ExCAPE-ML

Nested Cross Fold Validation Workflow

Contents

Funded by the European Union's Horizon 2020 Research and Innovation program.
Grant Agreement no. 671555

DATASET SPLIT

The dataset was split based on a structural clustering (**Fig. 1**). The structural clustering is a sphere exclusion algorithm which was run with binary ECFP (r=3) folded into 2048 bits. Resulting clusters were randomly distributed into three folds. This process was repeated until we found three folds containing data points for each of the 526 target proteins contained in the dataset.

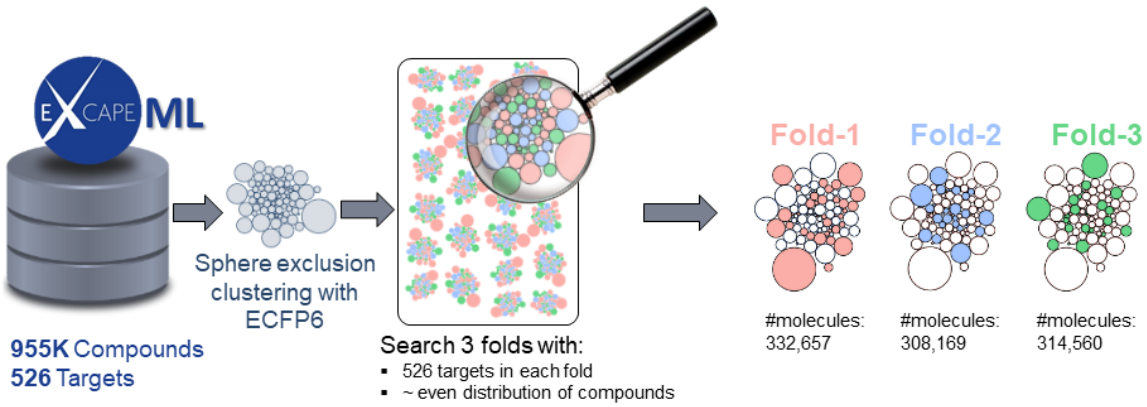

Figure 1. Data split performed on ExCAPE-ML dataset.

FOLD FILES

The ExCAPE-ML dataset is provided as three separate files, each of them represented one of the folds. Each fold contains around 300K molecules and all their protein activity annotations (**Fig. 2**).

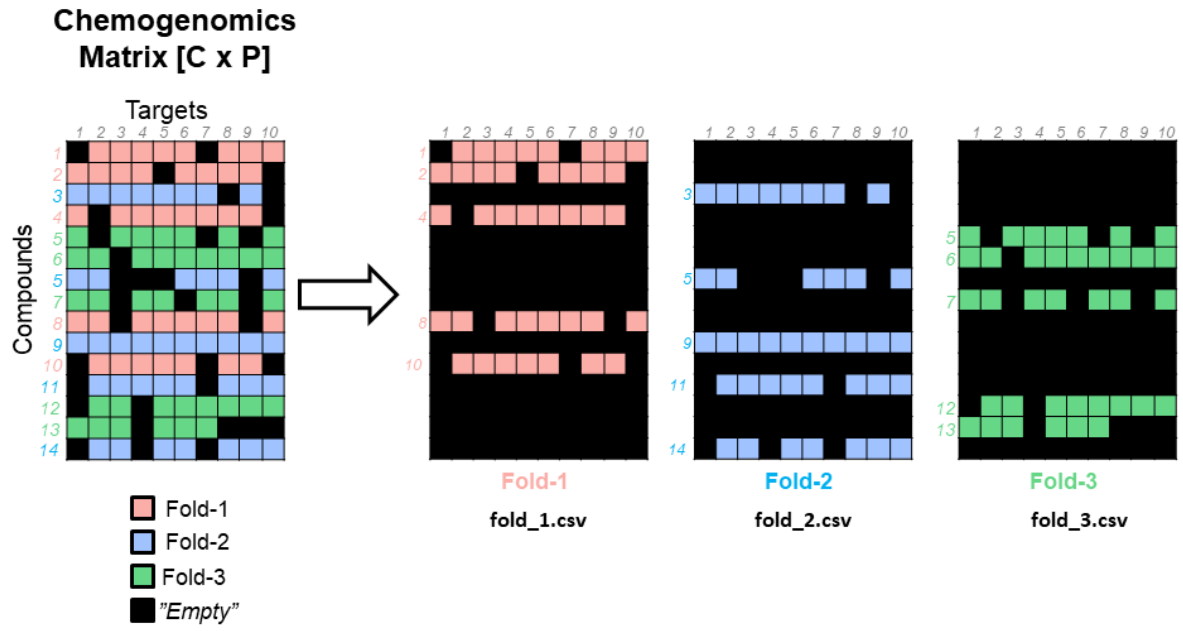

Figure 2. Scheme representing the dataset split in terms of chemogenomics matrix. Numbers are purely illustrative.

Each record in the fold files contain (compound, pXC50, target) triplets where compound is represented by InchI-key and SMILES strings. Records are ordered by target.

Example:

excapeml_index	compound	target	activity	SMILES
3981	ACPOUJIDANTYHO-YAQRNVERNA-N	ABL2	5.600	O=C1C=2C=3C(=NNC3C=CC2)C=4C1=CC=CC4
11057	AHDFVTJYNGXCEO-WVRSUYCFNA-N	ABL2	6.450	S(=O)(=O)(CC1=CC=C(NC=2N=C(N(C3=CC=C(NC(=O)NC4...
43059	BCFGMOOMADDAQU-CSKMVECVNA-N	ABL2	3.101	ClC=1C=C(NC2=NC=NC3=C2C=C(C=4OC(CNCCS(=O)(=O)C...
97314	CMKMGFAUKPAOMG-CQSZACIVNA-N	ABL2	5.600	CC=1C=NC=C2C=CC=C(C12)S(N3CCCN(C[C@H]3C)C(CN)=...
140239	DOKIBWCVABZHOH-LELJVTLKNA-N	ABL2	6.270	S(=O)(=O)(C=1C=C(NC=2OC(C=3C=C(C=4C(F)=CC=CC4)...

excapeml_index	Index of the table in the global chemogenomics matrix.
compound	InChI-key.
target	Target official gene symbol (only from human, mouse, rat).
activity	pXC50 = -log(activity), where log is the natural log; i.e. pXC50=6 => activity = 1μM.
SMILES	Standardized (ambit) SMILES string

NESTED CROSS FOLD VALIDATION WORKFLOW

The nested cross validation workflow is carried out similarly to a normal cross validation loop. In each iteration of the loop, one fold is left out (Fold-1 on **Fig.3**) and the rest of the data is used for training (Fold-2 and Fold-3 on **Fig.3**). However, in contrast to a normal cross validation, the nested cross validation uses the remaining folds (**Fig. 3**: Fold-2, Fold-3, also called the "inner-fold") to search hyperparameters. The hyperparameter search implies training a model with one inner-fold and validating it performance on the other inner-fold. This process is repeated inversely, which – at each iteration of the loop - results in two performance estimates per hyperparameter (Perf1.1, Perf1.2 on **Fig. 3**).

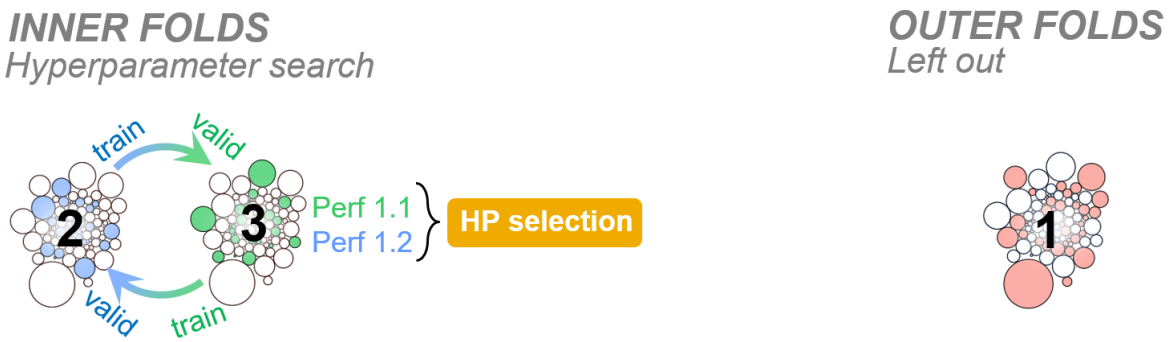

Figure 3. Hyperparameter search with Fold-2 and Fold-3 while Fold-1 is left-out.

Based on the obtained performance, a hyperparameter set can be selected (*e.g.* best mean performance over the two inner-fold validations) and used to train a subsequent model with the two inner-folds as training set (**Fig. 4**, concatenated file not provided). Ultimately, the model quality is evaluated based on its performance achieved when tested on the left-out fold (Fold-1 on **Fig. 4**).

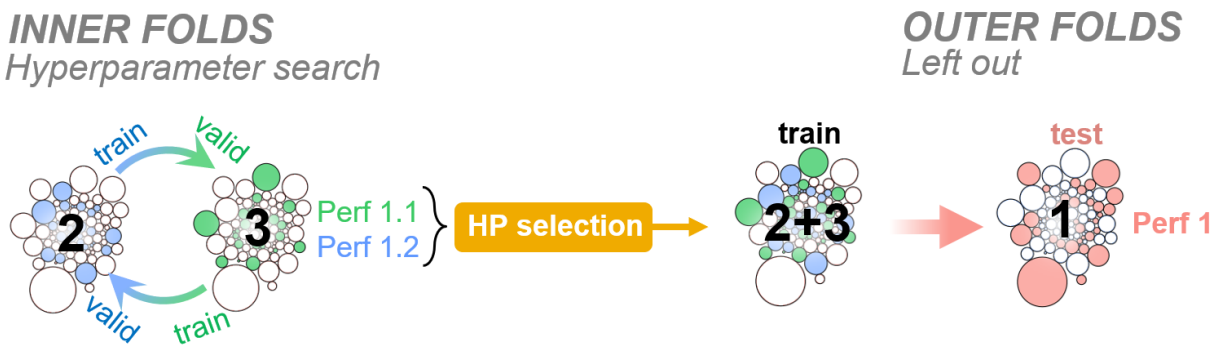

Figure 4. Model testing with Fold-1.

This process has to be repeated until each fold was used as a left-out fold (**Fig. 5**).

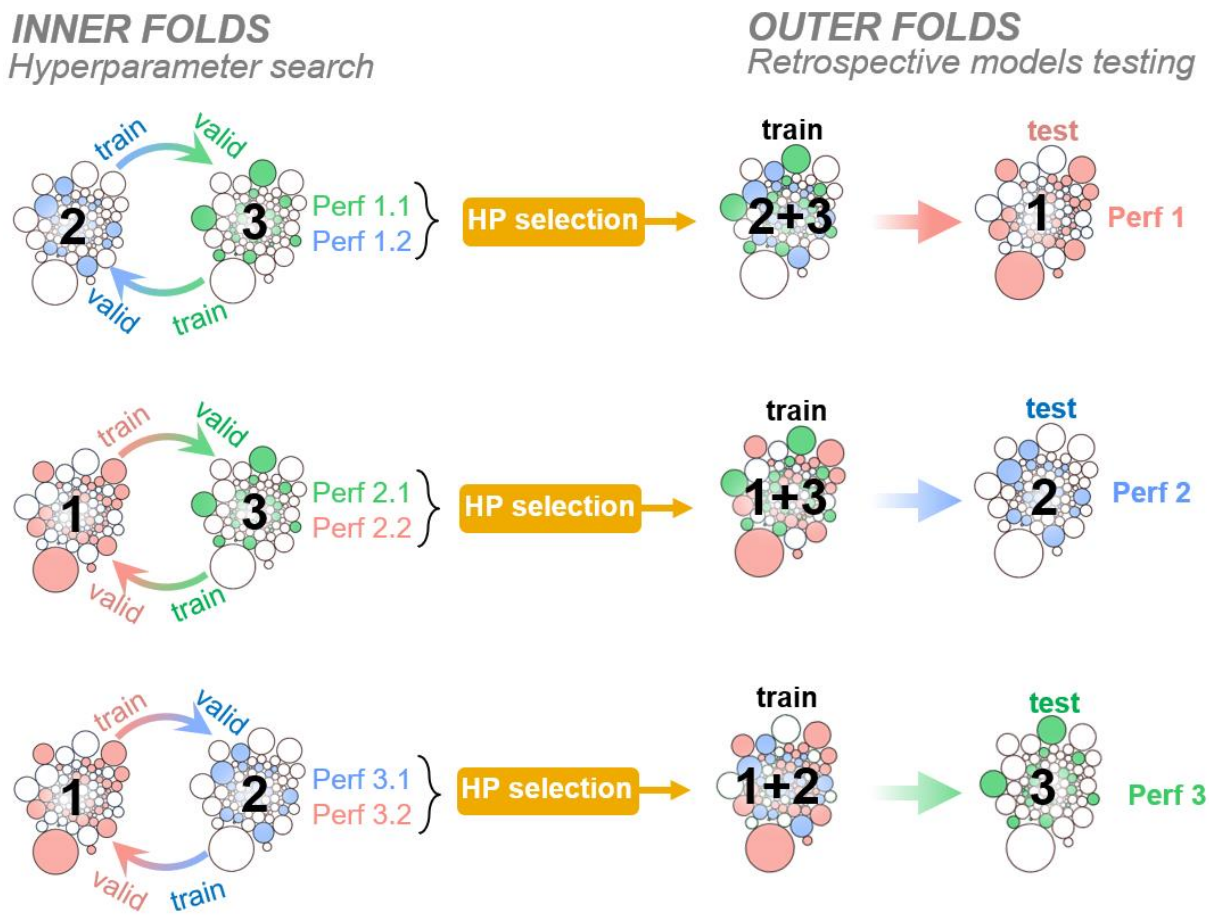

Figure 5. Full nested cross validation workflow scheme