LigandMapper.py

LIGAND BINDING SITE PREDICTOR

Explaining the theory behind the project

CONTENT

INTRODUCTION

Proteins perform diverse biochemical functions by binding to multiple molecules at different pockets on their surface. These distinct binding sites are crucial for structure-based drug design, making them valuable targets for drug discovery. Drug designers must search for small drug-like molecules that can block these pockets on specific proteins related to various diseases. Understanding and characterizing these binding pockets is essential to grasp the intricacies of molecular recognition and to provide functional annotation for orphan proteins. [8] Rational drug-design and help in drug side-effects prediction, as well as elucidation of protein function, are very important topics that have to be analysed.

One way of investigating about this topic is by protein-ligand binding site predictions from a 3D protein structure. 3D protein structure is important and has many applications, but it represents only one step in the range of many complex computational drug design efforts. There has been many methods published so far, but only a narrow selection of those methods are suitable for use in automated pipelines or for data processing of large datasets. More so, these approaches require high speed and stability, which disqualifies many of the recently composed tools that are either template based or available only as web servers. One of the main problems in the ligand-binding prediction computational approach is so called *pocket ranking* which is a question of how to score and sort candidate pockets in a way so that the best scored predictions correspond to true ligand binding sites. Since the approach that we have used is computational with machine learning as the background for the algorithm, this issue was a very important part of this project. [5]

DIFFERENT APPROACHES TO LIGAND BINDING PREDICTION

Identifying binding pockets is a first step in a structure-based drug discovery and it is followed by a more detailed description of the pocket. Pockets are concavities on the protein surface where a substrate may bind, but there are also pockets that are referred to as "druggable" pockets and those are the ones where small drug-like molecules can attach. Binding pockets can also be defined by their chemical space, relationships across different target classes, and their static, transient, or dynamic nature. Additionally, pockets can be classified as monomeric or multimeric (composed of several subunits) interfacial pockets. The understanding of binding pockets is rapidly evolving, and this progress is making developing novel therapies to advance faster which improves the treatment of human diseases.

There are two main categories of methods for identifying binding pockets. One of them is geometry-based, while another is energy-based. Geometry-based methods rely on the observation that binding pockets are often clefts or cavities in proteins and can be identified through geometrical criteria. They often incorporate physico-chemical information like polarity or charge. *The Voss Volume Voxelator* is one of the examples of this type and it is implemented as a web server. It does not require a starting point and it allows researchers to investigate the volumes of macromolecules and detect channels. *The CAVER method* is another one and it can be used to detect channels on molecular dynamics trajectories or conformational ensembles.

On the other hand, energy-based methods address the issue that not all binding sites are the largest pockets or clefts in a protein. Energetic methods build the approximations of free energy potentials by force fields, placing probes around the protein surface and calculating binding energies. These methods use changeable probes to detect different binding pockets and provide maximal flexibility to discriminate between different types of binding sites. [8]

Other methods include evolutionary, knowledge-based and consensus methods. Evolutionary methods include algorithms that make use of sequence conservation estimates, since functional residues are more evolutionary conserved, or protein threading. They utilize the idea that *functionally* important residues are conserved during evolution, making them valuable for identifying potential binding pockets. One approach, such as *LIGSITEcsc*, uses a sequence conservation measure of neighbouring residues to re-rank the top putative pockets calculated by *LIGSITEcsc*, leading to an improved success rate. [3]

Another example of an evolutionary-based method that considers structural information is *FINDSITE*. This algorithm selects ligand-bound structural templates from a database of known protein-ligand complexes using a threading algorithm that combines various scoring functions to match structurally related target/template pairs. Homologous structures are aligned with the target protein using a global structural alignment algorithm, and positions of ligands on superimposed template structures are clustered into consensus binding sites. [5]

Methods like *LIGSITE* that are based on evolutionary conservation can introduce bias towards binding sites with biological ligands. They assume that, since ligands have a biological function, then based on that, they are more prone to bind to some particular binding sites, then to some other binding sites. Thus, pockets that are not evolutionary conserved could be ignored by this approach which would be bad, since these are the pockets that can hold information for a novel binding site and function.

Method that we implemented is based only on local geometric and physico-chemical features of points near protein surface, so it should not be prone to previously explained bias. However, since this is a machine learning algorithm, so it has been trained on a particular dataset, a bias might be introduced there. This is a general and common issue when implementing machine learning techniques. Nevertheless, since the classifier has been trained to predict ligandability of pocket points that represent local chemical neighbourhood, rather than the whole pocket; the generalization has been applied to correctly predict ligandability of novel binding sites.

This approach takes a PDB structure as an input and then outputs a ranked list of predicted ligand binging sites defined by a set of points. The set of regularly spaced points lies on a protein's Connolly surface (Connolly surface is explained further down). These points are called *Connolly points*. The next step is calculating feature descriptors of Connolly points by computing property vectors for protein's solvent exposed atoms, projecting distance weighted properties of the adjacent protein atoms onto Connolly points and computing additional features afterwards. Eventually, Random Forest is used for the prediction of the ligandability score. Points with high ligandability score are clustered to form pocket predictions and pockets are ranked based on cumulative ligandability score of their points.

METHOD FOR RANKING

A scoring function is a way to order found pockets based on the probabilities of locating them on the particular places mapped on the protein. Since pocket identification algorithms are heuristic in nature, a scoring function is necessary, because it will provide a measure of confidence in the predictions. One way and the most common way for scoring putative pockets, pockets that have a really high probability of existing at a particular place on the protein, is to order them by a single descriptor such as volume, pocket depth, overall hydrophobicity or surface area. Additionally, pocket descriptors can be combined and used all together, as it was demonstrated by *Fpocket*. [1]

Fpocket is an open source pocket detection package based on Voronoi tessellation[1] and alpha spheres written in the C programming language. It is organised around a central library of functions in three main programs: Fpocket (to perform pocket identification), Tpocket (to organize pocket detection benchmarking on a set of known protein-ligand complexes) and Dpocket (to collect pocket descriptor values on a set of proteins). [6] This method has been used in our project to test for the ligand binding sites after training data with the corresponding datasets (see below).

ConCavity is another approach with the same idea, but on the other hand, it considers the overall pocket evolutionary conservation score projected onto pocket grid probes for ranking. **[5]** It makes specific predictions of positions in space that are likely to overlap ligand atoms and of residues that are likely to contact bound ligands. The pocket detection methods with the highest success rates in the benchmark appear to be those with more sophisticated ranking algorithms. **[1]** We used this method to compare it with previously explained Fpocket and to test for the significance of the results.

[1] Given a set *P := {p1, ..., pn}* of sites, a Voronoi Tessellation is a subdivision of the space into *n cells*, one for each site in *P*, with the property that a point *q* lies in the cell corresponding to a site *pi* iff *d(pi, q) < d(pj, q)* for *i* distinct from *j*.

CONNOLLY SURFACE

For studying the protein structure and function, the outer surface of a macromolecule must be defined, since that's the part of the molecule that binds ligands and different macromolecules. When it comes to smaller molecules, *the van der Waals surface* gives a reliable representation of the outer surface, but it is different for larger molecules, since most of the van der Waals surface is buried in the interior. In 1977, Richards presented a definition by which the molecular surface consists of the contact surface and the reentrant surface. **[7]** The former is the part of the van der Walls surface of the atoms and it is accessible to a probe sphere representing a solvent molecule. The latter represents the inward-facing surface of the probe sphere when it is in contact with more than one atom. There was, however, no method for calculating such surface.

In later years, there has been implemented a way of calculating the surface which assumes that the rolling of a probe sphere over a molecule is best understood in terms of translational degrees of freedom. If the probe is in contact with the molecule, it will have three degrees of freedom and it, additionally, looses one, for each atom that it touches. Thus, there are three cases of the number of atoms that a probe sphere may simultaneously touch and for each of the cases, a different shaped type of surface is generated. Each of the types is defined by the sphere it lies on and a boundary contour; each sphere defined by a centre and a radius. These pieces of surface form a joined network that cover the protein and they are hold together at common boundary arcs, referred to as edges, while the arcs meeting points are named vertices. At each point there is a well-defined tangent plane which makes these parts smooth and it is contrary to the van der Waals surface of a molecule where sharp cervices represent the atoms' intersections.

The Connolly surface is a computational representation of the molecular surface of a molecule, typically a protein. It is named after the scientist Andrew Connolly who first introduced this concept in 1983. The Connolly surface is constructed by rolling a probe sphere over the van der Waals surface of the molecule, as mentioned in the previous paragraph. The probe sphere is usually modelled as a water molecule or some other solvent. The surface points at which the probe sphere touches the van der Waals surface are then connected to form a continuous surface, which is the Connolly surface. The Connolly surface is useful in studying molecular interactions, such as ligand binding, because it provides a detailed representation of the protein surface topology, which is important for understanding protein-ligand interactions. [2]

The Connolly surface can be used in the computational algorithms for identifying pockets that may be suitable for small molecule binding. Pockets that are being identified are usually defined by a set of parameters such as shape, volume and different surface properties. One important feature of the Connolly surface is its ability to account for solvent accessibility. When modelled with a probe sphere, the Connolly surface can make a reliable distinction between exposed and buried regions of the protein surface. This was important in our case of study, because ligands typically bind to exposed regions of a protein, where they can interact with amino acid residues and other molecular features. There are many computational tools available for generating and analyzing Connolly surface, including software packages like PyMOL, Chimera, and VMD. Our output was made to be visualized using Chimera or PyMOL.

In general, these tools allow researchers to visualize and manipulate the surface in three dimensions, which can be useful for identifying potential binding sites or analyzing protein-ligand interactions. Overall, the Connolly surface is a valuable tool in the study of protein-ligand interactions and drug discovery. Its ability to accurately represent the molecular surface of a protein and account for solvent accessibility makes it an important tool for identifying and characterizing ligand binding sites.

REPRESENTING A POCKET

To represent a pocket, a way of choosing points on the molecule surface is needed. LigandMapper.py uses a group of inner points that it has selected by spacing points on the Connolly surface evenly. The spacings are made within 4 Å of the closest heavy pocket atom. The next step in the process is to make a feature vector to each of the chosen points. Such vector is made first by calculating feature vectors for specific pocket atoms, and then afterwards, they are aggregated into feature vectors of inner points. The vector is computed based on the properties of the pocket atoms, such as their distance from the ligand, element type and solvent accessibility. The aggregation process involves combining the computed vectors of the pocket atoms that are closest to each inner point and the resulting vector servers as a representation of the local environment around each inner point. This vector we used as an input for a machine learning algorithm to predict the ligandability of the point.

It should be explained that all the features included in the vectors are local, meaning that they are calculated based on the spatial neighbourhood of the points that are the nearest. Thus, the shape and properties of the whole pocket or protein are not considered in this model. Inner pocket points from different parts of the pocket can have very different feature vectors, so this locality has a positive impact on the model's generalization ability. However, when considering only local features, there is an possible issue of those features not being sufficient to cover the ligand binding quality of particular regions of the protein surface where some ligand where some ligand positions could be fixed by relatively distant non-covalent bonds. Based on other models that have been using a similar approach, it seems that this locality of features actually leads to improvements in terms of pocket prediction. At large, this implementation of feature vectors and locality seems to be a good way of converting complex objects or data points into a more easily manageable and standardized form.

SCORING FUNCTION AND RANDOM FOREST AS A PREDICTOR

The algorithm that we have proposed aims to determine whether certain points within a protein's inner pocket are capable of binding to ligands, using machine learning approach for prediction. This describes a binary classification problem for supervised learning where as the positive points are labeled those that are located within 2.5 Å distance to any ligand atom.

Previous studies that implement similar method reported a highly imbalanced set in terms of the ratio of positives and negatives, after training on a particular dataset like CHEN11 (different dataset are explained below). Additionally, they reported that even after trying a broad of compensation techniques such as oversampling or undersampling, they found reduced generalization ability of a classifier that has been trained. For this reason, in our training step we have decided not to perform any copmenzation techniques and chose a Random Forest as a predictive modelling tool. It seems that the Random Forest is fast and robust to the presence of a large number of irrelevant variables. Furthermore, it can also handle correlated variables.

Random forests classifier returns a histogram of class probabilities which is used to rescore the putative pockets. After executing the program via command line, a table-like representation of top predicted pockets, their scores and probabilities is given as the output, along with the file to be opened and observed via *Chimera* software. Since the point can either be seen as a pocket point or not, and so it is binary, the histogram is an ordered pair. The scores is the sum of predicted squared positive class probabilities of all inner pocket points.

$$\textbf{score} = \boldsymbol{\Sigma}(\boldsymbol{P_1}(\boldsymbol{V_i}))^2$$

Previously, there has been trials of scoring based on the mean probability based pocket score, but the cumulative one showed the best results. If a size of a correctly predicted pocket vary little from the true pocket, it should still be recognized as a true pocket. The higher the score of a putative pocket is, the higher the probability of it being a true pocket. Therefore, the last step requires reordering the putative pockets in the decreasing order based on their scores.

Different parameters can influence different steps of the algorithm. Even while not taking into account the hyperparameters of the classifier, there are still a variety of additional parameters that should be considered, since they can have the impact on the experiment. For this matter, the default values of those parameters have been optimized by linear and grid search and the performance of CHEN11 dataset was used as a reference to optimize parameters e.g. the probe radius of Connolly's surface or ligand distance threshold to denote positive and negative points.

DATASETS USED FOR MACHINE LEARNING

For the prediction, the model has been tested on several datasets which are described in this part. For each dataset predictions were generated using two algorithms, Fpocket and ConCavity. They have been explained in a section previously.

CHEN11 dataset includes 251 proteins and 476 ligands and it was designed with the intention to non-redundantly cover all SCOP families of ligand binding proteins from PDB.

DT198 is a dataset of 198 drug-target complexes.

MP210 is a benchmarking dataset of 210 proteins in bound state introduced in the MetaPocket study.

ASTEX is a collection of 85 proteins that was introduced as a benchmarking dataset for molecular docking methods.

UB48 contains a set of 48 proteins in a bound and unbound state and this dataset has been the most widely used for comparing pocket detection methods. It contains mainly small globular proteins.

EVALUATION

For the evaluation, ligand-centric approach has been used, instead of protein-centric approach. Since the goal is to identify every pocket on a particular protein for every relevant ligand in the data set, ligand-centric approach accomplishes this task. On the other side, a protein-centric approach would only require every protein to have at least one identified binding site. A pocket is considered successfully identified if at least one pocket, of all predicted pockets, passes a chosen detection threshold. D-ca is defined as the minimal distance between the centre of the predicted pocket and any atom of the ligand. A binding site is then considered correctly predicted if D-ca is not greater than an arbitrary threshold, which is usually set to 4 Å, because this value corresponds to the average radius of gyration for ligand molecules in the datasets (around 4.03 Å). [2] D-cc is defined as the distance between the centre of the predicted pocket and the centre of the ligand. It was introduced in the *Findsite study* to compensate for the size of the ligand.

Since the algorithm is based on physico-chemical properties, if some region on protein's surface is recognised as a true ligand-binding site, that means that there exists a ligand which binds at exactly that site. On the other hand then, when looking at the negatives, which correspond to the cases where there is no true binding site, but all other points within the putative pockets, it could mean two different scenarios. The first one would be that no ligand can, indeed, bind at that place, because of the physico-chemical properties that are not favorable for the binding. Taking into account that the other case would be that there is no crystal structure where the binding event could happen, but it could be that it has not been found yet, then some of the true positives are incorrectly labeled as negative, because of the lack of complete experimental data. **[6]**

REFERENCES

1. Capra JA, Laskowski RA, Thornton JM, Singh M, Funkhouser TA. Predicting protein ligand binding sites by combining evolutionary sequence conservation and 3D structure. PLoS
2. Connolly M. Solvent-accessible surfaces of proteins and nucleic acids. Science. 1983;221(4612):709–13.
3. Huang B, Schroeder M. LIGSITEcsc: predicting ligand binding sites using the Connolly surface and degree of conservation. BMC Struct Biol. 2006 Sep 24;6:19. doi: 10.1186/1472-6807-6-19.
4. Krivák R, Hoksza D. P2Rank: machine learning based tool for rapid and accurate prediction of ligand binding sites from protein structure. J Cheminform. 2018 Aug 14;10(1):39. doi: 10.1186/s13321-018-0285-8.
5. Krivák R, Hoksza D. Improving protein-ligand binding site prediction accuracy by classification of inner pocket points using local features. J Cheminform. 2015 Apr 1;7:12. doi: 10.1186/s13321-015-0059-5.
6. Le Guilloux V, Schmidtke P, Tuffery P. Fpocket: an open source platform for ligand pocket detection. BMC Bioinformatics. 2009 Jun 2;10:168. doi: 10.1186/1471-2105-10-168.
7. RICHARDS, E M. (1977). Ann. Rev. Biophys. Bioeng. 6, 151-176.
8. 1Zheng X, Gan L, Wang E, Wang J. Pocket-based drug design: exploring pocket space. AAPS J. 2013 Jan;15(1):228-41. doi: 10.1208/s12248-012-9426-6. Epub 2012 Nov 22.