriptors based principally on the chemical structure. In other words, once such descriptors have been designed, given a hit molecule, solving scaffold hopping problems is equivalent to search for candidates that are close to the hit in the chemical space resulting from this embedding.

1.7 Goals and Manuscript Summary

1.7.1 Goals

The main goals of the thesis are presented below.

Design chemical descriptors specifically tailored to the problem of scaffold hopping. As mentioned above, most classical molecular descriptors are not adapted to solving large-step scaffold hopping problems, because they mainly rely on the chemical structure of molecules, while isofunctional molecules should lie in remote regions of the embedding of the chemical space resulting from such descriptors.

In this context, molecular representations based on biological properties are expected to be better suited to large-step scaffold hopping. Several encodings have been

described in the literature [N. Muratov *et al.*(2020), Wassermann *et al.*(2015)]. Some, like CBFP [Xiong *et al.*(2021)], encode predicted activities for a profile of assays, thus defining bioactivity fingerprints for molecules. However, many of these bioactivity fingerprints need to be predicted for most molecules, because the corresponding assays were conducted on a limited number of molecules, and the corresponding prediction models may lack generalisation properties. Consistent with this remark, pre-training a convolution neural network that predicts protein-ligand interactions based on the PCBA dataset that contains a profile of 90 bioactivities for thousands of molecules, did not improve the prediction performances of the algorithm [Playe et Stoven(2020)]. The present thesis will therefore explore new types of bioactivity fingerprints.

Build a large-step scaffold hopping benchmark to provide a panel of cases on which the chemical descriptors can be evaluated. As a matter of fact, a few large-scale benchmark studies have been reported, comparing the performances of various topology-based (or other ligand-based) methods [Grisoni *et al.*(2018a), Nakano *et al.*(2020)]. However, they only considered proteins with a relatively large number of known ligands, so that these ligands can be used to train prediction models. In addition, they evaluated the performances based on the chemical diversity of known ligands retrieved among the top ranked molecules. This does not clearly specify to which extent the structures of retrieved ligands were distant from those of molecules in the training set, which prevents from drawing conclusions about the ability of these methods to specifically solve large-step scaffold hopping problems. Finally, because of their design, these benchmarks do not mimic real-life applications, where an active hit has been identified (or a small number of hits), and where a new active with very different chemical structure is searched. Hence, these benchmarks do not allow to anticipate the performances of the methods proposed in these studies in the general case. In addition to the degree of chemical novelty searched, one must distinguish 'easy' targets with many known ligands, and 'hard' targets with only one known ligand. Most available computational methods may fail on the latter [Bajorath(2019)]. Overall, one of the main challenges in the field of computational methods for scaffold hopping is the lack of appropriate benchmarks to evaluate those methods on 'hard' targets and large-step hops, because these settings are typically encountered in the design of new drugs, and correspond to the most difficult cases.

Propose a strategy to evaluate the relevance of these encodings to that problem. Most studies reporting large step scaffold hopping success cases using 2D or 3D ligand-based approaches considered a given protein under study, or a very small number of proteins [Dick et Cocklin(2020), Lovrics *et al.*(2019), Grisoni *et al.*(2018b), Nakano *et al.*(2021)]. These cases corresponds to unique stories with extensive prior knowledge, and *ad hoc* procedures to solve these specific cases. In fact, the performances of molecular representations and of computational methods for scaffold hopping problems is essentially unpredictable [Sun *et al.*(2012)] in the general case, and there is a crucial need to design benchmarks that span a variety of proteins, to improve performance evaluation of these approaches.

1.7.2 Manuscript Summary

In Chapter 2, I present the design a benchmark for large-step scaffold hopping that can be used to evaluate new molecular encodings for their use in computational approaches. This contribution lead to a publication that illustrates how difficult scaffold hopping is, and why new approaches are needed.

In Chapter 3, I introduce new molecular descriptors based on possible binding modes of molecules, called the Interaction Fingerprints Profile (IFPP). These descriptors are intended to encode bio-activity of molecules, and to implement the idea that molecules belonging to scaffold hopping pairs are expected to be similar based on this encoding. I detail the rationale behind this representation, and explain why it may be relevant for solving scaffold hopping. The interest of this representation is evaluated on the proposed benchmark. However, the promising results of this new representation of the chemical space needs to be nuanced with regard to its cost in terms of computation time.

In Chapter 4, I propose to use Deep Learning approaches to avoid the explicit computation of molecular IFPPs, while keeping in mind the idea that isofunctional molecules share similar IFPPs. We first show that predicting these IFPPs with Deep Learning approaches greatly reduces their computation time, but at the cost of degrading their quality because scaffold hopping pairs are "farther" in the resulting chemical space derived from predicted IFPPs than with the actual computed IFPPs.

Therefore, we then propose an alternative method based on Metric Learning and that appears far more promising. Interestingly, this approach directly predicts the IFPP similarity of molecules, without explicitly computing the molecules IFPPs. In other words, it directly implements the idea of searching scaffold hopping candidates for a given hit molecules among molecules that are close in the embedding space, without calculating the corresponding descriptors.

The performances of this approach is evaluated on both the scaffold hopping benchmark and on an external dataset, LIT-PCBA [Tran-Nguyen *et al.*(2020)], providing a glance of its interest in realistic virtual screening settings.

Finally, in Chapter 5, I summarize the results of the thesis, and provides future perspectives in the field of scaffold hopping.

2
Large-Hops Benchmark

Abstract:

In this Chapter, I detail the approach adopted to build the Large-Hops benchmark (LH), designed for problems of large-step scaffold hopping. Pairs of isofunctional molecules are gathered from PDBbind by selecting ligands targeting the same protein with dissimilar molecular structures but similar binding modes that are not solely explained by a common substructure. Then, I propose a strategy to evaluate molecular descriptors using the benchmark. It relies on the choice of appropriate decoy molecules for each pair, possessing similar global physical and chemical properties to those of the ligands, while being as distant from each ligand of the pair as these ligands are from each other, in terms of chemical structure. Classical 2D and 3D molecular descriptors are evaluated using this criterion, and display limited performances. Finally, I illustrate how this benchmark can also be used as test set for chemogenomic approaches.

Résumé:

Dans ce Chapitre, je détaille l'approche adoptée pour construire le benchmark Large-Hops (LH), conçu pour les problèmes de 'large-step scaffold hopping'. Des paires de molécules isofonctionnelles sont rassemblées à partir de PDBbind en sélectionnant des ligands ciblant la même protéine avec des structures moléculaires dissemblables mais des modes de liaison similaires qui ne sont pas uniquement expliqués par une sous-structure commune. Je propose ensuite une stratégie d'évaluation des descripteurs moléculaires à l'aide du benchmark. Elle repose sur le choix de molécules 'decoy' appropriées pour chaque paire, possédant des propriétés physiques et chimiques globales similaires à celles des ligands, tout en étant aussi éloignés de chaque ligand de la paire que ces ligands le sont l'un de l'autre, en termes de structure chimique. Des descripteurs moléculaires 2D et 3D classiques sont évalués à l'aide de ce critère et affichent des performances limitées. Enfin, je montre comment ce benchmark peut également être utilisé comme ensemble de tests pour les approches chémogénomiques.

Contents

Our main contribution in this chapter is to provide a flowchart to build a high-quality and well characterized large-step scaffold hopping benchmark for 'hard' targets, which is a prerequisite to develop and test new methods dedicated to these problems. We also propose a strategy to compare the performance of molecular descriptors for solving scaffold hopping, using this benchmark. Specifically, we illustrate this strategy for a few classical 2D and 3D molecular descriptors. In addition, we show that this benchmark can also be used as a test dataset to evaluate the performances of chemogenomic algorithms to solve scaffold hopping problems. This allows us to evaluate the difficulty of large-step scaffold hopping problems in a setting that corresponds to real-case studies.

These results are published in the article [Pinel *et al.*(2023)] and available online at `https://github.com/iktos/scaffold-hopping`:

P. Pinel, G. Guichaoua, M. Najm, S. Labouille, N. Drizard, Y. Gaston-Mathé, B. Hoffmann, V. Stoven (2023), *Exploring isofunctional molecules: Design of a benchmark and evaluation of prediction performance*, Molecular Informatics **42** (4), 2200216. doi:10.1002/minf.202200216

2.1 Building the Large-Hops Benchmark

As detailed in 1.6, the field of scaffold hopping lacks of a well characterised benchmark of large step scaffold hopping cases for a panel of diverse proteins.

Although the present thesis tackles the problem of scaffold hopping for proteins of unknown 3D structure, we built this benchmark from examples extracted from the PDBbind database [Wang *et al.*(2004b)] to ensure that the selected pairs of molecules are 'true' large-step scaffold hopping cases, i.e., highly dissimilar compounds that share similar binding modes with the same protein, as identified by the same UniProt ID. Indeed, as already pointed, there would not be any rationale to relate two inhibitors of an enzyme binding to two distinct and distant binding sites, and such 'false' scaffold hopping cases must not be present in the benchmark. Identification of such examples is not straightforward: some examples presented below show that it is not possible to use only one criterion based on a single molecular similarity measure. The next subsections present the subsequent steps that are used to perform this task.

2.1.1 Identifying Molecules with Dissimilar Structures

Filtering the PDBbind database

To identify scaffold hopping cases, we need to search for pairs of highly different molecules that bind to the same protein pocket with similar binding modes, because molecules that would present totally different binding modes in the same pocket, or bind to different pockets of the protein, do not meet the definition of scaffold hopping. To enable the selection of such pairs, we use the PDBbind database [Wang *et al.*(2004b)] that contains 17.652 PDB files (2019) of 3D crystallographic structures of protein-ligand complexes. We only keep structures with a resolution below 2.8 Å, to ensure that the binding modes of the ligands can be analysed with confidence. Second,

as some compounds can be co-crystallized with the protein in soaking experiments, even with unspecific affinities in the millimolar range, we remove all complexes with affinity above 10µM. This allows to only select 'successful' scaffold hops, for which both molecules present specific activities against the same target. Finally, we discard proteins for which only one ligand is available in PDBbind, since scaffold hopping examples cannot be searched for these proteins. This leads to 181.635 pairs of ligands for 997 proteins. Examples of large-step scaffold hopping cases are further searched among these pairs.

Selecting pairs of drug-like molecules

Because our study stands in the context of drug design, the selected molecules need to represent molecular characteristics encountered in drug-like molecules. Otherwise, the performances of computational methods on our benchmark may not be representative of those expected in drug design applications. We only keep pairs involving ligands of molecular weight between 200 and 900g/mol, to discard salts, solvent or other molecules present in crystallisation buffers, and large interacting partners like peptides. This leads to 6.494 PDB files, corresponding to 148.002 pairs of ligands and involving 856 different proteins. Among the 148.002 pairs, we select those in which both molecules have a quantitative estimate of drug-likeness (QED) [Bickerton *et al.*(2012)] above 0.5, which allows to remove molecules with unwanted physical-chemical properties, leading to 49.686 pairs involving 449 proteins.

Selecting pairs of large-step hops ligands

Among the 49.686 pairs of molecules, we use several criteria to exclude those corresponding to small- or medium-step hops cases. First, we determine the generic Murcko scaffolds of molecules because they characterize the core structure of molecules [Bemis et Murcko(1996)]. These scaffolds are obtained by removal of all substituents, while retaining ring systems and linker moieties between rings, and converting all bonds to single bonds. To remove small-step hops, we exclude pairs of ligands whose generic Murcko scaffolds have Morgan fingerprints Tanimoto similarities [Rogers et Hahn(2010)] above 0.6 (in the following, this similarity is called Murcko-based Morgan similarity), which selects 45.534 pairs. This single criterion does not always guarantee that the two molecules are highly dissimilar: in a few cases, they still display significant similarities, as illustrated in Figure 2.1 for one pair. To discard these cases, pairs of molecules with an overall Morgan fingerprints Tanimoto similarity above 0.3 are removed (in the following, this similarity is called molecular Morgan similarity), as they may represent medium-step hops rather than large-step hops. This leads to 44.386 pairs of molecules.

2.1.2 Identifying Molecules with Similar Binding Modes

Molecular Interactions

Before detailing how scaffold hopping pairs are selected, it is necessary to thoroughly describe how a molecule can bind to a protein target. Indeed, scaffold hopping molecules have similar binding modes, so in order to verify this property, we must be able to encode how a ligand interacts with a protein.

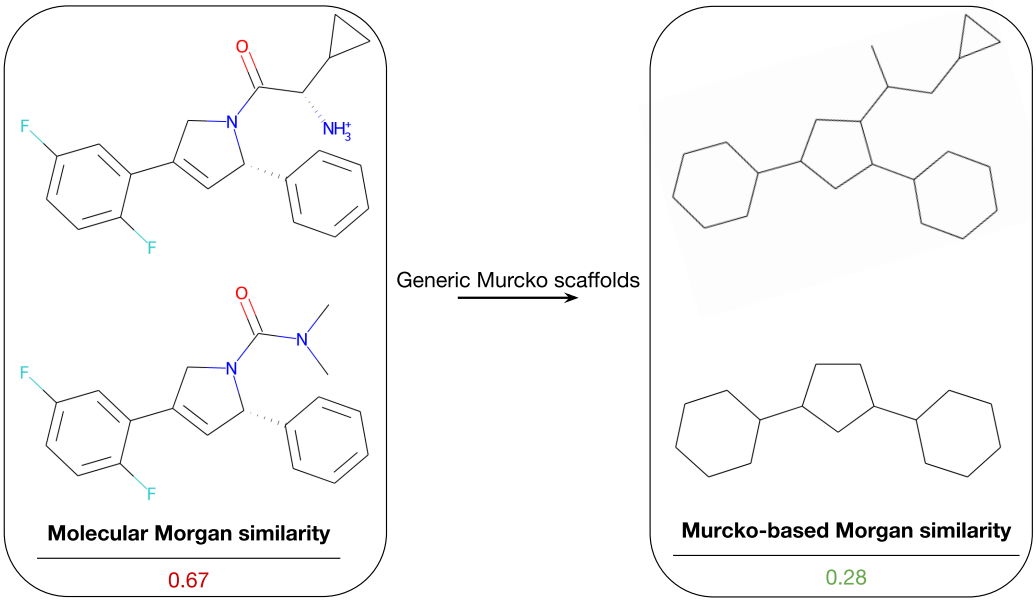


Figure 2.1: Example of a pair of molecules with low Murcko-based Morgan similarity but similar structures, leading to a higher molecular Morgan similarity. On the left the pair of molecules (PDBs: '2fl2' and '2fl6') is displayed and, on the right, their corresponding generic Murcko scaffolds are shown. This pair should not be present in the *LH* benchmark. It is excluded based on the molecular Morgan similarity between the molecules greater than the chosen threshold.

Protein-ligand interactions are reversible non-covalent interactions that do not involve the sharing of electrons [Bongrand(1999)]. We present the molecular interactions we considered when characterising ligand binding modes.

Hydrogen Bond. This interaction requires one hydrogen (H) bond donor, i.e. a polar hydrogen covalently bonded to an electronegative atom, and one H bond acceptor, an atom exhibiting a partial negative charge. However, this interaction is dependent on the geometry of the interacting atoms [Nittinger *et al.*(2017)]. Indeed, the hydrogen must be directed towards the lone pair of the H bond acceptor, which conversely must be directed towards the most polarized region of hydrogen. Variations of the angle between the H bond donor and acceptor have an important impact on the energy of the interaction [Li *et al.*(2011)].

Many studies describe hydrogen interactions as being either weak or strong depending on their energy estimation [Nittinger *et al.*(2017)]. Strong hydrogen bonds correspond to interactions whose energy is close to the optimal value. Weak hydrogen bonds result from imperfect geometry, a weak bond acceptor like sulfur, an aromatic cycle, or moderate polarization of the hydrogen bond donor.

Halogen Bond. Halogens are atoms belonging to the 17th group of the periodic table: fluorine (F), chlorine (Cl), bromine (Br) and iodine (I). They are covalently bonded to only one atom. For Cl, Br and I, there is in the extension of this bond, on the halogen, an electropositive zone called the σ-region, due to the anisotropic

distribution of electrons. The σ-region can interact with Lewis bases which can be hydrogen bond acceptors or other atoms with an electronegative moment, creating the halogen bond. In terms of geometry, the halogen bond behaves similarly to the hydrogen bond [Nittinger *et al.*(2017)].

Salt Bridge. Salt bridges correspond to any interaction involving an entity carrying a positive or negative charge. Salt bridges can sometimes be assimilated to hydrogen bonds in cases where a hydrogen bond donor faces an entity carrying a negative charge. Ionic bonds are generally identified as two elements carrying opposite charges in contact. The strength of this interaction is highly variable and depends both on the elements involved and the distance.

Multipolar. Similar to halogen bonds, multipolar interactions involve halogen atoms and carbonyl carbon or amide nitrogen [Paulini *et al.*(2005)]. These interactions entail favorable dipolar interactions between a C-X group (primarily with fluorine) and an electrophilic center, such as the amide group in the backbone or side chain of proteins. Rather than approaching the negatively polarized center in a head-to-head manner, the C-X interacts orthogonally to the carbonyl group.

π-Stacking. π-Stacking interactions involve two aromatic rings, i.e. rings having $4n + 2$ delocalized electrons, n being an integer starting from 0. The most common value is 1, for 6-membered aromatic rings. The electrons are evenly distributed along the alternation of single and double bonds of the ring. This phenomenon results in the creation of a specific dipole around the aromatic ring. The two electron-rich regions, called π regions, are located on either side of the plane of the aromatic ring and are centered on its center of mass. The periphery of the aromatic ring will be essentially positively charged due to the low density of the electron cloud. The two aromatic rings can interact with two different geometries:

- Face-to-face: both aromatic rings are parallel, with eventually a small offset between them.
- Edge-to-face: both aromatic rings are perpendicular.

Cation-π. This electrostatic interaction is created when the negatively charged electron cloud of a π system meets a positively charged electron cloud of a cation [Ferreira de Freitas et Schapira(2017)].

Amide-π. This interaction corresponds to when the π-surface of the amide bond stacks against the π-surface of the aromatic ring [Harder *et al.*(2013)]. The $sp2$ orbital of the carbon of the amide perfectly fits the electron-rich region of the ring.

Hydrophobic-π. Hydrophobic-π interactions occur between aromatic rings or other π-electron-rich systems and hydrophobic groups or surfaces. In these interactions, the delocalized π-electron cloud of the aromatic ring interacts with the hydrophobic environment, such as non-polar residues in proteins or hydrophobic regions on the surface of molecules.

Hydrophobic. Hydrophobic regions tend to avoid contact with water molecules due to the unfavorable energy associated with disrupting the hydrogen bonding network of water. As a result, hydrophobic molecules or regions tend to aggregate or associate with each other to minimize their exposure to water, leading to the formation of hydrophobic interactions. Any group is generally considered to be hydrophobic if it is apolar and/or alkyl. Since we distinguish hydrophobic-π interactions from hydrophobic interactions, aromatic rings are not considered in this definition.

We illustrate all those interactions in Figure 2.2.

Interactions Fingerprints

Interaction fingerprints (IFPs) are used to encode the binding modes of ligands. These fingerprints are target-focused binary vectors that incorporate, for each protein residue in the binding site, its interactions with the ligand. Bits are allocated for each residue, each encoding for the presence of one type of interaction with the ligand. To build those IFPs, detection of the protein-ligand interactions is needed.

Various binary target-focused protein-ligand interaction fingerprints have been proposed in the literature [Marcou et Rognan(2006),Chupakhin *et al.*(2014),Da et Kireev(2014), Salentin *et al.*(2014)]. They are an easily interpretable way to encode binding modes. However, they lack a few key interactions, which led us to develop our own tool to detect and encode them.

Starting from PLIP [Salentin *et al.*(2015)], a freely available algorithm that detects such interactions, including hydrogen bond, weak hydrogen bond, halogen bond, salt bridge, hydrophobic, pi-cation, and pi-stacking, we built an extended version adding several interactions [Bissantz *et al.*(2010), Freitas et Schapira(2017), Shinada *et al.*(2019),Kuhn *et al.*(2019),Nittinger *et al.*(2017)] that are missed by classical IFPs: hydrophobic-π, amide-π and multipolar. In particular, I optimized the interaction detection to make it faster. For this purpose, I used a Ball Tree algorithm [Liu *et al.*(2006)] to store the coordinates of the protein atoms. It organizes data points in a hierarchical tree structure, facilitating efficient nearest neighbor search, and range query operations in multidimensional space. This allows to use less memory, but more importantly, it enables us to quickly determine the protein atoms close to the ligand atoms. It allows to define the binding site of the protein, and to search for interactions only on this reduced list of atoms, which was then the most time-consuming step in the calculation process. Our tool made the interaction detection step five-time faster than PLIP. We released the code used to detect these interactions at:

`https://github.com/iktos/structure-interactions`

The detection criteria are described in Table 2.1. The thresholds used here are less restrictive than those of the original package, because we want to avoid missing the interaction detection in low resolution PDB structures. The interaction fingerprint associated to a given protein target is of fixed size for all molecules. Specifically, this size corresponds to the number of residues in the binding site times the number of considered interactions (10). The binding site is defined by all residues having at least one atom within a radius of 10 Å from the crystallographic ligand of the protein.

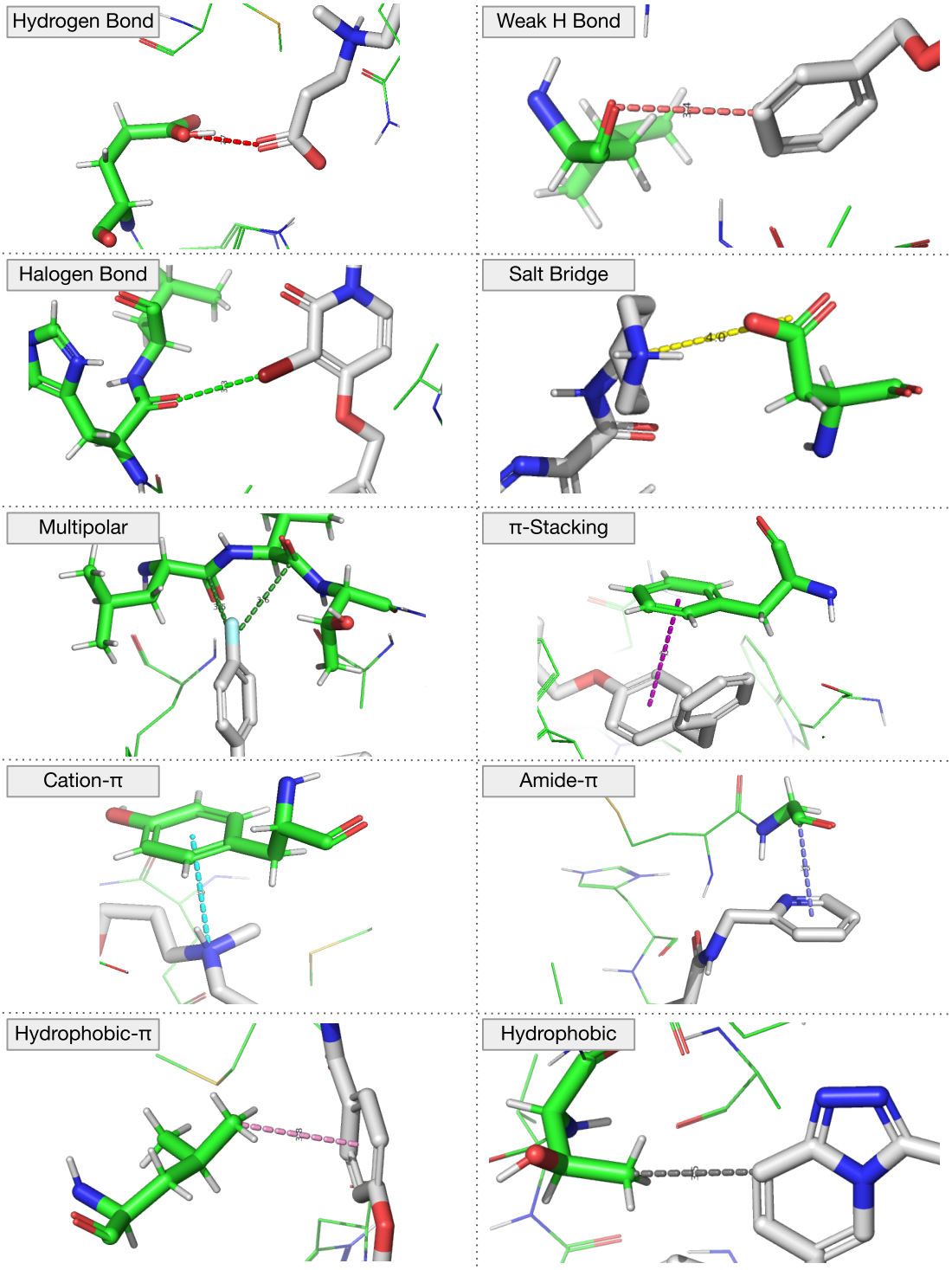


Figure 2.2: Illustration of the ten considered interactions. Those interactions were retrieved from analysing several PDBs ('1zyl', '1zzl', '3fhe', '3hp2', '4dfl'). The proteins are colored in green, and their natural ligands in white. For the sake of visibility, only the interacting residues and ligands are represented with sticks. The molecular interactions are represented through dash lines, colored according to the type of interaction.

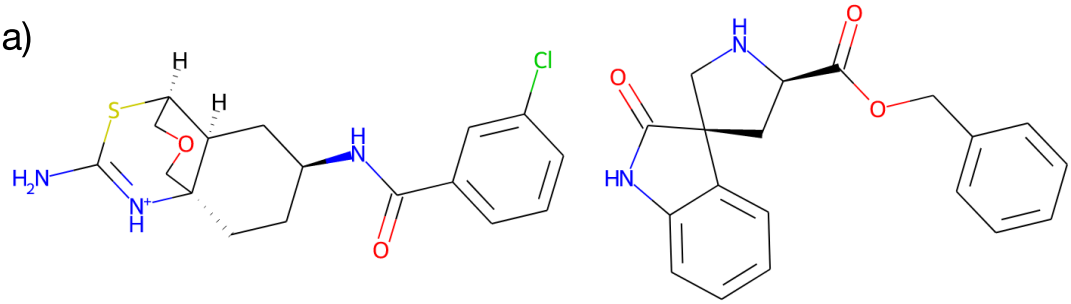
Interaction	Ligand	Protein	Distance	Angle
H-bond	H-bond donor H-bond acceptor	H-bond acceptor H-bond donor	$d_{D\ldots A} \leqslant 4.2$ Å	$\theta_{A\ldots H-D} \geqslant 130°$ $\theta_{R-A\ldots D} \geqslant 90°$ where R = A's neighbours
Weak H-bond	Weak H-bond donor Weak H-bond acceptor	Weak H-bond acceptor Weak H-bond donor	$d_{D\ldots A} \leqslant 4.0$ Å	$\theta_{A\ldots H-D} \geqslant 140°$ $\theta_{R-A\ldots D} \geqslant 90°$ where R = A's neighbours
Halogen bond	X-bond donor X-bond acceptor *sp2*	X-bond acceptor *sp2* X-bond donor	$d_{X\ldots A} \leqslant 4.5$ Å	$\theta_{AXD} \geqslant 120°$ $\theta_{RAX} \geqslant 90°$ where R = A's neighbours
Salt bridge	Anion Cation	Cation Anion	$d_{cation\ldots anion} \leqslant 5.5$ Å	-
Multipolar	X-bond donor Polar C *sp2*	Polar C *sp2* X-bond donor	$d_{X\ldots Csp2} \leqslant 4.5$ Å	$\vartheta \geqslant 70°$ where ϑ: angle between C-X and $X\ldots Csp2$ $\vartheta \leqslant 60°$ where ϑ: angle between $X\ldots Csp2$ and normal to amide plan
π-stacking	Aromatic ring	Aromatic ring	d $\leqslant 5.5$ Å offset $\leqslant 2.5$ Å	$0° \leqslant \vartheta \leqslant 30°$ ->parallel $60° \leqslant \vartheta \leqslant 90°$ ->T-shaped $30° < \vartheta < 60°$ ->face-to-face, face-to-edge
Cation-π	Aromatic ring Cation	Cation Aromatic ring	d $\leqslant 4.5$ Å offset $\leqslant 2.5$ Å	-
Amide-π	Aromatic ring Polar C *sp2*	Polar C *sp2* Aromatic ring	d $\leqslant 4.5$ Å offset $\leqslant 2.5$ Å	$0° \leqslant \vartheta \leqslant 30°$
Hydrophobic-π	Aromatic ring C *sp3*, S *sp3*, F, Cl	C *sp3*, S *sp3* Aromatic ring	d $\leqslant 4.5$ Å offset $\leqslant 2.0$ Å	-
Hydrophobic	C, S, F, Cl	C, S	d $\leqslant 3.5$ Å	-

Table 2.1: Interactions thresholds used for the detection. The offset for π-systems correspond to the projected distance between the center of masses of interacting groups. A and D correspond respectively to H bond acceptor and donor. X-bond donor is a C bonded to either a Cl, Br or I. X-bond acceptor corresponds to O, N or S with lone pair.

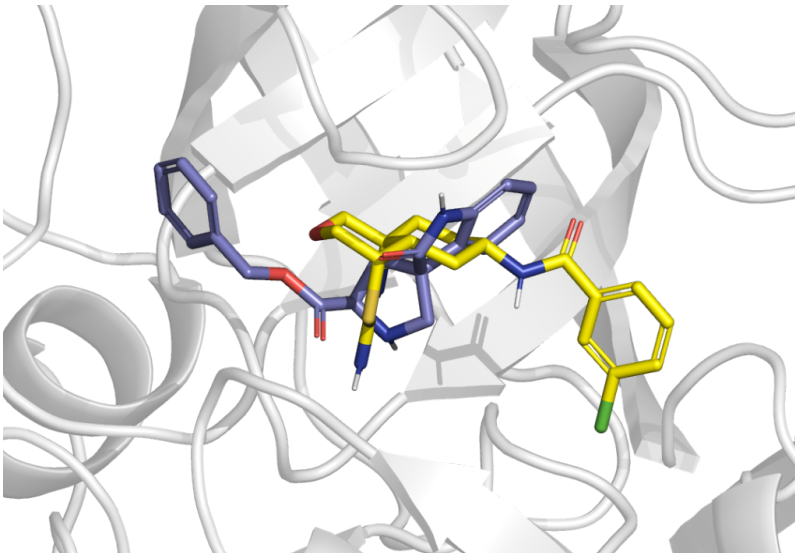
Selecting pairs with similar binding modes

Among the 44.386 pairs of highly dissimilar molecules, we need to identify those that correspond to a scaffold hopping case, i.e., to select those in which two molecules have similar binding modes within the same protein pocket. A Tanimoto similarity between IFPs is used to compare the binding modes of ligands and remove 'false' scaffold hopping cases, as illustrated in Figure 2.3. Ligands forming only few interactions with the protein (less than five) are removed, as the computation of Tanimoto similarities would not be reliable. We keep pairs of ligands with IFPs similarities above 0.6. This leads to 821 pairs of molecules with highly dissimilar chemical structures, but similar binding modes.

a)



b)



c)

Murcko-based Morgan similarity	Molecular Morgan similarity	IFP similarity
0.25	0.10	0.06

Figure 2.3: Example of a pair of dissimilar ligands for Beta-Secretase 1 (PDBs: '3udm' and '4zsq') occupying different areas of the binding site of the protein. The molecules are shown in a). The crystallographic conformations are displayed in b). Table c) compares the two molecules: they share little common binding modes and cannot be considered as a scaffold hopping case.

Discarding pairs based on Maximum Common Substructures

Among the 821 pairs, visual analysis allowed us to observe cases where the two molecules share a common substructure forming most of the interactions with the protein. These cases cannot be considered as scaffold hops if the common substructure is responsible for most of their interactions with the protein pocket, since these substructures can then be viewed as a common scaffold that drives binding to the protein. To remove these instances, we use the Maximum Common Substructure (MCS) concept, because it has been shown to help identify scaffold hopping cases [Barker *et al.*(2006)]. For each pair of molecules, we search for their MCS and compute the ratio between the number of common interactions arising from chemical groups in the MCS, and the total number of common interactions to the two molecules. A high ratio means that the MCS is responsible for most of the common interactions, and the corresponding pair should not be considered as a large-step scaffold hopping case, as described in Figure 2.4.

Concretely, the MCS between two molecules is searched based on three different types of MCS, as defined in RDKit [Landrum *et al.*(2021)]: MCS with matching of complete rings, MCS with partial matching of rings and MCS with allowed ring breaking. In particular, the first MCS searches for complete ring matches, allowing to discard pairs of molecules that would correspond to small-step scaffold hops.

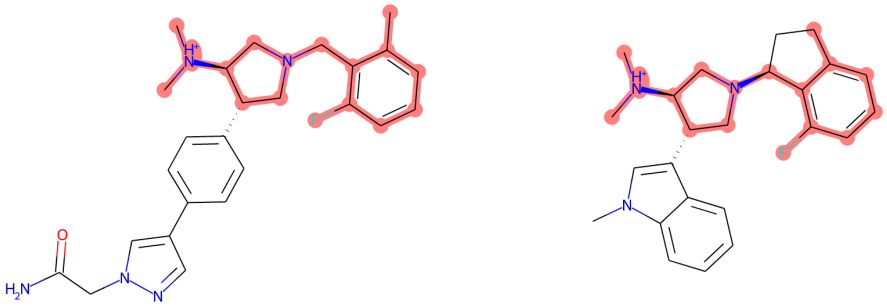
The maximum ratio of the number of common interactions formed by MCSs and the total of common interactions between the two molecules of a pair is computed as following. Each MCS type is matched on both ligands, and the ratio of the number of common interactions formed by the considered MCS and the total of common interactions is calculated as following:

$$ratio_{MCS\ interaction} = \frac{|Interactions_{Common} \cap Interactions_{MCS}|}{|Interactions_{Common}|} \tag{2.1}$$

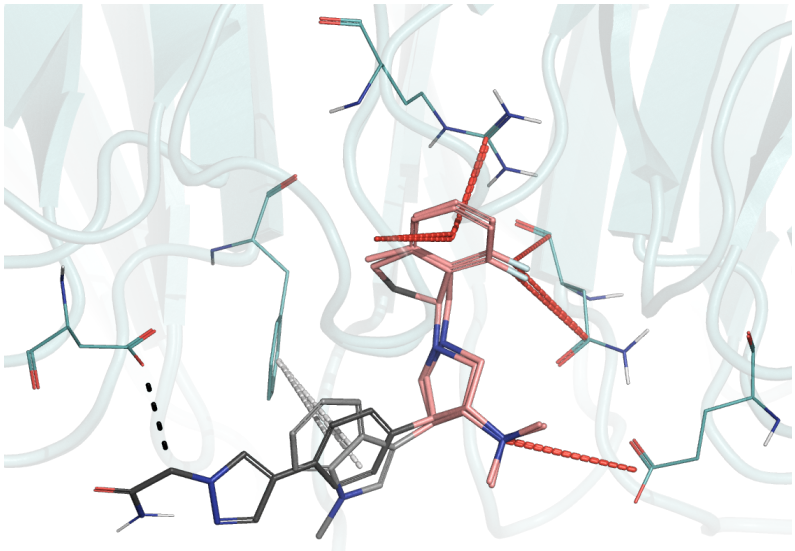
When several MCS matches are possible on a molecule, the match with the highest ratio is kept. The final ratio is defined as the highest of the ratios for all MCS types. Pairs with a final ratio above 0.8 were discarded, resulting in 531 pairs for 79 proteins.

Overall, the three types of substructure search are complementary, and the maximum ratio of common interactions formed by the MCSs over the total number of common interactions ensures to retrieve only large-step scaffold hopping cases. An example of MCS search between two molecules is given in Figure 2.5.

a)



b)

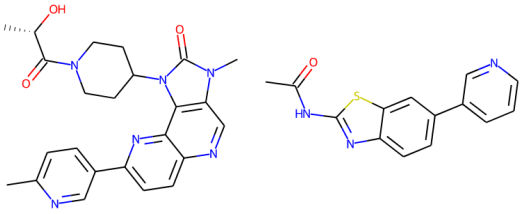


c)

Murcko-based Morgan similarity	Molecular Morgan similarity	IFP similarity	MCS interaction Ratio
0.23	0.22	0.86	0.83

Figure 2.4: Example for Polycomb protein EED of molecules (PDBs: '5u6d' and '5u8 f') with similar binding modes explained by a common substructure. The two ligands are displayed in a) with their common substructure highlighted in light red. Their crystallographic conformations are shown in b) along with their interactions with the protein. The red interactions corresponds to common interactions arising from the common substructure (colored in light red in the molecules), while the light grey interaction is the only common interaction arising from dissimilar parts of the molecules. Table c) compares the two molecules. As 5 out of the 6 common interactions are explained by the MCS, such a case cannot be considered as a scaffold hopping example.

a)



b)

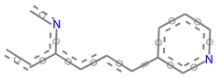
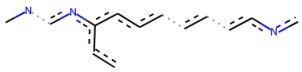
	MCS complete rings	MCS partial rings	MCS ring breaking
Properties	Only matches complete rings	Allows to match partial rings	Can break rings to match ring bonds to non ring bonds
SMARTS			
MCS interaction ratio	0.4	0.6	0.2

Figure 2.5: Illustration of the three different MCS searched. A pair of molecules (PDBs: '4hvb' and '4ps7') is displayed in a), and the table describing the three different MCS searched on this pair along with their ratios of common interactions is shown in b).

2.1.3 Discarding Redundant Pairs

We observe that for some of the 79 proteins, the selected pairs are strongly redundant and represent only slightly different examples of scaffold hopping cases: they involve two molecules that belong to the same chemical series (for instance, they differ by the addition of a small group not involved in the binding). A compelling example is given in Figure 2.6. To avoid redundancy in our dataset, which may lead to bias for performance evaluation of computational methods, we remove pairs in which both ligands are similar to both ligands of another pair, using a threshold of 0.5 on both their molecular Morgan similarities as detailed in the following. To discard redundant pairs, which differ by only slight molecular modifications, the Tanimoto similarities based on the Morgan fingerprints of the generic Murcko scaffolds and of the whole molecules are calculated. When two pairs of ligands have one of these Tanimoto coefficient above 0.5, only one pair is kept in the dataset, which finally leads to 178 pairs.

2.1.4 Resulting Large-step Scaffold Hopping Dataset

The global selection flowchart is shown in Figure 2.7. Overall, 178 large-step scaffold hopping cases of drug-like pairs binding to 79 different proteins are selected. On average, each protein is involved in 2.3 large-step scaffold hopping cases in the dataset. For the most represented protein, cell division protein kinase 2, 10 cases are selected. The most represented family of proteins is the kinases family, with 61 pairs involving 21 different kinases. This can be explained by the fact that kinases belong to a highly studied family of proteins, with many therapeutic targets against which many drug design projects have been devoted [Advani *et al.*(2013)]. However, the dataset still contains significant protein diversity, since the 79 proteins belong to 35 different super-families

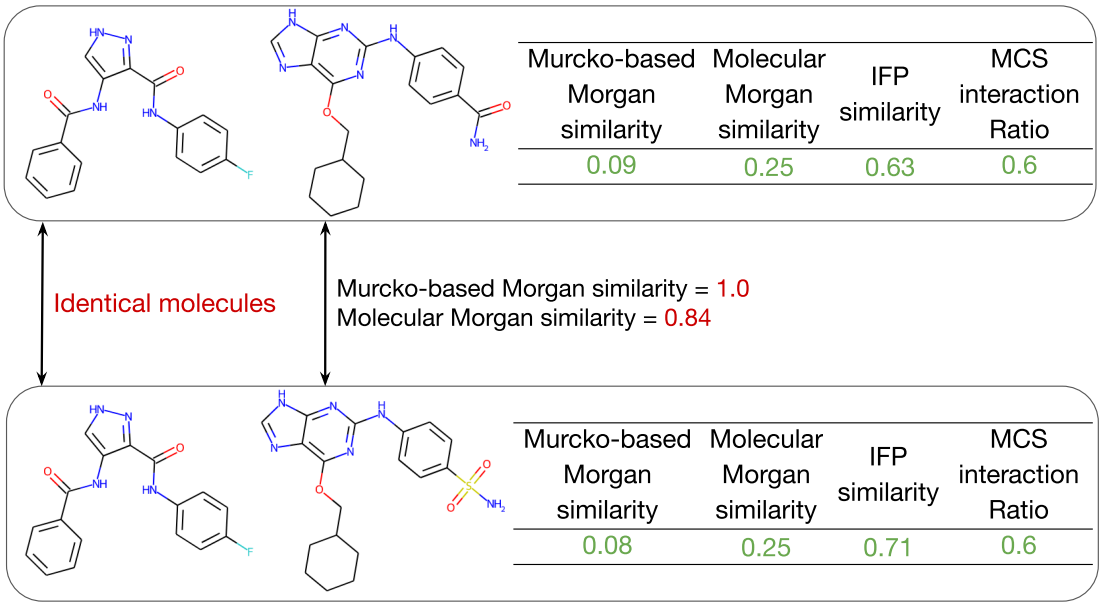


Figure 2.6: Example of redundant scaffold hopping cases for the cell division protein kinase 2: the two pairs are highly similar, since the second pair (PDBs: ('2vto', '4eok')) can be obtained from the first pair (PDBs: ('2vto', '1oiy')) by replacing the amide group on one of the molecule by a sulfonamide. In such cases, one of the two pairs was discarded.

of the SCOP protein family's hierarchy database [Murzin *et al.*(1995)]. On average, each super-family is involved in 5.1 scaffold hops.

Each selection step involves criteria with threshold values. The above paragraphs show that one must be careful to avoid 'false' large-step hops, or 'false' scaffold hopping cases, which has scarcely been discussed in previous benchmark studies. In the present work, the thresholds are chosen arbitrarily and somewhat stringently, to build a highly reliable dataset, as judged by visual analysis of the selected pairs. Of course, these thresholds can be changed. Examples of selected large-step scaffold hopping are provided in Figures 2.8 and 2.9.

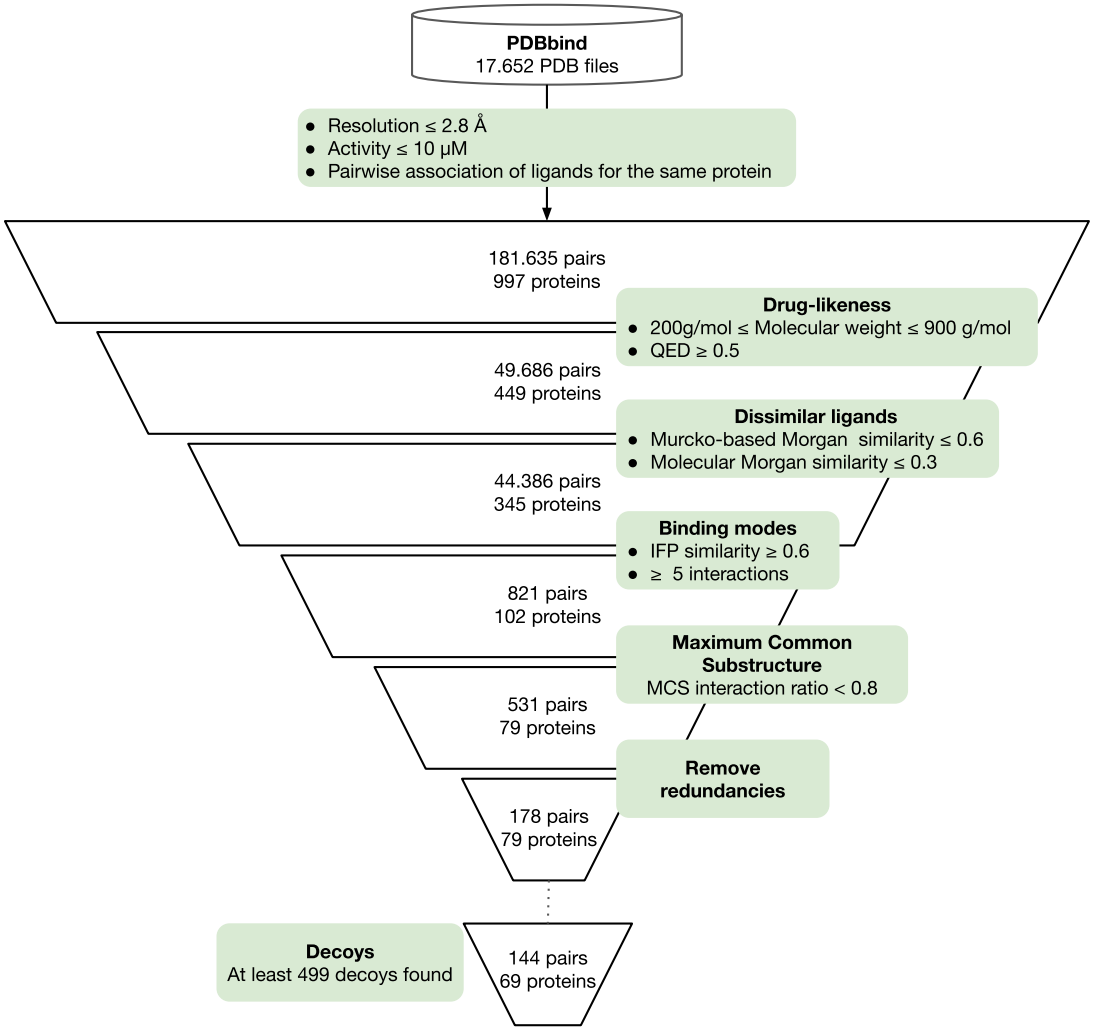
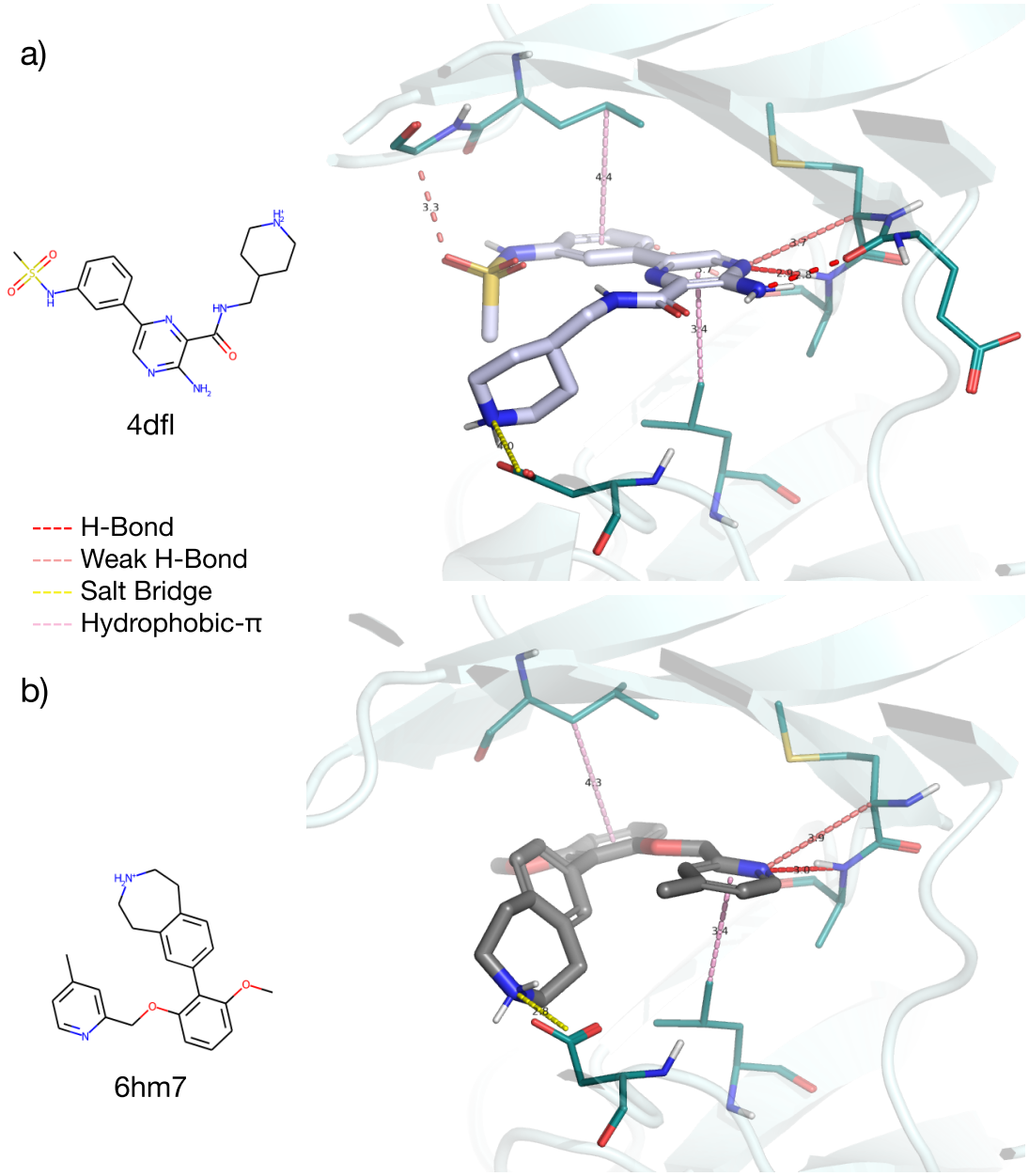


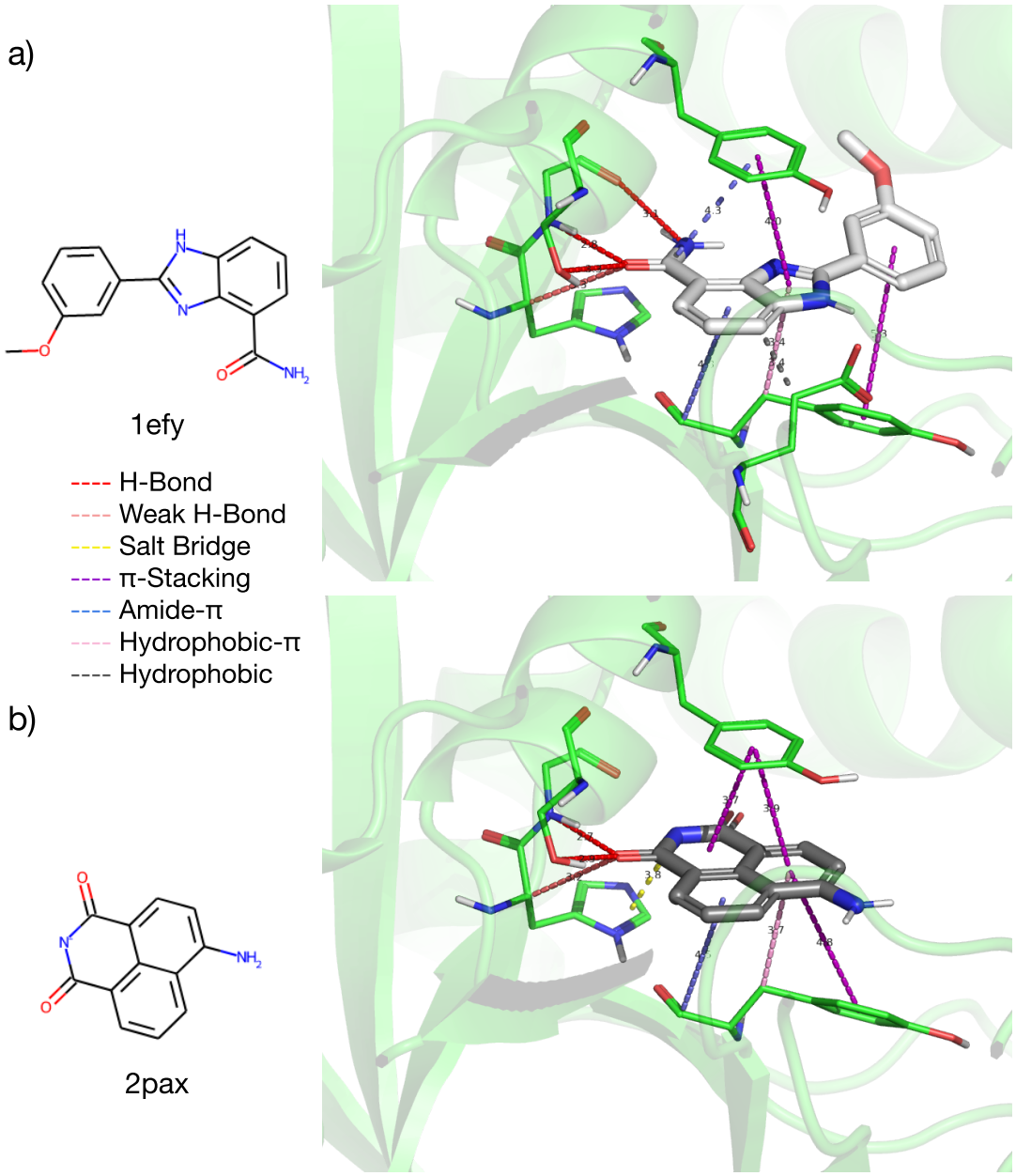
Figure 2.7: Flowchart describing the successive filters applied to identify large-step scaffold hopping cases. Starting from PDBbind crystal structures with good resolutions of proteins in complex with at least two ligands (181.635 pairs), we keep those involving drug-like molecules of dissimilar structures but similar binding modes. We removed pairs containing a common substructure responsible for most common interactions. We then discarded redundant pairs, leading to 178 large-step scaffold hopping cases. Among these cases, we keep those for which 499 decoy molecules could be found (see below). The chosen thresholds are arbitrary but ensured us to retrieve only confident large-step scaffold hopping cases, as detailed in subsection 2.5.



c)

	H-Bond		Weak H-Bond			Salt Bridge	Hydrophobic-π		IFP Similarity
	GLU449	ALA451	GLY378	MET450	ALA451	ASP512	LEU377	LEU501	
4dfl	1	1	1	1	1	1	1	1	0.625
6hm7	0	1	0	1	0	1	1	1	

Figure 2.8: Large-step scaffold hopping case for Tyrosine-protein kinase SYK. The binding modes of the ligand in PDB '4dfl' are illustrated in panel a), those of '6hm7' in panel b). The protein-ligand interactions are represented with dash lines and colors according to the type of molecular interaction. The interactions with high-spacing dash lines are interactions made only by one ligand of the pair, whereas those of low-spacing are common interactions. The resulting IFPs are displayed in panel c), as well as their Tanimoto similarity.



c)

	H-Bond		Weak H-Bond	Salt Bridge	π-Stacking		Amide-π		Hydrophobic-π	Hydrophobic	IFP Similarity
	GLY863	SER904	HIS862	HIS862	TYR897	TYR907	TYR896	TYR907	LEU377	GLU988	
1efy	1	1	1	0	1	1	1	1	1	1	0.7
2pax	1	1	1	1	1	1	1	0	1	0	

Figure 2.9: Large-step scaffold hopping case for Poly (ADP-ribose) polymerase. The binding modes of the ligand in PDB '1efy' are illustrated in panel a), those of '2pax' in panel b). The protein-ligand interactions are represented with dash lines and colors according to the type of molecular interaction. The interactions with high-spacing dash lines are interactions made only by one ligand of the pair, whereas those of low-spacing are common interactions. The resulting IFPs are displayed in panel c), as well as their Tanimoto similarity.

At this point, we have gathered 178 non redundant scaffold hopping pairs of active molecules, for a panel of diverse proteins. As detailed in 1.6, one of the main goals of the thesis is to design new molecular descriptors specifically tailored to help solving scaffold hopping problems. More precisely, the goal is to search for molecular descriptors according to which scaffold hopping molecules are similar, although they are dissimilar in terms of chemical structure. Relevant molecular descriptors should bring one of the active molecules closer to the other active than to randomly chosen molecules, in their corresponding embedding space. It is not straightforward to define a criterion to compare the relevance of various molecular encodings. We propose to define decoy molecules that are added to the *LH* benchmark, for each scaffold hopping pair. Then, for each pair, given one active (the known active), we will rank the other active (the unknown active) among the decoys, based on their similarity with respect to the known active. Thus, the better the molecular descriptors for the problem of scaffold hopping, the lower the rank of the unknown active (best rank being 1).

The global principle of the benchmark design is illustrated in Figure 2.10.

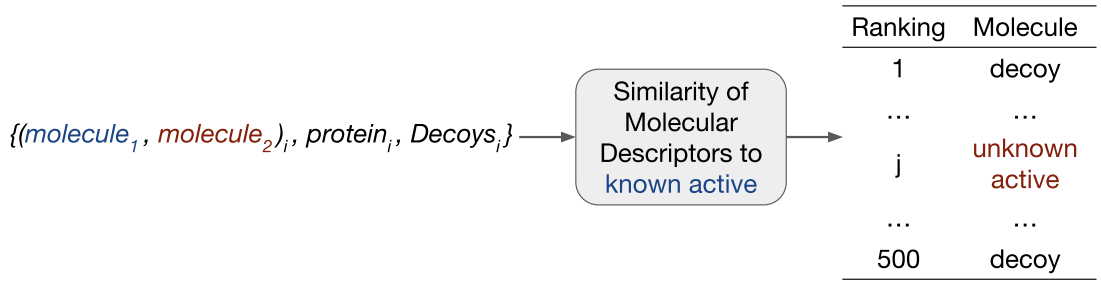


Figure 2.10: Principle of performance evaluation on the Large-Hops benchmark. For instance i, one molecule of the pair is set as the known active, and the other as the unknown active. The unknown active and the decoys are ranked according to their similarity of the evaluated molecular descriptors to the known active. The rank j of the unknown active is used to evaluate the considered molecular descriptors.

More precisely, we propose to compare molecular descriptors based on two criteria: (1) We draw Cumulative Histogram Curves (CHC), representing the number of cases for which the considered molecular descriptors ranked the unknown active below a given rank (as detailed in 2.5). The curves of the best performing molecular descriptors will stand above those of others. (2) In real-life screening campaigns, only the best ranked molecules are usually considered as candidate molecules for experimental tests. Thus, we also compare the relative positions of the CHC curves at best ranks and determine the proportion of cases where the unknown active is retrieved in the top 5% best ranked molecules [Grisoni *et al.*(2018a)], which can be seen as the success rate of the molecular descriptors.

Therefore, the following step consists in gathering decoy molecules for each scaffold hopping case in the benchmark.

2.2 Choice of Decoy Molecules

For each scaffold hopping pair, we selected 499 decoys to reach meaningful active/inactive ratio of 1/500, which is well below the frequently used ratio of 1/50 [Lagarde *et al.*(2015)]. These numbers reproduce real-world virtual screening campaigns, as detailed in 2.1. Thus, as illustrated in Figure 2.11, our final benchmark consists in a dataset of pairs of molecules representing large-step scaffold hopping cases, and their corresponding decoy molecules.

Selection of decoy compounds is not an easy problem, since we need to avoid statistical bias with respect to the active molecules [Réau *et al.*(2018)]. In particular, when decoys stand in regions of chemical space that are very distant from the two active molecules, the resulting benchmark may suffer from 'analogous bias' [Good et Oprea(2008)], and the success rate may be artificially overestimated. In addition, since we also want to mimic real-life applications, the decoys must be realistic scaffold hopping candidates that, in practice, would be searched among molecules sharing some physicochemical characteristics with the known active.

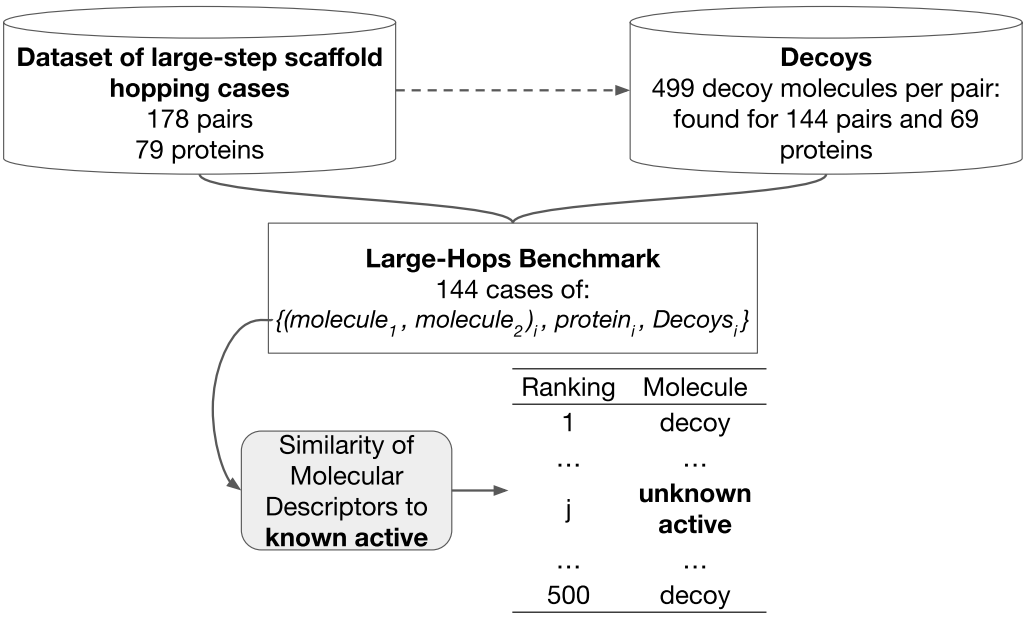


Figure 2.11: The Large-Hops benchmark was built from a dataset of large-step scaffold hopping cases extracted from PDBbind for which 499 decoy molecules were gathered. Overall, it comprises 144 cases defined by a pair of active molecules against the same protein target, and their corresponding decoys.

It has been shown that random selection of decoys from large chemical databases does not prevent the occurrence of statistical bias [Réau *et al.*(2018)]. Therefore, to avoid statistical bias between molecules in the active pairs and their corresponding decoys, decoys are selected from the ZINC database [Irwin et Shoichet(2004)], among molecules with physical and chemical properties similar to those of the active molecules, as detailed below. The considered physical and chemical descriptors are:

- Number of hydrogen bond donor and acceptor ;

- Number of aromatic and aliphatic rings ;
- Number of consecutive rotatable bonds ;
- Lipophilicity (ability to dissolve in fats, oils, lipids, and non-polar solvents);
- Topological polar surface area (surface sum over all polar atoms).

More precisely, molecules are selected if their physical and chemical descriptors fulfill the following criteria:

$$descriptor_{\,molecule} \in [min(descriptor_{\,ligands}) - c,\ max(descriptor_{\,ligands}) + c] \quad (2.2)$$

where $c = 1$ for integer descriptors, and

$$c = \frac{10}{100} |descriptor_{\,ligand_1} - descriptor_{\,ligand_2}| \quad (2.3)$$

for continuous descriptors.

As the decoys need to be realistic large-step scaffold hopping candidates, they are chosen at a Murcko-based Morgan similarity from the molecules in the active pairs below 0.6, since this threshold is used to select pairs of active molecules. In addition, the decoys should not either be too distant from the ligands, in order to avoid analogous bias, and to mimic real-life screens for the search of scaffold hop candidates. Thus, the decoys selected from the ZINC also have to be as similar to the ligands as the ligands are similar to each other, according to their overall structure Morgan fingerprints:

$$similarity_{\,molecule,\ ligands} \in \left[similarity_{\,ligand_1,\ ligand_2} - c',\ similarity_{\,ligand_1,\ ligand_2} + c'\right] \quad (2.4)$$

where $c' = 0.15$ to have an interval of size 0.3 and capture enough decoy molecules (i.e., a number of 499 decoys), neither too distant nor too close, with respect to the molecules in the active pair.

These criteria were successively applied to molecules in the ZINC database, and 499 decoys satisfying these criteria could be found for 144 pairs of active molecules.

Note that we cannot rule out the possibility of a few false negatives [Vogel *et al.*(2011)], because we may have accidentally picked decoys that bind to the same protein as the molecules in the pair. However, we assume that such cases are rare, and that their potential presence does not change the overall conclusions of the analyses.

The resulting benchmark finally consists in 144 pairs of molecules associated with their corresponding proteins and their 499 decoys (the 499 decoys are different for each of the 144 pairs). These 144 pairs of active molecules involve 69 different proteins, belonging to 31 different super-families of the SCOP [Murzin *et al.*(1995)]. This benchmark contains scaffold hops cases within protein families that have been more extensively studied than others. Nevertheless this panel of proteins is wide enough to set apart from a case study, and to get a broader glance at how well a computational method performs to solve large-step scaffold hopping problems. Taken together, the rules used to select the active pairs and their decoys are stringent, but this ensures to build a realistic, high-quality, and well characterized benchmark dedicated to the problem of large-step scaffold hopping.

2.3 Considered Molecular Descriptors

Many types of 2D descriptors have been proposed in the literature, and perform very well for the prediction of various molecular properties [Helguera *et al.*(2008)]. Since scaffold hopping relates to ligand binding to a protein, a phenomenon occurring in the 3D space, 3D descriptors are expected to be more relevant in this context. We first explore the performance of classical 2D descriptors (usually employed for small- to medium-step scaffold hops), and then study classical 3D descriptors. Finally, we also consider a more original chemogenomic algorithm, to show how the benchmark could be used for the development of new methods dedicated to large-step scaffold hops.

2.3.1 Baseline 2D Descriptors

Because they neither require the 3D structure of the target, nor the 3D conformations of the molecules, we first consider 2D structure descriptors. Although they are not meant to best encode ligand binding properties, it is interesting to see whether these simple methods capture some valuable information to solve scaffold hopping problems. We consider three types of 2D representations.

(1) Morgan fingerprints that encode 2D molecular structures. These descriptors are not expected to perform well on our benchmark, because solving large-step scaffold hopping problems requires to search for molecules with highly dissimilar chemical structures. In addition, the molecular Morgan Tanimoto similarity was used to select pairs of dissimilar active molecules, so that ranking the unknown active and the decoys according to this similarity is doomed to fail. Testing 2D Morgan fingerprints encoding is a kind of internal control to confirm that our benchmark is an interesting tool for the development of original methods dedicated to large-step scaffold hopping. This fingerprints implements the ECFP extended connectivity fingerprint [Rogers et Hahn(2010)] with radius 2 as a 4096-bit binary vector.

(2) MACCS keys fingerprints [Durant *et al.*(2002)], that in principle should suffer from the same limitations as the Morgan fingerprints. In fact, since the former encodes the presence or absence of particular chemical groups rather than the molecular graph itself, it is interesting to test if this can be beneficial to the current problem. This fingerprint corresponds to a binary 166-bit vector that encodes the presence of SMARTS-based (SMILES arbitrary target specification, a language for specifying substructural patterns in molecules) strings in the molecular structure.

(3) 2D pharmacophore fingerprints, that encode for the presence and relative positions in the 2D graph of the molecular structure of chemical groups able to drive different types of interactions with the protein, as defined in RDKit [Landrum *et al.*(2021)]. They are calculated using the distance separating 2- and 3-point pharmacophores defined as SMARTS strings, in a planar representation of molecules. Although these descriptors implement a notion of 2D (but not 3D) topology, they may improve over MACCS keys fingerprints.

These three types of 2D representations lead to a binary vector encoding for the molecules, allowing the definition of corresponding similarity measures based on Tanimoto coefficients. Thus, for each pair of active molecules in the benchmark, the unknown active and the decoys are ranked according to their molecular Morgan, MACCS keys and 2D pharmacophore Tanimoto similarities with respect to the known active.

2.3.2 3D Molecular Descriptors

We also tested 3D descriptors, since they capture molecular features that can be better related to ligand binding. We consider two types of representations: 3D molecular shape, and 3D pharmacophores, because both approaches have been described as useful tools to help solving scaffold hopping problems [Rush *et al.*(2005)]. We study the general case where the 3D structure of the protein is unknown, so that the 'active' conformations (i.e. the ligand conformation when bound to the protein pocket) of the active molecules are unknown.

In both cases, this first requires generating 3D molecular conformers. For all pairs of active molecules and their 499 decoys, a pool of 500 conformers is calculated, from which we keep up to ten conformers of local minimal energy that differ from a RMSD value of at least 1.5Å, under MMFF94 force field [Tosco *et al.*(2014)] using RDKit. Then, all conformers of the known active are aligned pairwise with those of the unknown active or those of the decoy molecules, to maximize their overlap. For the 3D-pharmacophore similarity, for each pair of active molecules and their decoys, the freely available Pharao software [Taminau *et al.*(2008)] is used to detect the pharmacophore groups for conformers. The Tanimoto coefficient quantifying the overlap between aligned conformers of the known active and those of the unknown active or of the decoys is calculated pairwise. The largest Tanimoto 3D pharmacophore coefficient observed is used to define the 3D-pharmacophore similarity between the corresponding known active and the unknown active or the decoys. The same method is used for the shape similarity [Taminau *et al.*(2008)], where the largest Tanimoto shape coefficient observed between conformers is used to define the shape similarity between the known active and the unknown active or the decoys. Finally, for each pair of active molecules, the unknown active and the decoys are ranked according to their Tanimoto 3D pharmacophore, or 3D shape, similarity.

In the previous sections, we have studied the interest of various molecular descriptors with respect to the scaffold hopping problem. In real-life applications, one would search to solve these problems not only by ranking candidate molecule according to their similarity with respect to the hit. One would also consider predicting candidate molecules based on more sophisticated computational methods that use these encodings.

However, the *LH* benchmark cannot be used as such to train QSAR or Machine Learning algorithms, because it only contains two actives per case. Nevertheless, chemogenomic algorithms offer an interesting option. Indeed, as detailed in the following, they can be trained using any protein-ligand information available in other databases. Therefore, we evaluated the interest of a chemogenomic approach to solve scaffold hopping problems, using the *LH* benchmark as a test set.

2.4 The *LH* Benchmark as a Test Set for Chemogenomic Algorithms

As stated above, the *LH* benchmark cannot be used as a training set because it contains too few active molecules. Chemogenomic algorithms can overcome this limitation if bindings involving other molecules and other proteins are known (these molecule-

protein pairs are noted (m, p) pairs in the following). Such (m, p) pairs can be collected from many public databases, such as the PubChem at NCBI [Bolton *et al.*(2008)]. Basically, the main difference between ligand-based and chemogenomic methods is that the former predicts ligands for a query protein given its known ligands (one known active in our case), while the latter predicts ligands for a query protein given its known ligands and those known for other proteins. In the case of the benchmark, chemogenomic algorithms can be trained with the (known active, query protein) binding pair and additional (ligand, protein) pairs known to bind, or not, gathered from an external database. Once trained, the prediction model provides a binding probability for the (decoys, query protein) and (unknown active, query protein) pairs, and the unknown active can be ranked among the 499 decoys according to these probabilities.

Kernel SVM. The chemogenomic approach used in the present study recasts the problem as a supervised learning binary classification over the space of pairs (m, p) of molecules and proteins, to separate binding pairs from a carefully selected set of non-binding pairs. We rely on a kernelized Support Vector Machines (SVM) classifier [Cortes et Vapnik(1995)] to perform this classification. Briefly, the SVM is trained on a dataset of (m, p) pairs and learns the optimal hyperplane that separates pairs that bind from those that do not. The kernel SVM leverages a kernel K encoding similarities between (m, p) pairs [Schölkopf *et al.*(2004)]. A general method to build a kernel on (m, p) pairs is to use the Kronecker product of molecule and protein kernels as done in [Schölkopf *et al.*(2004)] and [Vert et Jacob(2008)]. Given a molecule kernel $K_{molecule}$ and a protein kernel $K_{protein}$, the Kronecker kernel K_{pair} is defined by:

$$K_{pair}((m, p), (m', p')) = K_{molecule}(m, m') \times K_{protein}(p, p')$$

where m and m' are molecules and p and p' are proteins.

We chose the Local Alignment kernel for proteins [Saigo *et al.*(2004)] and the Tanimoto kernel with Morgan fingerprints for molecules [Swamidass *et al.*(2005)], whose hyperparameters are validated by cross validation in [Playe *et al.*(2018)]. The Local Alignment kernel for proteins sums up the contributions of all possible local alignments with gaps of the sequences which is efficient for detecting remote homology [Saigo *et al.*(2004)]. The Tanimoto kernel between two molecules is calculated as the Tanimoto similarity of their Morgan fingerprints [Swamidass *et al.*(2005)]. Protein and molecular kernels are centered and normalized.

The SVM algorithm also requires a regularisation parameter classically called C, which controls the trade-off between maximising the margin (the distance separating the hyperplane and the two classes' distributions) and minimizing classification error on the training points. To implement this algorithm, we use the sklearn [Pedregosa *et al.*(2011)] function SVC with the parameter $C = 10$ validated by cross validation in [Najm *et al.*(2021)]. Once the SVM is trained, it can be applied to any pair (m, p) to give a binding probability. This probability is computed by applying a sigmoid function to the SVM outputs, where the parameters are trained by cross validation as explained in [Platt(1999)]. It is implemented in the `predict_proba` method of `SVC`.

Training Dataset. To build our training set, we use the DrugBank database v1.5.1 [Wishart *et al.*(2018)] which defines a set of (m, p) pairs which bind together (i.e. m targets p). Indeed, DrugBank provides high quality bio-activity information regarding approved and experimental drugs, including their targets. It contains around 15.000 curated drug-target (m, p) pairs for 2.670 proteins. Although much smaller than other databases like PubChem or ChEMBL, DrugBank appears relevant to the benchmark because it contains drug-like ligands. We kept molecules with molecular weights between 100 and 800g/mol which is in the range of drug-like molecules [Lipinski(2000)]. Thus, the train dataset comprises 5.071 molecules, 2.670 proteins and 14.638 positive bindings. To complete the dataset, we need to select negative pairs. This selection should be designed with care to correct potential statistical bias in the database and reduce the number of false positive predictions. We use the greedy algorithm in [Najm *et al.*(2021)], which randomly chooses the same number (14.638) of negative pairs so that each molecule and each protein have the same number of positive and negative pairs in the training dataset.

Training Scheme. The Machine Learning (ML) chemogenomic algorithm is trained for each of the 288 scaffold hopping cases in the benchmark as follows: one molecule of the pair is considered as the only known active for the query protein. If this pair is not already present in the DrugBank database, it is added to the training set. All other pairs involving the query protein that are present in DrugBank are removed from the training set. This allows us to exclude the (m, p) pair between the unknown active of the pair with the query protein if this pair is in DrugBank. Hence, for each pair of ligands in the benchmark, the chemogenomic algorithm is trained with the same information about ligands of the query protein than the ligand-based algorithms: a single known active ligand. Once trained, the algorithm predicts the binding probabilities of $(molecule, query\ protein)$ pairs involving the 499 decoys and the unknown active molecule. In order to have a more robust score, this scheme is repeated 5 times for different sets of negative examples in the training set and the binding probabilities are averaged over these 5 versions. We observe that the variance across these repetitions is low (below 10^{-2}) which highlights the stability of the method. Finally, the unknown active molecule and the 499 decoys are ranked according to their averaged binding probabilities. Figure 2.12 summarizes the difference between the similarity searching experiments for molecular descriptors and the chemogenomic setups.

2.5 Results on LH Benchmark

For all the considered molecular descriptors and the chemogenomic algorithm, Cumulative Histogram Curves (CHC) corresponding to the rank of the unknown active molecules are plotted. The CHC curves of the most efficient approaches (molecular descriptors or chemogenomic algorithms) stand above the others. The x-axis represents the rank, and the y-axis represents the proportion of cases (i.e., the proportion of scaffold hopping cases, among the $144 * 2 = 288$ scaffold hopping problems in the Large-Hops benchmark) where the approach recovers the unknown active at a rank below the x-axis value. For instance, for the chemogenomics approach, the unknown active was ranked in top 50 molecules for 45% of cases, as seen in Figure 2.13. Ap-

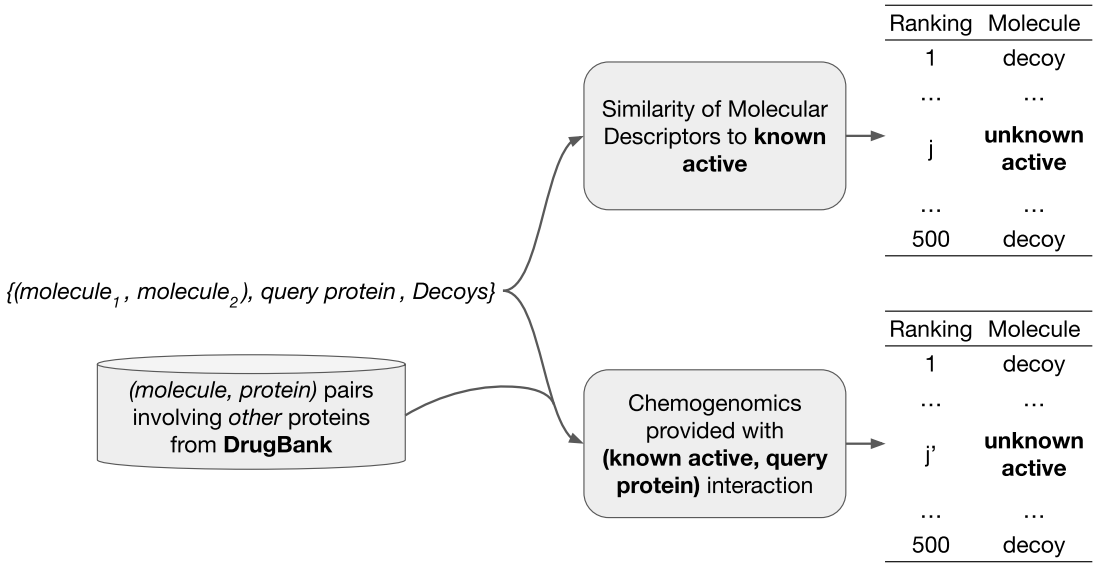
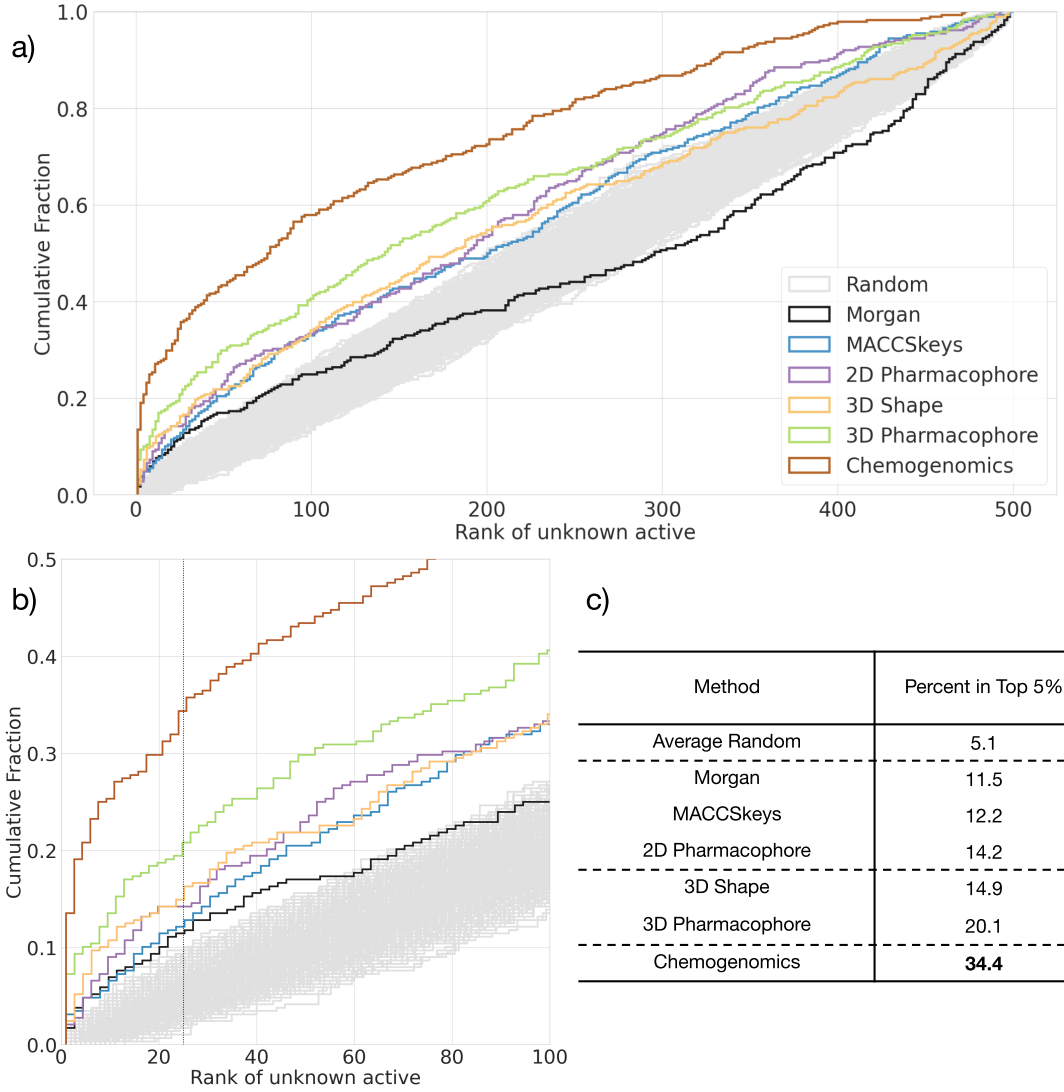


Figure 2.12: llustration of the schemes followed by molecular descriptors and by the chemogenomics approach to solve a scaffold hopping case.

proaches are also compared to random ranking: we perform one thousand random rankings for the unknown active for the 288 scaffold hopping problems. This leads to 1.000 CHC curves plotted in grey in Figure 2.13. For each approach, we compute the percentage of cases where the unknown active is ranked in the top 5%, i.e., in the first 25 molecules. This metric can be viewed as the percentage of successful cases, which replaces the Enrichment Factor that can not be computed since there is only one active in the datasets.

As shown in Figure 2.13, the molecular Morgan fingerprints display overall very poor performances in terms of success rate. At top ranks, its CHC curve stands only slightly above those random rankings, and the unknown active is retrieved in the top 5% in only 11.5% of the cases. Global failure of the Morgan fingerprints confirms that our benchmark mainly comprises large-step scaffold hopping cases. The MACCS keys and 2D pharmacophore fingerprints both improve over the molecular Morgan fingerprints. The 2D pharmacophore fingerprints were expected to perform better than the MACCS keys fingerprints, but their relative positions of CHC curves at high ranks, and success rates in the top 5% best ranked molecules are comparable. Overall, the performances of these two molecular descriptors remain modest since their success rate in the top 5% is below 15%.

The performance of the 3D pharmacophore and shape descriptors are also presented in Figure 2.13. 3D pharmacophore performs better than 3D shape on all criteria: relative positions of the CHC curves and success rate at 5%. This may be explained by the fact that 3D pharmacophore descriptors encode key information about chemical groups able to form interactions with a protein that are not present in the solely molecular shape. This result is in agreement with previous studies where 3D pharmacophore is depicted as a reference method for scaffold hopping [Hessler et Baringhaus(2010)]. The performances of 3D pharmacophore remain above those of the shape similarity, or



Method	Percent in Top 5%
Average Random	5.1
Morgan	11.5
MACCSkeys	12.2
2D Pharmacophore	14.2
3D Shape	14.9
3D Pharmacophore	20.1
Chemogenomics	**34.4**

Figure 2.13: Results on the *LH* benchmark. The cumulative histogram curves of each approach are plotted in a). A zoom of the same graphs is provided in b) with vertical grey lines corresponding to ranks of top 5% ranks. Table c) displays the percentage of successful scaffold hopping problems for methods using various molecular representations, according to a rank of the unknown active in the top 5%.

those of 2D methods. This is an interesting result, because some studies have reported that when the active conformations are unknown, performances of 2D methods might outperform those of 3D methods with calculated conformers, in the context of ligand binding prediction [Jacob *et al.*(2008)].

According to our results, a classical 3D pharmacophore appears as a good default similarity measure to solve large-step scaffold hopping problems. Note however that the achieved success rate at 5% lies around 20%. This allows to quantify the range in performance that can be expected, in general, with classical molecular descriptors on these types of problems, thus answering to the question raised by [Bajorath(2019)]. This leaves much room for the development of molecular descriptors more specifically designed to solve large-step scaffold hopping problems.

The performances of the chemogenomic algorithm are shown as well in Figure 2.13. It outperforms all tested molecular descriptors on all considered criteria. These performance improvements arise from the additional (ligand, protein) pairs provided during training, besides the (known active, query protein) pair. Similarity searching experiments cannot leverage such additional information, and the ML chemogenomic algorithm provides a computational framework to profit from such otherwise accessible prior knowledge. Note that the performances inside families of proteins are heterogeneous: on average, the families' success rate is about 37.8%, and for the most represented one, the kinases, the success rate is 35.1%. This means that the method depends little on the family of the proteins. However, the general success rate of 34.4% still leaves room for improvements. In particular, the kernel Support Vector Machine (SVM) algorithm used in the present study should be better adjusted to the scaffold hopping problem. Indeed, the SVM use a Tanimoto kernel for molecules that is calculated based on the molecular Morgan similarity [Swamidass *et al.*(2005)].

2.6 Conclusion

The scope of this Chapter is essentially: (1) to propose a flowchart to cover the need for a publicly available and well-characterized large-step scaffold hopping benchmark for the community; (2) to provide a method to assess the interest of molecular descriptors for solving large-step scaffold hopping problems.

To our knowledge, the benchmark is the first public high-quality benchmark dataset for large-step scaffold hopping. Starting from PDBbind, the proposed flowchart requires threshold values for various criteria. These thresholds were chosen in an expert-based manner to exclude irrelevant scaffold hopping cases. Some criteria enable the selection of pairs of highly different molecules, while others ensure that molecules in the same pair share similar binding modes, i.e., correspond to 'true' scaffold hopping cases. We use stringent thresholds for both types of criteria, because our goal is to build a high quality large-step scaffold hopping dataset. The resulting size for the benchmark is smaller than that reported for other less characterized benchmarks [Grisoni *et al.*(2018b), Nakano *et al.*(2020)], but this illustrates that the number of large-step scaffold hopping cases reported is much smaller than that of small- to medium-step scaffold hops. Note however that available benchmarks are not comparable to the present benchmark, because they were not conceived in a comparable setting. Should a large-step scaffold hopping benchmark of larger size be desired, the same flowchart could be followed with a lower drug-likeliness threshold, a larger range in molecular weights, or a higher threshold for redundancy between the pairs of molecules. Should an easier benchmark be designed, including medium-step hops, the thresholds in Murcko-based and molecular Morgan similarities could be increased. However, we advise not to relax the IFPs and MCS thresholds, to avoid selecting pairs of molecules that could correspond to 'false' scaffold hopping cases. An important contribution of this Chapter is to underline that building a reliable scaffold hopping benchmark must be a well-controlled multi-step process and cannot be achieved with the blind use of a few criteria. This important point, illustrated by the 'false' scaffold hopping examples shown in Figure 2.1, Figure 2.3 and Figure 2.4, has not been discussed in previous work reporting the construction of scaffold hopping benchmarks.

Based on the benchmark, all molecular representations tested display modest performances, which confirms that solving large-step scaffold hopping is a difficult problem. This was expected, but our study allows to quantify how difficult these problems are, in general.

Other promising topology-based descriptors not tested in the present study have been recently proposed [Grisoni *et al.*(2018b), Nakano *et al.*(2020), Nakano *et al.*(2021)], and future work could be to evaluate their performance on the Large-Hops benchmark.

Strategies based on descriptors that encode the bioactivity profiles of molecules have also been proposed [Petrone *et al.*(2012), Helal *et al.*(2016), Xiong *et al.*(2021), Hu *et al.*(2017)]. This is an interesting idea, because it allows to abstract from the chemical structure and address scaffold hopping issues. However, some of these profiles are not publicly available, but descriptors based on public domain HTS studies [Helal *et al.*(2016)] are interesting starting points to test their implementation in computational methods. In this context, we hope that the Large-Hops benchmark will be a convenient tool provided to the community, in order to test new strategies for the

difficult but important problem of large-step scaffold hopping.

The chemogenomic algorithm leads to the best performances, although the kernel SVM algorithm can be improved. Because our benchmark contained drug-like molecules for proteins belonging to diverse families, we trained the chemogenomic algorithm based on a DrugBank-derived dataset. However, other larger training sets can be used, for example derived from larger databases such as PubChem. For more focused problems like scaffold hopping problems involving a protein belonging to a specific well studied family, such as kinases or GPCRs, one can also use other training databases that gather (ligand, protein) molecular interactions known within these families of proteins [Carles *et al.*(2018), Okuno *et al.*(2006)]. As an illustration, although chemogenomics has been hardly explored in the field of large-step scaffold hopping, this approach was used in one study within the GPCR family, reporting identification of a new scaffold for an antagonist of Vasopressin 1A [Ratni *et al.*(2015)]. This underlines the interest to further explore these strategies in the field of scaffold hopping.

With those results in mind, we pursued the development of a new chemogenomic architecture trained on a wider custom dataset, not only as a tool to solve scaffold hopping, but as a way to identify hits for any protein. This article [Guichaoua *et al.*(2024)] is currently in revision:

G. Guichaoua, **P. Pinel**, B. Hoffmann, C.-A. Azencott, V. Stoven (2024), *Advancing Drug-Target Interactions Prediction: Leveraging a Large-Scale Dataset with a Rapid and Robust Chemogenomic Algorithm.*
doi:10.1101/2024.02.22.581599 (Currently in review.)

Details on this work are provided in Appendix A.

3

Identifying Isofunctional Molecules with the Interaction Fingerprints Profile

Abstract:

The performances of baseline molecular representations on the LH benchmark illustrated their limits in solving large-step scaffold hopping. Therefore, we propose the Interaction Fingerprints Profile (IFPP), a molecular representation that captures molecules binding modes based on docking experiments against a panel of diverse high-quality proteins structures. Evaluation on the LH benchmark demonstrates the interest of IFPP for identification of isofunctional molecules. Nevertheless, computation of IFPPs is expensive, which limits its scalability for screening very large molecular libraries.

Résumé:

Les performances des représentations moléculaires classiques sur le LH benchmark ont illustré leurs limites dans la résolution du 'large-step scaffold hopping'. Par conséquent, nous proposons l'Interaction Fingerprints Profile (IFPP), une représentation moléculaire qui capture les modes de liaison des molécules basés sur des expériences de docking contre un panel de structures protéiques diverses et de haute qualité. L'évaluation sur le LH benchmark démontre l'intérêt de l'IFPP pour l'identification de molécules isofonctionnelles. Cependant, le calcul des IFPP est coûteux, ce qui limite sa mise à l'echelle pour le criblage de bibliothèques moléculaires très volumineuses.

Contents

Most classical molecular descriptors are not adapted to solving large-step scaffold hopping problems, because they tightly depend on the chemical structure of molecules, while isofunctional molecules lie in remote regions of the chemical space.

Therefore, the problem of finding isofunctional molecules boils down to defining an encoding that maps molecules into a space where molecules that bind to the same pocket are close to each other. This encoding needs to be as much as possible agnostic of the 2D structure of molecules, in order to allow molecules that are dissimilar in terms of chemical structure to be close in the feature space. In this context, molecular representations based on biological properties are expected to be better suited to large-step scaffold hopping. Several encodings have been described in the literature [N. Muratov *et al.*(2020), Wassermann *et al.*(2015)]. Some, like CBFP [Xiong *et al.*(2021)], encode predicted activities for a profile of assays, thus defining bioactivity fingerprints for molecules. However, many of these bioactivity fingerprints need to be predicted for most molecules, because the corresponding assays were conducted on a limited number of molecules, and the corresponding prediction models may lack generalization properties. Consistent with this remark, pre-training a convolution neural network that predicts protein-ligand interactions based on the PCBA dataset that contains a profile of 90 bioactivities for thousands of molecules, did not improve the prediction performances of the algorithm [Playe et Stoven(2020)].

In this Chapter, we propose a novel biological representation of molecules inspired from such profiles. This encoding integrates information about protein-ligand interactions according to docking experiments against a panel of proteins for which a 3D structure of high quality is available. In the following, we refer to this molecular profile as the *Interaction Fingerprints Profile* (IFPP). We evaluate the interest of this representation in addressing large-step scaffold hopping problems using the Large-Hops (LH) benchmark.

3.1 The Interaction Fingerprints Profile Representation

In this section, we describe the motivation for the proposed IFPP as molecular representation, and explain how it is computed.

3.1.1 Rationale

As depicted in Chapter 2, none of the classical ligand-based methods relying on structural features of molecules display good performances in the challenging task of solving large-step scaffold hopping problems.

We suggest that molecular features derived from interactions that can be formed between a given molecule and protein pockets may be more relevant to solving this problem. Indeed, by definition, isofunctional molecules display dissimilar structures but share similar binding modes with a targeted protein. When the 3D structure of the target is available, docking can be used to search for candidates sharing similar binding modes. Our assumption is that even when the 3D structure of the target is unknown, the tendency to form similar interactions could be observed in other proteins.

The proposed approach is related to "ensemble methods" in Machine Learning, and particularly to "weak learners" methods [Kearns et Valiant(1989)]. A weak learner is a

model that performs only slightly better than the random prediction for a given task. In other words, it captures limited signal about the task at hand. Alone, it is not very useful, especially when compared to a "strong learner" that captures much of the signal, thus achieving good accuracy on the considered task. Unfortunately, such strong learner is often too hard to train, or even inaccessible. However, when aggregating several weak learners that may be easier to access, the performances of the resulting ensemble model can reach those of the strong learner. As an example, this general principle corresponds to the theory behind random forest algorithms [Breiman(2001)]. With this concept in mind, we introduce the following analogy. The strong learner would be docking in the protein target: based on the hit molecule binding mode, docking can be used to search for highly dissimilar molecules that present similar binding modes with this protein pocket. This strong learner is unavailable for a protein of unknown 3D structure. In such cases, the weak learners consist in docking the molecules in other proteins of known 3D structure. More precisely, we assume that two isofunctional molecules for a given protein would present a tendency to form similar interactions with proteins, in general, and that this tendency could be detected by docking. Docking a molecule in various proteins would allow to detect interactions between this molecule and protein pockets. Using "enough" weak learners, i.e. docking in "enough" proteins could be used to define the Interaction Fingerprint Profile of the molecule. The IFPP could then be used in ligand-based methods, replacing the unavailable strong learner (i.e. docking in the protein of interest).

It is important to note that the docking experiments used to build the IFPPs can be viewed as pure simulations. We are primarily interested in understanding how molecules would interact with the considered pockets, rather than whether they are true ligands with high affinity for these proteins.

In summary, our proposal relies on the idea that, within the space defined by IFPPs, two isofunctional molecules would be closer to each other than to randomly chosen decoys. They would also be closer to each other in this space than in the space defined by chemical descriptors.

3.1.2 Principle of IFPP Computation

The IFPP is built from weak learners that correspond to docking molecules into a panel of proteins of known 3D structures. The number and the nature of the proteins in this panel must be defined. We assume that a more diverse protein panel will cover a wider range of potential interactions that molecules can form.

The IFPP of a molecule is computed based on the interactions detected when docking this molecule in the panel of proteins. Details about the type of interactions as well as their retained detection thresholds are provided in 2.1.2.

We derive an Interaction Fingerprint for the molecule in the considered pocket, in the form of a fixed-size binary vector that incorporates, for each residue in the binding site, the types of interactions it forms with the molecule. The binding site is defined as all residues having at least one atom within a radius of 10 Å from the known crystallographic ligand of the protein.

The final IFPP is obtained by concatenating the fingerprint determined for each protein of the panel. Figure 3.1 illustrates how the IFPP of a molecule is obtained.

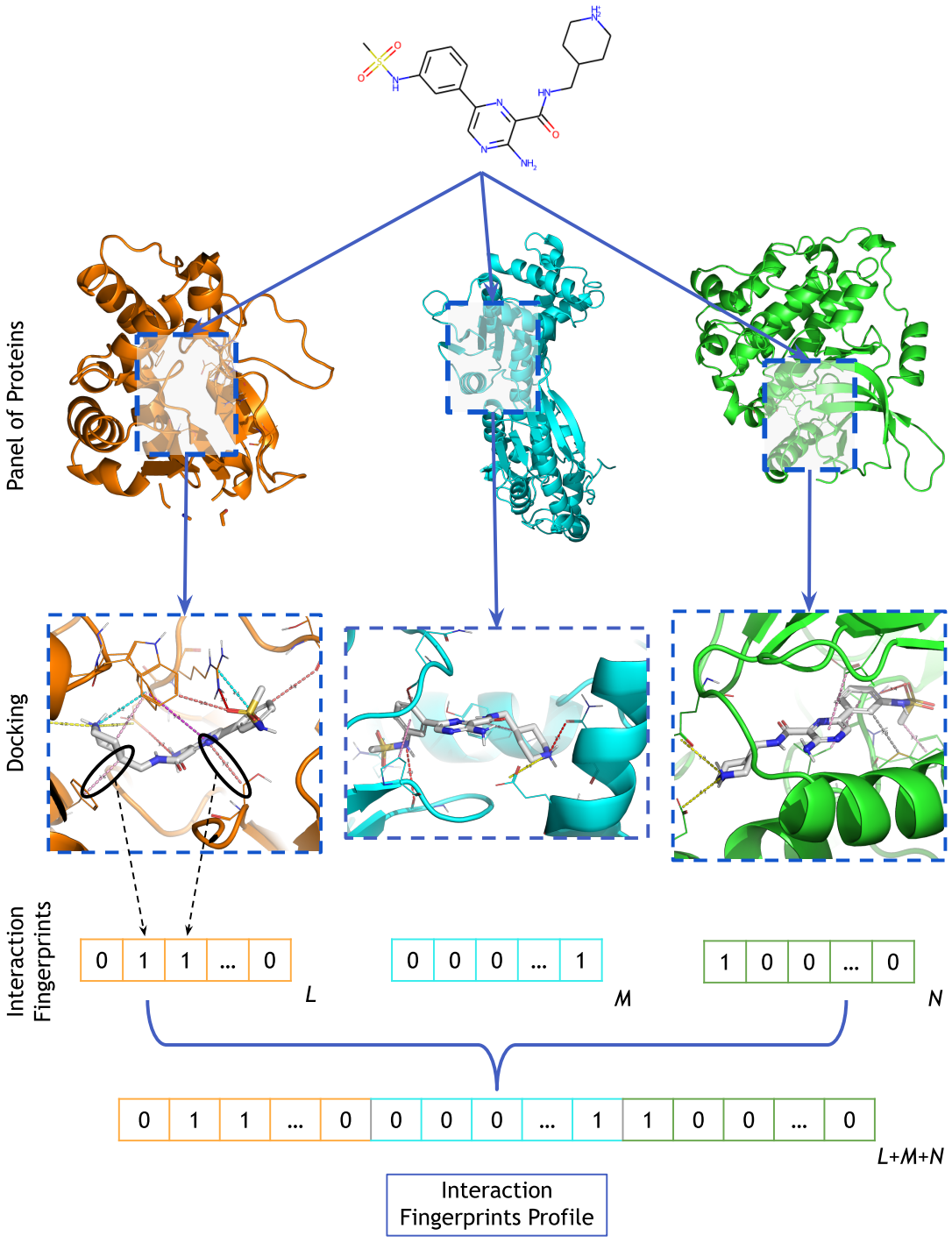


Figure 3.1: Illustration of how molecular IFPPs are built. In this example, a panel of 3 proteins is chosen. First, the molecule is docked in each of the proteins. Then, interactions of the best scoring pose are retrieved and encoded in a binary interaction fingerprint. L, M and N represent the length of the interaction fingerprints, dependent of the size of the binding site. Finally, interaction fingerprints are concatenated to form the final IFPP of length $L + M + N$.

However, as the binding sites' sizes vary from one protein to another, the length of the IFP also varies from one pocket to another. This results in the final IFPP to more bits allocated to some proteins than others. We did not perform any normalisation to have equal contributions for every protein because we wanted to keep the raw interactions. Besides, having a longer IFP for one protein does not lead to more "on" bits for a molecule, as it will only occupy a limited area in the binding site. It just leads to more possible binding poses. Thus, the number of "on" bits remains almost constant from one protein to another, regardless of the size of the binding site.

3.1.3 Choice of the Panel of Proteins

Ideally, we would like to dock in a large panel of diverse proteins as it would lead to the most complete sampling of possible binding modes for molecules. However, this is not realistic due to the cost of computing the resulting IFPP representation because:

- the proteins in the panel must be suitable for reliable docking experiments, which (among other criteria) requires careful preparation of its pocket and successful re-docking of the crystallographic ligand, to ensure that the docking protocol is adapted to this pocket.
- docking in too many pockets to compute the IFPP would be very costly.

Selecting Proteins with "drug-like" molecules

Therefore, to define a diverse but limited set of proteins, we considered the PDBbind database [Wang *et al.*(2004b)] containing 19.443 PDB files (2020) of 3D crystallographic structures of protein-ligand complexes. Only structures with resolution below 2.5 Å, and in which a drug-like ligand was bound, were kept, in order to keep only proteins with binding sites that are suitable for docking experiments with drug-like molecules. We selected complexes in which the ligand satisfied the following physicochemical parameters:

- No atoms other than H, C, N, O, F, P, S, Cl, Br
- 400g/mol $\leqslant$ Molecular weight $\leqslant$ 900g/mol
- -7 $\leqslant \log(lipophilicity) \leqslant$ 7
- Maximum ring size = 7
- No more than 7 rotatable bonds (to simplify docking and conformer sampling)
- Topological Polar Surface Area $\geqslant$ 30
- QED $\geqslant$ 0.3

These conditions exceed the criteria for drug-likeness, but they help removing unwanted chemotypes, such as salts, solvent or other molecules present in crystallisation buffers, and large interacting partners like peptides. This led to 1,248 PDB structures of complexes involving 378 different proteins.

These structures include pan-inhibitors or ligands belonging to the same chemical series. In order to avoid redundancies in terms of pockets, we clustered ligands according to their Morgan similarity with a threshold of 0.4. For each resulting cluster, we only keep the PDB with the best resolution. This led to 872 PDB structures for 358 different proteins. We retained only one structure per protein, keeping the PDB structure with the ligand of largest molecular weight, in order to define the largest binding site for the subsequent docking experiments, leading to 358 structures.

Protein Preparation for Docking

As explained in 1.4.1, prior to performing docking, each protein of the panel needs to be prepared. I describe in this subsection the different steps applied to prepare a PDB structure, as well as how the docking of a molecule in its pocket is performed.

To prepare a protein-ligand structure, we applied the following steps:

- To simplify, all water molecules are removed.
- The molecule and the protein are protonated using the softwares SimulationPlus and PDB2PQR [Dolinsky *et al.*(2007)] respectively.
- We parametrise the complex for the molecular dynamics engine using Gromacs [Van Der Spoel *et al.*(2005)] with an Amber force field [Wang *et al.*(2004a)].
- We minimise the complex in vacuo so that all the hydrogens are well positioned.

Dockings are performed using the DOCK6 software [Lang *et al.*(2009)]. It performs semi-flexible docking, considering the protein is rigid but exploring the conformational space of the molecule using the fragmentation method as explained in 1.4.1. For a molecule, only the pose with the best Grid-Based Score, an internal scoring function which relies on the non-bonded terms of the molecular mechanic force field, is retained.

Those preparation steps of PDBs and docking protocols were applied to all these structures.

Selecting Diverse Proteins

We then re-docked the crystallographic ligands in their corresponding pockets. We only kept those for which the best ranked docking poses had a RMSD below 2Å with respect to the crystallographic, ensuring that the docking protocol was adapted to these pockets. The final set of proteins should also avoid strong bias towards a few extensively studied families of proteins such as transferases or hydrolases, constituting almost 70% of crystallised PDBs [Burley *et al.*(2023)]. Therefore, to assess structural diversity of the proteins belonging to the final panel, we used the SCOP database [Murzin *et al.*(1995)]. This database provides a hierarchical classification of proteins according to their 3D fold. The proteins unrecognised by the SCOP database (using the UniProt ID) were discarded, leading to 283 different proteins. We observed that our set was still highly enriched in some protein structural families, such as some of the kinases folds. In order to form a panel of diverse proteins, we kept only one PDB structure per superfamily. A superfamily, as defined by the SCOP database, gathers proteins with different sequences, but that may have a common evolutionary origin according to their

structures and functional features. This filtering process left 73 superfamilies (therefore, 73 PDB structures) defined in the SCOP database. To evaluate the effectiveness of the IFPP using the *LH* benchmark, we excluded 4 proteins that were also included in the benchmark to prevent bias. Overall, these successive filters led to 69 PDB structures of proteins belonging to various superfamilies. Detailed information about this panel of proteins can be found in Appendix Table B.1.

Influence of the Size of the Protein Panel

However, real-life studies may require screening millions of molecules, and docking large chemical libraries against 69 proteins to derive the IFPPs would lead to high computational costs. Therefore, we undertook a preliminary study on the *LH* benchmark. The goal was to assess whether we could reduce the size of the IFPP (i.e. consider a smaller number of proteins to build the IFPP), without degrading the performance of its associated similarity measure. The corresponding similarity measure between molecules was defined as the Tanimoto similarity (1.2) of their IFPPs.

This preliminary study would require $34,569$ docking experiments for each scaffold hopping pair (2 actives and 499 decoys docked in 69 pockets), and consequently, almost 5 million docking experiments for the whole *LH* benchmark. To reduce these computational costs, for each scaffold hopping case, we kept 49 randomly picked decoys (out of 499), among which the unknown active was ranked according to its similarity with respect to the known active. We considered that a successful experiment corresponded to ranking the unknown active in the top 5% molecules (out of $49 + 1 = 50$ molecules, thus corresponding to a rank $\leqslant 2.5$). We tested different sizes of protein sets from the initial panel, ranging from 10 to 65. We performed 100 random draws of pockets for each set size from the 69 initially selected proteins.

The "Grid Score" scoring function was also used to dock molecules in this protein panel, to ensures that the docking score was adapted to predict the best poses of molecules in the protein of this panel, and compute the resulting IFPP molecular representations.

As shown in Figure 3.2, the performance of the IFPP increases with the size of the protein set. We do not seem to reach a plateau yet, suggesting that considering a higher number of proteins would improve the performances. However, as a compromise between computational cost of the IFPPs and performances of the associated similarity measure, we kept a panel of 37 proteins randomly picked from the 69 initially selected proteins. The list of these proteins is provided in Appendix Table B.1.

3.2 Performance of IFPP on *LH* Benchmark

3.2.1 Performance of IFPP

As introduced in Chapter 2, one method to evaluate the relevance of molecular representations for solving scaffold hopping cases is to compare the performance of their associated similarity measures in the *LH* benchmark.

In addition to the IFPP, we considered several classical structure-related encodings: Morgan fingerprints and 2D Pharmacophore fingerprints computed with RD-

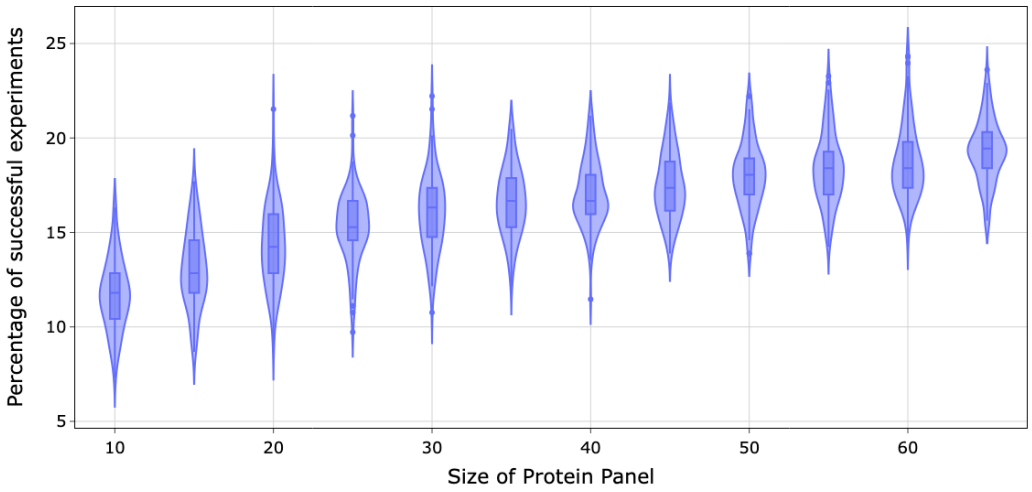


Figure 3.2: Influence of the size of the protein panel used to define the IFPP on the performance of the associated similarity measure on a reduced *LH* benchmark. Successful experiments are defined by a rank below 2.5 for the unknown active.

Kit [Landrum *et al.*(2021)], as well as 3D Pharmacophore and 3D Shape computed with Pharao [Taminau *et al.*(2008)].

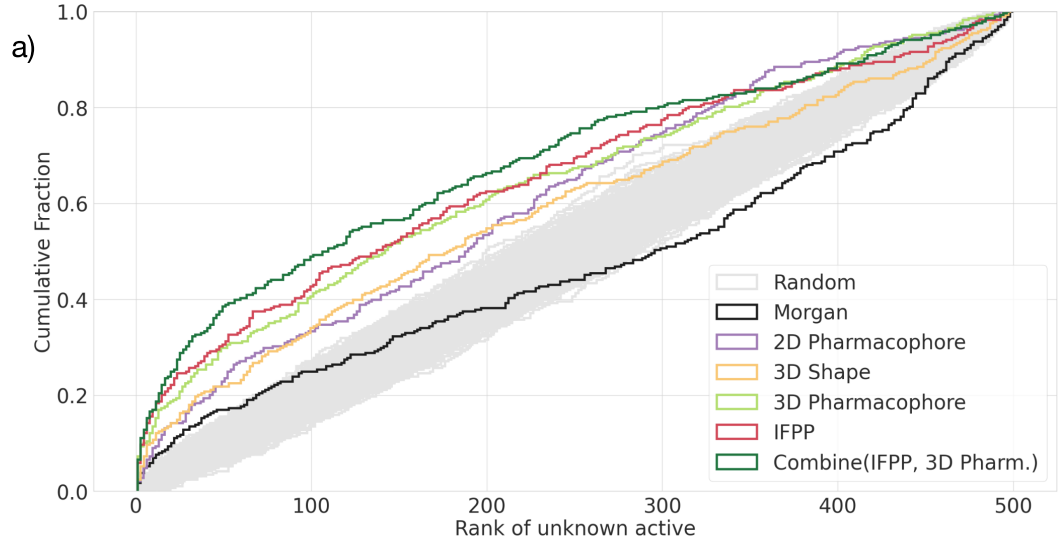
Figure 3.3 shows the performances of these 5 encodings. All molecular descriptors display modest performances on this benchmark, with success rates below 25%. Interestingly, the similarity in IFPP outperforms the others.

We performed the Kolmogorov-Smirnov test with the alternative hypothesis being that the cumulative distribution of ranks of the IFPP is greater than that of the 3D Pharmacophore. This resulted in a p-value of 0.038, a limited but significant improvement.

	Morgan	2D Pharm.	3D Pharm.	3D Shape	IFPP
Morgan	-	0.41	0.48	0.27	0.34
2D Pharm.	0.41	-	0.34	0.18	0.18
3D Pharm.	0.48	0.34	-	0.23	0.31
3D Shape	0.27	0.18	0.23	-	0.40
IFPP	0.34	0.18	0.31	0.40	-

Table 3.1: Spearman correlations between the rank of the unknown active for molecular descriptors.

We also computed the Spearman correlation of ranks between all methods, and summarised the results in Table 3.1. It shows that the IFPP is uncorrelated to the 3D Pharmacophore, and also to other methods. This indicates that this new representation captures information about the protein-ligand interactions that is absent in the other considered molecular representations. Because of their low correlation, we can combine the IFPP and 3D Pharmacophore representations, according to the minimum rank from both methods. This allowed to improve the performances, as illustrated in Figure 3.3 with a success rate of 27.4% in the top 5%.



Method	Percent in Top 5%
Morgan	11.5
2D Pharmacophore	14.2
3D Shape	14.9
3D Pharmacophore	20.1
IFPP	**24.7**
Combine (IFPP, 3D Pharm.)	27.4

Figure 3.3: Results on the *LH* benchmark. The cumulative histogram curves of each molecular descriptors are plotted in a). A zoom of the same graphs is provided in b) with vertical grey lines corresponding to ranks of top 5% ranks. Table c) displays the percentage of successful scaffold hopping problems for molecular descriptors, according to a rank of the unknown active in the top 5%.

We confirmed that the IFPP descriptors were not successful only in specific families of proteins, by computing the mean, standard deviation and quartiles ranks it assigns to the unknown active for scaffold hopping cases involving proteins from different superfamilies, as illustrated in Table 3.2. A superfamily, as defined by the SCOP database [Murzin *et al.*(1995)], gathers proteins with different sequences, but that may have a common evolutionary origin according to their structures and functional features. For example, in the case of the kinase superfamily, in 75% of the 94 related scaffold hopping cases (out of 288 cases), the unknown active was ranked above 36.8, i.e. not in the top 5% best ranked molecules.

Superfamily Name	Cases	Mean	STD	25%	50%	75%
ARID-like	2	85.5	29.0	75.2	85.5	95.8
ARM repeat-like	12	150.4	188.6	5.5	48.5	256.4
Acid proteases	4	48.2	58.2	18.5	27.0	56.8
Ankyrin repeat	12	256.6	104.7	218.0	274.5	319.5
Arginase/deacetylase-like	6	215.8	162.3	87.2	201.5	328.1
Bromodomain	16	189.1	136.0	90.0	152.5	288.4
DEATH domain	2	271.5	146.4	219.8	271.5	323.2
DPP6 N-terminal domain-like	8	217.4	181.8	53.8	234.8	302.0
Domain of poly(ADP-ribose) polymerase	10	144.8	137.3	28.2	146.2	183.0
FAT domain of focal adhesion kinase	2	381.0	113.1	341.0	381.0	421.0
GHKL (Gyrase, Hsp90, Histidine Kinase, MutL) domain-like	12	187.2	157.3	65.8	122.5	327.5
HD-domain/PDEase-like	28	162.6	147.1	25.5	122.5	230.8
Hemopexin-like domain	6	294.8	162.7	241.0	267.2	423.6
Inhibitor of apoptosis (IAP) repeat	4	85.0	136.4	13.0	23.5	95.5
Lipocalins	2	77.0	90.5	45.0	77.0	109.0
Macro domain-like	2	306.0	25.5	297.0	306.0	315.0
Metallo-hydrolase/oxidoreductase	4	68.5	22.8	60.2	73.5	81.8
Metalloproteases (zincins), catalytic domain	6	78.2	64.9	20.2	82.8	121.2
P-loop motor domain	4	372.0	140.3	250.5	371.8	493.2
PH domain-like	2	367.0	38.2	353.5	367.0	380.5
Polo-box domain	2	50.0	65.1	27.0	50.0	73.0
Protein kinase-like (PK-like)	94	173.5	157.4	36.8	123.5	273.2
Retrovirus capsid dimerization domain-like	2	13.0	17.0	7.0	13.0	19.0
Rudiment single hybrid motif	4	245.2	146.1	171.0	184.0	258.2
SH3-domain	2	212.8	97.2	178.4	212.8	247.1
Terpenoid synthases	4	147.8	166.2	41.1	99.8	206.4
Trypsin-like serine proteases	12	168.6	173.9	20.5	111.0	256.0
Type 2 solute binding protein-like	4	19.1	22.4	7.2	9.5	21.4
WGR domain-like	4	178.0	126.5	129.0	202.5	251.5
WW domain	4	23.2	29.9	2.8	12.5	33.0
Unknown	12	144.7	164.5	26.8	85.5	196.2

Table 3.2: Performances of IFPP across superfamilies. The number of scaffold hopping experiments, as well as the mean, standard deviation (STD) of the ranks of the unknown active are reported for each superfamily. We also provide the 3-quantiles (25%, 50% and 75%) of those distributions of ranks.

3.2.2 Contributions of Proteins in the Panel

In order to further characterise the results of this method, we quantified the influence of each protein pocket considered to build the IFPP in the retrieval of the unknown active. Indeed, some of these pockets could have a prejudicial effect on the performances, whereas other may significantly contribute to the better ranking of the ligands, when using the IFPP descriptors.

To evaluate the contribution of the proteins of the panel in the ranking of the unknown active, we first categorized each of the 288 similarity searching experiments according to its ranking of the unknown active with the IFPP similarity. We defined 7 categories:

- $Rank \leqslant 25$
- $25 < Rank \leqslant 50$
- $50 < Rank \leqslant 100$
- $100 < Rank \leqslant 200$
- $200 < Rank \leqslant 300$
- $300 < Rank \leqslant 400$
- $400 < Rank$

For each category and each protein, we computed the IFP similarity between the known active and the unknown active, as well as the median IFP similarity between the known active and its decoys, for all corresponding experiments. Comparing these two quantities provides useful information on the influence of the protein in the ranking.

Then, for each protein, we performed T-tests with the null hypothesis being that the similarity between ligands is either less or greater than the median similarity between the known active and the decoys. Thus, the p-values obtained provide an indication on how contributive or counteracting a protein is for each binned class of ranks.

P-values of the T-test with the alternative hypothesis being that the IFP similarity between the ligands is greater than between the known active and the decoys are displayed in panel a) of Figure 3.4. The lower the p-value, the more likely the alternative hypothesis is. As we can see, in all pockets the IFP similarity between the ligands is significantly greater (p-value $\leqslant 0.05$) than with the decoys when the experiment is successful (rank $\leqslant 25$), even though some proteins seem to contribute more: pockets '3sff' and '6mob' have very low p-values compared to '4gvm' and '2ovx' that are barely significant. This means that all pockets contribute to the successful experiments. In general, for the unsuccessful cases (rank of the unknown active $\geqslant 25$), only a few proteins bring a positive signal ('1t48', '6ccy', '4pv5', etc), while most show insignificant differences between the IFP similarity of ligands and the IFP similarity of the known active to the decoys.

On the contrary, when trying to explain why in some experiments the unknown active is ranked very high (rank $\geqslant 400$), a few pockets display low p-values for the alternative hypothesis that the IFP similarity between the ligands is less than between the known active and the decoys, displayed in panel b) of Figure 3.4. This means that

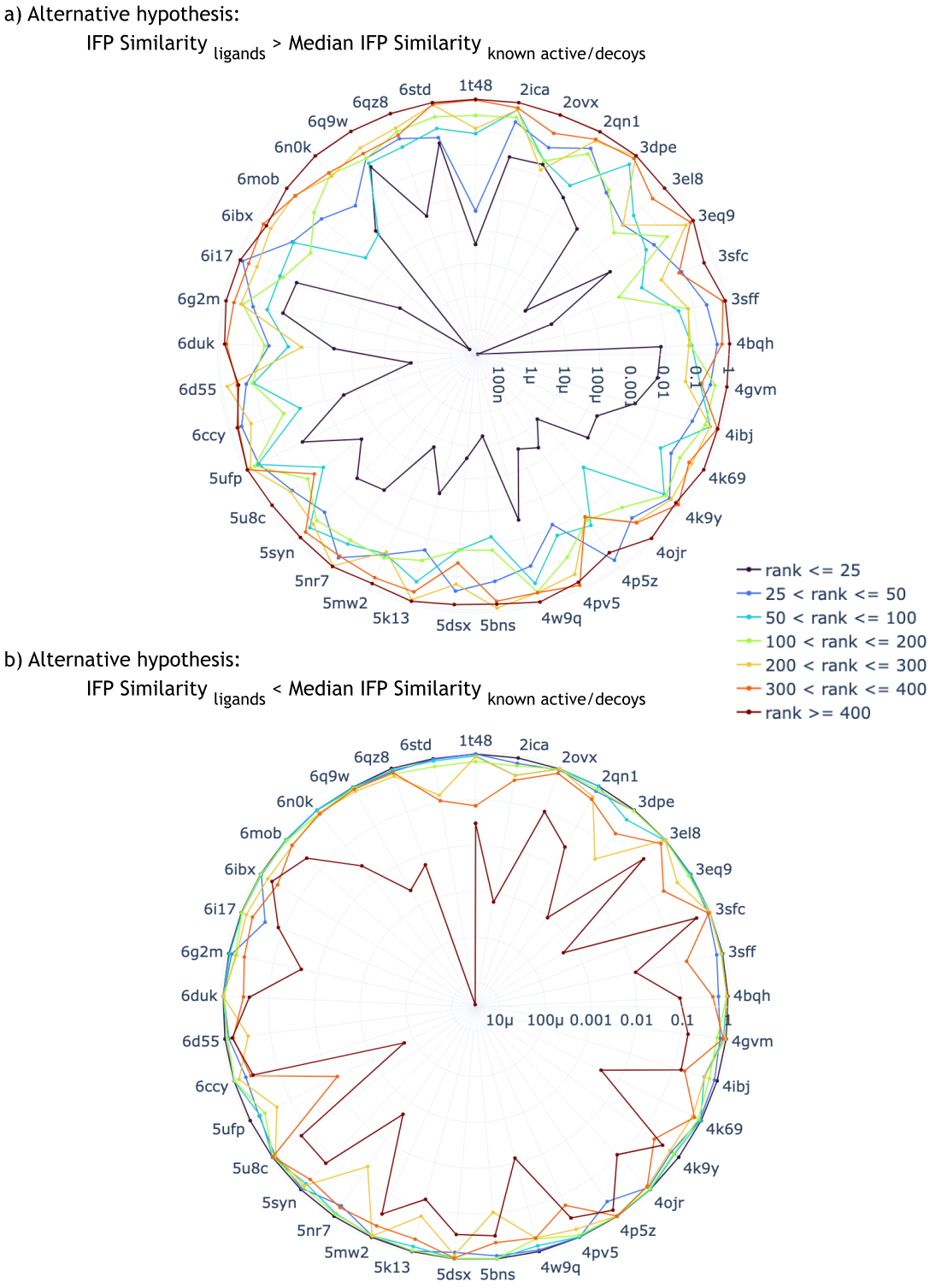


Figure 3.4: T-tests comparing the ligands IFP similarity and the IFP similarity between the known active and decoys across pockets represented through polar graphs. The radius corresponds to the p-value in logarithmic scale, and the edges are proteins of the panel. The lower the p-value, the more likely the alternative hypothesis stands for the considered protein in the studied rank category. Panel a) (resp. b)) displays the p-values of the alternative hypothesis "IFP similarity between the ligands is greater (resp. less) than between the known active and the decoys".

those pockets have a significant deleterious effect, and may explain the poor performances of the IFPP in some cases. It is not surprising, as pockets have been arbitrarily chosen to construct this molecular representation, and not to maximise the success rate on the benchmark. Proteins '5ufp' and '6std' are amongst them, and there is no doubt that removing them would increase the performance of our method.

One explanation of the contribution of proteins in accurately ranking the unknown active could be that, if the protein in which we dock is similar to the protein for which we observed the scaffold hopping pair, then for this protein the ligands will have a higher IFP similarity than the known active with the decoys. The rationale behind this claim is that, as the two proteins are close, the ligands might reproduce similar binding modes as initially observed. To evaluate this hypothesis, for each case and each protein, we computed:

- The sequence similarity between the protein used for docking and the protein of the scaffold hopping case using the Needleman-Wunsch algorithm [Needleman et Wunsch(1970)], which reconstructs the optimal alignment between sequences of amino acids by assigning scores to matches, mismatches and gaps. We used the BLOSUM62 substitution matrix [Eddy(2004)] to score the alignments. We did not penalise gaps during the scoring process here.
- The IFP similarity between the known active and the unknown active in the considered protein,
- The median IFP similarity between the known active and its corresponding decoys.
- Finally, we calculated the difference between those two values:
  $IFP\ Similarity_{\ ligands} - Median\ IFP\ Similarity_{\ known\ active,\ decoys}$.

We plotted in Figure 3.5 the variation of this difference with the sequence similarity for all cases and all proteins, splitting by the rank categories defined above. We observe that there is no correlation: high differences in IFP similarity are observed with any value of sequence similarity. Conversely, low differences in IFP similarity exist for high sequence similarity. We also notice the overall low sequence similarity between the proteins of the panel and the proteins of the *LH* benchmark (gaps were not penalised), demonstrating they were chosen arbitrarily and not to optimise the performance on the *LH* benchmark.

The signal is blurred for most, meaning that though all proteins contribute in the successful cases, the protocol applied for the protein selection might be improved. However, the criteria for such an enhancement are not clearly identified, and might be specific to the benchmark, hence lack generalisation.

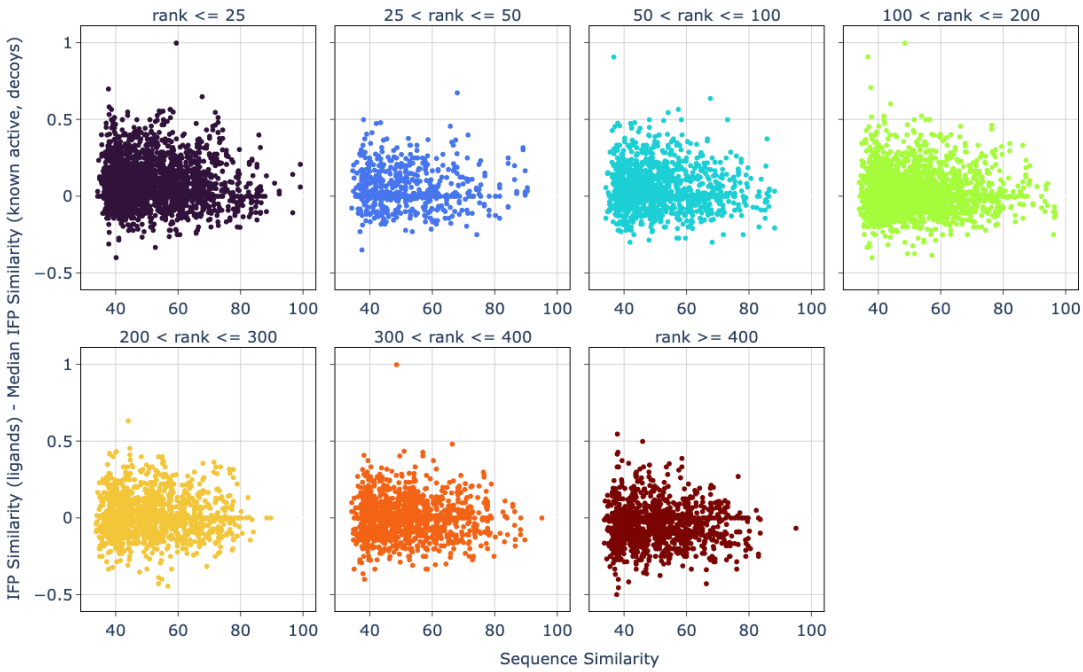


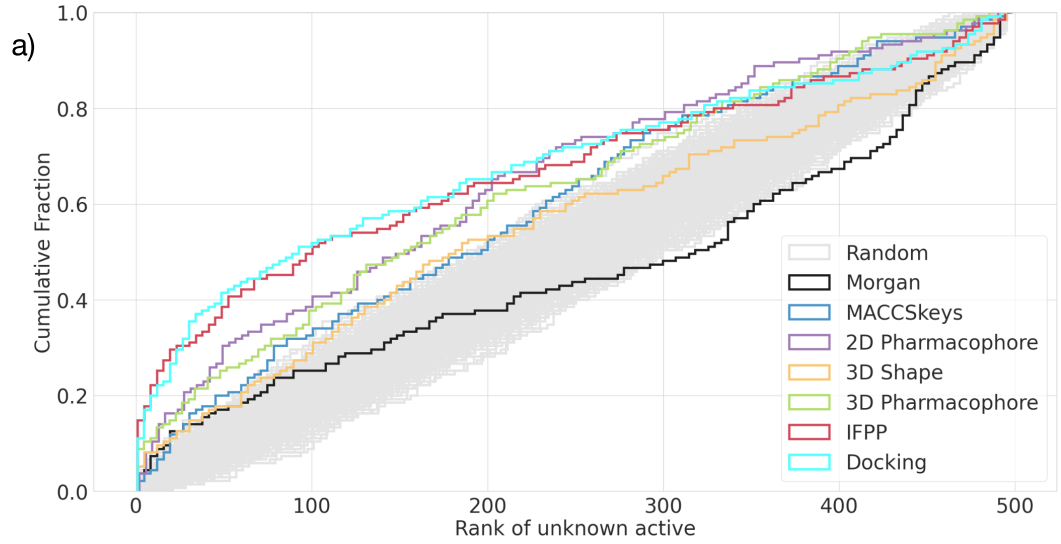
Figure 3.5: Variation of the difference between IFP Similarity(ligands) and Median IFP Similarity(known active, decoys) with the sequence similarity between the protein in which the molecules are docked and the protein for which we observed the scaffold hopping pair.

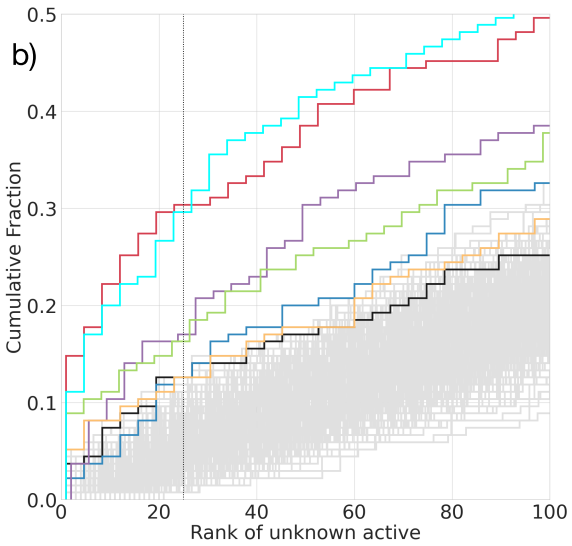
3.2.3 Comparison to Docking

Interestingly, in the *LH* benchmark, the 3D structures of the proteins targeted by the 144 scaffold hopping pairs are known, because these pairs were initially extracted from protein-ligand complexes of known 3D structures available in PDBbind [Wang *et al.*(2004b)]. In such cases, docking would be the reference method to solve scaffold hopping problems. Therefore, we assessed the performance of docking on this benchmark.

We prepared the structures for docking as explained in 3.1.3. In fact, for each scaffold hopping pair in the benchmark, two PDBs for the same protein are available, i.e. one complex for each of the two ligands. The crystallographic ligands were redocked inside their prepared pockets, and we computed the RMSD between the best docked pose according to the Grid Score and the crystallographic pose. Cases for which the RMSD was above 2.5Å were discarded as the preparations steps and docking protocol are not calibrated, and fixing the preparation steps in a case-dependant manner would have been extremely time-consuming. Thus, docking was assessed only for the 135 scaffold hopping experiments of the *LH* benchmark for which preparation of the PDBs succeeded. For each considered scaffold hopping case, the unknown active and the 499 decoys were ranked according to their docking score in the PDB structure of the known active to reproduce a real-life screening situation with docking. The performance of docking was assessed based on the percentage of cases for which the unknown active is ranked in the top 5%.

Docking retrieves the unknown active in the top 5% in 28.9% out of 135 considered scaffold hopping experiments. This performance would probably be higher in real-life cases, when a single protein target is considered. The docking protocol would then be finely tuned for this particular protein, which was not done in our benchmark application. Assuming that they would still remain in the same range, these performances are modest. This illustrates that large-step scaffold hopping is on average a difficult task, even when the structure of the protein is known. Note that on the same subset of 135 scaffold hopping experiments, the IFPP similarity measure had a success rate 30.4%, which is higher, although comparable to docking, as displayed in Figure 3.6.





c)

Method	Percent in Top 5%
Morgan	12.6
MACCSkeys	11.9
2D Pharmacophore	16.3
3D Shape	11.9
3D Pharmacophore	15.6
IFPP	**30.4**
Docking	28.9

Figure 3.6: Performance of docking on a subset of the *LH* benchmark. The cumulative histogram curves of each method are plotted in a). A zoom of the same graphs is provided in b) with vertical grey lines corresponding to ranks of top 5% ranks. Table c) displays the percentage of successful scaffold hopping problems for the approaches, according to a rank of the unknown active in the top 5%.

3.2.4 Application on a Kinase Subset

The IFPP represents an estimation of the possible binding modes of a molecule, according to the panel of diverse considered proteins. The higher the number of proteins, the better the estimation is expected to be. Without using any knowledge about the targeted protein and for which hits are searched, we demonstrated that the IFPP encoding from which a similarity-based method is applied provides a promising approach for solving large-step scaffold hopping cases.

However, when targeting a specific protein belonging to a well studied family, such a representation might not be optimal. Indeed, a more relevant approach may be to select a panel of proteins belonging to the same family, to derive an estimation of the

possible binding modes within the target protein family. Such a representation would take into account the intricacies and specific features of the allowed binding modes of the protein family. It may be more appropriate than a global estimation of possible binding modes based on proteins belonging to the whole human proteome.

Again, we illustrate this property on the protein Kinase family. Kinases are a type of enzyme that catalyze the transfer of phosphate groups from high-energy donor molecules, such as ATP (adenosine triphosphate), to specific target proteins. This phosphorylation plays a crucial role in cellular signaling pathways, regulating various cellular processes such as cell growth, differentiation, metabolism, and apoptosis (programmed cell death). By adding phosphate groups to proteins, kinases can modify their activity, localization, stability, and interactions with other molecules. Dysregulation of kinase activity is associated with various diseases, including cancer, inflammatory disorders, and neurodegenerative diseases. The *LH* benchmark contains 20 different kinases, corresponding to 47 scaffold hopping pairs, allowing to conduct this study according to the protocol described below.

Defining a Kinase Panel of Proteins. To compare IFPPs derived from a diverse protein panel to IFPPs derived from kinases to solve kinase-related scaffold hopping cases, we first selected the kinases that will define a kinase panel to compute the IFPP. We followed exactly the same steps as in 3.1.3, starting from PDBbind [Wang *et al.*(2004b)]:

- Only proteins belonging to the protein kinase superfamily as defined by the SCOP database [Murzin *et al.*(1995)] are considered,
- PDBs should have a resolution below 2.5Å,
- They should include "drug-like" ligands, as defined in 3.1.3,
- PDBs with redundant ligands are discarded,
- Only one PDB is kept for each protein,
- PDBs are subsequently prepared for docking,
- Only the structures with a successful redocking of the crystallographic ligand ($RMSD \leqslant 2.5$Å) are kept.

To limit the number of dockings to be performed, only 10 kinases were kept, leading to a total of 11 considered kinases (adding one kinase that was present in the panel of diverse proteins). Table 3.3 provides information on the retained kinases. All molecules (i.e. ligands and corresponding decoys) associated to the 47 kinase-specific scaffold hopping cases of the *LH* benchmark were subsequently docked in each of those selected kinases, leading to a total of 235,470 dockings.

Comparison of IFPPs. To compared the performances at solving scaffold hopping (i.e. ranking the unknown active in the top 5%) between the kinase-specific and the diverse proteins IFPPs, for different sizes of the panel of proteins. For each size, one hundred draws amongst available proteins (11 proteins for kinases, 37 for the diverse

PDB	UniProt	Protein
4i5h	P63086	MITOGEN-ACTIVATED PROTEIN KINASE 1
6slg	P28482	MITOGEN-ACTIVATED PROTEIN KINASE 1
6ccy	P31749	RAC-ALPHA SERINE/THREONINE-PROTEIN KINASE,PIFTIDE
6mob	P10721	MAST/STEM CELL GROWTH FACTOR RECEPTOR KIT
3wf8	P23443	RIBOSOMAL PROTEIN S6 KINASE BETA-1
5vee	O96013	SERINE/THREONINE-PROTEIN KINASE PAK 4
2ivu	P07949	PROTO-ONCOGENE TYROSINE-PROTEIN KINASE
5lmk	P24941	CYCLIN-DEPENDENT KINASE 2
5l2s	Q00534	CYCLIN-DEPENDENT KINASE 6
4eqc	Q13153	SERINE/THREONINE-PROTEIN KINASE PAK 1
3bi6	P30291	WEE1-LIKE PROTEIN KINASE

Table 3.3: All kinases used for the panel of proteins. Note that the two first belong to the same protein, but not in the same conformation: '4i5h' corresponds to the "DFG-out" conformation and '6slg' to the "DFG-in" conformation. This phenomenon translates in a flip of residues creating a new allosteric pocket, which leads to different binding sites and thus different IFPs.

set) are performed to get more meaningful comparisons and draw statistical conclusions. For each draw of proteins that define the IFPP, the proportion of experiments for which this representation ranked in the unknown active in the top 5% is computed for all the $47 \times 2 = 94$ kinase-related scaffold hopping experiments.

The box plots of success rates per sizes of protein panels for the kinase-specific and the diverse IFPP for all draws are gathered in Figure 3.7. Kinase-specific IFPP tend to better rank the unknown active than the diverse IFPP. The gap between both representations increases with the number of proteins used to define the IFPP, as shown by Figure 3.7. This demonstrates that, at least for kinases, defining an IFPP based on a panel of protein close to the targeted protein is expected to perform better than a panel of diverse proteins. However, this property might be somewhat overestimated because the above results were obtained of a highly structurally conserved family of proteins. We did not perform this analysis on other superfamilies of the benchmark for technical reasons (not enough scaffold hopping cases, high computational cost). Besides, for most proteins apart from kinases, we do not have enough scaffold hopping cases to provide a statistical conclusion.

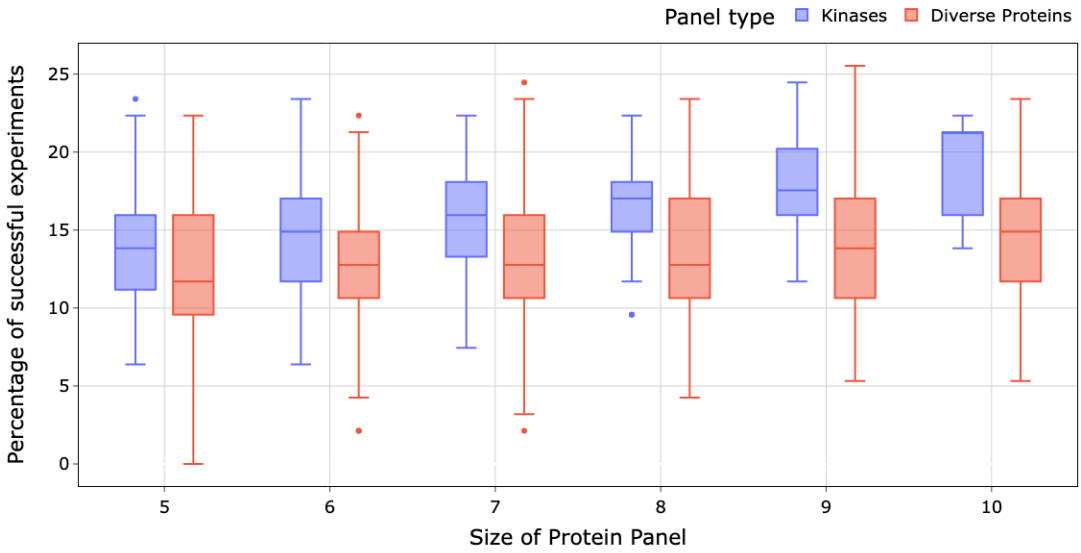


Figure 3.7: Comparison of success rates between the diverse IFPP and the kinase-specific IFPP with different sizes of the protein panel. The latter surpasses the former systematically, the median gap between both increasing with the size of the protein panel.

3.3 Conclusion

The main contribution in this Chapter was to propose a novel molecular representation dedicated to the scaffold hopping problem: the Interaction Fingerprints Profile (IFPP). This profile is a representation that intend to capture possible binding modes of molecules, based on docking experiments in a panel of 37 diverse proteins. The IFPP was computed using a single docking protocol that was not optimized for each of the 37 proteins. However, the successful re-docking of the crystallographic ligand present in the binding sites for these proteins indicates that the docking protocol was adapted to these structures, if not optimized. Future studies could consider evaluating several docking scores or using different docking software. This may allow to define consensus IFPPs with increased relevance with respect to the scaffold hopping problem.

We showed that increasing the number of proteins in the panel improves the performances of the IFPP descriptors, but one needs to find a compromise between improving the performances and increasing the cost of computing larger IFPP descriptors (see below). We consider that the Interaction Fingerprints Profile is essentially a new string to the bow of available methods for addressing these challenging problems. This representation should not replace others, but should rather be used in conjunction with other existing methods. Indeed, as shown in the Results 3.2.1, combining IFPP and 3D pharmacophore similarities increases the success rate on the *LH* benchmark.

Besides, as illustrated with the kinase subset in 3.2.4, when dealing with a specific protein with extensive knowledge of its family, choosing a panel of proteins belonging to the same family may improve the ability of IFPP to solve scaffold hopping. Such a representation would provide a more accurate understanding of potential binding modes within the family of target proteins, thus be more adapted to the protein at hand. This

bring a new rationale for the protein selection to build the IFPP in a real-life setting where hits are searched for only one protein target: picking proteins belonging to the same family.

However, a drawback of the IFPP is its computational cost. Indeed, building the IFPP of a molecule requires its docking in 37 protein pockets. For example, to perform our study on the *LH* benchmark, a total of $2,669,328$ docking experiments were required to compute the IFPPs of all molecules in the benchmark. This included 144 pairs of ligands and their 499 decoys docked in 37 pockets. In real applications, solving a given scaffold hopping problem by screening 100.000 molecules (i.e. medium-size chemical libraries) would require $3,700,000$ docking experiments, which is accessible.

Nevertheless, screening of very large chemical libraries (millions of molecules) using IFPP molecular representations would lead to computational burdens. Due to the cost of docking used to define the IFPP, as it stands, this encoding is not scalable to screen large molecular libraries (millions of molecules), although in the case where a large library would be screened recurrently, it would be feasible to calculate this embedding only once.

4

Overcoming Limitations Through Deep Learning

Abstract:

We illustrated how computationally expensive IFPP is, which limits its use for screening very large molecular libraries. We propose to overcome this limitation by leveraging two different Deep Learning approaches: one trying to predict IFPs for each protein of the panel, the other relying on Metric Learning concepts. The latter allows fast estimation of molecules IFPP similarities, thus providing an efficient pre-screening strategy that is applicable to very large molecular libraries. We illustrate on an external dataset, LIT-PCBA, how such a method can help identify new hits in a more realistic drug discovery setting.

Résumé:

Nous avons illustré à quel point l'IFPP est coûteux en calcul, ce qui limite son utilisation pour le criblage de bibliothèques moléculaires très volumineuses. Nous proposons de surmonter cet inconvénient en exploitant deux approches de Deep Learning différentes : l'une visant à prédire les IFPs pour chaque protéine du panel, l'autre reposant sur des concepts de Metric Learning. Cette dernière permet une estimation rapide des similarités entre les IFPPs des molécules, offrant ainsi une stratégie de pré-criblage efficace applicable aux bibliothèques moléculaires très volumineuses. Nous illustrons sur un ensemble de données externe, LIT-PCBA, comment une telle méthode peut aider à identifier de nouveaux hits dans un contexte plus réaliste de découverte de médicaments.

Contents

As it stands, the Interaction Fingerprints Profile is limited to small to medium sized chemical libraries of hundred thousands of compounds due to the cost of its computation. However, solving large-step scaffold hopping requires roaming remote chemical spaces in search for new hits. This strategy may require scoring a number of molecules several orders of magnitude higher than our method allows. In this Chapter, I describe two approaches relying on Deep Learning (DL) to overcome the computation cost limitations of the IFPP. I also illustrate how the most promising one, based on Metric Learning concepts, can be used in virtual screening through an external hit discovery benchmark.

4.1 Prerequisites

Before describing the models designed to democratise the Interaction Fingerprints Profile by addressing its cost and computation burdens, I recall some basic concepts employed throughout this Chapter. In particular, we built *Deep Learning* models with *Graph Neural Network* architectures. The goal is not to perform an extensive review of Deep Learning and Graph Neural Networks but rather recall key concepts before dwelling on more sophisticated architectures and ideas. For more information about those domains, the reader can browse [Goodfellow *et al.*(2016), Hamilton *et al.*(2018)].

4.1.1 Deep Learning

In Machine Learning (ML), a neural network is a model inspired by the structure and function of biological neural networks in animal brains. It consists in connected units that mimic the functioning of a neuron, which is an electrically excitable cell that transmits action potentials via synapses to other cells through the nervous system. In this analogy, artificial neurons are connected by edges, which model the synapses, and transmit "signals", which are real numbers, to other neurons of the network. Each artificial neuron outputs a value computed by some non-linear function of the sum of its inputs, called the activate function. The magnitude of the signal at each connection is governed by a weight (or parameter), which undergoes adjustment throughout the learning process.

Usually, neurons are organized into layers, each layer potentially executing distinct transformations on its inputs. Signals propagate from the initial layer (the *input layer*) to the last layer (the *output layer*), potentially traversing several intermediate layers (*hidden layers*). A neural network is commonly referred to as a deep neural network (DNN) when it possesses at least 2 hidden layers [Bishop(2006)].

This gave birth to Deep learning (DL), a sub-field of ML that focuses on artificial neural networks with multiple layers. These neural networks are capable of learning complex representations of data by composing multiple layers of non-linear transformations. DL has revolutionized various fields by enabling the development of highly accurate and flexible models for tasks such as image recognition [Taigman *et al.*(2014)], natural language processing [Hochreiter et Schmidhuber(1997)], speech recognition [Hannun *et al.*(2014)], and reinforcement learning [Mnih *et al.*(2015)].

We provide the mathematical formalism for an example of DNN architecture with 3 hidden layers in the following, summarised in Figure 4.1.

Let x denote the input vector of size n, and y denote the output vector of size k. The DNN consists of 3 hidden layers each with m neurons.

The input layer is denoted by $x = (x_1, x_2, \ldots, x_n)^T$, where x_i represents the i-th input feature.

For the l-th hidden layer, $l = 1, 2, 3$, the output vector $h^{(l)}$ is computed as follows:

$$z^{(l)} = W^{(l)} h^{(l-1)} + b^{(l)},$$
$$h^{(l)} = \sigma(z^{(l)}),$$

where $W^{(l)}$ is the weight matrix of size $m \times m$, $b^{(l)}$ is the bias vector of size m, $\sigma(\cdot)$ is the activation function that introduces non-linearity, and $h^{(0)} = x$.

Finally, the output layer computes the output vector y as:

$$y = W^{(4)} h^{(3)} + b^{(4)},$$

where $W^{(4)}$ is the weight matrix of size $k \times m$ and $b^{(4)}$ is the bias vector of size k.

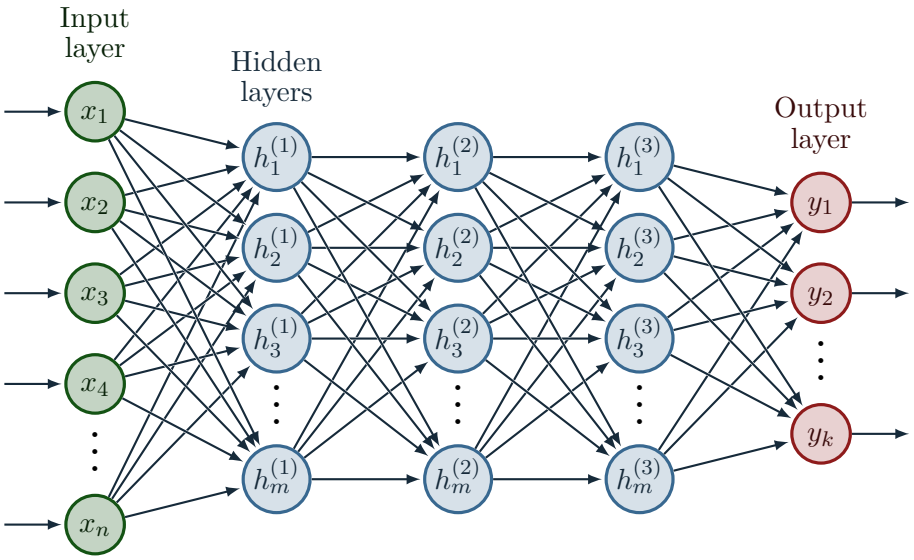


Figure 4.1: Illustration of Deep Neural Network architecture. In this example, the input layer receives external data, which successively go through three hidden layers with m neurons each. Finally, the output layer contains the ultimate predicted result.

During training, the DNN learns to optimize its parameters (weights and biases) to minimize the loss function, which measures the difference between the predicted and actual values. It is typically trained using a process called back-propagation. It involves comparing the predicted output of the network with the actual output (ground truth) using the loss function. Then, the gradient of the loss function with respect to the weights and biases of the network is computed, using the chain rule of calculus. This step involves propagating the error backward through the network. The weights and biases of the network are adjusted in the opposite direction of the gradient to minimize the loss function. This step is usually performed using optimization algorithms such as stochastic gradient descent (SGD) or its variants. Those steps are repeated for multiple epochs (iterations) until the model converges to a satisfactory solution or

reaches a predefined stopping criterion.

DL has demonstrated success across diverse tasks essential for drug discovery, including property and activity prediction [Mayr *et al.*(2016), Ma *et al.*(2015)], *de novo* design [Gómez-Bombarelli *et al.*(2018), Segler *et al.*(2018)], and prediction of ligand–protein binding modes [Krishna *et al.*(2024)].

4.1.2 Graph Neural Networks

Graph Neural Networks (GNN) are a class of neural networks designed to operate on graph-structured data. This class of algorithms has gained momentum since 2016 [Li *et al.*(2019b), Hamilton *et al.*(2018), Veličković *et al.*(2018), Xu *et al.*(2019)]. Graphs consist of nodes (vertices) and edges (connections), representing entities and relationships between them, respectively. Mathematically, a graph G can be defined as a tuple $G = (V, E)$, where:

- V is the set of vertices (nodes) in the graph.
- E is the set of edges (connections) between vertices.

Let $V = \{v_1, v_2, \ldots, v_n\}$ denote the set of vertices, and $E = \{e_1, e_2, \ldots, e_m\}$ denote the set of edges. Each edge e_i can be represented as a tuple (v_j, v_k) indicating the connection between vertices v_j and v_k.

A graph G can also be represented by an adjacency matrix A, where $A_{ij} = 1$ if there is an edge between vertices v_i and v_j, and $A_{ij} = 0$ otherwise.

The degree $d(v_i)$ of a vertex v_i is the number of edges incident to v_i. For an undirected graph, it is given by:

$$d(v_i) = \sum_{j=1}^{n} A_{ij}$$

For a directed graph, the in-degree $d_{\text{in}}(v_i)$ and out-degree $d_{\text{out}}(v_i)$ of a vertex v_i represent the number of incoming and outgoing edges, respectively.

GNNs extend traditional neural networks to handle such non-Euclidean data structures. In GNNs, each node in the graph is associated with a feature vector representing its attributes. The goal of GNNs is to learn either node or whole graph representations by aggregating information from neighboring nodes and their features.

The key components of GNNs include:

- *Node Embeddings*: GNNs learn low-dimensional representations (embeddings) for each node in the graph, capturing both structural and attribute information.
- *Message Passing*: GNNs propagate information between neighboring nodes through message passing. At each layer, nodes aggregate information from their neighbors and update their own representation based on the aggregated information.
- *Aggregation Function*: GNNs use aggregation functions to combine information from neighboring nodes. Common aggregation functions include summation, averaging, or weighted aggregation based on attention mechanisms.

- *Graph Convolutional Layers*: Graph convolutional layers are the building blocks of GNNs, performing message passing and aggregation operations. These layers typically consist in a message passing step followed by a node-wise aggregation step.

- *Pooling and Readout*: GNNs often incorporate pooling layers to aggregate information across nodes or subgraphs, as well as readout functions to generate graph-level representations for downstream tasks.

Graph Neural Networks have emerged as powerful tools in drug discovery due to their ability to model and analyze graph-structured data, which naturally represents molecules and their interactions [Duvenaud *et al.*(2015), Li *et al.*(2017), Brocidiacono *et al.*(2024)]. Indeed, a molecule can be represented by a 2D (or 3D in some contexts) graph $G = (V, E)$ where the nodes V represent the atoms and the edges E represent the bonds.

I provide an example of such an architecture, which achieved state-of-the-art predictions to a wide range of molecular properties [Xiong *et al.*(2020)], called *Attentive FP*.

Attentive FP This GNN architecture relies on the attention mechanism [Vaswani *et al.*(2017)], and takes into account edge embeddings in the message passing steps of the nodes. I detail the workflow for a molecule M, where v are atoms of M, u are neighbours of v, $\mathbf{h}$ corresponds to embeddings, W is a trainable matrix and i is the i-th attentive layer. Attentive FP consists in 3 steps:

(1) Initial atom and bond embedding to vectors of same length. Initial node embeddings are obtained by the concatenation of the two previous vectors.

(2) Stacked attentive layers performing message passing with an attention mechanism to update node embeddings, aggregating information from neighbouring nodes through the following process. For each node v of the graph:

- Compute the alignment for each neighbour u:

$$\mathbf{e}_{vu}^{i-1} = \text{LeakyReLU}(W \cdot [\mathbf{h}_v^{i-1}, \mathbf{h}_u^{i-1}]) \tag{4.1}$$

- Compute the weight (attention) of this neighbour with the softmax:

$$\mathbf{a}_{vu}^{i-1} = \frac{exp(\mathbf{e}_{vu}^{i-1})}{\sum_{u \in N(v)} exp(\mathbf{e}_{vu}^{i-1})} \tag{4.2}$$

- Compute the context of the node v:

$$\mathbf{C}_v^{i-1} = \text{elu}(\sum_{u \in N(v)} \mathbf{a}_{vu}^{i-1} \cdot W \cdot \mathbf{h}_v^{i-1}) \tag{4.3}$$

- The node embedding is updated using a Gated Recurrent Unit (GRU) [Chung *et al.*(2014)]:

$$\mathbf{h}_v^i = \text{GRU}(\mathbf{C}_v^{i-1}, \mathbf{h}_v^{i-1}) \tag{4.4}$$

(3) To get the final graph embedding of a molecule, the entire graph is treated as a supervirtual node (summing all node embeddings) that goes through stacked attentive layers (as detailed above) which output a state vector for the whole molecule.

To sum up, first, a molecule is encoded by a 2D graph $G = (V, E)$ where the nodes V represent the atoms and the edges E represent the bonds. Initial state vectors of same length for each node and edge are obtained with a fully connected input layer. Then, GNN layers perform message passing on the node embeddings using an attention mechanism to include local information of the relevant neighbouring atoms. The message passing mechanism relies on a context vector incorporating neighbouring node and edge embeddings that goes through a gated recurrent unit GRU that updates the state vector of the node. To get the final embedding of the molecule, an initial molecule state vector is obtained by summing all node embeddings. Then, a readout block consisting in two attentive pooling layers is applied. In each pooling layer, a context vector of the molecule is computed using an attention mechanism on all node embeddings, which goes through a GRU that updates the molecule embedding.

4.2 IFP Prediction per Protein

Docking and identifying interactions in each protein to define the IFPP require significant resources in terms of cost and time. In this section, we propose an alternative strategy to bypass these expenses: directly predict the IFPP for any molecule, using DL. The approach involves building a separate model for each protein belonging to the panel. These models are designed to predict how a given input molecule interacts with its corresponding protein. By predicting the interaction fingerprints (IFP) with the protein, these models supplant the need for docking, i.e. the bottleneck of IFPP. The aggregation of predictions from these models generates a predicted IFPP. Once the models are trained, predicting the IFPP for a new molecule becomes rapid and straightforward, easing the use of this representation in virtual screening.

I illustrate in the following subsections the architecture of a model that predicts the IFP for a single protein target.

4.2.1 Model Architecture

As explained in 2.1.2, the IFP is a target-focus binary vector encoding protein-ligand interactions between the target protein and a molecule. In particular, each residue of the protein is associated to a ten-long binary vector informing on how it interacts with a molecule considering the ten possible interactions described in 2.1.2. The concatenation of all those vectors forms the IFP of a molecule.

To recreate such a design, we built the IFP Predictor model so that it is composed of blocks, each specialised in predicting, for a molecule, how it interacts with a specific residue. The aggregation of the outputs of those residue-specific blocks forms the predicted IFP. To ensure that the model keeps a general idea of the whole molecule, and that residue-specific blocks share information, we also added a block outputting a graph embedding that is incorporated to all other blocks.

Concretely, we distinguish two types of blocks:

- *The shared block*: a GNN with a readout function to get a graph embedding of the molecule.
- *Residue-specific blocks*: each is composed of two parts. The first is a GNN with a readout function. To the output embedding of this GNN is concatenated the embedding of the shared block. Then, this aggregated embedding serves as input to a multilayer perceptron (MLP) that outputs the predicted protein-ligand interactions between the molecule and the residue expressed by the block.

Figure 4.2 summarises the architecture of the model. The number of residues in the binding site of the studied protein has to be determined, so that only possible interacting amino acids are considered. We used Attentive FP, the Graph Neural Network (GNN) described in 4.1.2 using the attention mechanism proposed by [Xiong *et al.*(2020)] to encode molecules with a readout function, in order to obtain a graph representation for each molecule. Each GNN of Figure 4.2 represents one Attentive FP architecture, composed of three GNN layers perform message passing, a GRU that updates the state vector of the nodes, and finally a readout block consisting in two attentive pooling layers. We used the python packages `dgl-life` [Li *et al.*(2021)] and PyTorch [Paszke *et al.*(2019)] to implement the corresponding architecture.

During the training phase, we chose to minimise the *Binary Cross Entropy* 4.5 of the concatenated predicted outputs:

$$BCE(y, \hat{y}) = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)] \tag{4.5}$$

Where:

- y is the ground truth (true interactions)
- $\hat{y}$ is the predicted probabilities
- N is the batch size

The weights of all blocks are updated simultaneously by back-propagating this loss.

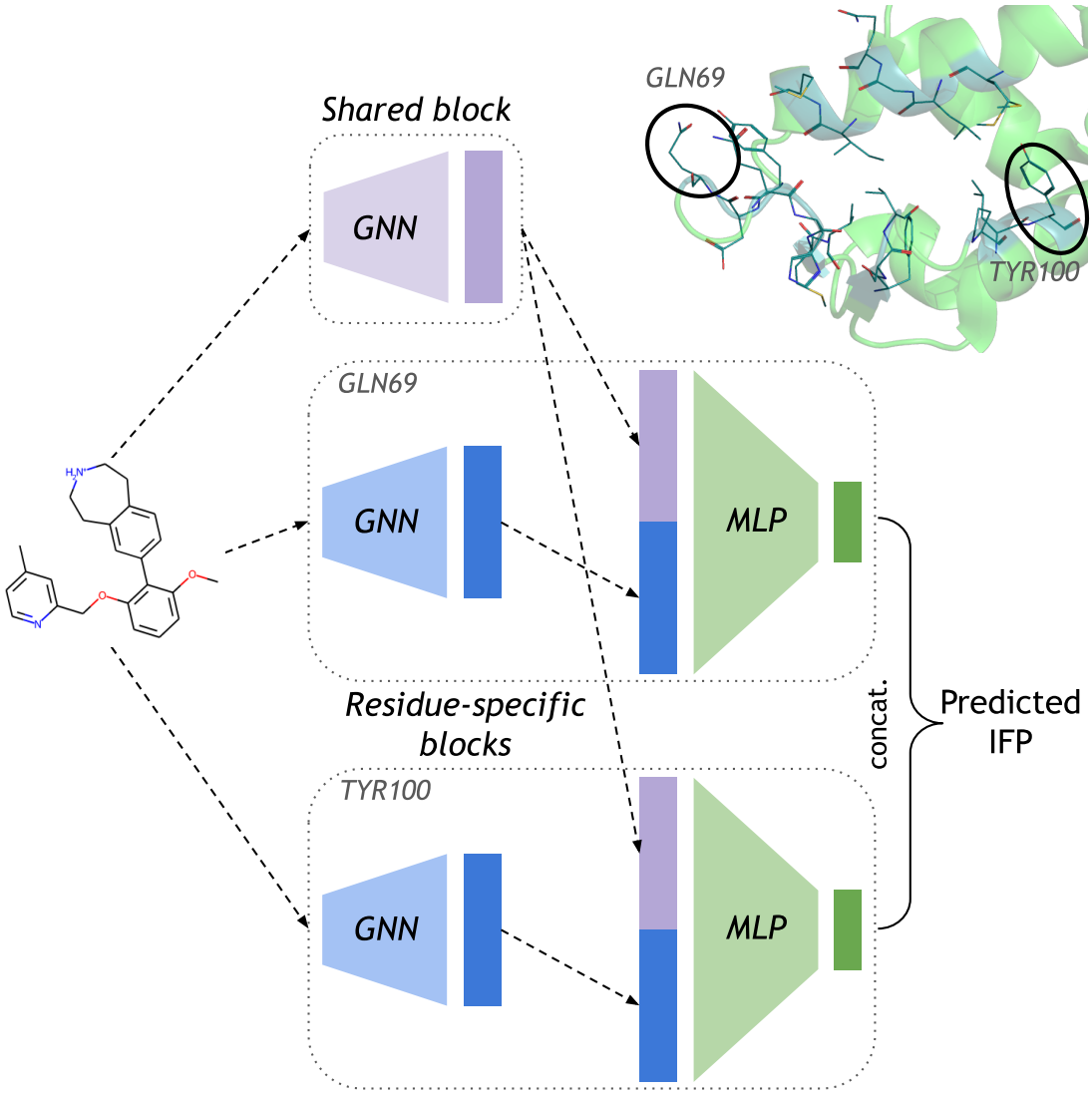


Figure 4.2: Architecture of IFP Predictor for a given protein. I illustrate the architecture for two residue-specific blocks, corresponding to residues *GLN69* and *TYR100*. The graph embedding obtained from the shared block is concatenated to each of the graph embeddings inside the residue-specific blocks, which then go through a MLP to output predicted interactions for each residue. The aggregation of those outputs creates the predicted IFP.

4.2.2 Training Dataset

The objective of this study is to develop a computational model capable of predicting the interactions between small molecules and a given protein pocket. To train this model, a comprehensive dataset containing crystallographic poses of diverse molecules within the target protein pocket would be required. Unfortunately, such a dataset is not readily available due to the extensive resources required for its compilation. An alternative approach involves leveraging docking datasets, wherein large collections of molecules have been virtually docked into the protein pocket. These datasets can be used to train the IFP prediction models.

Furthermore, using these docking datasets for model training is consistent with our prior use of docking poses to build the Interaction Fingerprints Profile.

A few ultra-large docking libraries have been published in the literature [Lyu *et al.*(2019), Stein *et al.*(2020), Sadybekov *et al.*(2020), Gorgulla *et al.*(2020)], gathering several millions to more than a billion compounds. Before handling such large datasets, we performed a proof of concept of the proposed approach. Indeed, in order to evaluate the interest of IFPP-derived similarity measures for solving scaffold hopping cases in the *LH* benchmark, we already performed docking of more than 70,000 molecules in 37 proteins pockets. Therefore, we started from these available docking results to train the IFP algorithm models.

We chose the protein MDM4 from the protein panel, a human protein contributing to TP53 regulation, because of its relatively small binding site involving few residues, as shown in Figure 4.3. The binding site comprises 23 residues, each can form up to ten possible protein-ligand interactions with a molecule, so that the IFP to predict is of length 230. The model trained on this dataset is thus composed of 23 residue-specific blocks, each aiming at predicting how a molecule would interact with this residue, as well as a shared block. More precisely, we train a multitask model that simultaneously predicts the binary values of the 230 bits defining the IFP for the MDM4 protein (one task corresponds to predicting one bit of the IFP).

The 71,856 molecules from the *LH* Benchmark are mostly decoys that cover a wide chemical space, and do not contain any chemical series. Therefore, in a first draft, we performed a 80/10/10 random split to create the train, validation and test sets.

4.2.3 Results

We trained the multi-task model to predict the IFP on MDM4 for 50 epochs. As a metric to evaluate the model, we computed the Tanimoto similarity between the predicted IFP and the true IFP (i.e. the IFP computed based on docking). This highly interpretable metric is more informative than the BCE loss to evaluate the performance of the model. We computed the Tanimoto similarity for molecules in the validation set, for each epoch of the training. We observed that this metric consistently converged, as well as the loss, to a low value (median of 0.3 at best, 1 being the maximum), even when changing different hyperparameters of the model (embedding size of blocks, number of GRU steps, etc).

Considering the nature of the IFP vector, where molecules usually are involved in around ten interactions with proteins, therefore displaying around ten "on" bits, we argue that the model was able to learn some information. Indeed, predicting a random

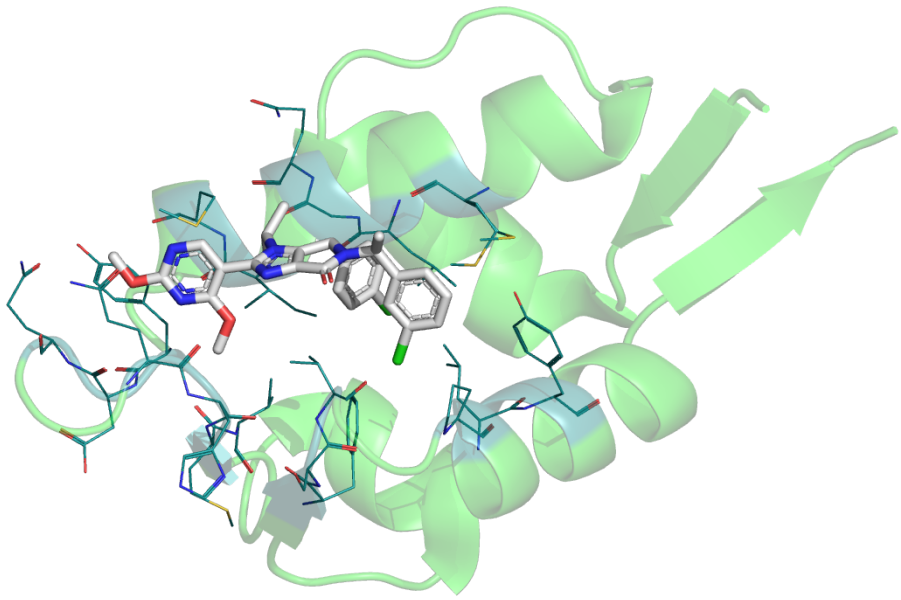


Figure 4.3: Protein MDM4 with its bound ligand (PDB '6q9w'). The dark blue residues are the amino acids comprised in the binding site.

IFP would result in a Tanimoto similarity near 0. However, the low value obtained, even when evaluating on the validation set, indicates that this approach is not reliable. We discuss the possible reasons that may prevent efficient training of the model in the next subsection.

4.2.4 Limits

We tried several strategies to improve learning of the model, changing the hyperparameters and the architecture. In particular, increasing the number of model parameters did not lead to overfitting the data, which indicated that this approach may be flawed because of two reasons.

Too many tasks? With a large number of tasks (230), the model architecture becomes complex, leading to increased computational requirements. The model needs to simultaneously learn representations for each task. This can result in slower convergence and problems in finding an optimal solution. Besides, having too many tasks can decrease the model's ability to generalize well to unseen data, as it may focus too much on each task rather than learning robust features.

Lack of Data. Although the network architecture may not be optimal, the low prediction performances appears to be probably due to the lack of data. Indeed, the 71k available training points in our study appears to be insufficient to train a 230-multitask prediction model. To tackle this issue we would need datasets comprising millions of molecules. Even if such data exist [Lyu *et al.*(2019), Stein *et al.*(2020), Sadybekov *et al.*(2020), Gorgulla *et al.*(2020)], they are related to a handful of proteins, involve

non overlapping chemical spaces and are obtained using different docking protocols and softwares. In the context of predicting the Interaction Fingerprints Profile, one would require several models trained on homogeneous data, i.e. the same molecules docked inside different proteins with the same protocol and software, so that the predictions of IFPs are consistent from one model to another.

Unfortunately, such datasets do not exist. Building them would require several million of dockings in many different proteins, since the ability of the IFPP to solve scaffold hopping increases with the size of its protein panel (see 3.1.3). We argue that the computational cost of building such a collection of datasets is too high for the reward, particularly when lacking proof of concept for this approach.

In the next section, we propose an alternative method that implements the idea that isofunctional molecules are expected to be close in the space of the IFPPs, although far in a space defined according to their chemical structure, without computing the IFPP itself. This leverages the computation limitations, allowing pre-screening of very large compound libraries.

4.3 Predicting IFPP Similarity through Metric Learning

We proposed to solve scaffold hopping problems by searching for molecules with similar IFPPs to that of a reference hit molecule, which requires to compute these IFPPs. One idea to bypass this calculation would be to train a ML algorithm that predicts the similarity of molecules in the space of IFPPs, without calculation of the IFPPs themselves. This approach uses concepts introduced in the domain of Metric Learning.

4.3.1 Metric Learning

Metric Learning is a subfield of ML which principle is to approximate a real-valued metric through an algorithm trained on available examples. The key idea behind metric learning is to transform the original feature space into a new space where the distances between data points are more meaningful, accessible or discriminative for the specific task at hand.

Its goal is to define a distance measure that can accurately capture the similarity or dissimilarity between pairs of data points in a given dataset. In practice, this distance metric is often learned in a supervised or semi-supervised manner, where the model is trained on labeled or pairwise similarity information. The learned metric is then used to compute distances or similarities between data points.

This strategy has been applied in many fields, from face recognition [Liu *et al.*(2018)] to representation learning [Kim *et al.*(2019)]. Recently, Deep Neural Networks (DNN) have been employed for this purpose: they are trained to learn an embedding space in which the distance between points mimics the real-valued metric in the original space.

The three important factors of Metric Learning architectures are:

- *The structure of the model*: not all structures are compatible with Metric Learning, and some may depend on the choice of the loss function.
- *The loss function*: as with any task, the loss function drives the training phase. It is prominent for the architecture as it serves to map similar data points closer

together and dissimilar data points farther apart in the learned embedding space.

- *The sampling protocol*: the accuracy of the model depends on the discriminating power of the samples that are presented. Even a state of the art model will have limited learning ability if provided with informative-less examples.

In our case, starting from the 2D structure of molecules as input, a Metric Learning approach would learn a new representation of the molecules such that, in this abstract space, the similarity between two molecules matches that of their similarity in Interaction Fingerprint Profiles. The learned embedding space serves as a surrogate of the IFPP space and can be interpreted as a dimension reduction of the IFPP.

4.3.2 Model Architecture

Siamese Networks

We chose a simple architecture that relies on the same principle as Siamese Networks [Koch *et al.*(2015), Bromley *et al.*(1993)]. This architecture consists in twin networks that take distinct inputs, here two molecules, and output representations for each input, for which similarity or distance metrics can be computed. Note that the twin networks share the same weights (they are identical), so that two identical molecules will be identical in the feature space, and thus, their distance in this space equals zero. Figure 4.4 displays the global architecture that was adopted.

Graph Neural Network for molecule embedding

We used Attentive FP, a Graph Neural Network (GNN) using the attention mechanism proposed by [Xiong *et al.*(2020)] to encode molecules with a readout function to obtain a graph representation for each molecule. Initial state vectors of length 128 for each node and edge are obtained with a fully connected input layer. Then, three GNN layers perform message passing on the node embeddings using an attention mechanism to include local information of the relevant neighbouring atoms. The message passing mechanism relies on a context vector incorporating neighbouring node and edge embeddings that goes through a gated recurrent unit GRU that updates the state vector of the node. To get the final embedding of the molecule, an initial molecule state vector is obtained by summing all node embeddings. Then, a readout block consisting in two attentive pooling layers is applied. In each pooling layer, a context vector of the molecule is computed using an attention mechanism on all node embeddings, which goes through a GRU that updates the molecule embedding. Finally, we get a graph embedding of length 256.

We used the python packages `dgl-life` [Li *et al.*(2021)] and PyTorch [Paszke *et al.*(2019)] to implement the corresponding architecture.

Training the DL model with a Loss function

The DL model is trained so that, in the learned feature space, the similarity between embeddings of molecules is similar to that computed with the IFPP of molecules.

Given a set of N molecules of known IFPP, we can define $\frac{N\times(N-1)}{2}$ pairs, and compute their IFPP similarity, defined as the Tanimoto coefficient of their fingerprints,

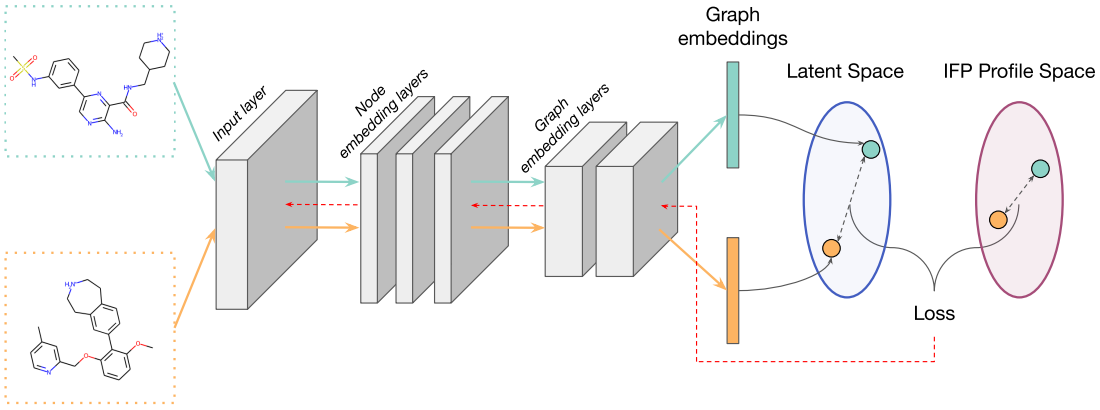


Figure 4.4: Illustration of the network architecture. A Siamese Neural Network is used to get the graph embeddings of pairs of molecules using Attentive FP. The GNN is composed of an input layer, node embedding layers and graph embedding layers. The similarity between molecules in the latent space are compared to their similarity in the IFPP space to compute the loss and train the model, as illustrated by the red arrow.

as above. In addition, for each pair, the DL model provides two graph embeddings (GE), for which we can also compute a similarity according to the following formula:

$$Similarity_{\ GE} = \frac{1}{1 + d_{Euclidean}} \tag{4.6}$$

where $d_{Euclidean}$ is the euclidean distance between the graph embeddings of a pair of molecules.

Training our DL model boils down to matching these two similarity measures. Therefore, we choose to compute the Root Mean Squared Log Error (RMSLE) between these two quantities, and use it as a part of the cost function to train our model:

$$RMSLE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\Big(\log\left(1 + Similarity_{\ GE}\right) - \log\left(1 + Similarity_{\ IFPP}\right)\Big)^2} \tag{4.7}$$

which can be rewritten as:

$$RMSLE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\Big(\log\frac{1 + Similarity_{\ GE}}{1 + Similarity_{\ IFPP}}\Big)^2} \tag{4.8}$$

Where n is the batch size. This loss can be broadly interpreted as the relative error between the predicted and the actual similarities.

We also included the Kullback-Leibler divergence (D_{KL}) to the RMSLE loss as a regularisation term to encourage the posterior distribution to be close to the prior distribution [Kingma et Welling(2022)]. Therefore, the final loss used to train the model is defined as:

$$Loss = RMSLE + D_{KL} \tag{4.9}$$

4.3.3 Training Dataset

The model requires a training dataset of pairs of molecules with known IFPP.

In subsection 3.2.1, we already computed the IFPPs for all actives and $144 \times 499 = 71,856$ decoys in the *LH* benchmark. From the $71,856$ decoys, we randomly picked $60,000$ molecules to define the training set, and $10,000$ molecules to define the validation set, and the network was trained as following. Pairs of molecules are formed during the training phase in each batch. We chose 64 as the batch size, so for each batch there are $\frac{64\times63}{2} = 2016$ pairs. Each epoch consists in 937 batches, leading to $1,890,000$ pairs formed at each epoch of the training. A learning rate of 0.0001 was used for *Adam* optimisation algorithm. We performed 200 epochs and kept the model with the lowest validation loss. Pairs are also formed inside batches for the validation, thus the model is evaluated on $\frac{10,000}{64} \times \frac{64\times63}{2} = 315,000$ pairs. The size of graph embedding was set at 256 to encompass the intricacies of the IFPP.

This allowed to train the model on pairs of molecules that are not considered with the *LH* benchmark. In this benchmark used to explore the efficacy of molecular representations, decoys and unknown actives are ranked according to their similarity with the known actives. Therefore, a training dataset containing only pairs of decoys ensures that the model is trained without any information about the actives, i.e. without using any pair of molecules taken into account for ranking. This limits potential bias in performance evaluation. Although the model will have seen decoys pairs during training, it will be tested only on pairs that include an unseen active molecule.

Figure 4.5 displays the evolution of the loss during training. The loss on the validation set is decreasing with the number of epochs, and seems to have reached a plateau. The training loss is still decreasing, indicating that the model is beginning to overfit. We chose to keep the weights of the model with the smallest validation error.

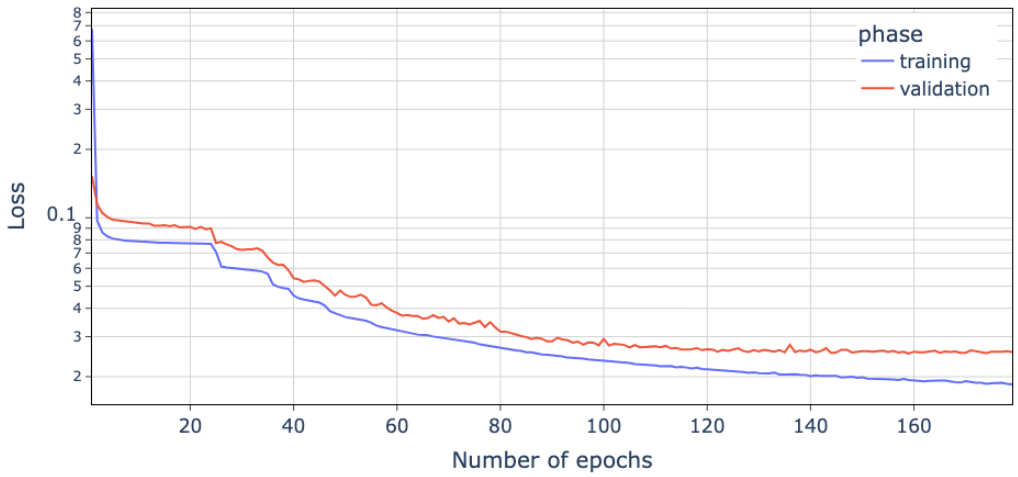


Figure 4.5: Evolution of both the training and validation losses in the logarithmic scale with number of epochs.

Once trained, this model can be employed to screen large compound libraries, allowing replacement of the expensive calculation of the IFPP by a quick inference of

molecule embeddings, from which a predicted IFPP similarity is computed.

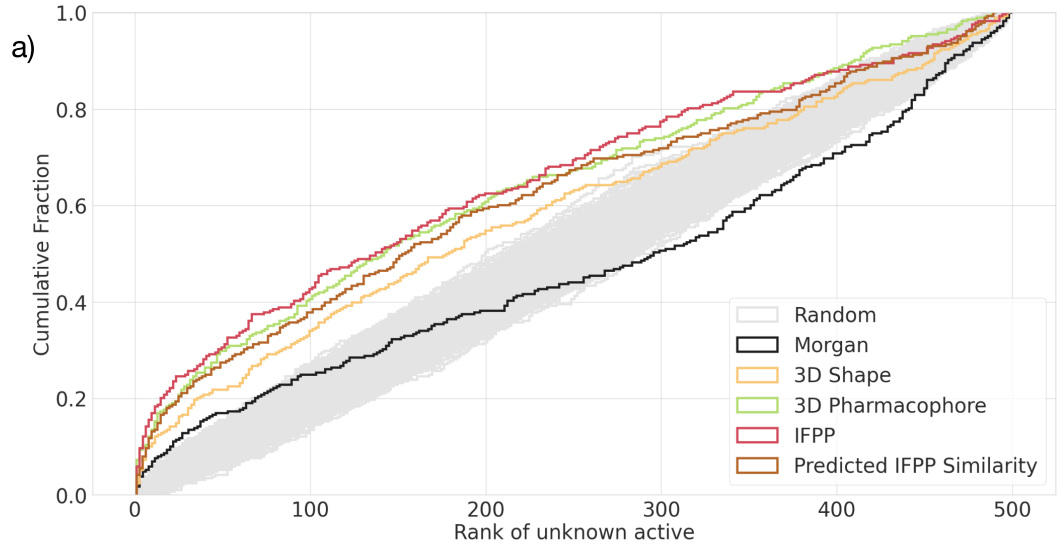
4.3.4 Performance of the Metric Learning Approach on *LH* Benchmark

The proposed DL model was evaluated on the *LH* benchmark to assess its ability to solve scaffold hopping cases. The protocol used to rank molecules is similar to that described in subsection 3.2.1:

- For each scaffold hopping pair, one of the ligand is set as the known active and the other, the unknown active, is joined to 499 decoys.
- The DL model is used to compute molecule embeddings.
- According to this embedding, the similarities of decoys and unknown active with respect to the known active are computed.
- Decoys and the unknown active molecules are ranked according to their similarity with the known active.
- The above steps are repeated by switching the active and unknown active roles to provide two scaffold hopping problems per pair of actives.

Figure 4.6 gathers the CHC of the predicted IFPP similarity measure, in addition to those of other similarity measures considered above.

As expected, the predicted IFPP similarity does not reach the performance of the similarity measure based on the true IFPP representation. We might argue that adjustments during the training phase and tuning of the model architecture may improve the performances. Still, this relatively simple model displays performances that reach those of the 3D Pharmacophore, while being much faster. Indeed, the 3D pharmacophore descriptors require computation of conformers and alignment of pharmacophores, leading to heavy calculations that are hardly compatible with screening of very large compounds libraries. On the contrary, once trained, the proposed Metric Learning approach quickly infers IFPP similarities, allowing large scale virtual screening campaigns. When screening very large compound libraries (millions of molecules), this approach could be used as a fast pre-screening campaign, keeping the best few percent ranked molecule. The top-scoring molecules (up to hundreds of thousands of molecules) could be screened based on the computed IFPP representation. Note that in real-case applications, this representation could be used as input of any ligand-based approach, and not only with the simple similarity measure used in the present study.



Method	Percent in Top 5%
Morgan	11.5
3D Shape	14.9
3D Pharmacophore	20.1
IFPP	**24.7**
Predicted IFPP Similarity	19.8

Figure 4.6: Results on the *LH* benchmark. The cumulative histogram curves of each similarity-based method are plotted in a). A zoom of the same graphs is provided in b). Table c) displays the percentage of successful scaffold hopping problems for molecular descriptors according to a rank of the unknown active in the top 5%.

4.3.5 Conclusion

The definition of the Interaction Fingerprints Profile, its performance as well as the Metric Learning framework were gathered into an article [Pinel *et al.*(2024)] that is currently being reviewed:

P. Pinel, G. Guichaoua, N. Devaux, Y. Gaston-Mathé, B. Hoffmann, V. Stoven (2024), *A molecular representation to identify isofunctional molecules.* doi:10.1101/2024.05.03.592355 (Currently in review.)

Based on the *LH* benchmark we demonstrated that Metric Learning can be applied to make accessible the costly IFPP thanks to simple heuristics. Indeed the architecture retained as well as the training protocol for the model are transparent and still leads to decent performances. In the case of the IFPP similarity predictor, its performance surpasses those of 2D baseline methods, and are as accurate than 3D Pharmacophores, a state of the art method for solving scaffold hopping in a ligand-based context. However, with a success rate 4.9% under the true IFPP representation, we argue that there is room for improvement.

As area of progress for the Metric Learning model we can think of different training protocols. Here, we chose to train the models using pairs of molecules, but other approaches have been described in the literature, for instance triplets of molecules [Coupry et Pogány(2022), Koge *et al.*(2021)]. The idea is to form triplets of molecules (anchor, positive, negative) within batches so that the two first molecules (anchor, positive) of the triplets are closer in terms of similarity than the pairs (anchor, negative) by a margin to be defined. That way, this Triplet model learns to distinguish between similar and dissimilar pairs.

However, [Gong *et al.*(2018)] argues that Siamese and Triplet networks neglect the structural information of training samples in each training step. During training, traditional Metric Learning methods update the model parameters based solely on the similarity relationships between pairs or triplets of examples within a single batch, without considering the structural information present in the entire dataset. This approach can neglect important information about the relationships between different samples, particularly when dealing with limited training data. Forming training examples results in a polynomial growth of training pairs/triplets which are highly redundant and less informative.

To address those challenges, other strategies resulting from different losses have been designed, such as the clustering loss introduced by [Song *et al.*(2017)]. It aims at clustering similar samples based on structural information. Besides, it does not require the training data to be preprocessed in a rigid paired format. Another possible approach has been developed by [Wang *et al.*(2019)]. They defined the multi-similarity loss, which aims to collect informative pairs, and weight these pairs through their own and relative similarities. It enables to reduce the computational burden of assembling pairs while limiting redundant information.

The choice of the loss is critical in Metric Learning. It drives the sampling method followed to train the model. In this subsection, we chose to use a simple Siamese Network architecture that still displayed promising performance. By employing a more suitable loss function, increasing the size of the dataset (i.e. compute IFPPs for ad-

ditional molecules) and developing sampling strategies to avoid redundancy, such a performance could be further improved.

4.4 Evaluation on LIT-PCBA

To overcome the cost limitation of IFPP, we proposed to leverage a Metric Learning approach to predict the IFPP similarity to the known active, which allows pre-screening of large molecular databases at a much lower computational cost. In a second step, screening using the computed IFPPs could be performed on pre-filtered, and thus reduced, chemical libraries.

In the previous section, the interest of the IFPP representation was assessed according to the performance of its corresponding similarity measure on the *LH* benchmark. However, we would like to point that the proposed IFPP molecular representation is meant to be used as input in more sophisticated ligand-based method. Indeed, in real-life applications, unlike in this benchmark, several known active and inactive molecules would usually be available for the target of interest. Encoding molecules with the IFPP would allow to train QSAR or ML algorithms dedicated to help solving scaffold hopping problems.

We illustrate this principle in the present section, and show how the IFPP Similarity Predictor could be used in realistic virtual screening setting, based on the LIT-PCBA benchmark [Tran-Nguyen *et al.*(2020)].

4.4.1 Dataset Description

LIT-PCBA was designed for both ligand- or structure-based virtual screening, and ML. It was assembled from processed dose-response PubChem bioassays [Wang *et al.*(2009)] to remove false positives and assay artifacts, and gather active and inactive compounds within similar molecular property ranges. The final dataset comprises active and inactive datasets of various sizes for 15 different targets. The dataset mimics experimental screening decks in terms of hit rate (ratio of active to inactive compounds). Additionally, each target is associated with 1 to 15 template PDBs for structure-based virtual screening.

Another interest of the LIT-PCBA dataset is that training and validation splits have been created using the asymmetric validation embedding (AVE) method [Wallach et Heifets(2018)]. It limits bias by measuring the pairwise distance in chemical space of active and inactive molecules in the training and validation sets to assemble the splits.

The LIT-PCBA dataset in summarised in Table 4.1.

The Enrichment Factor (EF) serves as the standard metric for evaluating the performance of methods in retrieving active compounds. It quantifies the degree of enrichment of active molecules within the top X% (typically the top 1%) compared to random selection. This metric is particularly informative and realistic, mirroring the scenario in real drug discovery projects where only the top-ranked molecules undergo *in vitro* testing. Hence, it provides valuable insights into the retrieval of active compounds among the tested molecules. It is computed using following formula:

$$Enrichment\ Factor_{x\%} = \frac{Number\ actives_{top\ x\%}}{Number\ molecules_{top\ x\%}} \frac{Number\ molecules_{dataset}}{Number\ actives_{dataset}} \quad (4.10)$$

Target	Target Name	Number PDBs	Actives Validation	Actives Training	Inactives Validation	Inactives Training
ADRB2	Beta2 adrenergic receptor	8	4	13	78,078	234,363
ALDH1	Aldehyde dehydrogenase 1	8	1,343	4,032	27,088	77,606
ESR1_ago	Estrogen receptor alpha	15	3	10	1,284	4,188
ESR1_ant	Estrogen receptor alpha	15	25	77	1,176	3,711
FEN1	Flap endonuclease 1	1	91	277	88,612	266,552
GBA	Glucocerebrosidase	6	41	125	73,636	222,039
IDH1	Isocitrate dehydrogenase	14	9	30	90,287	271,537
KAT2A	Histone acetyltransferase	3	48	146	86,750	261,411
MAPK1	Mitogen-activated protein kinase 1	15	77	231	15,657	46,972
MTORC1	Mechanistic target of rapamycin	11	24	73	8,267	24,729
OPRK1	Kappa opioid receptor	1	6	18	67,443	202,362
PKM2	Pyruvate kinase muscle isoform 2	9	136	410	61,467	184,143
PPARG	Peroxisome proliferator-activated receptor γ	15	6	21	1,227	3,909
TP53	Cellular tumor antigen p53	6	16	60	981	3,126
VDR	Vitamin D receptor	2	165	498	66,494	199,906

Table 4.1: Description of LIT-PCBA dataset. We removed some duplicated molecules present in the bioassays.

An EF of 1 means that the method is no better than random picking.

LIT-PCBA has been extensively used in the literature to evaluate performances of various structure-based or ML approaches [Berenger *et al.*(2021), Cai *et al.*(2022), Brocidiacono *et al.*(2024)]. The limited performances obtained (median EF1% of 0.0 when using the docking software Gold [Berenger *et al.*(2021)] on the validation set) demonstrate how difficult this dataset is.

Note that though LIT-PCBA was not built as a scaffold hopping benchmark, as it integrates actives of similar chemical structure and no information on binding modes is available, it still provides a suitable way to evaluate the ability of IFPP in retrieving active molecules.

However, to avoid docking the 2.6 millions molecules of LIT-PCBA in 37 proteins to build their true IFPP, we evaluated the ability of our IFPP Similarity predictor to enrich the top ranked molecules in actives. The decoys from the *LH* benchmark have *a priori* nothing in common with molecules of LIT-PCBA, as they originate from distinct databases, and were selected through different processes. Hence, this dataset is also a way to test the domain adaptability of the IFPP Similarity predictor, as the molecules it will perform inference for should be from a chemical space remote from its training set.

Indeed, we used the UMAP algorithm (Uniform Manifold Approximation and Projection) [McInnes *et al.*(2020)] to visualise the overlap of chemical spaces between decoys of the *LH* benchmark and molecules of LIT-PCBA for each of the 15 targets. This non-linear dimension reduction algorithm consists in learning the manifold structure of the data and finding a low dimensional embedding that preserves the essential topological structure of that manifold. We used the Tanimoto similarity of the Morgan fingerprints to evaluate the distance between molecules.

Figures 4.7 and 4.8 display the overlap in chemical spaces between decoys of the *LH* benchmark and molecules from 2 datasets of LIT-PCBA: **MAPK1** and **ESR1 antagonist**. The other 13 UMAP representations are provided in Appendix C. They show very little overlap, demonstrating that molecules used to train the IFPP Similarity Predictor are from distant chemical spaces to those of LIT-PCBA. Thus, LIT-PCBA

is outside of the theoretical applicability domain of the IFPP Similarity Predictor, and provides an interesting dataset to assess the generalisation properties of our IFPP Similarity Predictor.

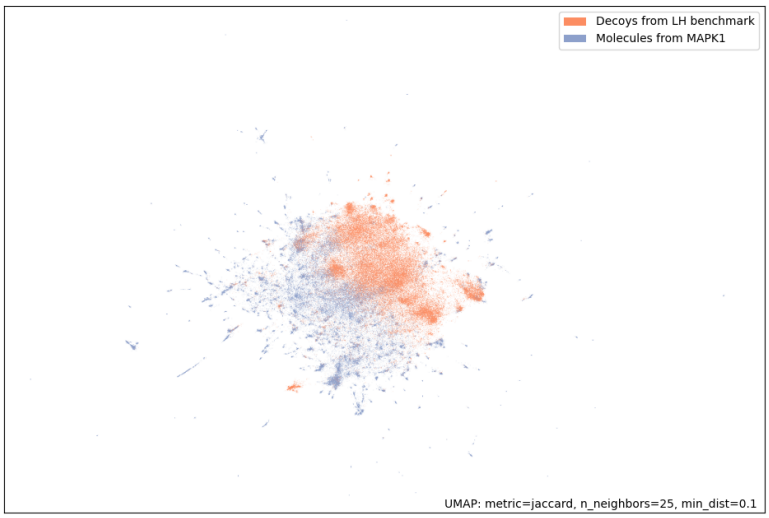


Figure 4.7: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from MAPK1 dataset.

We assessed the performance of our approach through *similarity searching* and *predictive models* as described in the following subsections.

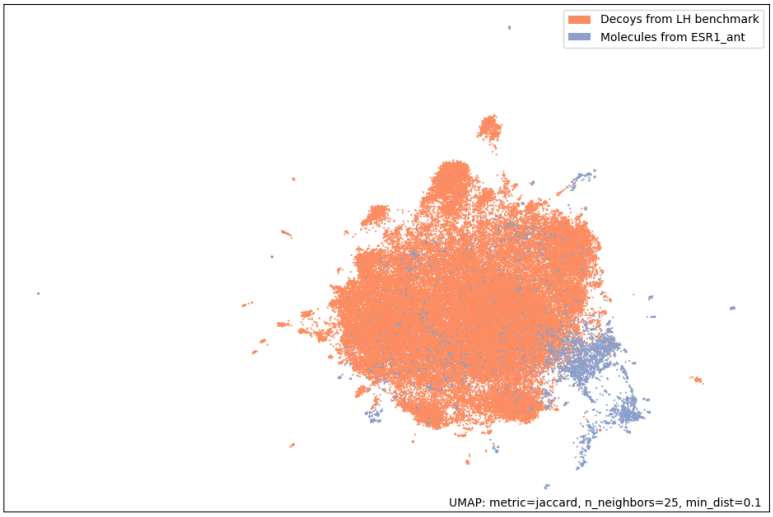


Figure 4.8: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from ESR1 antagonist dataset.

4.4.2 Similarity Searching

For each target of the dataset, between one and 15 template structures are provided, along with their respective ligands, which are absent from both the training and validation datasets. These ligands can serve as reference actives to rank other molecules based on their similarity, according to various encodings, as performed with the *LH* hopping benchmark.

Docking was conducted in the original article using Surflex-Dock v.3066 [Jain(2007)] on the available crystallographic structures, and molecules were ranked according to their docking scores. The structure-based approach resulted in low enrichment factors at 1%, averaging 1.8 across the 15 targets. We assessed ligand-based methods to determine whether they yield higher enrichment of active compounds among the top-ranked molecules. We used Morgan Fingerprints as the baseline method, wherein molecules are ranked based on their Tanimoto similarity to the reference ligands. For each protein of LIT-PCBA, we also performed such similarity searching experiments based on the predicted IFPP Similarity described in Section 4.3, and ranked molecules accordingly, using in turn each ligand as known active.

Figure 4.9 and Table 4.2 display the enrichment factors at 1% of the considered methods across all 15 targets. The low enrichment factors obtained illustrate how difficult this benchmark is. Still, the IFPP Similarity prediction shows better performances than docking, while only using information about one active. However, the higher performances on average of the Morgan fingerprints mitigate those encouraging results.

Nevertheless, due to the high imbalance between the number of active and inac-

tive molecules across the targets of LIT-PCBA, we argue that computing the average enrichment factors to compare methods is troublesome and not really reliable as the EF1% do not vary on the same scale across proteins. Rather, we propose to count the number of times a method performed better than others considering the same reference ligand. That way, we can compare broadly different methods in enriching the top molecules in actives. Out of the 129 similarity searching experiments, the IFPP Similarity prediction outperforms the Morgan fingerprint 53 times, and is beaten by the latter 49 times. Globally, both methods display the same performances, and both outperforms docking on most experiments.

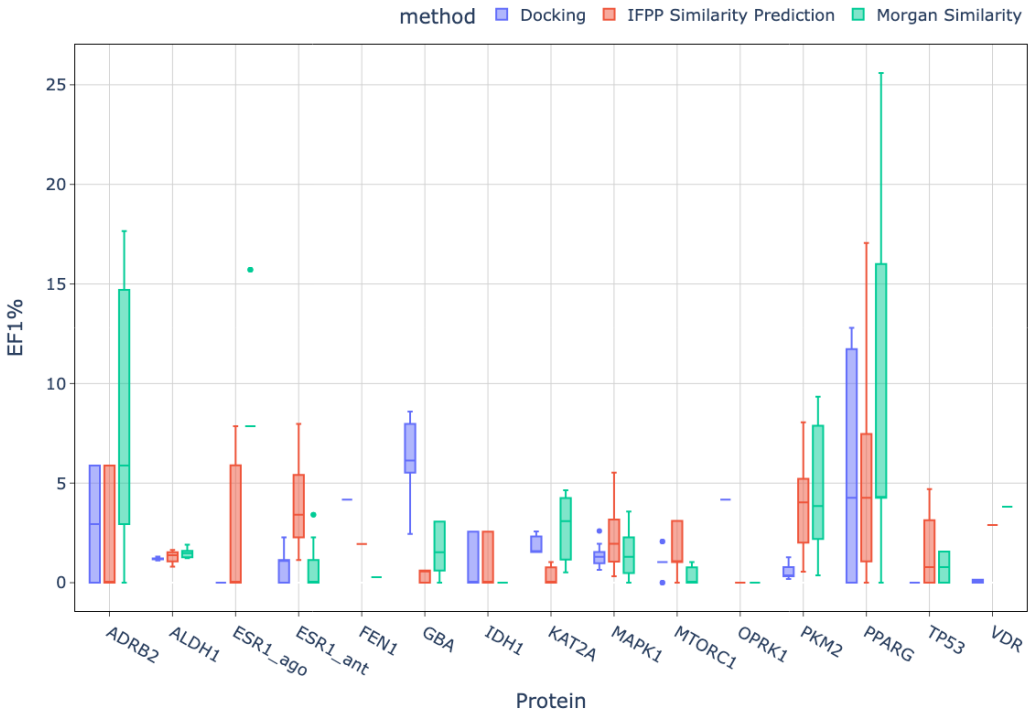


Figure 4.9: Enrichment Factors for similarity searching experiments across LIT-PCBA proteins. For each protein, between 1 and 15 experiments were conducted using the reference ligands in the available PDB structures.

Method	Mean	Standard Deviation	Median
Docking	1.81	2.86	1.033
Morgan Similarity	3.58	5.05	1.459
IFPP Similarity prediction	2.40	2.82	1.567

Table 4.2: Mean, standard deviation and median enrichment factors at 1% of tested methods across all similarity searching experiments.

The rather good performances of the Morgan fingerprints can be explained by the fact that this dataset is not a scaffold hopping benchmark, and some actives in the training or validation sets are closer to the reference ligands than most inactive molecules. Indeed, we computed the 2D structure similarity (using the Morgan Fingerprint) between the actives in the top 1% of the 3 method and the reference ligands.

Figure 4.10 and Table 4.3 display the structure similarity between top actives and reference ligands. Docking retrieves actives belonging to the most remote chemical spaces from the known ligands. It confirms the property we described in 1.4.3: docking is able to handle various chemical spaces, and its applicability domain is theoretically infinite in the chemical space. Its poor performances overall could be explained by an under-calibrated docking protocol, or poor quality structures.

The actives retrieved by the IFPP Similarity Predictor are consistently of low structural similarity to the reference ligands, confirming that this representation can retrieve actives from distant chemical space. It catches actives that are not identified by Morgan fingerprints, illustrating that combining the methods could help retrieve even more active compounds.

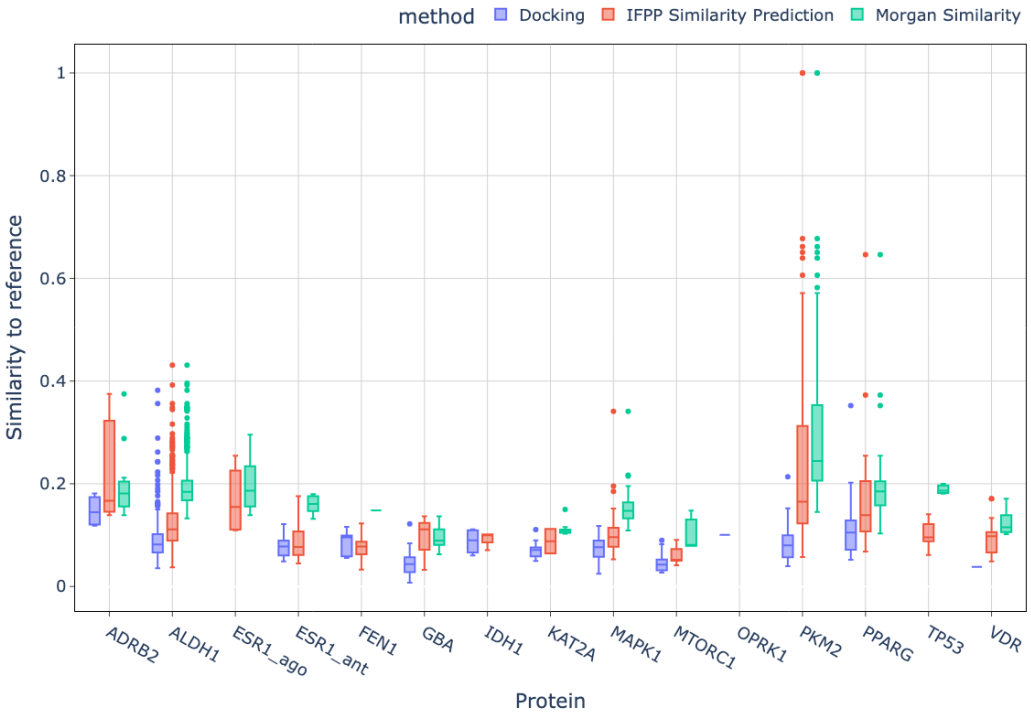


Figure 4.10: Structure similarity between top actives identified by the methods and reference ligands across LIT-PCBA proteins.

Method	Mean	Standard Deviation	Median
Docking	0.08	0.04	0.08
Morgan Similarity	0.20	0.09	0.18
IFPP Similarity prediction	0.14	0.10	0.11

Table 4.3: Mean, standard deviation and median structure similarity between top actives and reference ligands of tested methods across all similarity searching experiments.

4.4.3 Predictive Models

As outlined in subsection 4.4.1, separate training and validation splits have been constructed for every protein in LIT-PCBA. This facilitates the comparison of models trained on the training sets and assessed on the validation sets. Such studies have been conducted in several articles in the literature [Berenger *et al.*(2021), Cai *et al.*(2022)]. Especially [Cai *et al.*(2022)] compared different machine and DL frameworks on LIT-PCBA. Unfortunately, only ROC-AUC were computed on the models, and we argue that considering how imbalanced LIT-PCBA datasets are, with far more inactive than active molecules, it is not the most suited way to compare different approaches. Indeed, a dummy model classifying all molecules as inactive would get a high ROC-AUC while not being relevant. Again, we prefer to use the enrichment factor at 1% to confront methods as it is more informative as to what might be obtained in real-life experiments.

In this subsection, we illustrate the practical application of the IFPP Similarity Predictor in retrieving active compounds from datasets containing various active and inactive molecules for specific targets of interest. The Metric Learning model generates a graph embedding based on an input molecule. Up to this point, we have calculated the similarity of graph embeddings (refer to Equation 4.3.2) between molecules to rank them. However, this graph embedding can also be interpreted as a dimensional reduction of the IFPP, and serves as a molecular representation. This molecular representation, which we call in the following IFPP 'Predicted' to simplify, can be employed as input for ML models tasked with predicting activity.

[Cai *et al.*(2022)] compared different Machine and Deep Learning architectures combined with different molecular representations over the LIT-PCBA benchmark, and showed that DNNs with Morgan Fingerprints had one of the highest ROC-AUC overall. We kept the Morgan fingerprints as baseline molecular representation to compare the IFPP Predicted to, as it showed to perform the best combined with predictive models.

We chose basic Machine and Deep Learning frameworks to compare those representations: Deep Neural Networks (DNNs) with three layers as in [Cai *et al.*(2022)] and Logistic Regression. Indeed, the latter predicts the probability of belonging to a particular category, either active or inactive, and has not been tested yet on LIT-PCBA.

For each molecular representation, we trained a Logistic Regression model and a DNN for each target of LIT-PCBA using its training set. The training set was further split into two separate sets: one for the training (90%) and one as validation (10%) to monitor the generalisation power of the model during the training phase before evaluation on the testing set (also called validation set in [Tran-Nguyen *et al.*(2020)]). Thus, we conducted $Number_{Representations} \cdot Number_{Algorithms} \cdot Number_{Proteins} = 2 \cdot 2 \cdot 15 = 60$ training experiments. We also performed a hyperoptimisation phase for each experiment to select the best hyperparameters for each model, as done in [Cai *et al.*(2022)]:

- For Logistic Regressions, the best $\mathcal{C}$ (inverse of regularization strength) was sampled out of a Uniform Distribution $\mathcal{U}(-4, 2)$;
- For DNNs, we optimised several hyperparameters:
  - Weight decay for Adadelta backpropagation algorithm with a Uniform Distribution $\mathcal{U}(0, 0.01)$;

  - Dropout for all layers with Uniform Distribution $\mathcal{U}(0, 0.3)$;
  - For each of the three layers, the number of neurons are chosen from the set {64, 128, 256, 512} ;

- 50 hyperoptimisation steps were conducted.

We used the Python packages `hyperopt` for the hyperoptimisation phase, `scikit−learn` for Logistic Regressions models and Pytorch for the DNNs models.

We gather the results of all experiments in Figure 4.11. Molecules of the testing sets were ranked according to their probability of being active, and we calculated the resulting EF1%. Note that since protein targets of LIT-PCBA have various active rates, models have to be compared column-wise. Enrichment factors at 1% from one protein to another are not comparable because they do not evolve on the same scale. Overall, the best combination is the Morgan fingerprints with Logistic Regression, which exhibited best performance overall on 5 targets (**FEN1**, **GBA**, **MAPK1**, **PKM2**, **VDR**). It also ties at first place on 2 other targets (**IDH1**, **MTORC1**). For Logistic Regression models, none beat those with Morgan fingerprints. Still, models using IFPP Predicted ties those of the Morgan fingerprints for a few proteins (**ESR1 antogonist**, **IDH1**, **PPARG**, **TP53**).

Interestingly, the DNN models with IFPP Predicted challenge the Logistic Regression models with the Morgan, achieving best performance overall on 3 targets (**KAT2A**, **PPARG**, **TP53**) and tying at best for 4 proteins (**ESR1 antogonist**, **IDH1**, **MTORC1**). The former combinations still have lower enrichment factors on 6 other (**ALDH1**, **FEN1**, **GBA**, **MAPK1**, **PKM2**, **VDR**).

One possible explanation of the results showing Morgan Fingerprints with Logistic Regression outperforms other combinations is the fact that for some datasets, the actives in the testing set are from close chemical spaces to the actives in the training set. We illustrate this hypothesis in the following. For the two best combinations (DNN with IFPP Predicted and Logistic Regression with Morgan), we computed the structure similarity (expressed by their Morgan fingerprints) between the retrieved active molecules in the top 1% and the actives in the training sets. These similarities characterise for each target how close the chemical space of found hits is to the chemical space of the training set.

Figure 4.12 display the results. Models with IFPP Predicted show a tendency to retrieve actives of remote chemical spaces from the original active molecules in the training set. This property is especially true of the proteins **GBA**, **IDH1**, **KAT2A**, **PPARG** and **TP53**. The Logistic regressions with Morgan tend to find actives with high structure similarity to the known hits, indicating that they retrieved "easier" compounds. However, in large-step scaffold hopping, we search for molecules of distant chemical structures with biological activity. In this context, models with the IFPP Predicted molecular representation tend to be better suited as the identified hits show high structure dissimilarity compared to the known hits.

This study also illustrates that the models do not retrieve the same active compounds. This suggests that in a purely hit discovery approach, combining molecular representations could help identify more hits, and increase the enrichment factors.

To test this hypothesis, additional Logistic Regression models were trained using concatenated inputs of Morgan fingerprints and IFPP Predicted embeddings for each

Figure 4.11: EF1% of predictive models across LIT-PCBA proteins. The ratio of actives in the validation sets are reported for each protein. We compared the combination of different ML architectures (DNN or logistic regression) with different molecular representation (Morgan fingerprint or IFPP 'Predicted' from the Metric Learning approach).

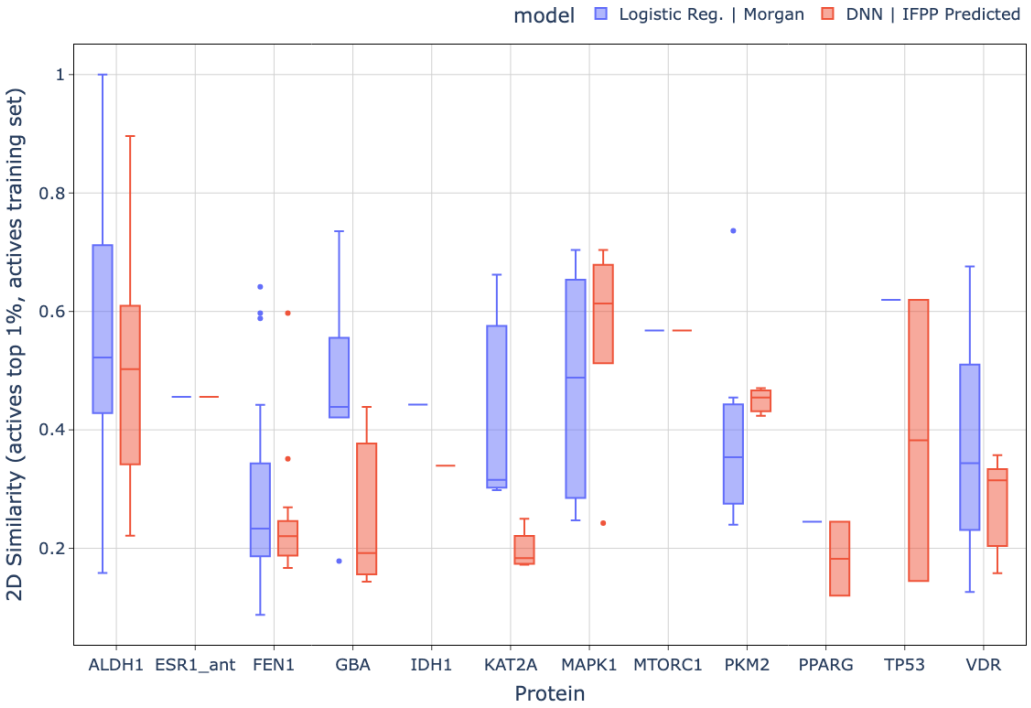


Figure 4.12: Structural similarity between actives retrieved by models and actives in training set across proteins. The two best model combinations are considered: DNN with IFPP Predicted and logistic regression with Morgan Fingerprints.

protein in LIT-PCBA. Once again, 50 hyperoptimisation steps were performed for each model.

Comparison of Enrichment Factors are gathered in Figure 4.13. The results clearly demonstrate that combining different descriptors undoubtedly increase the performances. The Logistic Regression models that integrate both Morgan fingerprints and IFPP Predicted outperform those relying solely on Morgan fingerprints for 5 targets (**FEN1**, **IDH1**, **KAT2A**, **OPRK1**, **PKM2**), while being surpassed for only one target (**ALDH1**). This aligns with the findings of [Cai *et al.*(2022)].

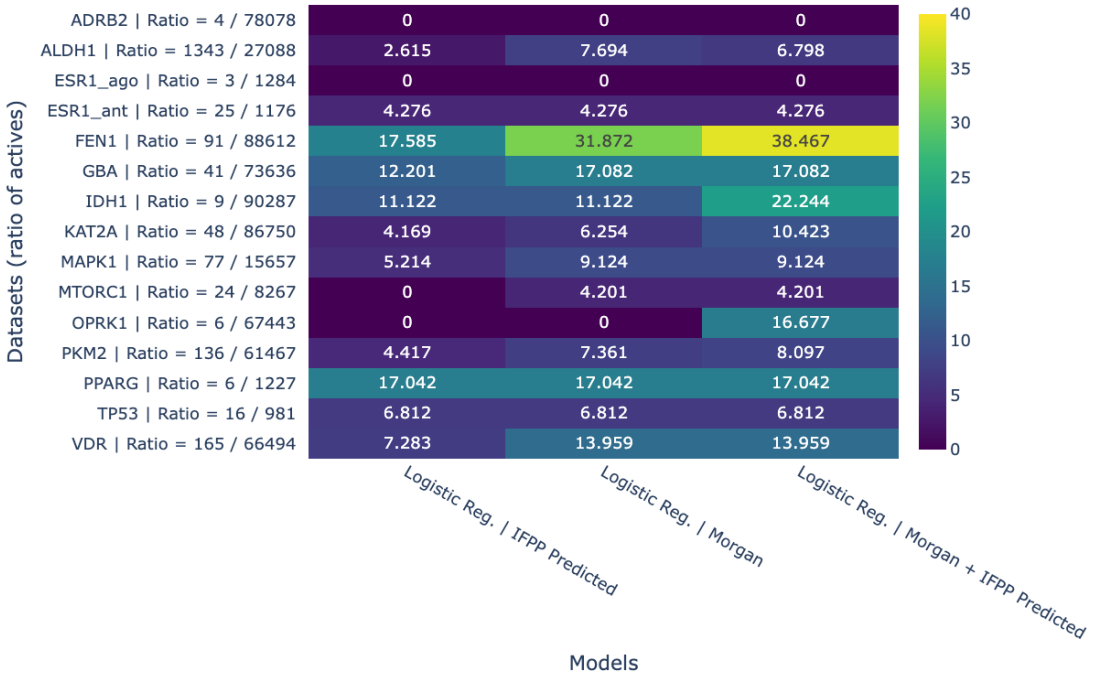


Figure 4.13: EF1% of logistic regression when combining Morgan fingerprints and IFPP Predicted across LIT-PCBA proteins.

4.4.4 Conclusion

Though LIT-PCBA is not a benchmark designed to address scaffold hopping, it provides a high quality dataset mimicking real virtual screening conditions for 15 protein targets. Several complex DL architectures have been tested on it, and displayed limited performances. In [Brocidiacono *et al.*(2024)], they computed the enrichment factors at 1% of their DL architecture *BANANA*, which combines the 3D pocket graph of the protein with the molecule graph to predict affinity. This model performed modestly, with a median EF1% of 1.81, comparable to another model that performs molecular docking with DL, *GNINA* [McNutt *et al.*(2021)], with a median EF1% of 2.58.

Those performances are in the same range of the IFPP Similarity predictor (median EF1% of 1.57), while only similarity searching was conducted. Besides, the complex models described above might have used external information about the studied proteins during their training phase, while for the latter only one active was used.

Based on the LIT-PCBA dataset, we illustrated the interest of the 'approximated' representation of the true IFPP obtained through the Metric Learning approach. It can quickly infer molecular representation for any molecule, which encodes prominent information on activity that is comparable to state of the art methods, while being uncorrelated. Hence, combining this molecular representation to others may help identify more hits, which is critical in Drug Discovery. Besides, its promising results on this dataset outside from its theoretical applicability domain show it can handle various chemical spaces, even far from those used for its training: it has great domain adaptability.

5

Conclusion and Perspectives

Contents

In this PhD thesis, our primary focus was addressing a challenging task often faced in the drug discovery process: solving large-step scaffold hopping problems, when the 3D structure of the protein target is unavailable or of poor quality. In this setting, docking is not reliable, and only ligand-based approaches remain applicable. We chose to tackle this demanding setting, because it represents a critical area where tailored computational methods are required the most. The problem is to find biologically active molecules displaying dissimilar structures to known hits for a protein target of interest.

5.1 Results of the Thesis

5.1.1 *LH* Benchmark and its use to Evaluate Molecular Descriptors

Chapter 2 describes the iterative process employed to construct a publicly available and well-characterized benchmark dataset for large-step scaffold hopping, in the context of drug discovery. The methodology involved gathering pairs of ligands from PDBbind [Wang *et al.*(2004b)] exhibiting dissimilar 2D structures but sharing similar binding modes with the same protein, without a common substructure that is responsible of akin interactions. The *LH* benchmark comprises high-quality and well characterized 144 pairs of molecules that are clear examples of large-step scaffold hopping cases. These cases were gathered from the PDBbind database, in order to ensure that they were examples of isofunctional molecules. This benchmark is an open resource that was missing in the community, to test new molecular encodings or new prediction algorithms dedicated to that problem, and that was meant to be used in ligand-based approaches. However, any docking algorithm could also be tested on this benchmark, since 3D structures of the targets are also available for the 144 pairs. Overall, this benchmark is a tool provided to the community in order to design and evaluate novel strategies for solving large-step scaffold hopping. Note that the *LH* benchmark is not a training dataset: no learning is possible due to the low number of actives, since it contains only 2 active molecules per case.

For each of the 144 pairs, the *LH* benchmark also comprises 499 corresponding decoy molecules that are used in the strategy proposed to evaluate molecular descriptors. For each scaffold hopping case, the 499 decoys are as "far" from the two molecules of the pair than these two molecules are from each other, for structure-based descriptors such as Morgan fingerprints. Therefore, we proposed that, given one molecule of the pair (the known active), ranking the other (the unknown active) among the 499 decoys according to new descriptors, provided a means to evaluate the interest of these new descriptors for solving scaffold hopping problems. The underlying idea was that, molecular descriptors allowing to rank the unknown active among the best ranked molecules in terms of similarity with respect to the known active, were well suited to solve these problems.

5.1.2 The Interaction Fingerprints Profile

In Chapter 3, a novel molecular representation called the Interaction Fingerprints Profile (IFPP) was introduced, specifically designed for addressing scaffold hopping challenges. The IFPP aims to capture potential binding modes of molecules through docking experiments across a panel of diverse proteins. We showed that it outperformed

the classical molecular representations on the *LH* benchmark.

Since IFPP descriptors contain information that is uncorrelated to classical representations, combining the IFPP to other state-of-the-art encodings may lead to improved success rates, as demonstrated when they were combined with 3D pharmacophore descriptors. The IFPP appears to be essentially a new string to the bow of available methods for solving large-step scaffold hopping.

Furthermore, increasing the diversity and size of the protein panel was found to improve the performance of the IFPP encoding. Additionally, for specific protein families, selecting a panel of proteins from the same family could enhance the IFPP's ability to solve scaffold hopping problems, as shown for kinases. This provides a rationale for the selection of proteins for the panel in the case of solving scaffold hopping for a target belonging to an extensively described family.

However, the IFPP has a high computational cost, as it requires docking in multiple protein pockets. While feasible for medium-sized chemical libraries, screening very large libraries would induce significant computational burdens.

5.1.3 Predicting the IFPP Similarity between Molecules

To address this computational constraint, we developed a Metric Learning approach to predict the IFPP similarity between molecules in Chapter 4, without computing the IFPP themselves. We trained a model to learn a new representation of the molecules such that, in the corresponding abstract embedding space, the similarity between two molecules matches that of their similarity in Interaction Fingerprint Profiles. Though imperfect, the model demonstrated performance comparable to the 3D pharmacophore similarity on the *LH* benchmark, while not requiring any docking for its inference. Hence, it offers a rapid, cost-effective screening solution for very large chemical libraries.

The interest of the approach was demonstrated using LIT-PCBA [Tran-Nguyen *et al.*(2020)]. This high quality benchmark comprises active and inactive compounds identified by HTS for 15 proteins and has shown to be extremely challenging in the literature [Brocidiacono *et al.*(2024), McNutt *et al.*(2021), Berenger *et al.*(2021), Cai *et al.*(2022)]. Despite operating beyond its theoretical applicability domain, the Metric Learning model's molecular embeddings proved competitive in retrieving active compounds compared to sophisticated state-of-the-art methods. These embeddings, whether utilised for similarity searching or employed as inputs for Machine Learning algorithms, displayed promising performance.

Besides, when compared to the Morgan fingerprint, the standard molecular representation for drug discovery, the Metric Learning model demonstrated enhanced diversity in hit identification, highlighting its efficiency for scaffold hopping scenarios.

5.2 Perspectives

5.2.1 From Test Dataset to a Train Dataset for Scaffold Hopping

As mentioned above, the *LH* benchmark is not a training dataset, because it contains only two active molecules per case. Therefore, it can only be used as a test dataset, in order to evaluate new protocols that would be proposed to solve scaffold hopping problems. In the era of Deep Learning and advanced predictive models, the creation of a

benchmark comprising both training and testing datasets for scaffold hopping retrieval would be a significant asset to the research community.

Assembling such a dataset presents challenges due to the scarcity of crystallographic poses and the limited number of tested molecules. Additionally, careful filtering steps would be required to avoid introducing bias in the chemical space of the dataset splits for instance.

Nevertheless, chemogenomic methods stand out as interesting ML algorithms that are able to use the *LH* benchmark as a test dataset. Indeed, chemogenomic algorithms can be trained on external datasets of protein-ligand interactions, leveraging interaction information available for any protein, in order to make predictions for a protein of interest, even if this protein is orphan or has very few known ligands (1 known active, for all cases in the *LH* benchmark).

5.2.2 Predicting the IFPPs

In Section 4.2, we described a method to predict interaction fingerprints for specific proteins, which can be used to compute the IFPPs by concatenation the predicted IFPs. This approach led to somewhat deceiving preliminary results, but it prompted us to pursue the Metric Learning strategy that proved to be more promising. However, there is an unexplored alternative worth considering. Recent literature has introduced Deep Learning models capable of predicting the 3D poses of molecules within proteins [Krishna *et al.*(2024), Cai *et al.*(2024)]. Leveraging these models to predict the poses of molecules within each protein in our panel, could enable our interaction detection algorithm to build the interaction fingerprints within each protein. Concatenating these results would then yield an alternative predicted IFPP.

While this approach may be slower than the Metric Learning method, it promises greater precision. Moreover, expanding the protein panel's size would be considerably easier and wouldn't require additional training, as those models are supposed to handle any protein.

5.2.3 Combining Chemogenomics with (predicted) IFPPs

The considered chemogenomic model exhibited promising performances on the *LH* benchmark, despite lack of any optimisation. This suggests that it still captures crucial information for retrieving actives, despite its simple architecture. However, we argue that the model could achieve better results by using better molecular descriptors as input. In particular, using the IFPPs as molecular descriptors, instead of the Morgan fingerprints (used in the present thesis) would be an interesting option to evaluate. Alternatively, because of their computational cost, IFPPs could be replaced by the predicted IFPPs introduced in Chapter 4, since this representation successfully identified hits of novel chemical structures on both the *LH* benchmark and LIT-PCBA. Although not as efficient as the IFPP for the scaffold hopping problem, predicted IFPPs may be an interesting option as input descriptors for chemogenomic algorithms, alone, or in combination with classical descriptors such as Morgan fingerprints.

In [Guichaoua *et al.*(2024)], we developed a more sophisticated chemogenomic architecture using a larger and unbiased training dataset. This model achieved state-of-the-art performance across various tasks, including active retrieval on the *LH* benchmark.

Integrating this model with IFPP Predicted as the molecular representation could further improve predictive performance and should be explored.

5.2.4 Perspectives on Metric Learning

In Section 4.3, we illustrated the interest of Metric Learning to reduce cost and computation time of the IFPP. We already discussed the area of improvement of the architecture in 4.3.5, including a better chosen loss, and a sampling method avoiding redundancy.

However, the scope of Metric Learning goes beyond predicting the IFPP similarity: it can be used to simplify any costly representation. This principle could be applied to the 3D Pharmacophore fingerprint, which requires both conformer generation of molecules, and the 3D alignment of molecules for comparison. Although some methods are agnostic of this latter step, like [Berenger et Tsuda(2023)] that rely on autocorrelation, a mathematical function which renders an object rotation-translation invariant, to build 3D pharmacophore fingerprints, they still suffer from long computation, which limits their downstream use in virtual screening. Exactly as for the IFPP, we could build models using Metric Learning to project molecules in a trained feature space that accurately mimics the 3D Pharmacophore space. Once trained, it become effortless to score molecules, which overcomes the scalability issue. More complex and costly molecular representation could also be considered. For instance, with Metric Learning, we could create an embedding space encoding quantum information of molecules by training a model to mimic the quantum-based similarity method introduced by [Al-Dabbagh *et al.*(2015)].

Overall, the work presented in this manuscript illustrates that the important field of scaffold hopping still presents many exciting challenges. However, it provides various routes that could be further followed to help solving these problems in the context of drug discovery programs.

5.3 Publications

First Author

- **P. Pinel**, G. Guichaoua, M. Najm, S. Labouille, N. Drizard, Y. Gaston-Mathé, B. Hoffmann, V. Stoven (2023), *Exploring isofunctional molecules: Design of a benchmark and evaluation of prediction performance*, Molecular Informatics **42** (4), 2200216. doi:10.1002/minf.202200216

- **P. Pinel**, G. Guichaoua, N. Devaux, Y. Gaston-Mathé, B. Hoffmann, V. Stoven (2024), *A molecular representation to identify isofunctional molecules.* doi:10.1101/2024.05.03.592355 (Currently in review.)

Co-Author

- G. Guichaoua, **P. Pinel**, B. Hoffmann, C.-A. Azencott, V. Stoven (2024), *Advancing Drug-Target Interactions Prediction: Leveraging a Large-Scale Dataset with a Rapid and Robust Chemogenomic Algorithm.* doi:10.1101/2024.02.22.581599 (Currently in review.)

Appendices

A

Advancing Drug-Target Interactions Prediction: Leveraging a Large-Scale Dataset with a Rapid and Robust Chemogenomic Algorithm

Advancing Drug-Target Interactions Prediction: Leveraging a Large-Scale Dataset with a Rapid and Robust Chemogenomic Algorithm

Gwenn Guichaoua,[*,†,‡,¶] Philippe Pinel,[†,‡,¶,§] Brice Hoffmann,[§] Chloé-Agathe Azencott,[†,‡,¶] and Véronique Stoven[†,‡,¶]

†*Center for Computational Biology (CBIO), Mines Paris-PSL, 75006 Paris, France*

‡*Institut Curie, Université PSL, 75005 Paris, France*

¶*INSERM U900, 75005 Paris, France*

§*Iktos SAS, 75017 Paris, France*

E-mail: gwenn.guichaoua@minesparis.psl.eu

Abstract

Predicting drug-target interactions (DTIs) is crucial for drug discovery, and heavily relies on supervised learning techniques. Supervised learning algorithms for DTI prediction use known DTIs to learn associations between molecule and protein features, allowing for the prediction of new interactions based on learned patterns. In this paper, we present a novel approach addressing two key challenges in DTI prediction: the availability of large, high-quality training datasets and the scalability of prediction methods. First, we introduce LCIdb, a curated, large-sized dataset of DTIs, offering extensive coverage of both the molecule and druggable protein spaces. Notably, LCIdb contains a much higher number of molecules than traditional benchmarks, expanding coverage of the molecule space. Second, we propose Komet (Kronecker Optimized

METhod), a DTI prediction pipeline designed for scalability without compromising performance. Komet leverages a three-step framework, incorporating efficient computation choices tailored for large datasets and involving the Nyström approximation. Specifically, Komet employs a Kronecker interaction module for (molecule, protein) pairs, which is sufficiently expressive and whose structure allows for reduced computational complexity. Our method is implemented in open-source software, leveraging GPU parallel computation for efficiency. We demonstrate the efficiency of our approach on various datasets, showing that Komet displays superior scalability and prediction performance compared to state-of-the-art deep learning approaches. Additionally, we illustrate the generalization properties of Komet by showing its ability to solve challenging scaffold-hopping problems gathered in the publicly available $\mathcal{LH}$ benchmark. Komet is available open source at `https://komet.readthedocs.io` and all datasets, including LCIdb, can be found at `https://zenodo.org/records/10731713`.

1 Introduction

Most marketed drugs are small molecules that interact with a protein, modulating its function to prevent the progression of a disease. Therefore, the development of computational methods for the prediction of drug-target interactions (DTIs) has been an active field of research in the last decades, intending to reduce the number of wet-lab experiments to be performed for solving various problems related to drug discovery.

Among current computational approaches, we focus on chemogenomic DTI prediction methods, i.e. methods that predict whether a (molecule, protein) pair interacts or not, based on known DTIs in a reference database of interactions. In the present paper, we formulate DTI prediction as a classification problem: (molecule, protein) pairs are classified as interacting (i.e. positive examples, labelled $+1$) or not interacting (i.e. negative examples, labelled -1). Chemogenomic methods offer a global framework to predict drugs' protein interaction profiles, or proteins' drug interaction profiles, at large scales both in the

molecule and protein spaces, which cannot be performed by other methods (mainly QSAR and docking) directly. Therefore, chemogenomic methods make it possible to tackle important problems in drug design. In particular, predicting a drug's protein interaction profiles allows for the prediction of deleterious off-targets responsible for unwanted side-effects and potentially leading to drug withdrawal or beneficial off-targets that may be of interest to treat other diseases thus offering drug repositioning opportunities. Conversely, the prediction of a protein's drug interaction profiles is an interesting tool to solve scaffold hopping problems in the context of drug design[1] .

Enhancing the performance of DTI predictions requires to use of ever-larger training datasets and the development of Machine-Learning (ML) algorithms capable of scaling to these dataset sizes. In this paper, we tackle these challenges by presenting a curated large-sized dataset LCIdb and Komet, a GPU-friendly DTI prediction pipeline. These two components complement each other, resulting in state-of-the-art performance achieved with minimal use of computer resources.

2 State-of-the-art in chemogenomic approaches

Most chemogenomic DTI prediction methods rely on the global framework comprising three main steps and presented in Figure 1. Therefore, we present a short review of state-of-the-art approaches used in these three steps.

2.1 Step 1: Feature representations for proteins and molecules

Various methods[2] have been designed to compute feature representations for proteins and molecules. For molecules, several types of features are considered, as discussed in recent papers[3,4] . They can globally be classified into: (1) string-based formats such as the Simplified Molecular-Input Line-Entry System[5] (SMILES), or the International Chemical Identifier[6] (InChI); (2) table-based formats that represent the chemical graph of the molecule such as

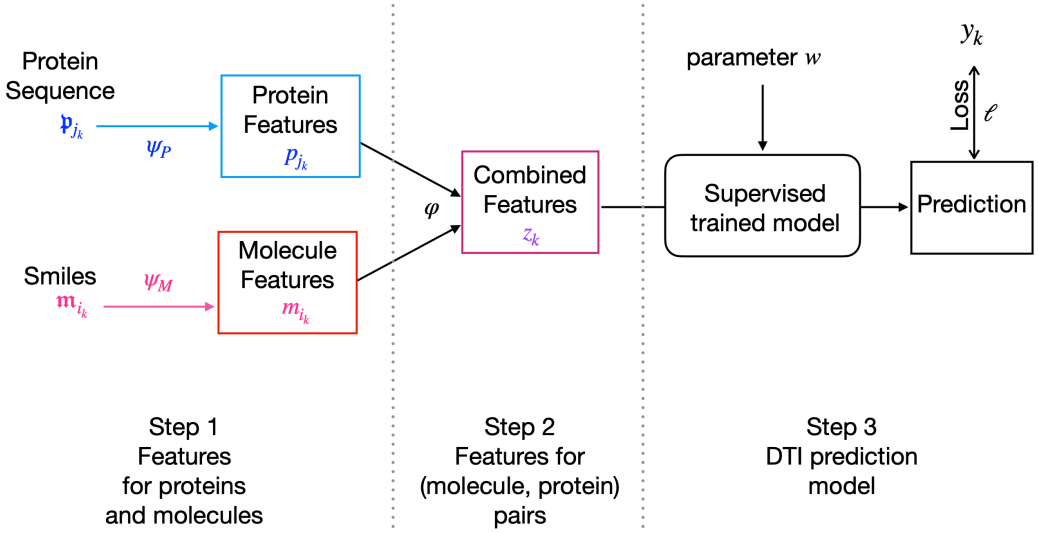


Figure 1: Global framework for DTI prediction in 3 key steps

the sdf format[7] ; (3) feature-based formats that consist in vectors whose elements encode various molecular characteristics. They include Morgan fingerprints or Extended-Connectivity fingerprints[8] (ECFP), as well as 2D and 3D pharmacophore fingerprints as described in the RDKit toolbox[9] ; (4) computer-learned representations that are derived by neural networks and used to encode molecules in deep learning approaches. These representations can be learned from recurrent neural networks or convolutional neural networks that use SMILES representations as input[10,11] . Graph convolutional networks have also been applied to 2D molecular graphs to learn small molecule representations[12,13] , and strategies to pre-train graph neural networks have been studied by Hu et al.[14] to compute molecule embeddings. Similar to natural language models, Mol2vec[15] and SMILES2vec[16] adapt the principles of the word2vec method[17] to learn embeddings for molecular structures. Additionally, transformer-based models like MolTrans[18] have emerged in this domain. Finally, other learned representation methods such as X-Mol[19] or MolGNet[20] use AutoEncoder (AE) techniques for molecular representation.

Similarly, proteins can globally be described by: (1) string-based representations corresponding to their primary sequence of amino-acids; (2) vector-based feature representations, where the elements of the vector are calculated according to various characteristics, as re-

viewed in Zhu et al.[21]. Such representations include the classically-used composition, transition, and distribution (CTD) descriptors[22] ; (3) computer-learned representations derived by neural networks in deep learning approaches. In this context, protein features can be acquired by a variety of deep learning architectures, including recurrent neural networks or convolutional neural networks[10,11] , as well as transformer models[18] . As in natural language models, protein embeddings can also be learned from pre-trained transformer-based models on external tasks such as ESM2[23] , or auto-encoder models such as ProtBert[24] and ProtT5XLUniref50[24] .

2.2 Step 2: Features for (molecule, protein) pairs

The second step of many DTI prediction pipelines consists of defining a representation for (molecule, protein) pairs, thus defining a latent space for pairs. The method that is used to define this latent space has a critical impact on the prediction performance, and a key aspect is that the features representing the (molecule, protein) pair should capture information about the interaction, which is not fully achieved by simple concatenation between molecule and protein features[25] . Therefore, step 2 usually consists of a non-linear mixing of the protein and molecule embeddings, to better encode information about interaction determinants. One common approach is to use the tensor product, which is equivalent to a Kronecker kernel[26,27] . Alternatively, in deep learning methods, the features for pairs can be learned from an interaction module that consists of fully connected multi-layer perceptrons[10,28–30] . Attention mechanisms applied to molecule and protein features constitute another option[11,18,31] . Then, the last layer of the network can be interpreted as an embedding for the (molecule, protein) pairs.

2.3 Step 3: DTI prediction model

The third step consists of a supervised classifier that is trained in the latent space of (molecule, protein) pairs, using a training dataset of positive and negative DTIs. These

classifiers include tree-based methods[32] and network-based inference approaches[33] . In linear models, step 3 consists of the optimization of the weights applied to the pair features calculated in step 2, according to a logistic loss, or a hinge loss for Support Vector Machines (SVM)[34] . For example, all methods of Pahikkala et al.[27], Nagamine and Sakakibara[35], Jacob and Vert[36], Playe et al.[37] rely on a linear model on a latent representation of pairs. In deep learning chemogenomic algorithms, step 3 relies on the pair features determined by the last layer of the neural network in step 2. The features' weights are optimized based on a loss function, typically binary cross-entropy, as the input progresses through the network in a feed-forward manner. This approach is used in several recent papers[10,11,18,28–31] .

2.4 Challenges in chemogenomic studies

Although different chemogenomic approaches have been proposed, as briefly reviewed above, all require a training dataset of positive and negative (molecule, protein) pairs. Recent ML chemogenomic algorithms have often been trained on small to medium-sized benchmarks that present various biases. Indeed, most classical benchmark datasets are extracted from a single biological database, and often favour drug and target families that have been more widely studied, and for which many known DTIs have been recorded[38,39] . Additionally, Bagherian et al.[40] highlights that most datasets use negative DTIs randomly chosen among pairs with unknown interaction status, and may therefore include false negative DTIs. One suggestion to overcome this problem is to derive training datasets from interaction databases that compile continuous values for binding affinities and choose stringent activity thresholds to derive confident positive and negative pairs, as suggested by Wang et al.[41] .

In addition, learning chemogenomic models that are broadly applicable and can generalize to many different families of proteins and drugs requires training on very large, high-quality, verified and well-established DTI datasets. This appears to be an important bottleneck since publicly available training datasets that meet these criteria are seldom.

However, training ML algorithms on very large datasets, potentially comprising hundreds

of thousands of molecules and therefore DTIs, leads to challenges in terms of computation times and memory requirements. In particular, the choice of the interaction module in step 2 has significant implications for computation time and memory resources in large-sized datasets. In the case of deep learning approaches, the complexity of neural network architectures, and the size of parameter spaces, may also contribute to the computational expense. Learning the interaction module requires iteratively adjusting the model parameters, leading to time-consuming training phases.

Overall, there is a critical need for chemogenomic approaches that can scale to very large datasets.

3 Contributions

In the present paper, we tackle the two important challenges mentioned above:

- in Section 4.2, we propose the Large Consensus Interaction dataset, called LCIdb hereafter, a new very large and high-quality dataset of DTIs that was designed to train chemogenomic ML algorithms for DTI prediction at large scale in the protein and molecule spaces. In particular, our dataset comprises a much larger number of molecules than commonly used datasets, offering a better coverage of the chemical space. Additionally, we paid attention to limiting potential bias among negative DTIs.

- in Sections 4.3 and 4.4, we propose Komet (Kronecker Optimized METhod), a simple yet efficient DTI prediction method that lies within the global pipeline presented in Figure 1. This method incorporates specific computation choices that provide scalability for very large training datasets, without compromising prediction performance.

We show that Komet competes with or outperforms state-of-the-art deep learning approaches for DTI prediction on medium-sized datasets, but that it scales much better to very large datasets in terms of prediction performances, computation time, and memory requirements (see Section 5.4).

Finally, we illustrate the performance of Komet trained on LCIdb using DrugBank as an external dataset for DTI prediction, and on a publicly available benchmark[42] designed to evaluate the performance of prediction algorithms in solving difficult scaffold hopping problems.

Komet adopts the global three-step framework shown in Figure 1, which aligns with recent computational pipelines, such as in Huang et al.[28]. However, Komet includes specific choices whose principles are presented below, while mathematical details are provided in Materials and Methods.

In step 1, molecule (resp. protein) features ψ_M (resp. ψ_P) are computed based on the distances of the considered molecule (resp. protein) to molecules in the training set, thus leveraging ideas from kernel methods. However, when the number of points in the training set becomes very large, the kernel matrix cannot be stored in memory. Therefore, from a small randomly chosen set of reference landmark molecules (resp. proteins) that are extracted for the training dataset, we use the Nyström approximation in addition to dimensionality reduction to efficiently compute embeddings ψ_M (resp. ψ_P) that approximate the feature maps corresponding to the chosen molecule and protein kernels. The parameters of the method are the numbers m_M (resp. m_P) of molecule (resp. protein) landmarks, and the dimension d_M (resp d_P) of the molecule (resp. protein) embeddings. The impact of these parameters is studied in Section 5.2.

In step 2, the interaction module consists of the tensor product between the protein and molecule spaces. One of the motivations for using the tensor product is that it offers a systematic way to encode correlations between molecule and protein features, independently from the choice of these features. A potential issue with this approach, however, is that the size of the resulting vector representation for the (molecule, protein) pair equals $d_M d_P$, and may be prohibitively large for computation time and memory. However, a classical property of tensor products is their factorization between inner products between the two tensor product vectors of molecules and proteins, called the Kronecker product. This allows

to avoid the explicit calculation of the interaction embedding, thus addressing the challenges posed by large datasets. Overall, as shown in Section 5, we found that this tensor product representation efficiently captured information about features interactions that govern the (molecule, protein) binding.

In step 3, Komet uses a simple SVM loss together with a BFGS optimization algorithm. This allows to leverage the Kronecker factorization of pairs' features, leading to a significant speedup of the training. It is important to note that, in the proposed approach, steps 2 and 3 are executed simultaneously. This is made possible by avoiding the implicit calculation of pairs' features, thanks to the Kronecker interaction module.

Our method is implemented in an open source software, leveraging parallel computation on GPU through a PyTorch[43] interface, and is available at `https://komet.readthedocs.io`. All datasets, including LCIdb, can be found at `https://zenodo.org/records/10731713`.

4 Materials and Methods

We first recall known and publicly available medium-sized DTI datasets that are used in the present paper (Section 4.1), and describe the construction of our large-sized DTI dataset LCIdb (Section 4.2). Then, we detail our computational approach for large-sized DTI prediction with Komet (Sections 4.3 and 4.4), and present the methodology used to compare the performance of Komet to those of a few state-of-the-art deep learning algorithms (Section 4.5). Finally, we introduce $\mathcal{LH}$, a publicly available benchmark dataset for scaffold hopping problems.

4.1 Medium-scale datasets

We first use medium-scale datasets to compare the performance of Komet to those of state-of-the-art algorithms: BIOSNAP, BIOSNAP_Unseen_drugs, BIOSNAP_Unseen_proteins, BindingDB, and DrugBank. The four first of these datasets are publicly available and were

established in Huang et al.[18]. They are used in various recent studies[28,44] . The last one is the DrugBank-derived dataset established in Najm et al.[45], from which we built an additional set called DrugBank (Ext) to be used as an external validation dataset, as detailed below.

Huang et al.[18] and Singh et al.[44] proposed to train and compare the performance of various DTI prediction algorithms based on splitting the datasets in training (Train), validation (Val), and test (Test) sets according to a 7:1:2 ratio. We followed this scheme to make fair comparisons. The number of drugs, targets, and interactions for all datasets used in the present study is given in Table 1. In addition, the number of positive and negative interactions across the Train, Val, and Test sets for all datasets used in the present paper is detailed in Table 2.

BIOSNAP in its three prediction scenarios The ChGMiner dataset from BIOSNAP[46] contains exclusively positive DTIs. Negative DTIs are generated by randomly selecting an equal number of positive DTIs, assuming that a randomly chosen (molecule, protein) pair is unlikely to interact. As proposed in Huang et al.[18], we considered three scenarios that are achieved based on different splits of BIOSNAP to build the Train, Val and Test sets. The first scenario, referred to as BIOSNAP, corresponds to random splitting of the DTIs in BIOSNAP. In the BIOSNAP_Unseen_targets scenario, the Train and Test sets do not share any protein. The BIOSNAP_Unseen_drugs dataset follows a similar process for molecules. The two last scenarios allow us to evaluate the generalization properties of the algorithm on proteins or molecules that were not seen during training.

BindingDB-derived dataset The BindingDB database[47] stores (molecule, protein) pairs with measured bioactivity data. We used a dataset derived from BindingDB and introduced by Huang et al.[18], where BindingDB is filtered to include only pairs with known dissociation constants (Kd). Pairs with Kd < 30 nM are considered positive DTIs, while those with Kd $>$ 30 nM values are considered negative. This leads to a much larger number of negative DTIs than positive DTIs. Although the resulting dataset does not include the whole BindingDB

database, for the sake of simplicity, it will be called BindingDB hereafter.

DrugBank-derived datasets We used the dataset provided in Najm et al.[45]. This dataset was built by filtering drug-like molecules and human protein targets in the DrugBank database, adding an equal number of negative DTIs through balanced sampling. More precisely, to avoid bias towards well-studied proteins for which many interactions are known, negative examples are randomly chosen among unlabeled DTIs in such a way as to ensure that each protein and each drug appear an equal number of times in positive and negative interactions, using a greedy algorithm. This dataset will be referred to as DrugBank in the following, for the sake of simplicity, and corresponds to the dataset called DrugBank (S1) in the original paper.

We created another dataset called DrugBank (Ext), derived from the above dataset, and used it as an external validation to compare the prediction performances of the considered algorithms when trained on BindingBD or on LCIdb. Positive interactions from DrugBank were selected, excluding those present in BindingDB and LCIdb, to gather a set of positive DTIs absent from the BindingDB and LCIdb datasets. All other DTIs in DrugBank are kept in DrugBank (Ext). As above, balanced negative interactions were added in DrugBank (Ext), using the greedy algorithm of Najm et al.[45].

4.2 Building the new large scale dataset LCIdb

To build a large-sized dataset of DTIs, we started from the database described by Isigkeit et al.[48], as it combines and curates data from prominent databases including ChEMBL[49], PubChem[50], IUPHAR/BPS[51], BindingDB[52], and Probes & Drugs[53]. We filtered the DTIs in this database according to 4 filters, as detailed below.

Filtering positive DTIs : (1) Chemical structure quality filter: for DTIs present in several of the source databases, we only retained those for which the SMILES representation of the molecule was identical in all sources, to exclude potential erroneous (molecule, protein)

pairs. We only kept molecules with molecular weights between 100 and 900 g.mol^{-1}, which is a standard choice for selecting drug-like molecules. Among these molecules, we selected those that target at least one human protein. These filters were used because the goal was to build a training dataset of DTIs that are relevant in the context of drug discovery projects.

(2) Bioactivity filter: we retained only DTIs for which inhibition constant Ki, dissociation constant Kd, or half maximal inhibitory concentration IC50 measurements were available in at least one of the source databases.

(3) Quantitative bioactivities filter: for DTIs with bioactivity measurements present in multiple source databases, we only retained those with bioactivities within one log unit from one another.

(4) Binary labelling of DTIs: Bioactivity measurements were converted into binary interactions based on a threshold. If the bioactivity value was less than 100 nM (10^{-7}M), the interaction was classified as positive DTI (binding). If the bioactivity value (Ki, Kd or IC50) was greater than 100μM (10^{-4}M), the interaction was classified as negative DTI (non-binding). When the bioactivity value was in the intermediate range, i.e. between 100 nM and 100μM, DTIs were classified as known non-conclusive.

This scheme leads to the selection of 274 515 molecules, 2 069 proteins, 402 538 positive interactions and 8 296 negative interactions. We then added negative interactions to build a balanced dataset, as described below.

Completion of a balanced negative DTI dataset: We randomly split the dataset into training (Train), validation (Val), and testing (Test) sets in a 7:1:2 ratio. We used unlabeled DTIs to include negative interactions in these three sets, assuming most unknown DTIs are negative. For the training set, the selection of additional negative interactions should be designed with care to tackle two classical issues: (1) reduce the number of false negative DTIs present in the training set; (2) correct potential statistical bias in the database towards highly studied molecules or proteins. To take into account the former, we excluded known

non-conclusive interactions, and for the latter, we applied the algorithm by Najm et al.[45] for selecting additional negative DTIs. In the Val and Test sets, remaining negative and randomly chosen unknown interactions are added. These sets form LCIdb, mirroring the DrugBank dataset scenario discussed in Section 4.1.

Different prediction scenarios: To evaluate performance in different prediction scenarios, we also derive different datasets from to LCIdbbased on specific splits of the Train, Val, and Test sets, as proposed in Huang et al.[18] and Singh et al.[44]. Datasets are built to correspond to LCIdb, LCIdb_Unseen_drug, LCIdb_Unseen_protein, and LCIdb_Orphan (unseen molecule and protein) scenarios. We added the Orphan case, which presents the greater difficulty for prediction tasks.

More precisely: (1) LCIdb is balanced in positive and negative pairs chosen at random; (2) LCIdb_Unseen_drugs is built so that (molecule, protein) pairs in one of the Train/Val/Test sets only contain molecules that are absent from the two other sets; (3) LCIdb_Unseen_targets is built so that (molecule, protein) pairs in one of the Train/Val/Test sets only contain proteins that are absent from the two other sets; (4) LCIdb_Orphan is built so that (molecule, protein) pairs in one of the Train/Val/Test sets only contain proteins and molecules that are absent from the two other sets. The number of drugs, targets, and interactions in these four datasets is given in Table 1. Table 2 provides the number of positive and negative interactions across the Train, Val, and Test sets in these four datasets.

4.3 Features for proteins and molecules in Komet

The initial step of our DTI prediction framework consists of computing simple and fixed features for molecules and proteins.

Nyström-based molecule and protein features ψ_M and ψ_P in Komet: In Komet, we encode molecules and proteins leveraging the Nyström approximation[54,55] and dimensionality reduction. For a molecule $\mathfrak{m}$ (for instance, represented as a SMILES string), let us explain

how we compute its embedding $\psi_M(\mathfrak{m})$ in $\mathbb{R}^{d_M}$. The same computation applies for the protein embedding $\psi_P(\mathfrak{p}) \in \mathbb{R}^{d_P}$ (where $\mathfrak{p}$ is for instance a FASTA string). ψ_M is built from a small set of landmark molecules $\{\hat{\mathfrak{m}}_\ell\}_{\ell=1}^{m_M}$ with $m_M \geq d_M$ that are randomly chosen in the training dataset. The other ingredient in Komet is a kernel $k_M(\mathfrak{m}, \mathfrak{m}')$ that can be viewed as a similarity measure between two molecules, and that is used to define molecule features (the choice of this kernel is discussed below). We first compute a small kernel matrix over the landmarks: $\hat{K}_M \in \mathbb{R}^{m_M \times m_M}$ where $(\hat{K}_M)_{i,j} \coloneqq k_M(\hat{\mathfrak{m}}_i, \hat{\mathfrak{m}}_j)$. Then, we define the extrapolation matrix $E \in \mathbb{R}^{m_M \times d_M}$ from the Singular Value Decomposition of $\hat{K}_M = U \operatorname{diag}(\sigma) U^\top$ as $E \coloneqq U[:, : d_M] \operatorname{diag}(\sigma_s^{-1/2})_{s=1}^{d_M}$. This extrapolation matrix allows to compute molecule embedding for any molecule $\mathfrak{m}$ (in particular for molecules in the training set that are not in the landmark set) as:

$$\psi_M(\mathfrak{m}) \coloneqq \Big(\sum_{\ell=1}^{m_M} E_{\ell,s}\, k_M(\hat{\mathfrak{m}}_\ell, \mathfrak{m}) \Big)_{s=1}^{d_M} \in \mathbb{R}^{d_M}.$$

Note that when no dimensionality reduction is performed ($d_M = m_M$), this embedding satisfies the relation $k_M(\hat{\mathfrak{m}}_i, \hat{\mathfrak{m}}_j) = \langle \psi_M(\hat{\mathfrak{m}}_i), \psi_M(\hat{\mathfrak{m}}_j) \rangle$ (see Appendix C for details). In this case, for any molecule $\mathfrak{m}$ that is not in the landmark set, $k_M(\mathfrak{m}, \hat{\mathfrak{m}}_i) \approx \langle \psi_M(\mathfrak{m}), \psi_M(\hat{\mathfrak{m}}_i) \rangle$, according to a so-called Nyström approximation (see Appendix C for details). Hence, E allows us to "extrapolate" the embedding ψ_M, which is the underlying kernel map of k_M, from the landmarks to new molecules.

Finally, we mean-center and normalize the features:

$$\psi_M(\mathfrak{m}) \leftarrow \frac{\psi_M(\mathfrak{m}) - \bar{m}}{\|\psi_M(\mathfrak{m}) - \bar{m}\|} \quad \text{where} \quad \bar{m} \coloneqq \frac{1}{m_M} \sum_{\ell=1}^{m_M} \psi_M(\mathfrak{m}_\ell).$$

We adopt a similar approach to build ψ_P but use all proteins from the data set as landmarks, as their number is much smaller. Again, because the number of proteins is small enough, we do not apply dimensionality reduction: $d_P = m_P = n_P$.

Choice of molecule and protein kernels: The embeddings ψ_M and ψ_P depend on the choice of molecule and protein kernels. We follow the choices made in Playe et al.[37] and adopt the Tanimoto kernel k_M for molecules. For each molecule $\mathfrak{m}$ represented in SMILES format, we calculate ECFP4 fingerprints, generating a 1024-bit binary vector using the RDKit package[9] . Values of the Tanimoto kernel between two molecules are then computed as the Jaccard index between their fingerprints. The Tanimoto kernel hence measures the similarity between two molecules based on the substructures they share. For each protein represented as a sequence $\mathfrak{p}$ of amino acids, we opt for the Local Alignment kernel (LAkernel)[56] . This kernel k_P detects remote homology by aggregating contributions from all potential local alignments with gaps in the sequences, thereby extending the Smith–Waterman score[57] . We used the same hyperparameters as Playe et al.[37] , where they were adjusted by cross-validation.

4.4 Large-scale chemogenomic framework with Komet

We address DTI prediction as a supervised binary classification problem, incorporating established steps, as outlined in Sections 2.2 and 2.3.

Features for molecule-protein pairs: Let us consider a DTI dataset containing molecules and proteins $(\mathfrak{m}_i)_{i=1}^{n_M}$ and $(\mathfrak{p}_j)_{j=1}^{n_P}$, where n_M and n_P are respectively the number of molecules and proteins in the dataset. To alleviate notations, in what follows, we denote by $m \coloneqq \psi_M(\mathfrak{m})$ the embedding of a molecule $\mathfrak{m}$ and by $p \coloneqq \psi_P(\mathfrak{p})$ the embedding of a protein $\mathfrak{p}$.

The training dataset consists of a set of n_Z (molecule, protein) pairs with indices $(i_k, j_k)_{k=1}^{n_Z}$ and their associated labels $y_k \in \{-1, 1\}$. If $y_k = 1$ (resp. -1), molecule $\mathfrak{m}_{i_k}$ and protein $\mathfrak{p}_{j_k}$ interact (resp. do not interact). The classification is performed in the space of pairs, which we define as the tensor product of the space of molecules and the space of proteins. Hence, the embedding for pairs is given by $\varphi(\mathfrak{m}, \mathfrak{p}) \coloneqq (m[s]p[t])_{1\leq s\leq d_M, 1\leq t\leq d_P} \in \mathbb{R}^{d_Z}$, where $m[s]$ is the s-th coordinate of m and $p[t]$ is the t-th coordinate of p.

Thus, the space of pairs has dimension $d_Z = d_M d_P$. This embedding corresponds to the use of a Kronecker kernel, already shown to be efficient in several publications[27,37,45] . Using a Kronecker kernel is crucial in our approach, not only because it is a state-of-the-art method, but also due to its favourable mathematical properties, which we will detail below. It is worth noting that our approach avoids explicitly calculating the embedding φ, which mitigates the computational burden associated with the large value of d_Z.

SVM classification: Our classification approach follows previous work (see Section 2.3), relying on a linear model with weight vector $w \in \mathbb{R}^{d_Z}$ and bias term $b \in \mathbb{R}$. The class decision for a pair feature vector $z \in \mathbb{R}^{d_Z}$ is determined by $\text{sign}(\langle w, z\rangle + b) \in \{-1, 1\}$. The parameters w and b are obtained by minimizing a penalized empirical risk:

$$\min_{w\in\mathbb{R}^{d_Z}} \sum_{k=1}^{n_Z} \ell(\langle w, z_k\rangle + b, y_k) + \frac{\lambda}{2}\|w\|^2. \tag{1}$$

In Komet, we employ a Support Vector Machine (SVM) classification where $\ell(y', y) = \max(0, 1 - yy')$.

The minimization of Equation (1) is computationally demanding, particularly when n_Z and d_Z are large. A conventional Stochastic Gradient Descent (SGD)[58] can result in slow convergence. Therefore, we use an alternative approach that leverages the specific structure of our embedding φ, as was previously done by Airola and Pahikkala[59]. Specifically, we exploit: (1) the tensor product nature of φ and (2) the fact that the sizes n_M and n_P of the input databases are much smaller than the number n_Z of interactions.

Efficient computation The core ingredient leading to a significant improvement in computational efficiency on a large-sized dataset is the efficient computation of the gradient by bypassing the evaluation of φ. Indeed, the function to be minimized in Equation (1) has the form $L(Zw + b) + \frac{\lambda}{2}\|w\|^2$, where the rows of $Z \in \mathbb{R}^{n_Z \times d_Z}$ are the vectors $z_k^\top$, and L takes into account ℓ and y. The main computational burden for evaluating this function and its

gradient is the computation of Zw. A naive implementation would require $n_Z d_Z$ operations just to compute Z, which would be unavoidable if one used a generic φ, such as a deep neural network. However, we bypass this bottleneck by directly computing Zw. This relies on the following identity:

$$(Zw)_k = \langle w, z_k \rangle_{\mathbb{R}^{d_Z}} \overset{(a)}{=} \langle m_{i_k}, W p_{j_k} \rangle_{\mathbb{R}^{d_M}} \overset{(b)}{=} \langle m_{i_k}, q_{j_k} \rangle_{\mathbb{R}^{d_M}}, \tag{2}$$

where $W \in \mathbb{R}^{d_M \times d_P}$ is such that it has w as flattened representation in $\mathbb{R}^{d_Z}$ and $q_j := W p_j$.

Equality (a) exploits the tensor product structure of φ. Please refer to the Appendix D for a detailed proof.

Equality (b) is interesting because all the $(q_j)_{j=1}^{n_P}$ can be computed in only $n_P d_Z$ operations. Once this has been computed, evaluating all n_Z values of $(Zw)_k = \langle m_{i_k}, q_{j_k} \rangle_{\mathbb{R}^{d_M}}$ require $n_Z d_M$ operations. We then minimize Equation (1) using a full batch method, which enables the use of efficient quasi-Newton methods. In practice, we use the BFGS method with limited memory[60] . The complexity of our algorithm is then $\mathcal{O}(n_P d_Z + n_Z d_M)$ where $\mathcal{O}(.)$ takes into account the number of iterations of the BFGS algorithm to reach a fixed accuracy. This number is quite small (10 to 50) in our numerical experiments. Note that we can exchange the role of the protein embeddings and the molecule embeddings in this calculation, resulting in a complexity of $O(n_M d_Z + n_Z d_P)$. In our setting $n_P \ll n_M$ so we prefer the initial formulation of Equation (2).

From classification to probability estimation Once the weight vector w has been computed, Platt scaling[61] computes a probability of belonging to the positive class using the formula $p_k := \sigma(-y_k(s\langle z_k, w\rangle + t))$, where σ is the logistic function $\sigma(u) = \frac{e^u}{1+e^u}$, and the scale s (which can be interpreted as a level of confidence) and the offset t need to be optimized. This is achieved by minimizing the same energy as in logistic regression:

$$\min_{s,t} E(s,t) := \sum_k \ell(-y_k(s\langle z_k, w\rangle + t))$$

where $\ell(u) := \log(1 + e^u)$. We use the BFGS method to solve this equation.

4.5 Evaluation of prediction performance

Comparing the prediction performances of various algorithms requires defining the evaluation strategies and the metrics used.

Metrics: We formulate the DTI prediction problem as a classification task, therefore, we use AUPR (area under the precision–recall curve), ROC-AUC (area under the ROC curve) and prediction accuracy, as metrics to compare prediction performances.

Evaluation strategies: There is only one hyperparameter in our model, as shown in Equation (1). We select the best $\lambda \in \{10^{-11}, 10^{-10}, ..., 10, 100\}$ based on AUPR performance from the validation (Val) set. This value is used to train the parameters of the model 5 times on the training set, each time with new landmark molecules and approximated molecule features, and we calculate the mean prediction probability. The final computed model is then evaluated on the Test set.

Implementation details and data availability: We use a server with 2 CPUs and 1 NVIDIA A40 GPU with 48 GB of memory. We provide a Python implementation of Komet and the code used to build LCIdb at `https://komet.readthedocs.io`. We provide the LCIdb itself at `https://zenodo.org/records/10731713` and other files at `https://github.com/Guichaoua/komet/tree/main/data`.

4.6 Application to the scaffold hopping problem

To assess computational methods for solving large-step scaffold hopping problems, Pinel et al.[42] built a high-quality benchmark called Large-Hops ($\mathcal{LH}$) comprising 144 pairs of highly dissimilar molecules that are active against diverse protein targets. In $\mathcal{LH}$, one active molecule is considered as known, and the second active molecule must be retrieved

among 499 decoys carefully selected to avoid statistical bias. This dataset is available at `https://github.com/iktos/scaffold-hopping`.

For each case, the considered algorithms were trained with one molecule of the pair considered as the only known active for the query protein. If the known interaction was absent from the training dataset, it was added to it, and all other interactions involving the query protein potentially present in the database were removed. After training, the algorithms ranked the unknown active and the 499 decoy molecules, according to the predicted binding probabilities of the (molecule, query protein) pairs. The lower the rank of the unknown active, the better the prediction performance.

As in[62] , we employ three criteria to compare ranking algorithms: (1) Cumulative Histogram Curves (CHC) are drawn to represent the number of cases where a method ranks the unknown active below a given rank, with better-performing methods having curves above others; (2) Area Under the Curve (AUC) of CHC curves provide a global quantitative assessment of the methods; (3) the proportion of cases where the unknown active was retrieved in the top 1% and 5% best-ranked molecules.

5 Results

In the following, we first present the new LCIdb DTI dataset, analyze its coverage of the molecule and protein spaces, and compare it to other available and widely used datasets. Next, we explore different parameters within the Komet pipeline, to find a balance between speed and prediction performance. We then show that Komet displays state-of-the-art DTI prediction performance capabilities on the considered medium- and large-sized datasets, and on the external dataset DrugBank (Ext). Finally, we highlight the efficiency of our approach on the publicly available ($\mathcal{LH}$) benchmark dataset designed to address challenging scaffold hopping problems.

5.1 Coverage of the protein and molecule spaces in the LCIdb dataset

Different reviews introduce numerous biological databases that can be used to derive large-sized training datasets[2,40] , to best cover the protein and molecule spaces. Following Isigkeit et al.[48], we combine and filter curated data from prominent databases including ChEMBL[49] PubChem,[50] IUPHAR/BPS,[51] BindingDB[52] , and Probes & Drugs[53] , and built LCIdb, a large-sized high-quality DTI database, as detailed in Section 4.2. Table 1 provides the numbers of molecules, proteins, and interactions in all the DTI training datasets considered in the present study.

Table 1: Numbers of molecules, proteins, and positive/negative DTIs in the considered datasets. "random" indicates that negative DTIs were randomly chosen among unlabeled DTIs. "balanced" indicates that negative DTIs were randomly chosen among unlabeled DTIS, but in such a way that each protein and each drug appears in the same number of positive and negative DTIs.

Datasets	Molecules	Proteins	Positive DTIs	Negative DTIs
BIOSNAP	4,510	2,181	13,836	(13,647 random)
Unseen_drugs			13,836	(13,647 random)
Unseen_targets			13,836	(13,647 random)
BindingDB	7,161	1,254	9,166	23,435
DrugBank	4,813	2,507	13,715	(13,715 balanced)
DrugBank (Ext)	4,257	1,216	10,838	(10,838 balanced)
LCIdb	274,515	2,069	402,538	8,296 (+ 394,242 balanced)
Unseen_drugs	274,515	2,069	402,538	8,296 (+ 394,242 balanced)
Unseen_targets	232,018	2,069	431,011	8,296 (+ 422,715 balanced)
Orphan	143,255	2,069	151,690	8,296 (+ 143,394 balanced)

Table 1 reveals that DrugBank- or BIOSNAP-derived datasets and BindingDB share a few characteristics: their numbers of proteins are similar (in the range of one to two thousand), their numbers of molecules are modest (in the range of a few thousand), their number of known positive DTIs are similar (in the range of thousands). BindingDB contains true negative DTIs, while the DrugBank- or BIOSNAP-derived datasets use DTIs of unknown status as negative DTIs, randomly chosen for BIOSNAP-derived datasets, and randomly chosen in such a way that all molecules and proteins appear in the same number of posi-

tive and negative DTIs (labelled "balanced" in Table 1) for the DrugBank-derived datasets. Overall, these observations underline the need for a larger dataset, as required for chemogenomic studies. As shown in Table 1, LCIdb includes 40 times more molecules and 30 times more positive DTIs than the other considered datasets, the number of human proteins being in the same order of magnitude.

However, it is important to evaluate whether this larger number of molecules corresponds to better coverage of the chemical space and whether the different datasets are comparable in terms of biological space coverage. Indeed, the chemical space is estimated to be extremely large[63] , and efficient sampling of this space by the training dataset is expected to have a great impact on the generalization properties of the prediction models.

We use the t-SNE algorithm[64] on the molecule features ψ_M derived from the Tanimoto kernel, as defined in Section 4.3, to visualize the resulting high-dimensional molecular space in a two-dimensional space, thus facilitating analysis. Figure 2 shows not only that LCIdb contains a much larger number of molecules than BIOSNAP, DrugBank, and BindingDB, but also that the molecules it contains are more diverse,

While it is far from covering the entire vast and unknown chemical space, LCIdb spans a much larger area on the t-SNE plot, therefore providing a better sampling of this space overall. In addition, it shows that LCIdb also covers the chemical space more uniformly than the other datasets. Figure 2 also highlights that the BIOSNAP dataset was built from DrugBank, displaying similar patterns of red clusters of molecules.

We also ran the t-SNE algorithm based on Tanimoto features computed using an alternative set of molecule landmarks, and based on other molecule features (see Figure 2 of the Appendix A). In all cases, plots confirmed the above conclusions that LCIdb presents a wider and more uniform coverage of the chemical space, underscoring their robustness.

Isigkeit et al.[48] analyze the space formed by the five databases from which LCIdb originates. Specifically, they examined distributions of common drug-like features such as molecular weight, the number of aromatic bonds, the number of rotatable bonds, and predicted

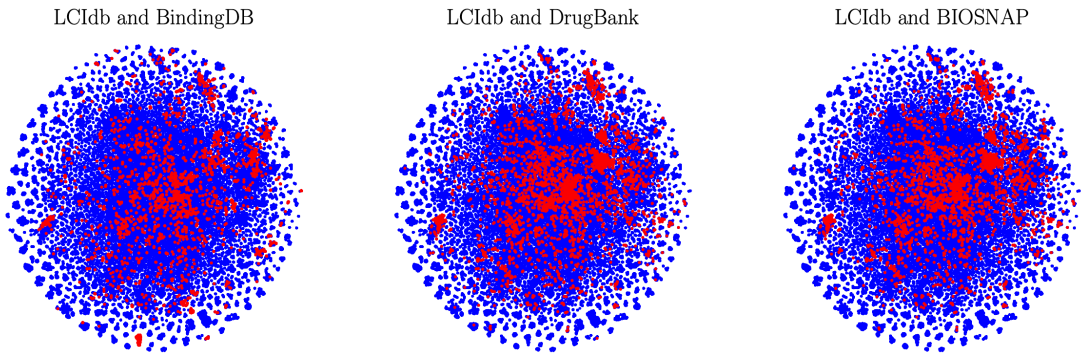


Figure 2: 2D representation of the molecular space with the t-SNE algorithm based on molecule features. In blue: the large-sized LCIdb dataset, and in red: the medium-sized DrugBank, BIOSNAP, and BindingDB datasets.

octanol-water partition coefficients. The authors observed that these distributions are similar across all sources. In Appendix A, we present plots illustrating the distribution of drugs in our LCIdb dataset, based on the five databases from which they originate.

By contrast, the number of human proteins is comparable across all considered datasets, although not identical (see Figure 3). We also used t-SNE plots based on protein features defined in Section 4.3 to explore the coverage of the protein space by LCIdb . As shown in the resulting 2D representation presented in Figure 4, the protein space covered by LCIdb contains clusters that align with functional families of proteins. This was expected when using features calculated using the LAkernel (see Section 4.3), since proteins that share high sequence similarity usually belong to the same protein family. Thus, we can leverage this representation to discuss the diversity of proteins in our datasets. As shown in Figure 5, although LCIdb contains slightly fewer proteins than the DrugBank dataset, their coverage of the biological space is similar. BIOSNAP appears to have a lower coverage of a few protein clusters (such as protein kinases), while BindingDB focuses more on a few clusters corresponding to specific protein families.

As detailed in Section 4.1, for BIOSNAP and LCIdb, additional datasets are derived, as suggested in various studies[10,11,27,37,65] , as well as in Huang et al.[18] and Singh et al.[44], two papers that respectively introduced the MolTrans and ConPLex algorithms. They cor-

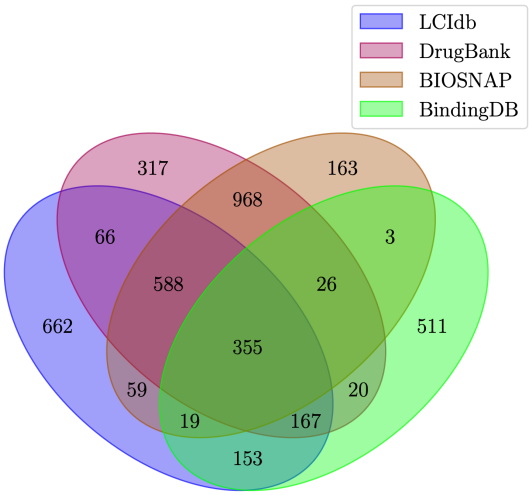


Figure 3: Overlap between LCIdb , DrugBank, BIOSNAP, and BindingDB datasets in terms of proteins.

respond to scenarios of varying difficulties encountered in real-life situations in drug discovery projects: (1) the Unseen_drugs case is typical of new drugs identified in phenotypic screen and for targets are searched to elucidate the drug's mechanism of action; (2) the Unseen_targets case is typical of newly identified therapeutic targets against for which repositioning opportunities if known drugs are searched; (3) The Orphan case is typical of a new therapeutic target has been identified, and against which ligands (inhibitors or activators) are searched at large scale in the molecule space.

The composition of the corresponding datasets is provided in Table 1. In Huang et al.[18] and Singh et al.[44], only the Unseen_drugs and Unseen_targets were considered, but we added the Orphan case for LCIdb.

Finally, following Huang et al.[18] and Singh et al.[44], in all the prediction experiments reported in the Results, the prediction performances of all considered algorithms are computed based on the Test set, after optimization of the parameters on the Train/Val sets built from the considered DTI datasets. Details about the Train/Val/Test sets are given in Section 4.1). The number of molecules, proteins and interactions in these sets are provided in Table 2.

Figure 4: Representation of the protein space in LCIdb according to the t-SNE algorithm based on protein features derived from the LAkernel. A few protein families are labelled and coloured.

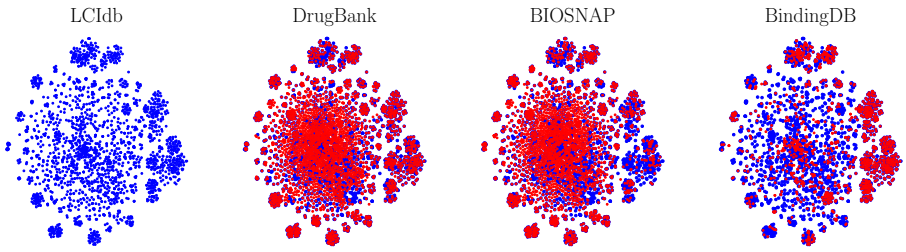


Figure 5: Representation of the protein space according to the t-SNE algorithm based on protein features derived from the LAkernel. In blue: LCIdb, in red: DrugBank, BIOSNAP, and BindingDB.

Table 2: Full specification of the Train/Val/Test sets for all datasets. DrugBank (Ext) is only used as an external validation dataset when algorithms are trained on BindingDB or LCIdb (see Section 5.4.3). Therefore, no Train, Val, or Test sets were built for DrugBank (Ext)

Datasets	#Train	#Val	#Test
BIOSNAP	9,670/9,568	1,396/1,352	2,770/2,727
Unseen_drugs	9,535/9,616	1,383/1,353	2,918/2,675
Unseen_targets	9,876/9,499	1,382/1,386	2,578/2,762
BindingDB	6,334/6,334	927/5,717	1,905/11,384
DrugBank	10,972/10,972	1,098/1,098	1,645/1,645
DrugBank (Ext)	-	-	10,838/10,838
LCIdb	161,015/161,015	32,204/32,204	48,304/48,304
Unseen_drugs	156,942/156,942	32,326/32,326	56,328/56,328
Unseen_targets	154,683/161,015	32,349/32,349	60,822/60,822
Orphan	59,132/59,132	10,145/10,145	22,503/22,503

5.2 Parameters set-up of the model

Due to the vast number of molecules in LCIdb (see Table 1), our Komet algorithm incorporates the Nyström approximation to calculate molecular features as well as a dimension reduction, which involved parameters m_M (number of landmark molecules) and d_M (dimension of molecular features). By contrast, for proteins, we retain all the proteins in the Train set as protein landmarks ($n_P = m_P = d_P$). It is therefore crucial to evaluate the potential impact of the m_M and d_M parameters on the prediction performance of Komet, the resulting gain in calculation time, and to study whether good default values can be determined. This study was performed on LCIdb_Orphan and BindingDB, respectively large- and medium-

sized datasets. LCIdb_Orphan was chosen as the large dataset for exploring the impact of m_M and d_M because it corresponds to the most difficult dataset, on which it is critical not to degrade the prediction performances. Figure 6 shows that for both datasets, we can significantly reduce the number of landmark molecules (m_M) and the dimension (d_M) of molecular features without losing performance, while saving time and computational resources. In particular, results on BindingDB illustrate that reducing m_M from the total number of molecules (7 161) to 5 000 or 3 000 does not significantly affect precision-recall curves. In addition, for the large-sized datasets like LCIdb_Orphan, reducing m_M from 10 000 to 5 000 or 3 000 does not degrade the prediction performance.

Moreover, the precision-recall curves reach a plateau for d_M values between 1 000 and 2 000, suggesting that we can limit the number of molecular features without a loss in performance. This observation is confirmed with the medium-size dataset BindingDB, for which a plateau is also reached for similar values of d_M, particularly when no approximation was made ($n_M = m_M = 7\,161$). This suggests that d_M values in the range of 1 000-2 000 could be good default values for the number of features used in molecular representations. In addition, Figure 6 illustrates that, as expected, reducing m_M and d_M significantly reduces computational time and GPU memory usage. Consequently, we choose $d_M = 1\,000$ and $m_M = 3\,000$ as a good compromise to design a rapid and less resource-intensive algorithm, without majorly compromising performance.

5.3 Impact of different molecule and protein features on Komet prediction performances

We explored the impact of molecule and protein features on the prediction performances of Komet. For molecule features, we consider the features extracted from the Tanimoto kernel between ECFP4 fingerprints, as described in Section 4.3, with the ECFP4 fingerprints themselves. This is equivalent to using the dot product between ECFP4 fingerprints, rather than the Tanimoto kernel, and no approximation (neither through the choice of a reduced set of

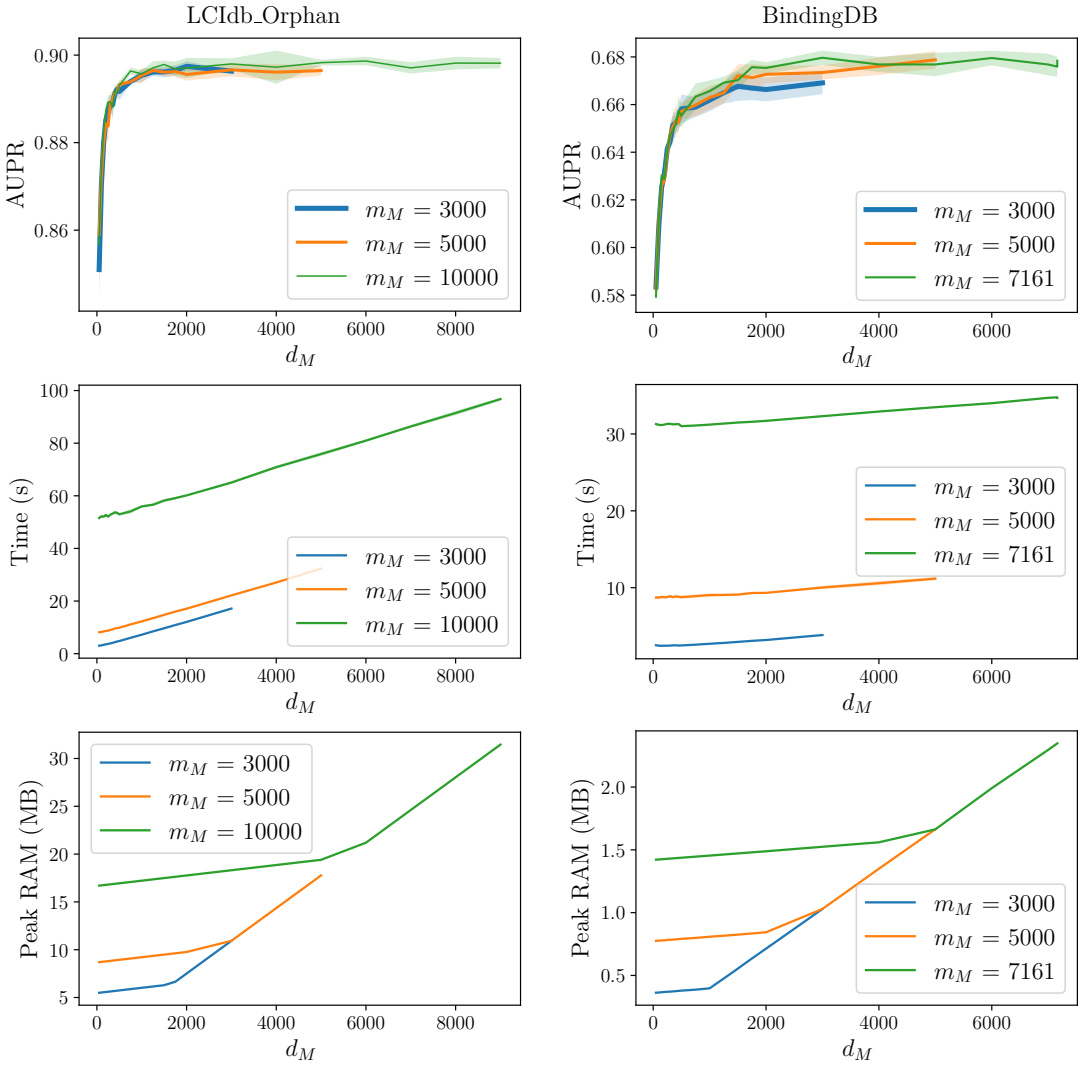


Figure 6: Influence of m_M and d_M on AUPR on the validation set of LCIdb_Orphan, computation time (in seconds) and usage and peak GPU RAM (in Gb). In each graph, the three curves correspond to three values of m_M, i.e. the number of random molecules used by the Nyström approximation of the molecular kernel. Error bars correspond to the choice of different landmark molecules. Graphs on the left refer to the large-sized dataset (LCIdb_Orphan) and on the right to the medium-sized dataset (BindingDB).

landmark molecules nor through dimensionality reduction). Previous studies have shown that ECFP4 fingerprints perform as well as state-of-the-art fingerprint-based 3D models[66], and are not significantly outperformed by embeddings learned from deep learning methods[67]. Therefore, we also considered pre-trained Graph Neural Networks (GNNs) for the generation of molecule features. Specifically, Hu et al.[14] outline several pre-training strategies for GNNs using a dataset of two million molecules. These strategies include supervised learning for molecular property prediction and semi-supervised learning methods such as context prediction, mutual information maximization between local and global graph representations, encouraging similarity in representations of adjacent nodes while differentiating distant nodes, and predicting masked node and edge attributes. We use the trained models adapted by Li et al.[68] to calculate the molecular embeddings and we present in Table 3 only the features giving the best results. These features correspond to a model for supervised learning for molecular property prediction combined with semi-supervised learning on context prediction.

For proteins, we compare features extracted from the LAkernel, as described in Section 4.3, with features computed similarly, but using the 20 605 proteins of the UniProt human proteome[69] as landmark proteins, with a dimension reduction step ($d_P = 1\,200$). In addition, we used three embeddings from deep learning models: ESM2[23] which is based on transformers, and ProtBert[24] and ProtT5XLUniref50[24] which are based on variational autoencoders trained on very large data sets of proteins.

Results are displayed in Table 3 for LCIdb_Orphan, the most challenging large-sized dataset. They show that the features proposed for Komet in the present study lead to the best prediction performance. However, replacing the molecular embeddings built from the Tanimoto kernel between ECFP4 fingerprints with the ECFP4 fingerprints themselves barely degrades the performance. This could indicate that the molecular information lost by approximations (using a subset of landmark molecules and performing dimensionality reduction) is compensated by the Tanimoto kernel being a more appropriate kernel than

Table 3: AUPR of Komet using different molecule and protein features on the LCIdb_Orphan dataset. "Tanimoto" features are built from the Tanimoto kernel between ECFP4 fingerprints as described in Section 4.3, and the "GNN supervised contextpred" features are available in the DGL-LifeSci package[68] . "LAkernel" features are built from the Local Alignment kernel between proteins as described in Section 4.3. "UniProt LAkernel" features are built in the same way, but considering all human proteins from UniProt as landmarks proteins and using dimensionality reduction.

		Protein embedding				
		LAkernel	UniProt LAkernel	ProtBert	ProtT5XLUniref50	ESM2
Molecule embedding	Tanimoto	0.897	0.873	0.834	0.632	0.864
	ECFP4	0.893	0.861	0.829	0.630	0.866
	GNN supervised contextpred	0.887	0.857	0.834	0.618	0.858

the dot product. The protein embedding derived from the LAkernel on the 2 069 druggable proteins[69] , i.e. human proteins for which at least one drug-like ligand is known, leads to the best prediction performances. One explanation could be that the human druggable proteins present some sequence and family bias, and do not span the whole human proteome space. As a consequence, generic embeddings learned in deep learning approaches on very large sets of proteins from multiple species (ProtBert, ProtT5XLUniref50, ESM2), may be less appropriate for the specific problem DTI prediction in the context of drug-like molecules and human druggable proteins. This may also explain why features derived from the LAkernel computed on 20 605 human proteins also degrade the prediction performance. For this latter case, using the whole human proteome comes with the necessity of dimensionality reduction ($d_P = 1\,200$), which may also contribute to reducing the prediction performance.

As a consequence, the molecule features derived from the Tanimoto kernel on and the ECFP4 fingerprints and the protein features derived from the LAkernel on the 2 069 druggable proteins are used in all the following prediction experiments performed with Komet. However, one should note that except for the ProtT5XLUniref50 protein features, the prediction performances of Komet remain relatively stable to molecule and protein features.

5.4 Comparison of the prediction performances between Komet and deep learning algorithms

Because LCIdb is large, deep learning methods are expected to perform well on it[70] . Therefore, we compare Komet to the recently proposed ConPLex[44] algorithm, a deep learning approach that was shown to achieve state-of-the-art performance on medium-sized datasets.

ConPLex uses as input molecules encoded with Morgan fingerprints and proteins encoded by pre-trained Protein Language Model ProtBert[24] . The latent space for (molecule, protein) pairs is learned through a non-linear transformation into a shared latent space. This learning phase combines a binary DTI classification phase with a contrastive divergence phase, in which the DUD-E database[71] , comprising 102 proteins together with ligands and non-binding decoys, is used to compute a loss that minimizes the target-ligand distances (corresponding to positive DTIs) and maximizes the target-decoy distances (corresponding to negative DTIs).

We also compared Komet to MolTrans[18] , another recent and state-of-the-art deep learning framework. MolTrans uses a representation of molecules (resp. proteins) based on frequent subsequences of the SMILES (resp. amino acid) strings, combined through a transformer module.

5.4.1 DTI prediction performances on medium-sized datasets

We first compare the performance of Komet to those of ConPLex and MolTrans on the medium-sized datasets BIOSNAP, BindingDB and DrugBank introduced in Section 4.1. We only use the AUPR score because most negative interactions in the considered datasets are unknown interactions. The results are presented in Table 4. Note that the performance of a random predictor would correspond to an AUPR score of 0.5 (except for BindingDB in which the number of negative DTIs is much larger than the number of positive DTIs, and for which the performance of a random predictor would be equal to 0.4). We report the average and standard deviation of the area under the precision-recall curve (AUPR) for 5 random

initializations of each model. Interestingly, in all cases, Komet's AUPR performances (with $d_M = 1\,000$ and $m_M = 3\,000$) are similar to or higher than those of the two deep learning methods. This is consistent with the expectation that deep learning methods only outperform shallow learning methods when training data are abundant, due to their larger number of parameters to fit.

Table 4: AUPR performances of Komet, ConPLex, and MolTrans on medium-sized datasets BIOSNAP, BindingDB, and DrugBank. The ConPLex and MolTrans algorithms were re-run on these three datasets, and the resulting AUPR are very close (in fact slightly better) to those in the original paper.

Dataset	Komet	ConPLex	MolTrans
BIOSNAP	0.940 ± 0.001	0.921 ± 0.002	0.893 ± 0.001
Unseen_drugs	0.914 ± 0.001	0.899 ± 0.011	0.871± 0.002
Unseen_targets	0.891 ± 0.001	0.863 ± 0.005	0.683 ± 0.005
BindingDB	0.667 ± 0.005	0.669 ± 0.003	0.611 ± 0.004
DrugBank	0.939 ± 0.001	0.935 ± 0.002	0.809 ± 0.004

In the Unseen_drugs and Unseen_targets scenarios on BIOSNAP, as expected, the AUPR performances decrease for all algorithms but remain high, except for MolTrans which overall tends to display lower performances than the two other algorithms.

5.4.2 DTI prediction performances on large-sized datasets

Then, we trained Komet, ConPlex, and MolTrans on the four large-sized LCIdb-derived datasets. The results demonstrate that Komet achieves state-of-the-art prediction performance in all cases (see Table 5) at a much lower cost in terms of training time (see Table 6).

Table 5: Comparison of AUPR scores on large-sized datasets

	Komet	ConPLex	MolTrans
LCIdb	0.990 ± 0.001	0.969 ± 0.002	0.967 ± 0.001
Unseen_drugs	0.994 ± 0.0003	0.978 ± 0.003	0.968 ± 0.002
Unseen_targets	0.915 ± 0.001	0.894 ± 0.031	0.591 ± 0.007
Orphan	0.896 ± 0.0008	0.846 ± 0.003	0.552± 0.013

Overall, the performance of Komet is consistently high, with AUPR scores above 0.9 in most cases. Because the number of molecules is still very large in the LCIdb Unseen_drugs

Table 6: Comparison of training time for the considered algorithms

	Komet	ConPLex	MolTrans
LCIdb	15s	907.3s	69838s
Unseen_drugs	15s	1734s	68400s
Unseen_proteins	15s	888s	64800s
Orphan	8s	1329s	25200s

dataset, thus covering a broad chemical space, the performance remains excellent, although molecules in the Test set are absent in the Train set. In LCIdb Unseen_targets and LCIdb_Orphan, where the proteins in the Test set are absent in the Train set, the performances are slightly lower but remain high. The ConPLex algorithm also displays high performances (although lower than those of Komet) in all cases, while MolTrans appears to be less stable.

We conducted a comparison using various performance measures, and the outcomes consistently align with the above results. For these additional insights, please refer to the Appendix B.

5.4.3 Validation on DrugBank (Ext) as external dataset

In the above sections, the performances of the algorithms are compared based on Train/Val /Test splits on all the considered datasets. To better assess the generalization properties of the algorithms, we used as an external dataset the DrugBank (Ext) introduced in Section 4.1.

The prediction performance of the three considered algorithms on DrugBank (Ext), when trained on BindingDB or on LCIdb, are reported in Table 7, from which two conclusions can be drawn. First, all ML algorithms perform better when trained on LCIdb compared to BindingDB. This improvement is attributed to LCIdb's more large coverage of both chemical and protein spaces. Indeed, according to Figure 2, the molecule space covered by LCIdb globally includes that covered by DrugBank, but this does not appear to be the case for the BindingDB dataset. Similarly, according to Figure 4, the protein space of LCIdb globally covers that of DrugBank, whereas the protein space of BindingDB does not seem to cover

that of DrugBank.

Second, Komet always outperforms the two deep learning algorithms. Overall, Komet trained on LCIdb displays the best generalization performances on DrugBank (Ext).

Table 7: AUPR performance for considered algorithms trained on BibndingDB and LCIdb

Training set \ Algorithm	Komet	ConPLex	MolTrans
LCIdb	0.848	0.822	0.558
BindingDB	0.659	0.611	0.503

5.5 Case Study: solving scaffold hopping problems

Finally, we evaluate the ability of the pipeline that leads to the best performance, i.e. Komet trained on the LCIdb dataset, to solve scaffold hopping problems. This requires highly demanding generalization properties and corresponds to an important challenge in drug discovery[1] . Indeed, various problems can restrain the downstream development of a new candidate drug such as inadequate ADME profile, poor selectivity potentially resulting in unacceptable toxicity, or an expensive synthesis route. The hit molecular scaffold may also be protected by patents, which poses problems for its industrial exploitation. To circumvent these limitations, other active molecules with different molecular scaffolds are searched. The difficulty of the problem posed by this search depends on the degree of "dissimilarity" that is required for the new active molecule concerning the known hit. Although various examples of successful scaffold hopping cases have been reported, these types of problems remain difficult and new concepts are required to help *in silico* approaches efficiently solve these difficult cases[72] .

Pinel et al.[42] proposed the $\mathcal{LH}$ benchmark to assess the performance of computational methods to solve scaffold hopping problems. They focused on the most difficult case, i.e. the "large-step" scaffold hopping scenario, where one ligand molecule for a given target is known, and another ligand molecule of a highly dissimilar structure is searched for the same target. The $\mathcal{LH}$ benchmark comprises 144 pairs of highly dissimilar molecules that are active

against diverse protein targets. Computational methods are evaluated as follows: for each pair, one active molecule is considered as known, and the second active has to be retrieved among decoys that were carefully selected to avoid statistical bias. Since either molecule of the pair can be chosen as the known active, this leads to 288 scaffold hopping cases to solve. More precisely, given one molecule of the pair, the objective is to rank the other (considered as unknown active) among a pool of decoy molecules. The lower the rank of the unknown active, the better the prediction performance.

In Figure 7, we compare the performance of Komet and ConPLex prediction algorithms trained on LCIdbor BindingDB, using Cumulative Histogram Curves (CHC). This criterion illustrates the frequency of cases where the method ranked the unknown active molecule below a specific rank. Table 8 supplements this evaluation by providing the Area Under the Curve (AUC) of CHC curves, offering a quantitative comparison of methods, along with the proportion of cases where the unknown active was retrieved within the top 1% and 5% of best-ranked molecules. These metrics serve as indicators of the success rate of the methods. We also re-computed the results obtained by the Kronecker kernel with an SVM calculated with the scikit-learn toolbox, using the same kernels as in Komet, and trained on the DrugBank dataset. These results align with those of the original paper by Pinel et al. [42].

As shown in Figure 7 and Table 8, Komet trained on LCIdb leads to the best performances on all criteria. The ConPLex deep learning algorithm trained on LCIdb (and fine-tuned with DUD-E) performs better on all criteria than when trained on BindingDB (and fine-tuned with DUDE-E), while the Kernel SVM trained on DrugBank of the original paper displays performances that are intermediates with those of ConPlex on the two considered training datasets. The fact that ConPLex does not outperform Komet specifically on the $\mathcal{LH}$ benchmark is somewhat puzzling. Indeed, one of the reasons why we chose ConPLex is that it incorporates a contrastive learning step based on DUD-E, which should help separate the unknown positive from the decoys in $\mathcal{LH}$. One explanation may reside in the fact that DUD-E presents a hidden bias that was shown to mislead the performance of deep learning

algorithms[73] . The use of an unbiased database for contrastive learning may improve the performance of ConPLex on the $\mathcal{LH}$ benchmark.

Table 8: Prediction performances on the $\mathcal{LH}$ benchmark.

Dataset	Komet on LCIdb	Kernel SVM on DrugBank	ConPLex on BindingDB and contrastive on DUD-E	ConPLex on LCIdb and contrastive on DUD-E
ROC-AUC	0.85	0.77	0.70	0.75
Top 1%	32%	22%	12%	24%
Top 5 %	52%	36%	26%	43%

Notably, in 50% of cases, our pipeline involving Komet trained on LCIdb successfully ranks the unknown active in the top 5%. This performance surpasses those of all ligand-based methods tested in the original paper by Pinel et al.[42], the best of which, involving 3D pharmacophore descriptors, ranked the unknown active in the top 5% in 20% of cases.

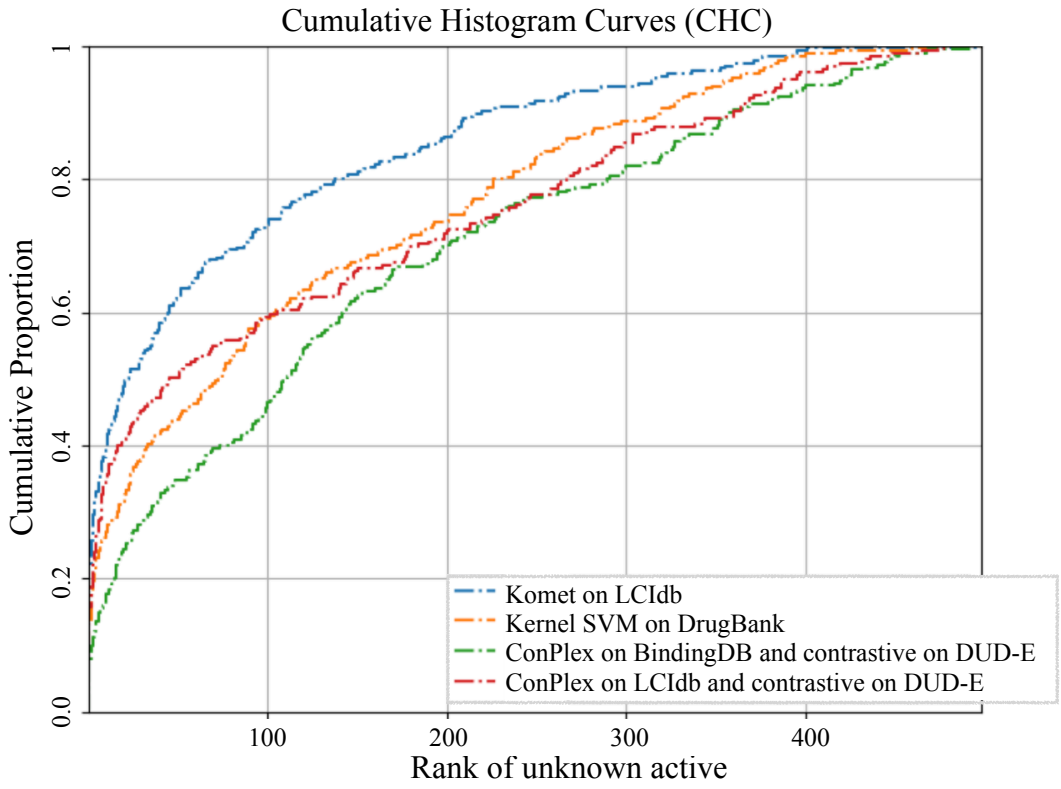


Figure 7: Cumulative Histogram Curves of the considered algorithm, measuring the cumulative proportion of cases the unknown active is retrieved below a given rank.

The fact that Komet trained on LCIdb outperforms ConPLex trained on the same dataset

may again be explained by more expressive features for the (molecule, protein) pairs in Komet. In addition, the facts that (1) the performances of ConPLex are improved when trained on LCIdb over those obtained with BindingDB, and that (2) the performances of Komet trained on LCIdb over than those obtained with Kernel SVM trained on DrugBank, may be explained by a better coverage of the active molecules space in $\mathcal{LH}$ by LCIdb than by BindingDB and DrugBank. Indeed, we used the t-SNE algorithm to visualize the molecule space coverage of the LCIdb , DrugBank, BindingDB and superposed with the space of active molecules in $\mathcal{LH}$. As shown in Figure 8, LCIdb uniformly spans the entire space of active molecules in $\mathcal{LH}$, which is not the case for the DrugBank and the BindingDB datasets.

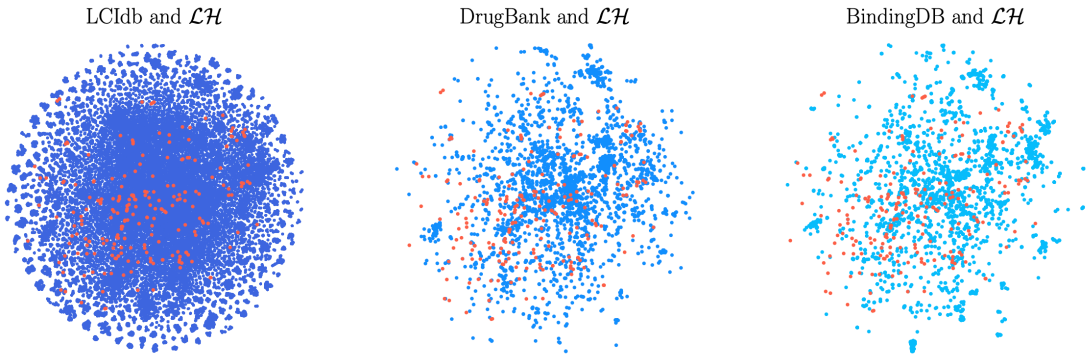


Figure 8: t-SNE on molecule features. In blue and from left to right: LCIdb, DrugBank and BindingDB, in orange: active molecules of $\mathcal{LH}$.

6 Discussion

An important contribution of the present work resides in providing the LCIdb DTI dataset, which appears much larger than most public datasets used in the recent literature. A key feature of this dataset is a wider and more uniform coverage of the molecular space. A recurrent problem when building DTI datasets for training ML algorithms is that negative interactions are usually not reported. One way to circumvent this problem is to use reference databases that provide quantitative bioactivity measurements and choose threshold values to define positive and negative interactions. In previous studies[18,44] , other authors chose

a common and rather low threshold value of 30 nM for both types of DTIs, leading to a modest number of positive (9 166) and three times more negative DTIs (23 435), as shown in Table 1. The notion of positive and negative DTIs is not absolute, because bioactivities are continuous, and threshold values are somewhat arbitrary. In the present paper, we chose distinct thresholds for positive and negative interactions, respectively under 100 nM (10^{-7}M) and above 100μM (10^{-4}M). This leads to a limited number of known negative DTIs in the dataset (8 296) compared to known positives (402 538). Overall, our goal was to limit the potential false negative DTIs and the bias towards well-studied molecules and proteins. Therefore, true negative DTIs were completed by randomly chosen DTIs according to the algorithm in Najm et al. [45], while excluding all DTIs with activities falling in the $10^{-4}-10^{-7}$M range. However, we are aware that using a lower threshold value for the negative DTIs in LCIdb would have allowed us to select a high number of DTIs considered as known negatives.

Another important contribution is the proposal of the Komet pipeline, a DTI prediction algorithm designed to learn on very large training datasets such as LCIdb. This algorithm has two parameters, m_M (number of landmark molecules) and d_M (dimension of molecular features). We were able to define good default values for these parameters ($d_M = 1\,000$ and $m_M = 3\,000$), significantly reducing the computational time and memory requirements. Interestingly, computational resources will not increase drastically if the size of the Train set increases (if new DTIs are added), as can be judged from Figure 6.

We also showed that the performance of the algorithm was robust for the choice of the landmark molecules and the molecule and protein features, although learned features tended to decrease the performance, as shown in Table 3.

Importantly, Komet belongs to the family of shallow ML algorithms and proved to outperform ConPLex and MolTrans, two recently proposed deep learning algorithms, at a much lower computational cost. One explanation for the good performance of Komet could be that features for the (molecule, protein) pairs derived by Komet in Step 2, simply based on the Kronecker product, may better capture determinants of the interaction than the com-

bined learned features in the considered deep learning algorithms. The Kronecker product strategy to combine molecule/protein features to encode interactions seems more important than the choice of features for (molecule, protein) pairs since different molecule features did not significantly impact performance (see Table 3), and since ConPLex does not reach the performance of Komet when both are trained on LCIdb (see Table 7). In addition, the architectures of ConPLex and MolTrans may not yet be fully optimized for the DTI prediction problem. Furthermore, our study focuses on DTI prediction in the human druggable space of proteins, because our goal is to propose a tool for drug discovery projects. The dimension of this space is modest, as illustrated by the number of proteins in LCIdb (2 069), concerning that of the human proteome (above 20 000, but expected to be in the order of 90 000 when including splicing variants). Therefore, the druggable human proteins may present some sequence bias, and the protein features used in ConPLex and MolTrans and learned based on a much wider space of proteins may not be optimal for the DTI prediction problem at hand. This is consistent with the results in Table 3, showing that learned features did not improve the performances of Komet.

Komet proved to display state-of-the-art performances on various prediction scenarios, including the most difficult problems. In particular, it proved to be efficient in solving scaffold hopping cases. Although it was not designed and tuned for this specific scenario, it appears as an effective tool to guide medicinal chemists in solving such problems. One possible future improvement would be to use other molecule kernels. Indeed, the Tanimoto molecule kernel used in Komet is a measure of structure similarity between molecules, which is a priori not well suited to the scaffold hopping problem. Other molecule kernels based on pharmacophore features may improve the prediction performances of Komet on the specific problem of scaffold hopping.

Acknowledgement

This work has been supported by the Paris Île-de-France Region in the framework of the "DIM AI4IDF".

References

(1) Schneider, G.; Neidhart, W.; Giller, T.; Schmid, G. "Scaffold-hopping" by topological pharmacophore search: a contribution to virtual screening. *Angewandte Chemie International Edition* **1999**, *38*, 2894–2896.

(2) Lim, S.; Lu, Y.; Cho, C. Y.; Sung, I.; Kim, J.; Kim, Y.; Park, S.; Kim, S. A review on compound-protein interaction prediction methods: data, format, representation and model. *Computational and Structural Biotechnology Journal* **2021**, *19*, 1541–1556.

(3) Wigh, D. S.; Goodman, J. M.; Lapkin, A. A. A review of molecular representation in the age of machine learning. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **2022**, *12*, e1603.

(4) Kim, J.; Park, S.; Min, D.; Kim, W. Comprehensive survey of recent drug discovery using deep learning. *International Journal of Molecular Sciences* **2021**, *22*, 9983.

(5) Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences* **1988**, *28*, 31–36.

(6) Heller, S. R.; McNaught, A.; Pletnev, I.; Stein, S.; Tchekhovskoi, D. InChI, the IUPAC international chemical identifier. *Journal of Cheminformatics* **2015**, *7*, 1–34.

(7) Dalby, A.; Nourse, J. G.; Hounshell, W. D.; Gushurst, A. K.; Grier, D. L.; Leland, B. A.; Laufer, J. Description of several chemical structure file formats used by computer pro-

grams developed at Molecular Design Limited. *Journal of Chemical Information and Computer Science* **1992**, *32*, 244–255.

(8) Rogers, D.; Hahn, M. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling* **2010**, *50*, 742–754.

(9) Landrum, G. et al. rdkit/rdkit: Release_2023.09.5. 2024; `https://doi.org/10.5281/zenodo.10633624`.

(10) Lee, I.; Keum, J.; Nam, H. DeepConv-DTI: Prediction of drug-target interactions via deep learning with convolution on protein sequences. *PLOS Computational Biology* **2019**, *15*, e1007129.

(11) Zhao, Q.; Zhao, H.; Zheng, K.; Wang, J. HyperAttentionDTI: improving drug–protein interaction prediction by sequence-based deep learning with attention mechanism. *Bioinformatics* **2022**, *38*, 655–662.

(12) Duvenaud, D. K.; Maclaurin, D.; Iparraguirre, J.; Bombarell, R.; Hirzel, T.; Aspuru-Guzik, A.; Adams, R. P. Convolutional networks on graphs for learning molecular fingerprints. Advances in Neural Information Processing Systems. 2015.

(13) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molecular graph convolutions: moving beyond fingerprints. *Journal of Computer-Aided Molecular Design* **2016**, *30*, 595–608.

(14) Hu, W.; Liu, B.; Gomes, J.; Zitnik, M.; Liang, P.; Pande, V.; Leskovec, J. Strategies for pre-training graph neural networks. 8th International Conference on Learning Representations, ICLR 2020. 2020.

(15) Jaeger, S.; Fulle, S.; Turk, S. Mol2vec: unsupervised machine learning approach with chemical intuition. *Journal of Chemical Information and Modeling* **2018**, *58*, 27–35.

(16) Goh, G. B.; Hodas, N.; Siegel, C.; Vishnu, A. Smiles2vec: Predicting chemical properties from text representations. **2018**,

(17) Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; Dean, J. Distributed representations of words and phrases and their compositionality. Advances in Neural Information Processing Systems. 2013.

(18) Huang, K.; Xiao, C.; Glass, L. M.; Sun, J. MolTrans: molecular interaction transformer for drug–target interaction prediction. *Bioinformatics* **2021**, *37*, 830–836.

(19) Xue, D.; Zhang, H.; Chen, X.; Xiao, D.; Gong, Y.; Chuai, G.; Sun, Y.; Tian, H.; Wu, H.; Li, Y.; Liu, Q. X-MOL: large-scale pre-training for molecular understanding and diverse molecular analysis. *Science Bulletin* **2022**, *67*, 899–902.

(20) Li, P.; Wang, J.; Qiao, Y.; Chen, H.; Yu, Y.; Yao, X.; Gao, P.; Xie, G.; Song, S. An effective self-supervised framework for learning expressive molecular global representations to drug discovery. *Briefings in Bioinformatics* **2021**, *22*, bbab109.

(21) Zhu, L.; Davari, M. D.; Li, W. Recent advances in the prediction of protein structural classes: Feature descriptors and machine learning algorithms. *Crystals* **2021**, *11*, 324.

(22) Dubchak, I.; Muchnik, I.; Holbrook, S. R.; Kim, S.-H. Prediction of protein folding class using global description of amino acid sequence. *Proceedings of the National Academy of Sciences* **1995**, *92*, 8700–8704.

(23) Rives, A.; Meier, J.; Sercu, T.; Goyal, S.; Lin, Z.; Liu, J.; Guo, D.; Ott, M.; Zitnick, C. L.; Ma, J.; others Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences* **2021**, *118*, e2016239118.

(24) Elnaggar, A.; Heinzinger, M.; Dallago, C.; Rehawi, G.; Wang, Y.; Jones, L.; Gibbs, T.; Feher, T.; Angerer, C.; Steinegger, M.; others Prottrans: Toward understanding the

language of life through self-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2021**, *44*, 7112–7127.

(25) Nguyen, N.-Q.; Jang, G.; Kim, H.; Kang, J. Perceiver CPI: a nested cross-attention network for compound–protein interaction prediction. *Bioinformatics* **2023**, *39*, btac731.

(26) Jacob, L.; Hoffmann, B.; Stoven, V.; Vert, J.-P. Virtual screening of GPCRs: an in silico chemogenomics approach. *BMC Bioinformatics* **2008**, *9*, 1–16.

(27) Pahikkala, T.; Airola, A.; Pietilä, S.; Shakyawar, S.; Szwajda, A.; Tang, J.; Aittokallio, T. Toward more realistic drug–target interaction predictions. *Briefings in Bioinformatics* **2015**, *16*, 325–337.

(28) Huang, K.; Fu, T.; Glass, L. M.; Zitnik, M.; Xiao, C.; Sun, J. DeepPurpose: a deep learning library for drug–target interaction prediction. *Bioinformatics* **2020**, *36*, 5545–5547.

(29) Öztürk, H.; Özgür, A.; Ozkirimli, E. DeepDTA: deep drug–target binding affinity prediction. *Bioinformatics* **2018**, *34*, i821–i829.

(30) Sledzieski, S.; Singh, R.; Cowen, L.; Berger, B. Adapting protein language models for rapid DTI prediction. *bioRxiv* **2022**, 2022–11.

(31) Tsubaki, M.; Tomii, K.; Sese, J. Compound–protein interaction prediction with end-to-end learning of neural networks for graphs and sequences. *Bioinformatics* **2019**, *35*, 309–318.

(32) Shi, H.; Liu, S.; Chen, J.; Li, X.; Ma, Q.; Yu, B. Predicting drug-target interactions using Lasso with random forest based on evolutionary information and chemical structure. *Genomics* **2019**, *111*, 1839–1852.

(33) Cheng, F.; Liu, C.; Jiang, J.; Lu, W.; Li, W.; Liu, G.; Zhou, W.; Huang, J.; Tang, Y.

Prediction of drug-target interactions and drug repositioning via network-based inference. *PLOS Computational Biology* **2012**, *8*, e1002503.

(34) Rosasco, L.; De Vito, E.; Caponnetto, A.; Piana, M.; Verri, A. Are loss functions all the same? *Neural Computation* **2004**, *16*, 1063–1076.

(35) Nagamine, N.; Sakakibara, Y. Statistical prediction of protein–chemical interactions based on chemical structure and mass spectrometry data. *Bioinformatics* **2007**, *23*, 2004–2012.

(36) Jacob, L.; Vert, J.-P. Protein-ligand interaction prediction: an improved chemogenomics approach. *Bioinformatics* **2008**, *24*, 2149–2156.

(37) Playe, B.; Azencott, C.-A.; Stoven, V. Efficient multi-task chemogenomics for drug specificity prediction. *PLOS ONE* **2018**, *13*, e0204999.

(38) Sieg, J.; Flachsenberg, F.; Rarey, M. In need of bias control: evaluating chemical data for machine learning in structure-based virtual screening. *Journal of Chemical Information and Modeling* **2019**, *59*, 947–961.

(39) Chen, L.; Tan, X.; Wang, D.; Zhong, F.; Liu, X.; Yang, T.; Luo, X.; Chen, K.; Jiang, H.; Zheng, M. TransformerCPI: improving compound–protein interaction prediction by sequence-based deep learning with self-attention mechanism and label reversal experiments. *Bioinformatics* **2020**, *36*, 4406–4414.

(40) Bagherian, M.; Sabeti, E.; Wang, K.; Sartor, M. A.; Nikolovska-Coleska, Z.; Najarian, K. Machine learning approaches and databases for prediction of drug–target interaction: a survey paper. *Briefings in Bioinformatics* **2021**, *22*, 247–269.

(41) Wang, Z.; Liang, L.; Yin, Z.; Lin, J. Improving chemical similarity ensemble approach in target prediction. *Journal of Cheminformatics* **2016**, *8*, 1–10.

(42) Pinel, P.; Guichaoua, G.; Najm, M.; Labouille, S.; Drizard, N.; Gaston-Mathé, Y.; Hoffmann, B.; Stoven, V. Exploring isofunctional molecules: Design of a benchmark and evaluation of prediction performance. *Molecular Informatics* **2023**, *42*, 2200216.

(43) Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; others Pytorch: An imperative style, high-performance deep learning library. Advances in Neural Information Processing Systems. 2019.

(44) Singh, R.; Sledzieski, S.; Bryson, B.; Cowen, L.; Berger, B. Contrastive learning in protein language space predicts interactions between drugs and protein targets. *Proceedings of the National Academy of Sciences* **2023**, *120*, e2220778120.

(45) Najm, M.; Azencott, C.-A.; Playe, B.; Stoven, V. Drug Target Identification with Machine Learning: How to Choose Negative Examples. *International Journal of Molecular Sciences* **2021**, *22*, 5118.

(46) Zitnik, M.; Sosič, R.; Maheshwari, S.; Leskovec, J. BioSNAP Datasets: Stanford Biomedical Network Dataset Collection. `http://snap.stanford.edu/biodata`, 2018.

(47) Liu, T.; Lin, Y.; Wen, X.; Jorissen, R. N.; Gilson, M. K. BindingDB: a web-accessible database of experimentally determined protein–ligand binding affinities. *Nucleic Acids Research* **2007**, *35*, D198–D201.

(48) Isigkeit, L.; Chaikuad, A.; Merk, D. A consensus compound/bioactivity dataset for data-driven drug design and chemogenomics. *Molecules* **2022**, *27*, 2513.

(49) Mendez, D.; Gaulton, A.; Bento, A. P.; Chambers, J.; De Veij, M.; Félix, E.; Magariños, M. P.; Mosquera, J. F.; Mutowo, P.; Nowotka, M.; others ChEMBL: towards direct deposition of bioassay data. *Nucleic Acids Research* **2019**, *47*, D930–D940.

(50) Kim, S.; Chen, J.; Cheng, T.; Gindulyte, A.; He, J.; He, S.; Li, Q.; Shoemaker, B. A.; Thiessen, P. A.; Yu, B.; others PubChem in 2021: new data content and improved web interfaces. *Nucleic Acids Research* **2021**, *49*, D1388–D1395.

(51) Harding, S. D.; Armstrong, J. F.; Faccenda, E.; Southan, C.; Alexander, S. P.; Davenport, A. P.; Pawson, A. J.; Spedding, M.; Davies, J. A.; NC-IUPHAR The IUPHAR/BPS guide to PHARMACOLOGY in 2022: curating pharmacology for COVID-19, malaria and antibacterials. *Nucleic Acids Research* **2022**, *50*, D1282–D1294.

(52) Gilson, M. K.; Liu, T.; Baitaluk, M.; Nicola, G.; Hwang, L.; Chong, J. BindingDB in 2015: a public database for medicinal chemistry, computational chemistry and systems pharmacology. *Nucleic Acids Research* **2016**, *44*, D1045–D1053.

(53) Škuta, C.; Southan, C.; Bartůněk, P. Will the chemical probes please stand up? *RSC Medicinal Chemistry* **2021**, *12*, 1428–1441.

(54) Scholkopf, B.; Mika, S.; Burges, C. J.; Knirsch, P.; Muller, K.-R.; Ratsch, G.; Smola, A. J. Input space versus feature space in kernel-based methods. *IEEE Transactions on Neural Networks* **1999**, *10*, 1000–1017.

(55) Williams, C.; Seeger, M. Using the Nyström method to speed up kernel machines. Advances in Neural Information Processing Systems. 2000.

(56) Saigo, H.; Vert, J.-P.; Ueda, N.; Akutsu, T. Protein homology detection using string alignment kernels. *Bioinformatics* **2004**, *20*, 1682–1689.

(57) Smith, T. F.; Waterman, M. S. Identification of common molecular subsequences. *Journal of Molecular Biology* **1981**, *147*, 195–197.

(58) Bottou, L. Large-scale machine learning with stochastic gradient descent. Proceedings of COMPSTAT'2010: 19th International Conference on Computational StatisticsParis

France, August 22-27, 2010 Keynote, Invited and Contributed Papers. 2010; pp 177–186.

(59) Airola, A.; Pahikkala, T. Fast Kronecker product kernel methods via generalized vec trick. *IEEE Transactions on Neural Networks and Learning Systems* **2017**, *29*, 3374–3387.

(60) Nocedal, J.; Wright, S. J. *Numerical optimization*; Springer, 1999; Chapter 6.

(61) Platt, J.; others *Advances in Large Margin Classifiers*; Cambridge, MA, 1999; Vol. 10; pp 61–74.

(62) Grisoni, F.; Merk, D.; Byrne, R.; Schneider, G. Scaffold-hopping from synthetic drugs by holistic molecular representation. *Scientific Reports* **2018**, *8*, 16469.

(63) Ruddigkeit, L.; Van Deursen, R.; Blum, L. C.; Reymond, J.-L. Enumeration of 166 billion organic small molecules in the chemical universe database GDB-17. *Journal of Chemical Information and Modeling* **2012**, *52*, 2864–2875.

(64) Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *Journal of Machine Learning Research* **2008**, *9*.

(65) Kim, Q.; Ko, J.-H.; Kim, S.; Park, N.; Jhe, W. Bayesian neural network with pre-trained protein embedding enhances prediction accuracy of drug-protein interaction. *Bioinformatics* **2021**, *37*, 3428–3435.

(66) Gao, K.; Nguyen, D. D.; Sresht, V.; Mathiowetz, A. M.; Tu, M.; Wei, G.-W. Are 2D fingerprints still valuable for drug discovery? *Physical Chemistry Chemical Physics* **2020**, *22*, 8373–8390.

(67) Sabando, M. V.; Ponzoni, I.; Milios, E. E.; Soto, A. J. Using molecular embeddings in QSAR modeling: does it make a difference? *Briefings in Bioinformatics* **2022**, *23*, bbab365.

(68) Li, M.; Zhou, J.; Hu, J.; Fan, W.; Zhang, Y.; Gu, Y.; Karypis, G. DGL-LifeSci: An Open-Source Toolkit for Deep Learning on Graphs in Life Science. *ACS Omega* **2021**,

(69) Boutet, E.; Lieberherr, D.; Tognolli, M.; Schneider, M.; Bansal, P.; Bridge, A. J.; Poux, S.; Bougueleret, L.; Xenarios, I. UniProtKB/Swiss-Prot, the manually annotated section of the UniProt KnowledgeBase: how to use the entry view. *Plant Bioinformatics: Methods and Protocols* **2016**, 23–54.

(70) Playe, B.; Stoven, V. Evaluation of deep and shallow learning methods in chemogenomics for the prediction of drugs specificity. *Journal of Cheminformatics* **2020**, *12*, 11.

(71) Mysinger, M. M.; Carchia, M.; Irwin, J. J.; Shoichet, B. K. Directory of useful decoys, enhanced (DUD-E): better ligands and decoys for better benchmarking. *Journal of Medicinal Chemistry* **2012**, *55*, 6582–6594.

(72) Hu, Y.; Stumpfe, D.; Bajorath, J. Recent advances in scaffold hopping: miniperspective. *Journal of Medicinal Chemistry* **2017**, *60*, 1238–1246.

(73) Chen, L.; Cruz, A.; Ramsey, S.; Dickson, C. J.; Duca, J. S.; Hornak, V.; Koes, D. R.; Kurtzman, T. Hidden bias in the DUD-E dataset leads to misleading performance of deep learning in structure-based virtual screening. *PLOS ONE* **2019**, *14*, e0220113.

A Molecule space coverage of various datasets

This section shows cases of a 2D visualization of the chemical space covered by various datasets considered in the paper, using the t-SNE algorithm on various molecule features.

Figure 9 shows the drug distribution in LCIdb across the five databases from which the initial dataset[48] is extracted. It highlights a significant contribution from the ChEMBL and PubChem databases, enhanced mainly by data from Probes&Drugs.

Figure 10 shows the t-SNE visualizations of the molecular space for various considered datasets, based on Tanimoto features (as in Figure 2) for one choice of 3000 landmark molecules, for another choice of 3000 landmark molecules, and ECFP4 features. It confirms that LCIdb offers broader and more uniform coverage of the chemical space than BindingDB, DrugBank, or BIOSNAP.

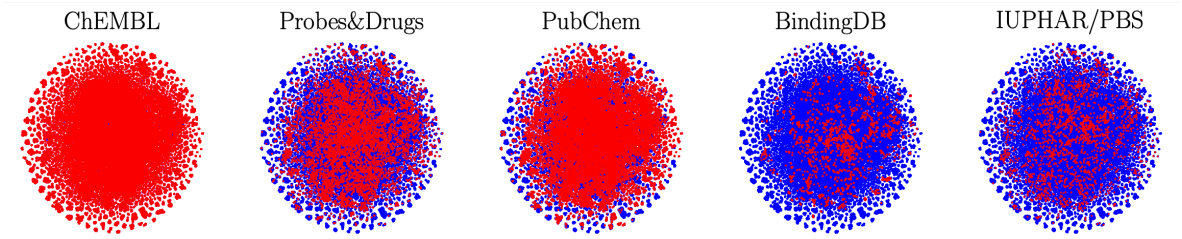


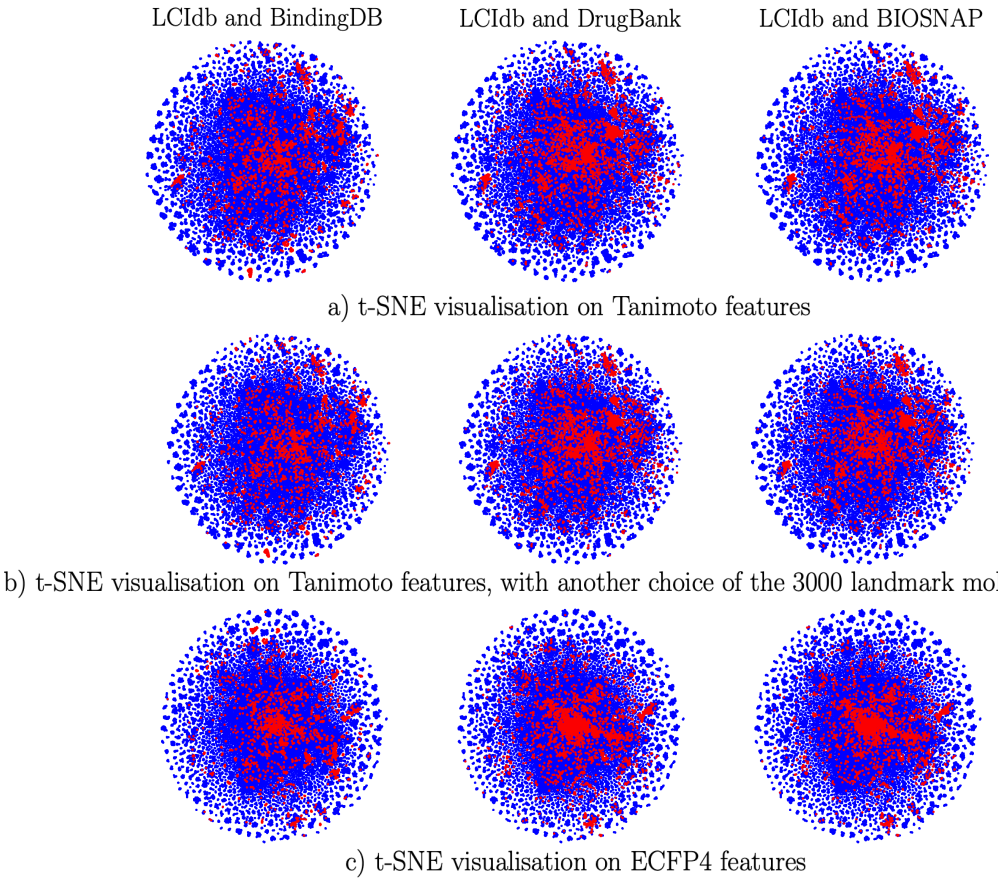
Figure 9: t-SNE on molecule features. In blue: large-sized benchmark LCIdb, in red: 5 databases from which the initial dataset[48] is extracted.

B Several metrics to compare prediction performances

Table 9 presents various metrics for comparing prediction performances on the four LCIdb-datasets. While ConPlex has better accuracy in two cases, overall, Komet outperforms the other algorithms in most cases according to AUPR, ROC-AUC and Accuracy prediction performances, supporting the main conclusions in the paper.

Table 9: AUPR, ROC-AUC and Accuracy prediction performances

	Komet			ConPLex			MolTrans	
	AUPR	ROC-AUC	Accuracy	AUPR	ROC-AUC	Accuracy	AUPR	ROC-AUC
LCIdb	0.990	0.990	0.966	0.970	0.971	0.917	0.967	0.970
Unseen_drugs	0.994	0.994	0.976	0.980	0.977	0.934	0.968	0.969
Unseen_targets	0.915	0.896	0.714	0.893	0.874	0.763	0.591	0.584
Orphan	0.896	0.879	0.682	0.845	0.834	0.689	0.552	0.536



a) t-SNE visualisation on Tanimoto features

b) t-SNE visualisation on Tanimoto features, with another choice of the 3000 landmark molecules

c) t-SNE visualisation on ECFP4 features

Figure 10: 2D representation of the molecular space, based on the t-SNE algorithm on molecule features. In blue: large-sized LCIdb dataset, and in red: medium-scale DrugBank, BIOSNAP, and BindingDB datasets.

C Nyström approximation

In Komet, we encode molecules leveraging the Nyström approximation [54,55] . In the following, we present the mathematical details of Section 4.3.

Let us consider a set of landmark molecules $\{\hat{\mathfrak{m}}_1, \dots \hat{\mathfrak{m}}_{m_M}\}$, a new molecule $\mathfrak{m}$, and a kernel k_M over molecules. The kernel matrix $K \in \mathbb{R}^{(m_M+1)\times(m_M+1)}$ over these m_{M+1} molecules can be written as $K = \begin{bmatrix} \hat{K}_M & \kappa^\top \\ \kappa & k_M(\mathfrak{m},\mathfrak{m}) \end{bmatrix}$ with $\hat{K}_M \in \mathbb{R}^{m_M \times m_M}$ being the kernel matrix over the landmark molecules and $\kappa = (k_M(\mathfrak{m},\hat{\mathfrak{m}}_1), \dots, k_M(\mathfrak{m},\hat{\mathfrak{m}}_{m_M})) \in \mathbb{R}^{m_M}$ the vector of kernel values between $\mathfrak{m}$ and the landmark molecules.

The Nyström's approximation consists in approximating K as $K \approx C\hat{K}_M^{-1}C^\top = \begin{bmatrix} \hat{K}_M & \kappa^\top \\ \kappa & \kappa\hat{K}_M^{-1}\kappa^\top \end{bmatrix}$ with $C = \begin{bmatrix} \hat{K}_M \\ \kappa \end{bmatrix} \in \mathbb{R}^{(m_M+1)\times m_M}$.

Writing the Single Value Decomposition of $\hat{K}_M$ as $\hat{K}_M = U\,\mathrm{diag}(\sigma)U^\top$, the approximation of K can be rewritten as $K \approx \Phi\Phi^\top$ with $\Phi = CU\,\mathrm{diag}(\sigma)^{-1/2} \approx CE$. When no dimensionality reduction is performed ($d_M = m_M$), $E = U\,\mathrm{diag}(\sigma)^{-1/2}$ and $\Phi = CE$.

The last line of matrix Φ is $\Phi_{m_M+1} = (\sum_{l=1}^{m_M} C_{m_M+1,l}E_{ls})_{s=1}^{m_M} = \psi_M(\mathfrak{m})$. Similarly, its m_M first lines are $\psi_M(\hat{\mathfrak{m}}_1)$, …, $\psi_M(\hat{\mathfrak{m}}_{m_M})$. Hence $k_M(\mathfrak{m},\hat{\mathfrak{m}}_i) \approx \langle \psi_M(\mathfrak{m}), \psi_M(\hat{\mathfrak{m}}_i)\rangle$ for any molecule $\mathfrak{m}$ (including one of the landmark molecules), which justifies our proposition of ψ_M.

Furthermore, if we do not use dimensionality reduction, because the Nyström approximation is an equality on the upper-left block $\hat{K}_M$, $k_M(\hat{\mathfrak{m}}_i,\hat{\mathfrak{m}}_j) = \langle \psi_M(\hat{\mathfrak{m}}_i), \psi_M(\hat{\mathfrak{m}}_j)\rangle$ for any pair of landmark molecules.

D Efficient computation

We explicit here the details for equality (a) of Eq (2) in paragraph 4.4.

$$(Zw)_k = \langle w, z_k\rangle_{\mathbb{R}^{d_Z}} \overset{(a)}{=} \langle m_{i_k}, Wp_{j_k}\rangle_{\mathbb{R}^{d_M}} \overset{(b)}{=} \langle m_{i_k}, q_{j_k}\rangle_{\mathbb{R}^{d_M}}.$$

We use the matrix representation $W \in \mathbb{R}^{d_M \times d_P}$ instead of $w \in \mathbb{R}^{d_Z}$ in a way that w is the flattened representation of W.

$$\begin{aligned}\forall k = 1..n_Z,\ (Zw)_k &= \langle w, z_k \rangle_{\mathbb{R}^{d_Z}} \\ &= \langle W, m_{i_k} p_{j_k}^\top \rangle_{\mathbb{R}^{d_M \times d_P}} \\ &= \mathrm{tr}\left(W (m_{i_k} p_{j_k}^\top)^\top\right) = \mathrm{tr}\left(W p_{j_k} m_{i_k}^\top\right) = \langle W p_{j_k}, m_{i_k} \rangle_{\mathbb{R}^{d_M}}\end{aligned}$$

B

Protein Superfamilies

Table B.1 displays the 69 proteins used to define the IFPP, spanning different superfamilies of protein structures, according to the SCOP database [Murzin *et al.*(1995)].

PDB	UniProt	Protein	Superfamily
2qn1	P00489	glycogen phosphorylase	Type B glycosyltransferase-like
3eq9	P23687	prolyl endopeptidase	Peptidase/esterase 'gauge' domain
4bqh	Q386Q8	udp-n-acetylglucosamine pyrophosphorylase	Nucleotide-diphospho-sugar transferases
3el8	P00523	tyrosine-protein kinase src	SH3-domain
6ibx	Q16875	6-phosphofructo-2-kinase/fructose-2,6-bisphosphatase 3	Histidine phosphatase-like
6i17	Q16658	fascin	Actin-crosslinking proteins
6d55	P01112	gtpase hras	Ras-like P-loop GTPases
4w9q	Q13451	peptidyl-prolyl cis-trans isomerase fkbp5	TPR-like
5dsx	Q8TEK3	histone-lysine n-methyltransferase, h3 lysine-79 specific	S-adenosyl-L-methionine-dependent methyltransferases
3sff	Q9BY41	histone deacetylase 8	Arginase/deacetylase-like
4p5z	P29320	eph receptor a3	galactose-binding domain-like
1t48	P18031	protein-tyrosine phosphatase, non-receptor type 1	(Phosphotyrosine protein) phosphatases II
6duk	P00533	epidermal growth factor receptor	L domain-like
6ccy	P31749	rac-alpha serine/threonine-protein kinase,piftide	PH domain-like
3sfc	P00797	renin	Acid proteases
6mob	P10721	mast/stem cell growth factor receptor kit	Protein kinase-like (PK-like)
5bns	P0A6R0	3-oxoacyl-[acyl-carrier-protein] synthase 3	Thiolase-like
4k9y	Q05397	focal adhesion kinase 1	FAT domain of focal adhesion kinase
6n0k	O00625	pirin	RmlC-like cupins
5u8c	P35439	glutamate receptor ionotropic, nmda 1	Type 2 solute binding protein-like
5k13	P10276	retinoic acid receptor alpha	Nuclear receptor ligand-binding domain
2ovx	P14780	matrix metalloproteinase-9 (mmp-9)	PGBD-like
5syn	O95372	acyl-protein thioesterase 2	alpha/beta-Hydrolases
4k69	P23946	chymase	Trypsin-like serine proteases

5nr7	P9WKE1	thymidylate kinase	P-loop nucleotide/nucleoside kinase-like
6g2m	Q9NPB1	5'(3')-deoxyribonucleotidase	HAD-like
2ica	P20701	integrin alpha-l	vWA-like
4pv5	Q9CPU0	lactoylglutathione lyase	Glyoxalase/Bleomycin resistance protein/Dihydroxybiphenyl dioxygenase
4gvm	P12497	gag-pol polyprotein	N-terminal Zn binding domain of HIV integrase-like
4ojr	P03366	hiv-1 integrase	DNA-binding domain of retroviral integrase
6std	P56221	scytalone dehydratase	NTF2-like
6qz8	Q07820	induced myeloid leukemia cell differentiation protein mcl	Bcl-2-like inhibitors of programmed cell death
3dpe	P22894	neutrophil collagenase	Metalloproteases (zincins), catalytic domain
5mw2	P41182	b-cell lymphoma 6 protein	POZ domain
4ibj	Q05127	polymerase cofactor vp35	Ebola VP35 IID-like
5ufp	Q99814	endothelial pas domain-containing protein 1	PYP-like sensor domain (PAS domain)
6q9w	O15151	protein mdm4	SWIB/MDM2 domain-like
5t4b	P27487	dipeptidyl peptidase 4	DPP6 N-terminal domain-like
4wp7	P00352	retinal dehydrogenase 1	ALDH-like
5ovg	Q07889	son of sevenless homolog 1	ENTH/VHS domain-like
4yz9	O75460	serine/threonine-protein kinase/endoribonuclease ir	Ire1-RNaseL RNase domain-like
6te6	Q8TEK3	histone-lysine n-methyltransferase, h3 lysine-79 specific	S-adenosyl-L-methionine-dependent methyltransferases
5ur1	P11362	fibroblast growth factor receptor 1	Immunoglobulin (Ig) domain-like
6qed	P50579	methionine aminopeptidase 2	Creatinase/aminopeptidase catalytic domain-like
5n9r	Q93009	ubiquitin carboxyl-terminal hydrolase 7	Cysteine proteinases
3vhe	P35968	vascular endothelial growth factor receptor 2	Immunoglobulin (Ig) domain-like
4l7n	Q9Y6F1	poly [adp-ribose] polymerase 3	WGR domain-like
5twl	Q14680	maternal embryonic leucine zipper kinase	UBA-like
3hb4	P14061	estradiol 17-beta-dehydrogenase 1	SDR-like
6nss	P04629	high affinity nerve growth factor receptor	Immunoglobulin (Ig) domain-like

6f3i	Q9NWZ3	interleukin-1 receptor-associated kinase 4	DEATH domain
2qcg	P11172	uridine 5'-monophosphate synthase (ump synthase)	Ribulose-phoshate binding barrel
2vwz	P54760	ephrin type-b receptor 4	SAM/Pointed domain
6g2r	P08191	type 1 fimbrin d-mannose specific adhesin	Bacterial adhesin-like
4at4	Q16620	bdnf/nt-3 growth factors receptor	Immunoglobulin (Ig) domain-like
3qrk	P00519	tyrosine-protein kinase abl1	alpha-Catenin/Vinculin-like
4tsx	F2WR39	integrase	Ribonuclease H-like
4c9x	P36639	7,8-dihydro-8-oxoguanine triphosphatase	Nudix
6qlr	P17931	galectin-3	Concanavalin A-like lectin-s/glucanases
3tc5	Q13526	peptidyl-prolyl cis-trans isomerase nima-interactin	WW domain
6as8	Q1RBS0	fml fimbrial adhesin fmld	Bacterial adhesin-like
6r8w	P30405	peptidyl-prolyl cis-trans isomerase f	Cyclophilin-like
4kow	Q9HV14	uncharacterized protein	Acyl-CoA N-acyltransferases (Nat)
5d47	P15090	fatty acid-binding protein, adipocyte	Lipocalins
5m6m	P98170	e3 ubiquitin-protein ligase xiap	Inhibitor of apoptosis (IAP) repeat
6q96	Q00987	e3 ubiquitin-protein ligase mdm2	SWIB/MDM2 domain-like
4z6i	Q9H8M2	bromodomain-containing protein 9	Bromodomain
4lwu	P56273	e3 ubiquitin-protein ligase mdm2	SWIB/MDM2 domain-like
1utr	P17559	uteroglobin	Uteroglobin-like

Table B.1: All 69 proteins used for the panel of proteins with the superfamily they belong to. The first 37, also in bold, constitute the panel for the IFPP evaluated on the whole *LH* Benchmark.

C

UMAP LIT-PCBA

Following Figures display the overlap in chemical spaces between decoys of the *LH* benchmark and molecules from 13 datasets of LIT-PCBA: ADRB2, ALDH1, ESR1 agonist, FEN1, GBA, IDH1, KAT2A, MAPK1, MTORC1, PKM2, PPARG, TP53, VDR.

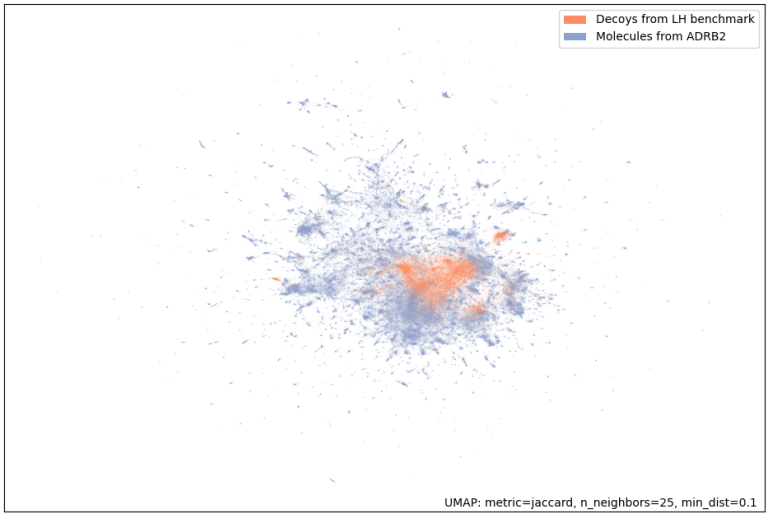


Figure C.1: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from ADRB2 dataset.

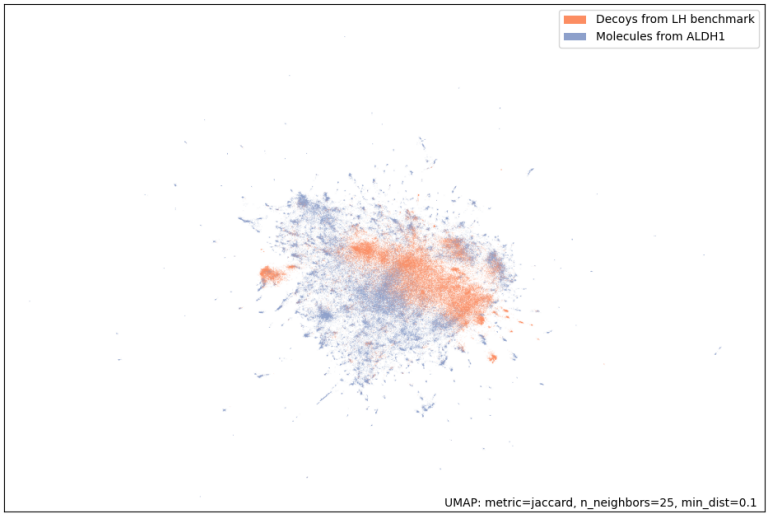


Figure C.2: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from ALDH1 dataset.

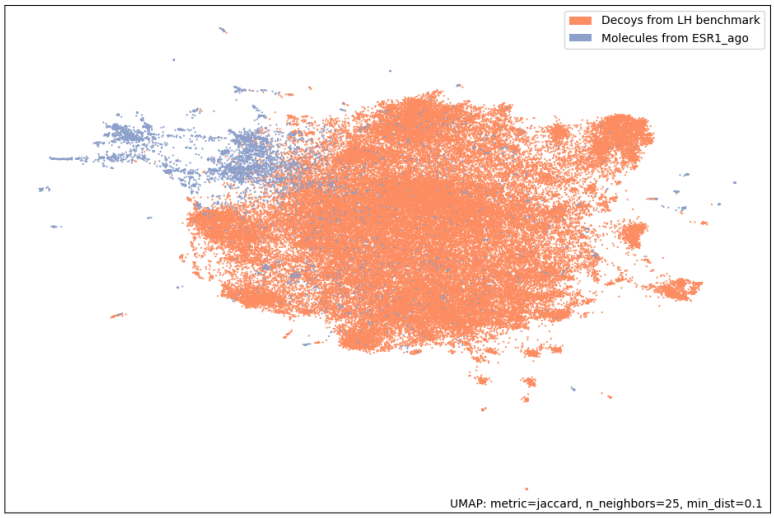


Figure C.3: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from ESR1 agonist dataset.

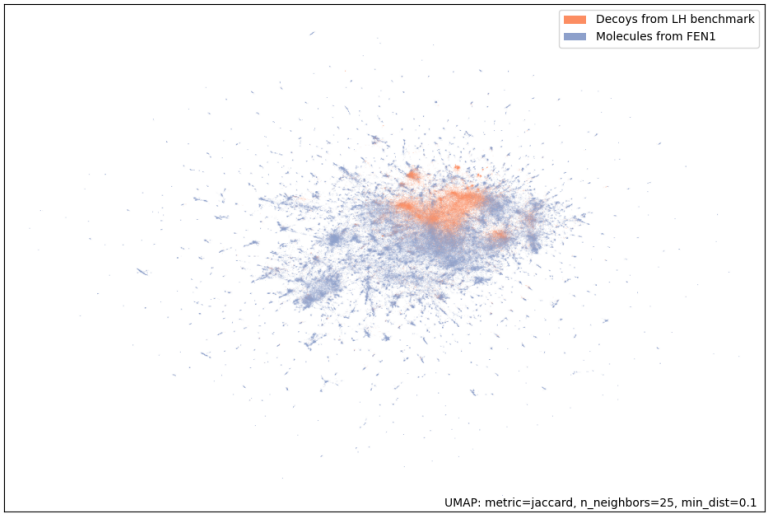


Figure C.4: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from FEN1 dataset.

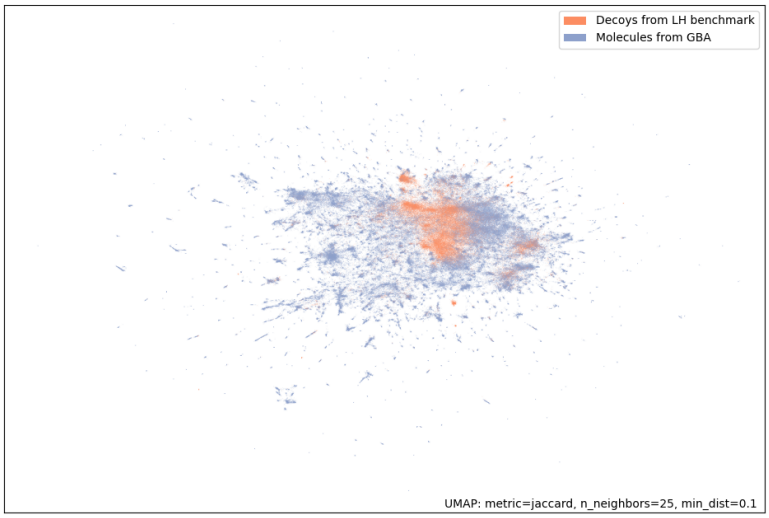


Figure C.5: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from GBA dataset.

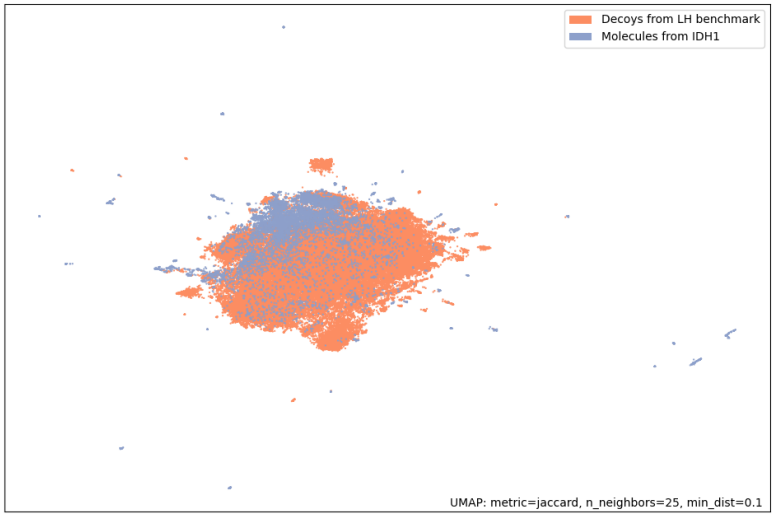


Figure C.6: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from IDH1 dataset.

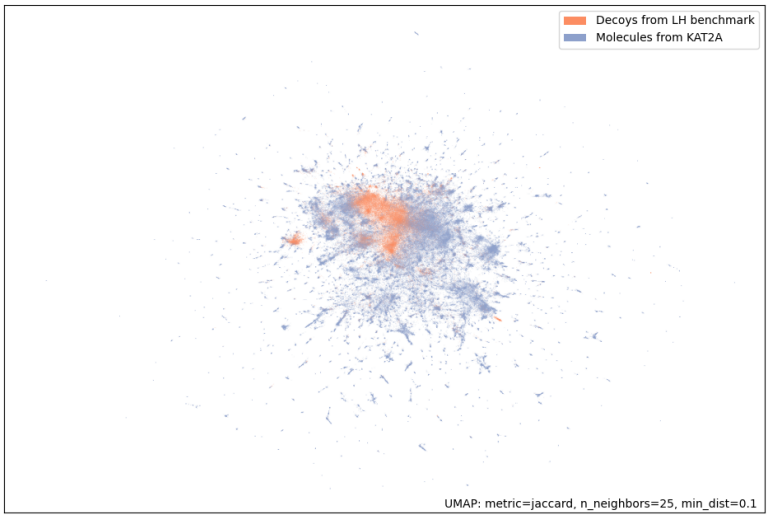


Figure C.7: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from KAT2A dataset.

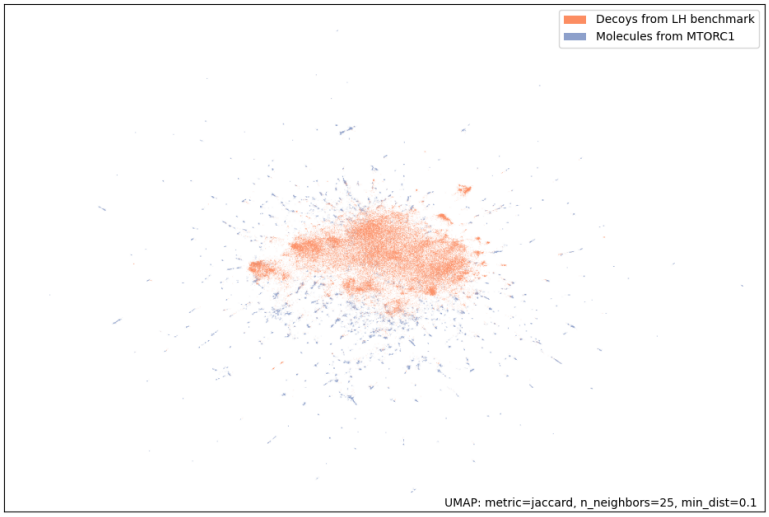


Figure C.8: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from MTORC1 dataset.

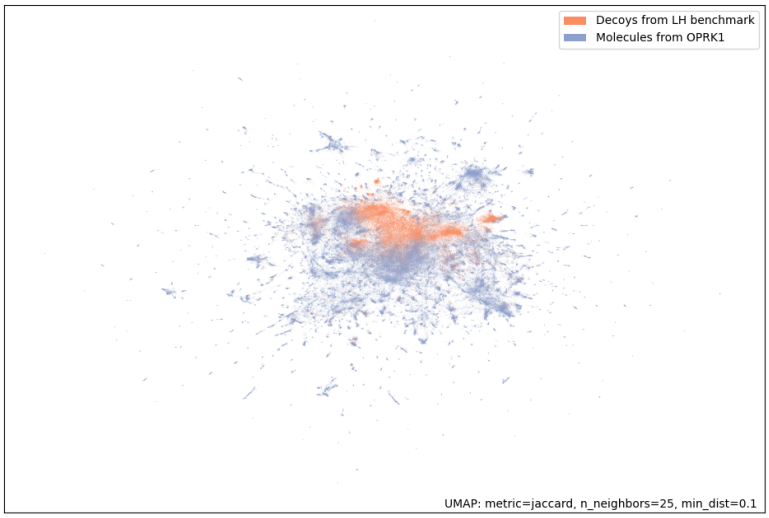


Figure C.9: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from OPRK1 dataset.

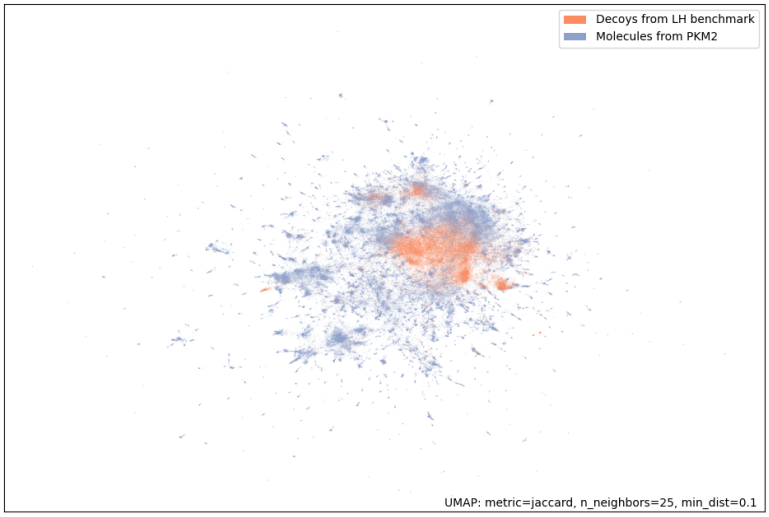


Figure C.10: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from PKM2 dataset.

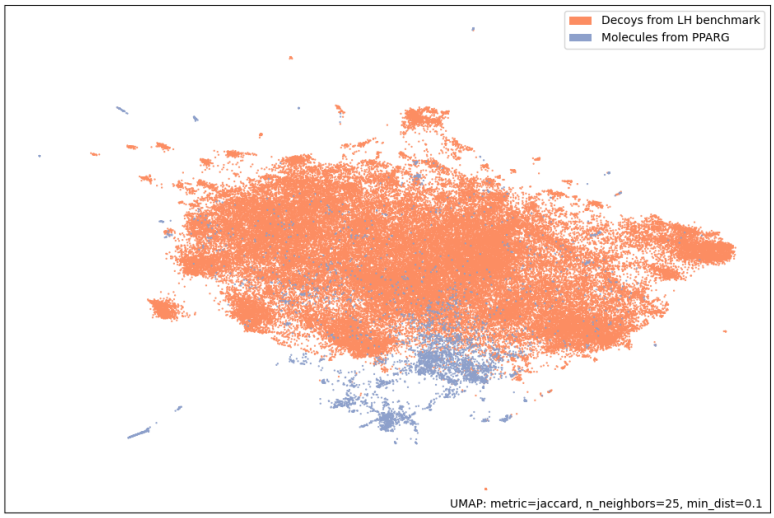


Figure C.11: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from PPARG dataset.

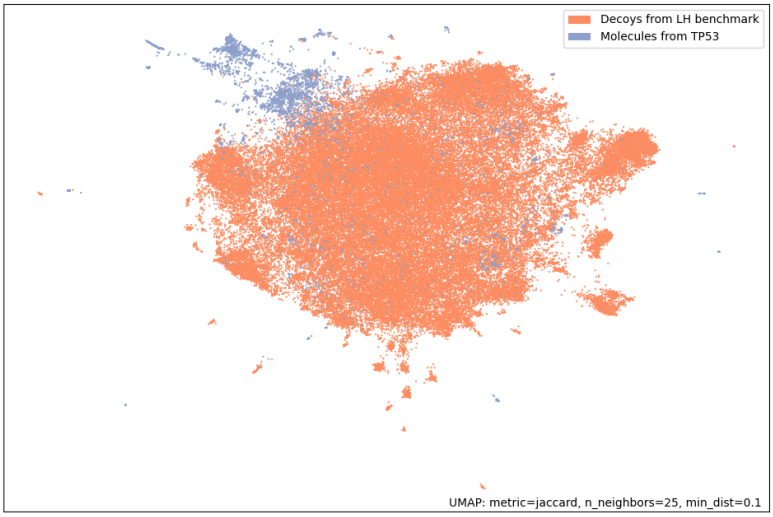


Figure C.12: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from TP53 dataset.

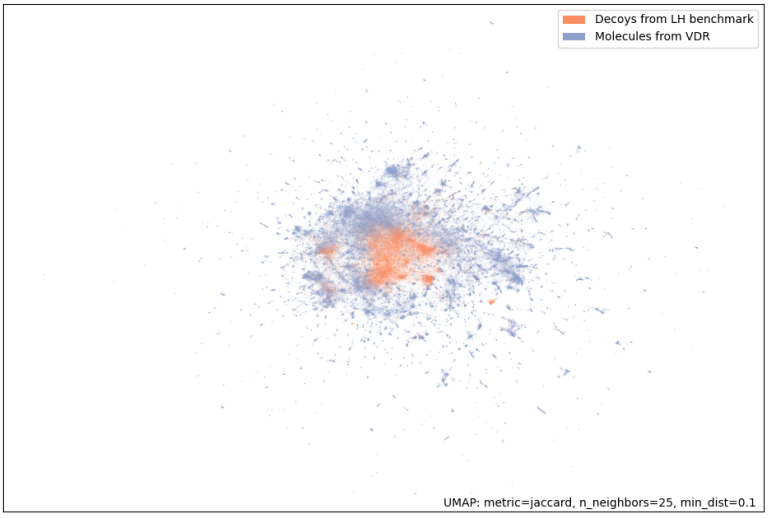


Figure C.13: 2D representation of the chemical space with the UMAP algorithm based on Morgan fingerprints. In orange: decoys from the *LH* benchmark, in blue: molecules from VDR dataset.

Bibliography

[Advani *et al.*(2013)] R. H. Advani, R. T. Hoppe, et al.*Efficacy of abbreviated Stanford V chemotherapy and involved-field radiotherapy in early-stage Hodgkin lymphoma: mature results of the G4 trial†*, Annals of Oncology **24** (4), 1044 (2013).

[Al-Dabbagh *et al.*(2015)] M. M. Al-Dabbagh, N. Salim, et al., *A Quantum-Based Similarity Method in Virtual Screening*, Molecules **20** (10), 18107 (2015).

[Bajorath(2019)] Bajorath (2019), in *IB Chemistry Revision Guide* (Anthem Press) pp. 222–238.

[Barker *et al.*(2006)] E. J. Barker, D. Buttar, et al., *Scaffold Hopping Using Clique Detection Applied to Reduced Graphs*, Journal of Chemical Information and Modeling **46** (2), 503, publisher: American Chemical Society (2006).

[Bemis et Murcko(1996)] G. W. Bemis et M. A. Murcko, *The Properties of Known Drugs. 1. Molecular Frameworks*, Journal of Medicinal Chemistry **39** (15), 2887, publisher: American Chemical Society (1996).

[Bento *et al.*(2014)] A. P. Bento, A. Gaulton, et al., *The ChEMBL bioactivity database: an update*, Nucleic Acids Research **42** (D1), D1083 (2014).

[Berenger *et al.*(2021)] F. Berenger, A. Kumar, et al., *Lean-Docking: Exploiting Ligands' Predicted Docking Scores to Accelerate Molecular Docking*, Journal of Chemical Information and Modeling **61** (5), 2341 (2021).

[Berenger et Tsuda(2023)] F. Berenger et K. Tsuda, *3D-Sensitive Encoding of Pharmacophore Features*, Journal of Chemical Information and Modeling **63** (8), 2360, publisher: American Chemical Society (2023).

[Berman *et al.*(2000)] H. M. Berman, J. Westbrook, et al., *The Protein Data Bank*, Nucleic Acids Research **28** (1), 235 (2000).

[Bickerton *et al.*(2012)] G. R. Bickerton, G. V. Paolini, et al., *Quantifying the chemical beauty of drugs*, Nature Chemistry **4** (2), 90, publisher: Springer Science and Business Media LLC (2012).

[Bishop(2006)] C. M. Bishop (2006), *Pattern Recognition and Machine Learning*.

[Bissantz *et al.*(2010)] C. Bissantz, B. Kuhn, et M. Stahl, *A Medicinal Chemist's Guide to Molecular Interactions*, Journal of Medicinal Chemistry **53** (14), 5061, publisher: American Chemical Society (ACS) (2010).

[Bolton *et al.*(2008)] E. E. Bolton, Y. Wang, et al. (2008), in *Annual Reports in Computational Chemistry*, Vol. 4, édité par R. A. Wheeler et D. C. Spellmeyer (Elsevier) pp. 217–241.

[Bongrand(1999)] P. Bongrand, *Ligand-receptor interactions*, Reports on Progress in Physics **62** (6), 921 (1999).

[Boström *et al.*(2007)] J. Boström, K. Berggren, et al., *Scaffold hopping, synthesis and structure–activity relationships of 5,6-diaryl-pyrazine-2-amide derivatives: A novel series of CB1 receptor antagonists*, Bioorganic & Medicinal Chemistry **15** (12), 4077 (2007).

[Breiman(2001)] L. Breiman, *Random Forests*, Machine Learning **45** (1), 5 (2001).

[Brocidiacono *et al.*(2024)] M. Brocidiacono, P. Francoeur, et al., *BigBind: Learning from Nonstructural Data for Structure-Based Virtual Screening*, Journal of Chemical Information and Modeling **64** (7), 2488 (2024).

[Bromley *et al.*(1993)] J. Bromley, I. Guyon, et al., *Signature Verification using a "Siamese" Time Delay Neural Network*, International Journal of Pattern Recognition and Artificial Intelligence (1993).

[Burley *et al.*(2023)] S. K. Burley, C. Bhikadiya, et al., *RCSB Protein Data Bank (RCSB.org): delivery of experimentally-determined PDB structures alongside one million computed structure models of proteins from artificial intelligence/machine learning*, Nucleic Acids Research **51** (D1), D488 (2023).

[Böhm *et al.*(2004)] H.-J. Böhm, A. Flohr, et M. Stahl, *Scaffold hopping*, Drug Discovery Today: Technologies **1** (3), 217 (2004).

[Cai *et al.*(2024)] H. Cai, C. Shen, et al., *CarsiDock: a deep learning paradigm for accurate protein–ligand docking and screening based on large-scale pre-training*, Chemical Science **15** (4), 1449, publisher: Royal Society of Chemistry (2024).

[Cai *et al.*(2022)] H. Cai, H. Zhang, et al., *FP-GNN: a versatile deep learning architecture for enhanced molecular property prediction*, Briefings in Bioinformatics **23** (6), bbac408 (2022).

[Carles *et al.*(2018)] F. Carles, S. Bourg, et al., *PKIDB: A Curated, Annotated and Updated Database of Protein Kinase Inhibitors in Clinical Trials*, Molecules **23** (4), 908, number: 4 Publisher: Multidisciplinary Digital Publishing Institute (2018).

[Carosati *et al.*(2007)] E. Carosati, R. Mannhold, et al., *Virtual Screening for Novel Openers of Pancreatic KATP Channels*, Journal of Medicinal Chemistry **50** (9), 2117, publisher: American Chemical Society (2007).

[Chung *et al.*(2014)] J. Chung, C. Gulcehre, et al. (2014), *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*, arXiv:1412.3555 [cs].

[Chupakhin *et al.*(2014)] V. Chupakhin, G. Marcou, et al., *Simple Ligand–Receptor Interaction Descriptor (SILIRID) for alignment-free binding site comparison*, Computational and Structural Biotechnology Journal **10** (16), 33, publisher: Elsevier BV (2014).

[Cortes et Vapnik(1995)] C. Cortes et V. Vapnik, *Support-vector networks*, Machine Learning **20** (3), 273 (1995).

[Coupry et Pogány(2022)] D. E. Coupry et P. Pogány, *Application of deep metric learning to molecular graph similarity*, Journal of Cheminformatics **14** (1), 11 (2022).

[Da et Kireev(2014)] C. Da et D. Kireev, *Structural Protein–Ligand Interaction Fingerprints (SPLIF) for Structure-Based Virtual Screening: Method and Benchmark Study*, Journal of Chemical Information and Modeling **54** (9), 2555, publisher: American Chemical Society (ACS) (2014).

[Dalby *et al.*(1992)] A. Dalby, J. G. Nourse, et al., *Description of several chemical structure file formats used by computer programs developed at Molecular Design Limited*, Journal of Chemical Information and Computer Sciences **32** (3), 244 (1992).

[Dick et Cocklin(2020)] A. Dick et S. Cocklin, *Bioisosteric Replacement as a Tool in Anti-HIV Drug Design*, Pharmaceuticals **13** (3), 36, number: 3 Publisher: Multidisciplinary Digital Publishing Institute (2020).

[Dolinsky *et al.*(2007)] T. J. Dolinsky, P. Czodrowski, et al., *PDB2PQR: expanding and upgrading automated preparation of biomolecular structures for molecular simulations*, Nucleic Acids Research **35** (Web Server issue), W522 (2007).

[Durant *et al.*(2002)] J. L. Durant, B. A. Leland, et al., *Reoptimization of MDL Keys for Use in Drug Discovery*, Journal of Chemical Information and Computer Sciences **42** (6), 1273, publisher: American Chemical Society (ACS) (2002).

[Duvenaud *et al.*(2015)] D. Duvenaud, D. Maclaurin, et al., *Convolutional Networks on Graphs for Learning Molecular Fingerprints*, (2015).

[Eddy(2004)] S. R. Eddy, *Where did the BLOSUM62 alignment score matrix come from?*, Nature Biotechnology **22** (8), 1035, publisher: Nature Publishing Group (2004).

[Elton *et al.*(2019)] D. C. Elton, Z. Boukouvalas, et al., *Deep learning for molecular design—a review of the state of the art*, Molecular Systems Design & Engineering **4** (4), 828, publisher: The Royal Society of Chemistry (2019).

[Ferreira de Freitas et Schapira(2017)] R. Ferreira de Freitas et M. Schapira, *A systematic analysis of atomic protein–ligand interactions in the PDB †Electronic supplementary information (ESI) available. See DOI: 10.1039/c7md00381a*, MedChemComm **8** (10), 1970 (2017).

[Freitas et Schapira(2017)] R. F. d. Freitas et M. Schapira, *A systematic analysis of atomic protein–ligand interactions in the PDB*, MedChemComm **8** (10), 1970, publisher: Royal Society of Chemistry (RSC) (2017).

[Gong *et al.*(2018)] Z. Gong, P. Zhong, et al., *Diversity-Promoting Deep Structural Metric Learning for Remote Sensing Scene Classification*, IEEE Transactions on Geoscience and Remote Sensing **56** (1), 371, conference Name: IEEE Transactions on Geoscience and Remote Sensing (2018).

[Good et Oprea(2008)] A. C. Good et T. I. Oprea, *Optimization of CAMD techniques 3. Virtual screening enrichment studies: a help or hindrance in tool selection?*, Journal of Computer-Aided Molecular Design **22** (3), 169 (2008).

[Goodfellow *et al.*(2016)] I. Goodfellow, Y. Bengio, et A. Courville (2016), *Deep Learning* (MIT Press) google-Books-ID: omivDQAAQBAJ.

[Gorgulla *et al.*(2020)] C. Gorgulla, A. Boeszoermenyi, et al., *An open-source drug discovery platform enables ultra-large virtual screens*, Nature **580** (7805), 663, publisher: Nature Publishing Group (2020).

[Grisoni *et al.*(2018a)] F. Grisoni, D. Merk, et al., *Scaffold-Hopping from Synthetic Drugs by Holistic Molecular Representation*, Scientific Reports **8** (1), 16469, number: 1 Publisher: Nature Publishing Group (2018a).

[Grisoni *et al.*(2018b)] F. Grisoni, D. Merk, et al., *Scaffold hopping from natural products to synthetic mimetics by holistic molecular similarity*, Communications Chemistry **1** (1), 1, number: 1 Publisher: Nature Publishing Group (2018b).

[Guichaoua *et al.*(2024)] G. Guichaoua, P. Pinel, et al. (2024), *Advancing Drug-Target Interactions Prediction: Leveraging a Large-Scale Dataset with a Rapid and Robust Chemogenomic Algorithm*.

[Gómez-Bombarelli *et al.*(2018)] R. Gómez-Bombarelli, J. N. Wei, et al., *Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules*, ACS Central Science **4** (2), 268 (2018).

[Hamilton *et al.*(2018)] W. L. Hamilton, R. Ying, et J. Leskovec (2018), *Representation Learning on Graphs: Methods and Applications*, arXiv:1709.05584 [cs].

[Hannun *et al.*(2014)] A. Hannun, C. Case, et al. (2014), *Deep Speech: Scaling up end-to-end speech recognition*, arXiv:1412.5567 [cs].

[Harder *et al.*(2013)] M. Harder, B. Kuhn, et F. Diederich, *Efficient Stacking on Protein Amide Fragments*, ChemMedChem **8** (3), 397 (2013).

[Helal *et al.*(2016)] K. Y. Helal, M. Maciejewski, et al., *Public Domain HTS Fingerprints: Design and Evaluation of Compound Bioactivity Profiles from PubChem's Bioassay Repository*, Journal of Chemical Information and Modeling **56** (2), 390, publisher: American Chemical Society (2016).

[Helguera *et al.*(2008)] A. M. Helguera, R. D. Combes, et al., *Applications of 2D Descriptors in Drug Design: A DRAGON Tale*, Current Topics in Medicinal Chemistry **8** (18), 1628 (2008).

[Hendrickson(1991)] J. B. Hendrickson, *Concepts and Applications of Molecular Similarity.*, Science **252** (5009), 1189, publisher: American Association for the Advancement of Science (1991).

[Hertzberg et Pope(2000)] R. P. Hertzberg et A. J. Pope, *High-throughput screening: new technology for the 21st century*, Current Opinion in Chemical Biology **4** (4), 445 (2000).

[Hessler et Baringhaus(2010)] G. Hessler et K.-H. Baringhaus, *The scaffold hopping potential of pharmacophores*, Drug Discovery Today: Technologies 3D Pharmacophore Elucidation and Virtual Screening, **7** (4), e263 (2010).

[Hochreiter et Schmidhuber(1997)] S. Hochreiter et J. Schmidhuber, *Long Short-Term Memory*, Neural Computation **9** (8), 1735, conference Name: Neural Computation (1997).

[Hu *et al.*(2016)] Y. Hu, D. Stumpfe, et J. Bajorath, *Computational Exploration of Molecular Scaffolds in Medicinal Chemistry*, Journal of Medicinal Chemistry **59** (9), 4062, publisher: American Chemical Society (2016).

[Hu *et al.*(2017)] Y. Hu, D. Stumpfe, et J. Bajorath, *Recent Advances in Scaffold Hopping*, Journal of Medicinal Chemistry **60** (4), 1238, publisher: American Chemical Society (2017).

[Hughes *et al.*(2011)] J. Hughes, S. Rees, et al., *Principles of early drug discovery*, British Journal of Pharmacology **162** (6), 1239 (2011).

[Irwin et Shoichet(2004)] J. J. Irwin et B. K. Shoichet, *ZINC - A Free Database of Commercially Available Compounds for Virtual Screening*, Journal of Chemical Information and Modeling **45** (1), 177, publisher: American Chemical Society (ACS) (2004).

[Jacob *et al.*(2008)] L. Jacob, B. Hoffmann, et al., *Virtual screening of GPCRs: An in silico chemogenomics approach*, BMC Bioinformatics **9** (1), 363 (2008).

[Jain(2007)] A. N. Jain, *Surflex-Dock 2.1: Robust performance from ligand energetic modeling, ring flexibility, and knowledge-based search*, Journal of Computer-Aided Molecular Design **21** (5), 281 (2007).

[Jiang *et al.*(2021)] D. Jiang, Z. Wu, et al., *Could graph neural networks learn better molecular representation for drug discovery? A comparison study of descriptor-based and graph-based models*, Journal of Cheminformatics **13** (1), 12 (2021).

[Jumper *et al.*(2021)] J. Jumper, R. Evans, et al., *Highly accurate protein structure prediction with AlphaFold*, Nature **596** (7873), 583, number: 7873 Publisher: Nature Publishing Group (2021).

[Kaplan *et al.*(2022)] A. L. Kaplan, D. N. Confair, et al., *Bespoke library docking for 5-HT2A receptor agonists with antidepressant activity*, Nature **610** (7932), 582, publisher: Nature Publishing Group (2022).

[Kearns et Valiant(1989)] M. Kearns et L. G. Valiant (1989), *Crytographic limitations on learning Boolean formulae and finite automata*, in *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, STOC '89 (Association for Computing Machinery, New York, NY, USA) pp. 433–444.

[Kim *et al.*(2019)] S. Kim, M. Seo, et al. (2019), *Deep Metric Learning Beyond Binary Supervision*, arXiv:1904.09626 [cs].

[Kingma et Welling(2022)] D. P. Kingma et M. Welling (2022), *Auto-Encoding Variational Bayes*, arXiv:1312.6114 [cs, stat].

[Koch *et al.*(2015)] G. Koch, R. Zemel, et R. Salakhutdinov, *Siamese Neural Networks for One-shot Image Recognition*, ICML Deep Learning Workshop (2015).

[Koge *et al.*(2021)] D. Koge, N. Ono, et al., *Embedding of Molecular Structure Using Molecular Hypergraph Variational Autoencoder with Metric Learning*, Molecular Informatics **40** (2), 2000203, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/minf.202000203 (2021).

[Krishna *et al.*(2024)] R. Krishna, J. Wang, et al., *Generalized biomolecular modeling and design with RoseTTAFold All-Atom*, Science **384** (6693), eadl2528 (2024).

[Kuhn *et al.*(2019)] B. Kuhn, E. Gilberg, et al., *How Significant Are Unusual Protein–Ligand Interactions? Insights from Database Mining*, Journal of Medicinal Chemistry **62** (22), 10441, publisher: American Chemical Society (ACS) (2019).

[Kuntz *et al.*(1982)] I. D. Kuntz, J. M. Blaney, et al., *A geometric approach to macromolecule-ligand interactions*, Journal of Molecular Biology **161** (2), 269 (1982).

[Lagarde *et al.*(2015)] N. Lagarde, J.-F. Zagury, et M. Montes, *Benchmarking Data Sets for the Evaluation of Virtual Ligand Screening Methods: Review and Perspectives*, Journal of Chemical Information and Modeling **55** (7), 1297, publisher: American Chemical Society (2015).

[Landrum *et al.*(2021)] G. Landrum, P. Tosco, et al. (2021), *rdkit/rdkit: 2021_03_5 (Q1 2021) Release*.

[Lang *et al.*(2009)] P. T. Lang, S. R. Brozell, et al., *DOCK 6: Combining techniques to model RNA–small molecule complexes*, RNA **15** (6), 1219 (2009).

[Li *et al.*(2017)] J. Li, D. Cai, et X. He (2017), *Learning Graph-Level Representation for Drug Discovery*, arXiv:1709.03741 [cs, stat].

[Li *et al.*(2019a)] J. Li, A. Fu, et L. Zhang, *An Overview of Scoring Functions Used for Protein–Ligand Interactions in Molecular Docking*, Interdisciplinary Sciences: Computational Life Sciences **11** (2), 320 (2019a).

[Li *et al.*(2021)] M. Li, J. Zhou, et al., *DGL-LifeSci: An Open-Source Toolkit for Deep Learning on Graphs in Life Science*, ACS Omega **6** (41), 27233, publisher: American Chemical Society (2021).

[Li *et al.*(2011)] X.-Z. Li, B. Walker, et A. Michaelides, *Quantum nature of the hydrogen bond*, Proceedings of the National Academy of Sciences **108** (16), 6369, publisher: Proceedings of the National Academy of Sciences (2011).

[Li *et al.*(2019b)] Z. Li, Z. Cui, et al. (2019b), *Fi-GNN: Modeling Feature Interactions via Graph Neural Networks for CTR Prediction*, in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 539–548, arXiv:1910.05552 [cs].

[Lipinski(2000)] C. A. Lipinski, *Drug-like properties and the causes of poor solubility and poor permeability*, Journal of Pharmacological and Toxicological Methods Current Directions in Drug Discovery:A Review of Modern Techniques, **44** (1), 235 (2000).

[Liu *et al.*(2006)] T. Liu, A. W. Moore, et A. Gray (2006), in *Nearest-Neighbor Methods in Learning and Vision*, édité par G. Shakhnarovich, T. Darrell, et P. Indyk (The MIT Press) pp. 75–102.

[Liu *et al.*(2018)] W. Liu, Y. Wen, et al. (2018), *SphereFace: Deep Hypersphere Embedding for Face Recognition*, arXiv:1704.08063 [cs] version: 4.

[Lovrics *et al.*(2019)] A. Lovrics, V. F. S. Pape, et al., *Identifying new topoisomerase II poison scaffolds by combining publicly available toxicity data and 2D/3D-based virtual screening*, Journal of Cheminformatics **11** (1), 67 (2019).

[Lyu *et al.*(2019)] J. Lyu, S. Wang, et al., *Ultra-large library docking for discovering new chemotypes*, Nature **566** (7743), 224, publisher: Nature Publishing Group (2019).

[Ma *et al.*(2015)] J. Ma, R. P. Sheridan, et al., *Deep Neural Nets as a Method for Quantitative Structure–Activity Relationships*, Journal of Chemical Information and Modeling **55** (2), 263 (2015).

[Mahé *et al.*(2006)] P. Mahé, L. Ralaivola, et al., *The Pharmacophore Kernel for Virtual Screening with Support Vector Machines*, Journal of Chemical Information and Modeling **46** (5), 2003 (2006).

[Marcou et Rognan(2006)] G. Marcou et D. Rognan, *Optimizing Fragment and Scaffold Docking by Use of Molecular Interaction Fingerprints*, Journal of Chemical Information and Modeling **47** (1), 195, publisher: American Chemical Society (ACS) (2006).

[Mayr *et al.*(2016)] A. Mayr, G. Klambauer, et al., *DeepTox: Toxicity Prediction using Deep Learning*, Frontiers in Environmental Science **3**, 10.3389/fenvs.2015.00080, publisher: Frontiers (2016).

[McGregor et Muskal(1999)] M. J. McGregor et S. M. Muskal, *Pharmacophore fingerprinting. 1. Application to QSAR and focused library design*, Journal of Chemical Information and Computer Sciences **39** (3), 569 (1999).

[McInnes *et al.*(2020)] L. McInnes, J. Healy, et J. Melville (2020), *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*, arXiv:1802.03426 [cs, stat].

[McNutt *et al.*(2021)] A. T. McNutt, P. Francoeur, et al., *GNINA 1.0: molecular docking with deep learning*, Journal of Cheminformatics **13** (1), 43 (2021).

[Mnih *et al.*(2015)] V. Mnih, K. Kavukcuoglu, et al., *Human-level control through deep reinforcement learning*, Nature **518** (7540), 529 (2015).

[Morgan(1965)] H. L. Morgan, *The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service.*, Journal of Chemical Documentation **5** (2), 107, publisher: American Chemical Society (1965).

[Murzin *et al.*(1995)] A. G. Murzin, S. E. Brenner, et al., *SCOP: A structural classification of proteins database for the investigation of sequences and structures*, Journal of Molecular Biology **247** (4), 536 (1995).

[Najm *et al.*(2021)] M. Najm, C.-A. Azencott, et al., *Drug Target Identification with Machine Learning: How to Choose Negative Examples*, International Journal of Molecular Sciences **22** (10), 5118, number: 10 Publisher: Multidisciplinary Digital Publishing Institute (2021).

[Nakano *et al.*(2020)] H. Nakano, T. Miyao, et K. Funatsu, *Exploring Topological Pharmacophore Graphs for Scaffold Hopping*, Journal of Chemical Information and Modeling **60** (4), 2073, publisher: American Chemical Society (2020).

[Nakano *et al.*(2021)] H. Nakano, T. Miyao, et al., *Sparse Topological Pharmacophore Graphs for Interpretable Scaffold Hopping*, Journal of Chemical Information and Modeling **61** (7), 3348, publisher: American Chemical Society (2021).

[Needleman et Wunsch(1970)] S. B. Needleman et C. D. Wunsch, *A general method applicable to the search for similarities in the amino acid sequence of two proteins*, Journal of Molecular Biology **48** (3), 443 (1970).

[Nittinger *et al.*(2017)] E. Nittinger, T. Inhester, et al., *Large-Scale Analysis of Hydrogen Bond Interaction Patterns in Protein–Ligand Interfaces*, Journal of Medicinal Chemistry **60** (10), 4245, publisher: American Chemical Society (ACS) (2017).

[N. Muratov *et al.*(2020)] E. N. Muratov, J. Bajorath, et al., *QSAR without borders*, Chemical Society Reviews **49** (11), 3525, publisher: Royal Society of Chemistry (2020).

[Okuno *et al.*(2006)] Y. Okuno, J. Yang, et al., *GLIDA: GPCR-ligand database for chemical genomic drug discovery*, Nucleic Acids Research **34** (suppl_1), D673 (2006).

[Pang *et al.*(2021)] J. Pang, S. Gao, et al., *Discovery of small molecule PLpro inhibitor against COVID-19 using structure-based virtual screening, molecular dynamics simulation, and molecular mechanics/Generalized Born surface area (MM/GBSA) calculation*, Structural Chemistry **32** (2), 879 (2021).

[Paszke *et al.*(2019)] A. Paszke, S. Gross, et al. (2019), *PyTorch: An Imperative Style, High-Performance Deep Learning Library*, in *Advances in Neural Information Processing Systems*, Vol. 32 (Curran Associates, Inc.).

[Paulini *et al.*(2005)] R. Paulini, K. Müller, et F. Diederich, *Orthogonal Multipolar Interactions in Structural Chemistry and Biology*, Angewandte Chemie International Edition **44** (12), 1788, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/anie.200462213 (2005).

[Pedregosa *et al.*(2011)] F. Pedregosa, G. Varoquaux, et al., *Scikit-learn: Machine Learning in Python*, MACHINE LEARNING IN PYTHON , 6 (2011).

[Pence et Williams(2010)] H. E. Pence et A. Williams, *ChemSpider: An Online Chemical Information Resource*, Journal of Chemical Education **87** (11), 1123, publisher: American Chemical Society (2010).

[Petrone *et al.*(2012)] P. M. Petrone, B. Simms, et al., *Rethinking Molecular Similarity: Comparing Compounds on the Basis of Biological Activity*, ACS Chemical Biology **7** (8), 1399, publisher: American Chemical Society (2012).

[Pinel *et al.*(2024)] P. Pinel, G. Guichaoua, et al. (2024), *A molecular representation to identify isofunctional molecules*, pages: 2024.05.03.592355 Section: New Results.

[Pinel *et al.*(2023)] P. Pinel, G. Guichaoua, et al., *Exploring isofunctional molecules: Design of a benchmark and evaluation of prediction performance*, Molecular Informatics **42** (4), 2200216 (2023).

[Platt(1999)] J. Platt, *Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods*, Advances in large margin classifiers **10** (3), 61, publisher: Cambridge, MA (1999).

[Playe *et al.*(2018)] B. Playe, C.-A. Azencott, et V. Stoven, *Efficient multi-task chemogenomics for drug specificity prediction*, PLOS ONE **13** (10), e0204999, publisher: Public Library of Science (2018).

[Playe et Stoven(2020)] B. Playe et V. Stoven, *Evaluation of deep and shallow learning methods in chemogenomics for the prediction of drugs specificity*, Journal of Cheminformatics **12** (1), 11 (2020).

[Ratni *et al.*(2015)] H. Ratni, M. Rogers-Evans, et al., *Discovery of Highly Selective Brain-Penetrant Vasopressin 1a Antagonists for the Potential Treatment of Autism via a Chemogenomic and Scaffold Hopping Approach*, Journal of Medicinal Chemistry **58** (5), 2275, publisher: American Chemical Society (2015).

[Rogers et Hahn(2010)] D. Rogers et M. Hahn, *Extended-Connectivity Fingerprints*, Journal of Chemical Information and Modeling **50** (5), 742, publisher: American Chemical Society (ACS) (2010).

[Rush *et al.*(2005)] T. S. Rush, J. A. Grant, et al., *A Shape Based 3D Scaffold Hopping Method and Its Application to a Bacterial Protein Protein Interaction*, Journal of Medicinal Chemistry **48** (5), 1489, publisher: American Chemical Society (2005).

[Réau *et al.*(2018)] M. Réau, F. Langenfeld, et al., *Decoys Selection in Benchmarking Datasets: Overview and Perspectives*, Frontiers in Pharmacology **9** (2018).

[Sadybekov *et al.*(2020)] A. A. Sadybekov, R. L. Brouillette, et al., *Structure-Based Virtual Screening of Ultra-Large Library Yields Potent Antagonists for a Lipid GPCR*, Biomolecules **10** (12), 1634, number: 12 Publisher: Multidisciplinary Digital Publishing Institute (2020).

[Saigo *et al.*(2004)] H. Saigo, J.-P. Vert, et al., *Protein homology detection using string alignment kernels*, Bioinformatics **20** (11), 1682 (2004).

[Salentin *et al.*(2014)] S. Salentin, V. J. Haupt, et al., *Polypharmacology rescored: Protein–ligand interaction profiles for remote binding site similarity assessment*, Progress in Biophysics and Molecular Biology **116** (2-3), 174, publisher: Elsevier BV (2014).

[Salentin *et al.*(2015)] S. Salentin, S. Schreiber, et al., *PLIP: fully automated protein–ligand interaction profiler*, Nucleic Acids Research **43** (W1), W443 (2015).

[Scardino *et al.*(2023)] V. Scardino, J. I. Di Filippo, et C. N. Cavasotto, *How good are AlphaFold models for docking-based virtual screening?*, iScience **26** (1), 105920 (2023).

[Schneider *et al.*(1999)] G. Schneider, W. Neidhart, et al., *"Scaffold-Hopping" by Topological Pharmacophore Search: A Contribution to Virtual Screening*, Angewandte Chemie International Edition **38** (19), 2894, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/%28SICI%291521-3773%2819991004%2938%3A19%3C2894%3A%3AAID-ANIE2894%3E3.0.CO%3B2-F (1999).

[Schölkopf *et al.*(2004)] B. Schölkopf, K. Tsuda, et J.-P. Vert (2004), *Kernel Methods in Computational Biology* (MIT Press).

[Segler *et al.*(2018)] M. H. S. Segler, T. Kogej, et al., *Generating Focused Molecule Libraries for Drug Discovery with Recurrent Neural Networks*, ACS Central Science **4** (1), 120 (2018).

[Shinada *et al.*(2019)] N. K. Shinada, A. G. d. Brevern, et P. Schmidtke, *Halogens in Protein–Ligand Binding Mechanism: A Structural Perspective*, Journal of Medicinal Chemistry **62** (21), 9341, publisher: American Chemical Society (ACS) (2019).

[Song *et al.*(2017)] H. O. Song, S. Jegelka, et al. (2017), *Deep Metric Learning via Facility Location*, in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Honolulu, HI) pp. 2206–2214.

[Stanzione *et al.*(2021)] F. Stanzione, I. Giangreco, et J. C. Cole (2021), in *Progress in Medicinal Chemistry*, Vol. 60, édité par D. R. Witty et B. Cox (Elsevier) pp. 273–343.

[Stein *et al.*(2020)] R. M. Stein, H. J. Kang, et al., *Virtual discovery of melatonin receptor ligands to modulate circadian rhythms*, Nature **579** (7800), 609, publisher: Nature Publishing Group (2020).

[Sun *et al.*(2022)] D. Sun, W. Gao, et al., *Why 90% of clinical drug development fails and how to improve it?*, Acta Pharmaceutica Sinica. B **12** (7), 3049 (2022).

[Sun *et al.*(2012)] H. Sun, G. Tawa, et A. Wallqvist, *Classification of scaffold-hopping approaches*, Drug Discovery Today **17** (7), 310 (2012).

[Sutherland *et al.*(2023)] J. J. Sutherland, D. Yonchev, et al., *A preclinical secondary pharmacology resource illuminates target-adverse drug reaction associations of marketed drugs*, Nature Communications **14** (1), 4323, publisher: Nature Publishing Group (2023).

[Swamidass *et al.*(2005)] S. J. Swamidass, J. Chen, et al., *Kernels for small molecules and the prediction of mutagenicity, toxicity and anti-cancer activity*, Bioinformatics **21** (suppl_1), i359 (2005).

[Taigman *et al.*(2014)] Y. Taigman, M. Yang, et al. (2014), *DeepFace: Closing the Gap to Human-Level Performance in Face Verification*, in *2014 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, Columbus, OH, USA) pp. 1701–1708.

[Taminau *et al.*(2008)] J. Taminau, G. Thijs, et H. De Winter, *Pharao: Pharmacophore alignment and optimization*, Journal of Molecular Graphics and Modelling **27** (2), 161 (2008).

[Tosco *et al.*(2014)] P. Tosco, N. Stiefl, et G. Landrum, *Bringing the MMFF force field to the RDKit: implementation and validation*, Journal of Cheminformatics **6** (1), 37 (2014).

[Tran-Nguyen *et al.*(2020)] V.-K. Tran-Nguyen, C. Jacquemard, et D. Rognan, *LIT-PCBA: An Unbiased Data Set for Machine Learning and Virtual Screening*, Journal of Chemical Information and Modeling **60** (9), 4263, publisher: American Chemical Society (2020).

[Van Der Spoel *et al.*(2005)] D. Van Der Spoel, E. Lindahl, et al., *GROMACS: Fast, flexible, and free*, Journal of Computational Chemistry **26** (16), 1701, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/jcc.20291 (2005).

[Vaswani *et al.*(2017)] A. Vaswani, N. Shazeer, et al., *Attention is All you Need*, (2017).

[Veličković *et al.*(2018)] P. Veličković, G. Cucurull, et al. (2018), *Graph Attention Networks*, arXiv:1710.10903 [cs, stat].

[Vert et Jacob(2008)] J.-P. Vert et L. Jacob, *Machine Learning for In Silico Virtual Screening and Chemical Genomics: New Strategies*, Combinatorial Chemistry & High Throughput Screening **11** (8), 677 (2008).

[Vieth *et al.*(2004)] M. Vieth, M. G. Siegel, et al., *Characteristic Physical Properties and Structural Fragments of Marketed Oral Drugs*, Journal of Medicinal Chemistry **47** (1), 224 (2004).

[Vogel *et al.*(2011)] S. M. Vogel, M. R. Bauer, et F. M. Boeckler, *DEKOIS: Demanding Evaluation Kits for Objective in Silico Screening — A Versatile Tool for Benchmarking Docking Programs and Scoring Functions*, Journal of Chemical Information and Modeling **51** (10), 2650, publisher: American Chemical Society (ACS) (2011).

[Wallach et Heifets(2018)] I. Wallach et A. Heifets, *Most Ligand-Based Classification Benchmarks Reward Memorization Rather than Generalization*, Journal of Chemical Information and Modeling **58** (5), 916 (2018).

[Walters(2019)] W. P. Walters, *Virtual Chemical Libraries*, Journal of Medicinal Chemistry **62** (3), 1116, publisher: American Chemical Society (2019).

[Wang *et al.*(2004a)] J. Wang, R. M. Wolf, et al., *Development and testing of a general amber force field*, Journal of Computational Chemistry **25** (9), 1157, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/jcc.20035 (2004a).

[Wang *et al.*(2004b)] R. Wang, X. Fang, et al., *The PDBbind Database:\hspace0.167em Collection of Binding Affinities for Protein-Ligand Complexes with Known Three-Dimensional Structures*, Journal of Medicinal Chemistry **47** (12), 2977, publisher: American Chemical Society (ACS) (2004b).

[Wang *et al.*(2019)] X. Wang, X. Han, et al. (2019), *Multi-Similarity Loss With General Pair Weighting for Deep Metric Learning*, in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Long Beach, CA, USA) pp. 5017–5025.

[Wang *et al.*(2009)] Y. Wang, J. Xiao, et al., *PubChem: a public information system for analyzing bioactivities of small molecules*, Nucleic Acids Research **37** (Web Server), W623 (2009).

[Wassermann *et al.*(2015)] A. M. Wassermann, E. Lounkine, et al., *The opportunities of mining historical and collective data in drug discovery*, Drug Discovery Today **20** (4), 422 (2015).

[Weininger(1988)] D. Weininger, *SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules*, Journal of Chemical Information and Computer Sciences **28** (1), 31 (1988).

[Wishart *et al.*(2018)] D. S. Wishart, Y. D. Feunang, et al., *DrugBank 5.0: a major update to the DrugBank database for 2018*, Nucleic Acids Research **46** (D1), D1074 (2018).

[Xiong *et al.*(2021)] G.-L. Xiong, Y. Zhao, et al., *Computational Bioactivity Fingerprint Similarities To Navigate the Discovery of Novel Scaffolds*, Journal of Medicinal Chemistry **64** (11), 7544, publisher: American Chemical Society (2021).

[Xiong *et al.*(2020)] Z. Xiong, D. Wang, et al., *Pushing the Boundaries of Molecular Representation for Drug Discovery with the Graph Attention Mechanism*, Journal of Medicinal Chemistry **63** (16), 8749, publisher: American Chemical Society (2020).

[Xu *et al.*(2019)] K. Xu, W. Hu, et al. (2019), *How Powerful are Graph Neural Networks?*, arXiv:1810.00826 [cs, stat].

[Zhao *et al.*(2020)] J. Zhao, Y. Cao, et L. Zhang, *Exploring the computational methods for protein-ligand binding site prediction*, Computational and Structural Biotechnology Journal **18**, 417 (2020).

RÉSUMÉ

La découverte de médicaments, de l'identification de candidats jusqu'au développement clinique, implique parfois de résoudre des problèmes de 'scaffold hopping', dans le but d'optimiser l'activité biologique, la sélectivité, les propriétés ADME, ou de réduire les préoccupations toxicologiques des molécules. Ils consistent à identifier des molécules actives dont les modes de liaison sont similaires mais dont les structures chimiques sont différentes de celles des actifs connus. Le 'large-step scaffold hopping', qui correspond au degré le plus élevé de différence structurelle avec la molécule initiale, nécessite l'aide de méthodes calculatoires. Le docking est considéré comme la méthode de choix pour l'identification de telles molécules isofonctionnelles. Cependant, la structure de la protéine peut ne pas être adaptée au docking en raison d'une faible résolution, voire être inconnue. Dans de tels cas, les approches 'ligand-based' sont prometteuses mais souvent insuffisantes car basées sur des descripteurs moléculaires n'ayant pas été spécifiquement développés pour le 'large-step scaffold hopping'. La résolution de ces problèmes se résume à l'identification de descripteurs correspondant à une représentation de l'espace chimique dans laquelle deux molécules qui sont des cas de 'scaffold hopping' sont similaires, bien qu'elles soient dissemblables dans l'espace représenté par les descripteurs basés principalement sur la structure chimique. Afin d'évaluer la capacité des descripteurs à les résoudre, nous avons constitué un ensemble de cas de 'scaffold hopping' de haute qualité comprenant des paires de molécules actives pour une variété de protéines. Nous avons ensuite proposé une stratégie pour évaluer la pertinence des descripteurs pour résoudre ces problèmes, correspondant à des cas réels où une molécule active est connue, et la seconde active est recherchée parmi un ensemble de molécules leurres choisies de manière à éviter les biais statistiques. Nous avons ainsi illustré les limites des descripteurs classiques 2D et 3D. Par conséquent, nous proposons l'Interaction Fingerprints Profile (IFPP), une représentation moléculaire qui capture les modes de liaison des molécules via des dockings sur un panel de protéines diverses. L'évaluation de cette représentation sur le benchmark démontre son intérêt pour l'identification de molécules isofonctionnelles. Cependant, son calcul coûteux limite sa mise à l'échelle pour le criblage de bibliothèques moléculaires très larges. Nous avons remedié à cela en tirant parti du Metric Learning, qui permet une estimation rapide des similarités des IFPP des molécules, fournissant ainsi une stratégie de pré-criblage efficace applicable à de larges bibliothèques. Nos résultats suggèrent que l'IFPP est un outil intéressant et complémentaire aux méthodes existantes afin de résoudre le 'scaffold hopping'.

MOTS CLÉS

Docking, Interactions moléculaires, Machine Learning, Représentation moléculaire, Scaffold hopping

ABSTRACT

The challenges of drug discovery from hit identification to clinical development sometimes involves addressing scaffold hopping issues, in order to optimise molecular biological activity or ADME properties, improve selectivity or mitigate toxicology concerns of a drug candidate. They consist in identifying active molecules of similar binding modes but of different chemical structures to that of known active molecules. Large-step scaffold hopping, which corresponds to the highest degree of structural dissimilarity with the original hit, cannot be easily solved without the aid of computational methods. Docking is usually viewed as the method of choice for identification of such isofunctional molecules. However, the structure of the protein may not be suitable for docking because of a low resolution, or may even be unknown. In such cases, ligand-based approaches offer promise but are often inadequate to handle large-step scaffold hopping, because they are based on molecular descriptors that were not specifically developed for it. Solving those problems boils down to the identification of molecular descriptors corresponding to an embedding of the chemical space in which two molecules that are examples of large-step scaffold hopping cases are similar (i.e. close), although they are dissimilar (i.e. far) in the space embedded by molecular descriptors based principally on the chemical structure. To evaluate molecular descriptors to solve this particular challenging task, we built a high quality dataset of scaffold hopping examples comprising pairs of active molecules and including a variety of protein targets. We then proposed a strategy to evaluate the relevance of molecular descriptors to that problem, corresponding to real-life applications where one active molecule is known, and the second active is searched among a set of decoys chosen in a way to avoid statistical bias. We assessed how limited classical 2D and 3D descriptors are at solving these problems. Therefore, we introduced the Interaction Fingerprints Profile (IFPP), a molecular representation that captures molecules' binding modes based on docking experiments against a panel of diverse high-quality protein structures. Evaluation on the benchmark demonstrated its interest for identifying isofunctional molecules. Nevertheless, its computation is expensive, which limits its scalability for screening very large molecular libraries. We proposed to overcome this limitation by leveraging Metric Learning approaches, allowing fast estimation of molecules IFPP similarities, thus providing an efficient pre-screening strategy that is applicable to very large molecular libraries. Overall, our results suggest that IFPP provides an interesting and complementary tool alongside existing methods, in order to address challenging scaffold hopping problems effectively in drug discovery.

KEYWORDS

Docking, Molecular interactions, Machine Learning, Molecular representation, Scaffold hopping