Statistical Machine Learning

Hilary Term 2025

Group-Assessed Practical

Protein-Ligand Binding Affinity Prediction

Description. This project aims to predict the binding affinity of a molecule to a protein target, based on some pre-computed features. You have been provided with two different sets of pre-computed features: 'fps' and 'embed' (further details of both featurisations are provided later, but there is no expectation of a good understanding of how the pre-computed features were obtained).

The dataset consists of $p_{\text{fps}} = 1024$ and $p_{\text{embed}} = 224$ pre-computed features extracted from 1157 molecules. Each molecule i is represented by an input vector $x_i = (x_{i1}, \ldots, x_{ip})$ where $x_{ij} \in \mathbb{R}$ represents the j'th feature for molecule i. For the 'embed' featurisation, the features are real-valued, while for 'fps' they are binary. Each molecule has an associated label that represents its binding affinity to the protein target of interest. Binding affinity is a quantitative measure of how strongly a molecule binds to its target, with larger values indicating stronger binding.

The dataset is split into a training set of 975 molecules, a public test set of 109 molecules, and a private test set of 73 molecules. For the training and public test observations, you have access to both the inputs (X_{fps,embed}_train, X_{fps,embed}_public_test) and outputs (y_train, y_public_test). For the private test set, you have only access to the inputs (X_{fps,embed}_private_test), and the objective is to predict the binding affinity.

Tasks. You are asked to perform the following tasks. You will be assessed based on the quality of your report.

1. (5 marks) Conduct exploratory data analysis for the data provided.
2. (15 marks) Construct regression models that predict the binding affinity of a molecule based on (i) the 'fps' features only and (ii) the 'embed' features only. Analyse the performance of the regression models based on suitable evaluation metrics.
3. (5 marks) Construct a final model that you believe will perform best on the private test set. Provide an estimate of its expected performance with respect to an appropriate evaluation metric and provide the predictions for the private test set.

Methods. You are free to use any machine learning technique you wish, as long as you describe clearly in the report all the steps and choices that you have made. While getting a good predictive performance for your method will be important, remember that you will be assessed based on the quality of your report; so explaining your steps and choices clearly and discussing all the issues you have faced in this practical will be essential. Besides explaining your final predictor, you should also describe some of the other techniques you have tried and include a brief description of the more computational aspects of your work. It is particularly important to discuss the potential advantages/disadvantages of the different methods considered, in terms of interpretability, computational cost, etc. You can use any programming language you wish (Python, R, etc.), and any available library/toolbox, as long as you understand and can describe the methods used. In Python, most of the methods covered in the course (except convolutional neural networks) are implemented in Scikit-learn. In R, many machine learning methods are implemented in the (meta)-package caret.

Report. The report has a limit of 2,500 words. Please be as concise as you can. You should work in teams of 4 participants. Remember to place your team name, which consists of the collated anonymous IDs of all group members, on the cover page of the report. Please include the code you used to obtain your final predictor, performance estimate, and predictions on the private test set as an appendix (this does not count towards the 2,500 words limit). Make sure the code is readable (i.e. it contains comments explaining what you are doing). Only one student from each group is required to make the submission.

Submissions. Together with your report, you should also submit a csv file, containing the prediction for the 73 observations in the private test set. The submission file (csv format) should contain two columns: Id and Prediction. The file should contain a header, followed by the 73 real-valued predictions, and have the following format:

```
Id, Prediction
0, 7.11
1, 8.22
...
72, 9.33
```

A sample submission file is available.

Files available:

- This pdf with the instructions
- Training inputs: `X_{fps,embed}_train.csv`
- Training outputs: `y_train.csv`
- Public test inputs: `X_{fps,embed}_public_test.csv`
- Public test outputs: `y_public_test.csv`
- Private test inputs: `X_{fps,embed}_private_test.csv`
- Sample submission file: `myprediction.csv`
- Sample Python code: `AssessedPracticalSamplePythonCode.ipynb`

Sample Python code. A sample Python notebook is provided. The code loads the data, fits a k-nearest neighbour regression model on the training set and predicts the label of examples the test set. It then exports a csv file of the correct format. For your information, the 3-nearest neighbour regression model has a mean squared error of about 1.14 on the public test set and 1.34 on the private test set with the 'fps' features. You can use these values as a benchmark, and a lower bound for the performance of your model; you should be able to achieve higher performances with other methods.

Deadline. The deadline to submit your pdf report and csv file is Wednesday 26 March noon UK time (week 10).

Additional information about the two featurisations.
This section is just provided for your information. There is no expectation that you should have a good understanding of how the pre-computed features were obtained.

‘fps’ featurisation. The ‘fps’ featurisation was obtained by calculating the Morgan fingerprint of each molecule. Morgan fingerprints are a way to represent the structure of a molecule by encoding its connectivity information into a numerical fingerprint. They achieve this by first determining all molecular substructures with a maximum radius from each atom in the molecule. Then the molecular substructures are condensed into a fixed-length bit vector. A ‘1’ at a given position in the fingerprint indicates the presence of a particular substructure in a molecule, while a ‘0’ indicates its absence. Here, the ‘fps’ featurisation used Morgan fingerprints of radius 2 and length 1024.

‘embed’ featurisation. The ‘embed’ featurisation was obtained from the latent representation of a convolutional neural network applied to a voxel-grid representation of the 3D structure of the molecule in complex with the protein target. The ‘embed’ featurisation has dimension 224.