FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

BACHELOR THESIS

Name Surname

Thesis title

Name of the department

Supervisor of the bachelor thesis: Supervisor's Name
Study programme: study programme
Study branch: study branch

Prague YEAR

I declare that I carried out this bachelor thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In .... date ....
Author's signature

Dedication.

Title: Thesis title

Author: Name Surname

Department: Name of the department

Supervisor: Supervisor’s Name, department

Abstract: Abstract.

Keywords: key words

Contents

Introduction

[TK: AI] The field of drug discovery and bioinformatics increasingly leverages Machine Learning (ML) models for predicting protein-ligand interaction sites. Understanding these interactions forms a crucial aspect of rational drug design and aids in the development of new therapeutics. While several ML models exist for this purpose, challenges persist. The accurate prediction of ligand binding sites on proteins remains a complex problem due to the vast conformational space sampled by both the protein and potential ligands, along with the need for incorporation of diverse chemical and biological information related to the protein structure.

In this thesis I'm expanding P2Rank, a state-of-the-art algorithm for ligand binding sites prediction, by introducing new models in place of the original random forest classifier described in the original paper [TK: CITE].

1. ML Models

In this thesis, I have implemented and tested 3 various models:

1. Baseline - Random forest classifier
2. Random forest classifier on surroundings
3. REFINED: https://doi.org/10.1038/s41598-021-90923-y

1.1 Model interface

All of these models have a common interface of a method `predict(protein)`, which takes a protein as an input and returns the predicted probability of each input point being a ligand binding site.

1.1.1 Input protein

The input protein is a 3D cloud of points with features, where each point represents an atom [TK: TRUE?] in the protein and features characterize chemical and biological characteristics of the described area.

In the program, this is represented by a 2D array of features by locations. The physical position is one of the features, using which we can recreate the points cloud.

1.1.2 Output probabilities

The output of the models is the predicted probability of each location to be a ligand binding site. Using the input order, the output is simply an array of predicted probabilities.

1.2 Baseline Random Forest Model

As a baseline model, we are using a random forest classifier as described in the original paper. It does training and prediction point-wise.

This means that it predicts the probability of a site being ligand binding by only the features logged for the given site.

1.3 Surroundings extraction

For the following models, we need to have features for the surroundings of the site as well as the features logged for it.

To achieve this, we run a simple quadratic algorithm for finding the k nearest neighbors. Then we just append their features to the original ones. For the following algorithms, we use these features instead of the original ones.

1.4 Random Forest Model on surroundings

This model uses RFC as well, but it uses not only the given site's features but also features of the surrounding k sites, extracted as described above[TK, link].

1.5 REFINED

This model follows the following steps

1. Input: Set of samples $X = x_1, ...x_n$ in form of matrices of size *single point features* x *surroundings size* = N.
2. We regulize each feature to follow $f \sim N(0,1)$
3. We minimize over all permutations of features the function of

   $$f(X) = \sum_{s=1}^{n} \sum_{i,j,k,l=1}^{n} D(x_{s,ij}, x_{s,kl}) d((i,j),(k,l)^{-1}$$

   Where d is the euclidian distance and D absolute difference.
   As this is computationally exponential, we approximate this using the hill climbing algorithm.
   Tk: In the original paper this is done with basian noise and a lot of continuous math - but in principle the same. I don't know why. Maybe because it allows them to run this on GPU for some reason? Or it help the HCA?
4. Now we train a CNN-based classifier on the created dataset. Tk: Explain CNNs