**Summary updated**

**21st June 2019**

**Objective:** Developing a prediction model to predict Hammett parameters of different ligands attached to Benzoic Acid.

**Database:** Hammett constant values are studied for meta and para isomers of benzoic acid. The database comprises of 996 compounds.

**Quantative Structural Information:**

Molecules can be described in SMILES representation created in IQmol. SMILES give a 1D representation of the molecule. OpenBabel and RDkit tools have been used to get fingerprints and other properties of the compounds from input structural files.

**Model:**

An attempt has been made to predict the dissociation constants of different ligands on benzoic acid using the structural information of the ligand. The following set of input parameters have been used for the same:

1. **Extended Connectivity Fingerprints (ECFP):**

Extended-Connectivity Fingerprints are circular topological fingerprints designed for molecular characterization, similarity searching, and structure-activity modeling. ECFPs are frequently applied in ligand-based virtual screening studies to distinguish between actives and in-actives. The ECFP generation process systematically records the neighborhood of each non-hydrogen atom into multiple circular layers up to a given diameter. These atom-centered sub structural features are then mapped into integer codes using a hashing procedure. It is the set of the resulting identifiers that defines the extended-connectivity fingerprint. A 1024 bit long fingerprint has been used in this model.

1. **Type of isomer:**

A value of 1 or 2 has been assigned for ligands attached to the meta or para positions respectively.

**Random forests (RF):**

Random forests are an ensemble machine learning model based on a number of decision trees, say m trees. Every tree in the forests has an independent subset containing training data which is withdrawn from an original dataset. The way withdrawing the data is called bootstrap resampling. Those subsets are trained in the corresponding trees. Every single tree shares a same set of parameters to ensure that all small learners are essentially the same. A special parameter ‘random\_state = None’ indicates that the process of a tree growing from the parent node which is in the top of the tree to the bottom are different even with the same training set, which means the learning procedures are totally random. From m trees which make a so-called random forests, to each molecule, m predictions are given. These predictions from which the statistic errors are calculated constitute an analytical parity plot which contains the mean predictions and experimental values.

**Bootstrap Resampling:**

Bootstrap resampling is an effective way to make use a limit dataset. The subset is withdrawn randomly form the whole dataset with replacement. With an extremely large number of trials, almost all possible subsets are created and all molecules are trained and tested.

**Statistic errors:**

Since the bootstrapping method is utilized in the model. The error estimation is much more complicate than the usual case. The factor called occurrence is considered. In the Efron’s equation

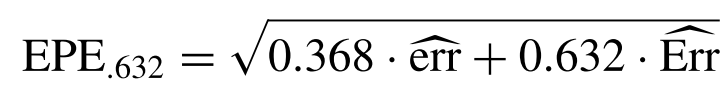
A picture containing object

Description automatically generated

is the testing error (out-of-bag error), which is based on every single sample μ, of a forest. In the equation, are subsets that do not contain μ, the number of subsets that do not contain μ, the number of the sample μ. In the ﻿parentheses, the differences between predictions and target values are calculated.

To get the meaningful prediction error of a forest, the training error are needed which is . For

, the best training model should be found. Considering the special situation in our case, no best model can be found. Then the training error is the mean square error between the mean in-bag predictions and target values. Mean in-bag predictions are given by the predictions of molecules which are included in the training sets. The true procedures are 1) To a sample α, we find the predictions of it from models that used α as training data 2) Calculate the mean valued of these in-bag predictions 3) Calculate the difference between the mean value and target value 4) Sum the squared differences up across all αs and take the average. Then the estimate error of a forest is given by

**

Explanation of the code:

1) Load the original data containing the ecfp and isomer and target sigma values

2) The same procedure for ndonors without target value.

3) Set up the range of depths and the number of trees, say N, for the model.

4) For a certain depth, say m, N corresponding bootstrapping subsamples are created.

5) To the nth subsample, first, the model would find all the error-complexity values which are alphas starting from 0. According to ascending alphas, the trees are getting pruned and becoming smaller ending up with a single parent node which is in the top.

6)Save these alphas, assume Q alphas are given and create machine learning models using all alphas.

7) Use every model to fit the nth bootstrapping subsample. Predict the training data and test data. Get the estimate error using 0.632/0.368 rule. (still need to think about whether to take the repeated training data into account)

8) From all models, select the best model which has the lowest error.

9) After using all N subsamples, N best models are given. Then those best trees constitute the random forests.

10) Use all trees based the best alphas to predict the complete dataset. Take the average as the final prediction.

11)Calculate the error based on the procedures in ‘**statistic error**’

12) Plot it out

13) Run it over again for (m+1)th depth