Machine Learning for Molecular Engineering

Problem Set 6

Date: May 7, 2021
Due: 10 pm ET on Thursday, May 21, 2021

Instructions This is the final problem set for the undergraduate version of this course (3.100, 10.402, 20.301). This exercise is a supervised learning competition hosted on Kaggle. Here is the link to the competition. For the submission, you will need to submit a notebook containing your code and a short writeup to describe your solutions; the writeup can be integrated with your notebook in a markdown cell. We have included a coarse grading rubric at the end of this document. You can find the template notebook here. You can also submit a separate notebook as the solution to the machine learning competition. As always, you are encouraged to ask for clarification on Piazza if you find any of the questions statements unclear. **Because this is the final pset, you will need to use your discretion in deciding how to tackle these problems; we have not specified exactly how each part should be approached.**

Background

Solvation Free Energies

Solvation free energies (SFEs) are differences in the thermodynamic potential of a mixed solute molecule and a solvent reservoir and a separated solute molecule and solvent reservoir. The SFE can be used to describe the relative population of a chemical species in solution and air (really, in a vacuum) at thermodynamic equilibrium. Solvation free energies also provide insight into how a solvent behaves in different environments and how it partitioning between two different environments, e.g., hydrophilic blood and hydrophobic tissue for a drug. Some of the most important physicochemical and drug metabolism properties are solubility, permeability, clearance, volume of distribution, and half-life [1], which all depend on partition coefficients that can be derived from SFEs. Improving our capability to estimate SFEs accurately will help us better predict these properties and design more efficient drugs.

However, the accuracy of predictions for solvation free energies can be poor, leading to fold changes in the prediction of solubility and partitioning. This is true of both *ab initio* and data-driven approaches. Predicting the solvation energy of ionic solutes (i.e., charged species) is particularly challenging [2]. In this pset, you will explore a machine learning solution to predict partition coefficients given a pair of solvent and solute. This is similar to other quantitative structure-property relationship tasks we've seen in the course, but crucially it depends on the *interaction* of two molecules, not just one molecule.

The partition coefficient data we will use are from the "Solv@TUM" database. [3]. The authors calculated the partition coefficients from published experimental infinite dilution activity coefficient, Henry's law constant, and mole fraction solubility data using thermodynamic relationships. We the provide a training set with four columns: Solvent (SMILES), Solute (SMILES), $\log_{10} K$ (partition coefficient), ΔG (solvation free energy). The solvation free energy and $\log K$ are related by the follow mathematical relation:

$$\Delta G = -kT \log(K) \tag{1}$$

where K, the equilibrium constant, is defined as the relative population:

$$K = \frac{[solute]_{solvent}}{[solute]_{air}} \tag{2}$$

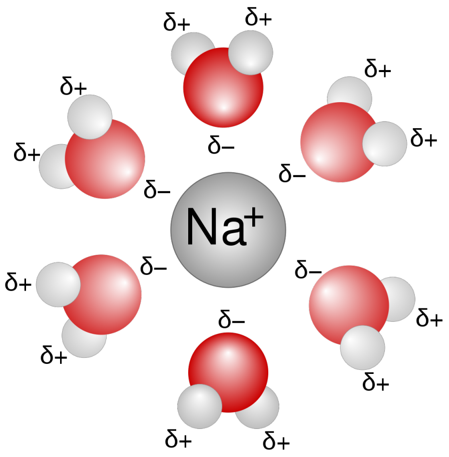

Figure 1: A sodium ion solvated by water molecules. Source: Wikipedia

Registering for the competition

Go to kaggle.com to register for an account. After successful registration, click here to join the competition. Your can submit your solutions to the test set and your result will be displayed on the leaderboard. If you prefer anonymity, you can choose a nickname like `molmlmaster` rather than your real name. Note that you are not graded based on your ranking but a combination of different factors which will be described below. Our competition will have no cash prizes, but you can explore other competitions on Kaggle for potential cash prizes and employment opportunities.

Part 1: (30 points) Baseline regression methods

This section provide instructions to get you started, and will walk you through to train a couple of baseline ML methods for the prediction task. You don't need to submit your predictions to Kaggle for this part.

Part 1.1 (10 points) Prepare dataset

To get the data, you can either to the Data section in Kaggle or directly use the code we provided to download the data. There are 3 files you can download:

`solvation_train.csv`: training set . There are four columns: `"Solvent"`, `"Solute"`, `"logK"`, `"delG_solv"`, which are the Solvent molecule SMILES, solute molecule SMILES, partition coefficient, and SFE.

`solvation_test.csv`: The solvent/solute SMILEs used to generate solutions for the test set

`mol_prop.csv`: For each molecule, we also provided its molecular weight, dipole moment, and polarizabilities, which you can use to generate features to train a model.

We provided some utility functions for you to load and generate features which you can use to train your models. The first step is to get familiar and load your data. Write code to build a feature set that combines physical descriptors for solvents and solutes. You need to concatenate solvent and solute features into a feature vector for each solvation/solute pair. Remember to normalize your feature set. The code you need to develop is similar to what you did for pset 3.

Part 1.2 (10 points) Linear regression

Apply linear regressions on the training data to predict $logK$. Combine normalized physical descriptors for solvents and solutes into one vectors as inputs. Report 5-fold cross-validated R^2 score.

Part 1.3 (10 points) MLP regression

Use the same input features as previous part to train an MLP regressor to predict $logK$. Report 5-fold cross-validated R^2 score.

Part 2: (70 points) Machine Learning Competition and Report

In this part, you will apply the techniques you learned in this class to propose machine learning solutions to a supervised regression problem. Based on the training data we provide to you, you will try to make predictions for the held-out test data. We provided the test feature set in `solvation_test.csv`. You will use your model to predict $logK$ using this test feature set. The evaluation of test performance will be handled by Kaggle, and you can see your results in a Leaderboard. The goal is find the best model possible. The teaching team will also participate in the competition.

You can submit up to 20 solutions to Kaggle per day. We provided a utility function for you to generate a submission file. Your answers will be compared with the true test labels on Kaggle and your model performance will be ranked in a Leaderboard. The metric we use is the R^2 score, ranging from 0 to 1. The score on Kaggle is calculated with only 40% percent of the test data, so your final performance might change during the final evaluation; this is to prevent you from tuning performance to the test data (which is bad practice!). We will release the final performance values when the competition is over.

Please note that in your representation or in your model architecture, you will need some way of learning about the *interactions* between the solute and solvent molecules. How you've accomplished this must be addressed explicitly in your writeup.

You need to submit your commented code (Jupyter notebook) with some markdown cells/descriptions that address the following points:

1. Data preprocessing
2. Choice of feature representations

   You are provided with molecular SMILES and a very small number of physical properties for both solvent and solute molecules. You can represent them with SMILES strings, circular fingerprints, molecular graphs, other RDKit descriptors, or look at descriptor-calculating packages like Mordred. It's up to you! Describe your choice of molecular representation and briefly explain your reason.

3. Choice of model architecture

   You are expected to try at least two methods and select the best-performing model to submit the best solution. If you are designing a novel model in pytorch, describe your model architecture. For each of your models, report cross-validation scores. You need to provide brief description of your model choice or design and mention any open-sourced code/packages you used. if you decide to adopt a model architecture or method from a paper, please include the reference in your writeup.

4. Model evaluation and selection

   Describe how you performed hyperparameter search. Describe how the model performance is evaluated to select the best hyperparameter.

Grading Rubric

(25 points) Creativity

There are many modelling choices for classification questions like this. You can use logistic regressions, random forests, support vector machines, gradient boosted trees, and neural network, etc. You are encouraged to survey some literature to get some inspiration. For example, in ref [4], the authors applied two separate solvent and solute encoder networks that can quantify structural features of given compounds via word embedding and recurrent layers. The two networks are also coupled with attention mechanism. A graph-based approach is described here.

We also encourage you to apply use PyTorch in your solutions. This will let you build more complex models that are optimizable by gradient descent. For example, you can augment your MLP architecture with an attention mechanism, or play around with how solute and solvent features are combined (e.g., concatenated? added? outer product?). You can also try interesting methods to quantify feature importance. For example, you can apply feature masking to quantify how much it influences your prediction performance.

(30 points) Technical correctness

We will examine your code and text to evaluate your solution on technical correctness and the appropriateness of your chosen methods. Please comment and document your code so that it is easier for us to understand each process from data preprocessing, train/validation/test split, hyperparameter optimization, and cross-validation.

(15 points) Model performance

We will provide a linear regression baseline, if your model performs the same as the logistic baseline, you will automatically get half credit at least. We will grade based the prediction performance between your solutions and the solutions submitted by the teaching team. We may award extra points depending on how strong your performance is.

References

[1] Kroger, L. C., Muller, S., Smirnova, I. & Leonhard, K. Prediction of solvation free energies of ionic solutes in neutral solvents. *The Journal of Physical Chemistry A* **124**, 4171–4181 (2020).

[2] Subramanian, V. *et al.* Multisolvent models for solvation free energy predictions using 3d-rism hydration thermodynamic descriptors. *Journal of chemical information and modeling* **60**, 2977–2988 (2020).

[3] Hille, C. *et al.* Generalized molecular solvation in non-aqueous solutions by a single parameter implicit solvation scheme. *The Journal of chemical physics* **150**, 041710 (2019).

[4] Lim, H. & Jung, Y. Delfos: deep learning model for prediction of solvation free energies in generic organic solvents. *Chemical science* **10**, 8306–8315 (2019).