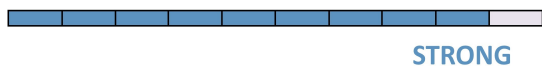# R²OBERT
AUTOMATED ML PROTOCOLS

ROBERT v 1.0.6 2025/02/04 09:00:22

How to cite: Dalmau, D.; Alegre Requena, J. V. ChemRxiv, 2023, DOI: 10.26434/chemrxiv-2023-k994h

ROBERT SCORE

This score is designed to analyze the predictive ability of the models using different metrics.

No PFI (all descriptors):

ML model: MVL
Proportion Train:Validation = 70:30

STRONG

The model has a score of 9/10

- ● The valid. set shows an R^2 of 0.78
- ●● The valid. set has 4.0% of outliers
- ●● Using 166:8 points(train+valid.):descriptors
- ●●●● The valid. set passes 4 VERIFY tests
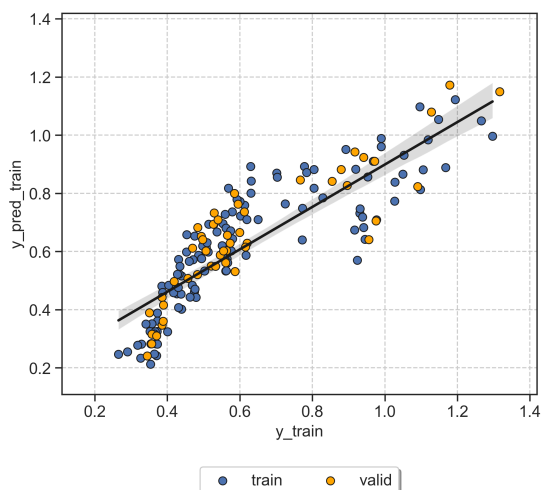
Train : R^2 = 0.73, MAE = 0.11, RMSE = 0.13
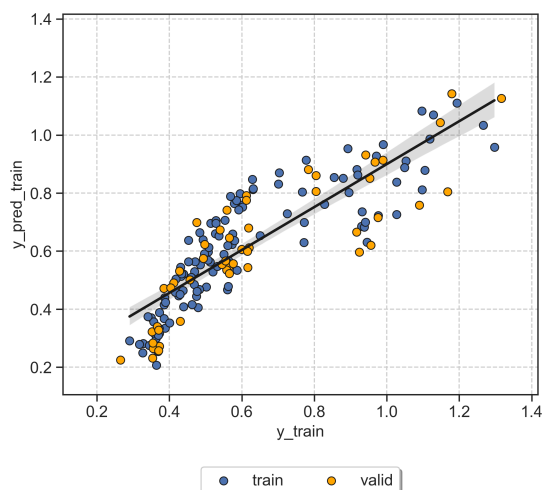Valid. : R^2 = 0.78, MAE = 0.087, RMSE = 0.12

PFI (only important descriptors):

ML model: MVL
Proportion Train:Validation = 70:30

MODERATE

The model has a score of 8/10

- ● The valid. set shows an R^2 of 0.74
- ● The valid. set has 10.0% of outliers
- ●● Using 166:6 points(train+valid.):descriptors
- ●●●● The valid. set passes 4 VERIFY tests

Train : R^2 = 0.74, MAE = 0.098, RMSE = 0.13
Valid. : R^2 = 0.74, MAE = 0.1, RMSE = 0.14

Score thresholds *(detailed in https://robert.readthedocs.io/en/latest/Score/score.html)*

R^2		Outliers		Points:descriptors		VERIFY tests	
●●	$R^2 > 0.85$	●●	< 7.5% of outliers	●●	> 10:1 p:d ratio	Up to ●●●● (tests pass)	
●	$0.85 > R^2 > 0.70$	●	7.5% < outliers < 15%	●	10:1 > p:d ratio > 3:1	- (all tests failed)	
-	$R^2 < 0.70$	-	> 15% of outliers	-	p:d ratio < 3:1		

Some tips to improve the score

✔ A ROBERT score of 9 or 10 suggests that the predictive ability of your model is strong, congratulations!

How to predict new values with these models?

1. Create a CSV database with the new points, including the necessary descriptors.

2. Place the CSV file in the parent folder (i.e., where the module folders were created)

3. Run the PREDICT module as 'python -m robert --predict --csv_test FILENAME.csv'.

4. The predictions will be shown at the end of the resulting PDF report and will be stored in the last column of two CSV files called MODEL_SIZE_test(_No)_PFI.csv, which are in the PREDICT folder.

♻ **REPRODUCIBILITY**

This section provides all the instructions to reproduce the results presented.

1. Download these files *(the authors should have uploaded the files as supporting information!)*:

   - CSV database (smooth_sin_rGO_sin_DMF_clima_noise_ratio_m16.csv)

2. Install and adjust the versions of the following Python modules:

   - Install ROBERT and its dependencies: conda install -c conda-forge robert

   - Adjust ROBERT version: pip install robert==1.0.6

   - scikit-learn-intelex: not installed

   (if scikit-learn-intelex is installed, slightly different results might be obtained)

3. Run ROBERT using this command line in the folder with the CSV database:

python -m robert --model "[MVL]" --train "[70]" --ignore "[indice,area,cell]" --names "indice" --y "PCE" --auto_test
"False" --csv_name "smooth_sin_rGO_sin_DMF_clima_noise_ratio_m16.csv"

4. Execution time, Python version and OS:

Originally run in Python 3.10.12 using Darwin Darwin Kernel Version 22.5.0: Thu Jun 8 22:22:19 PDT 2023; root:xnu-8796.121.3~7/RI

Total execution time: 4.1 seconds *(the number of processors should be specified by the user)*

⊙ TRANSPARENCY

This section contains important parameters used in scikit-learn models and ROBERT.

1. Parameters of the scikit-learn models (same keywords as used in scikit-learn):

No PFI (all descriptors):

sklearn model: LinearRegression

random_state: 233

names: indice

PFI (only important descriptors):

sklearn model: LinearRegression

random_state: 70

names: indice

2. ROBERT options for data split (KN or RND), predict type (REG or CLAS) and hyperopt error (RMSE, etc.):

No PFI (all descriptors):

split: RND

type: reg

error_type: rmse

PFI (only important descriptors):

split: RND

type: reg

error_type: rmse

⊞ ABBREVIATIONS

Reference section for the abbreviations used.

|		
ACC: accuracy	**KN:** k-nearest neighbors	**REG:** Regression
ADAB: AdaBoost	**MAE:** root-mean-square error	**RF:** random forest
CSV: comma separated values	**MCC:** Matthew's correl. coefficient	**RMSE:** root mean square error
CLAS: classification	**ML:** machine learning	**RND:** random
CV: cross-validation	**MVL:** multivariate lineal models	**SHAP:** Shapley additive explanations
F1 score: balanced F-score	**NN:** neural network	**VR:** voting regressor
GB: gradient boosting	**PFI:** permutation feature importance	
GP: gaussian process	**R2:** coefficient of determination	

CURATE

This module takes care of data curation, including filters for correlated descriptors, noise, and duplicates, as well as conversion of categorical descriptors.

The complete output (CURATE_data.dat) and curated database are stored in the CURATE folder.

Time CURATE: 0.31 seconds

**------- Images generated by the CURATE module -------**

Pearson's r heatmap
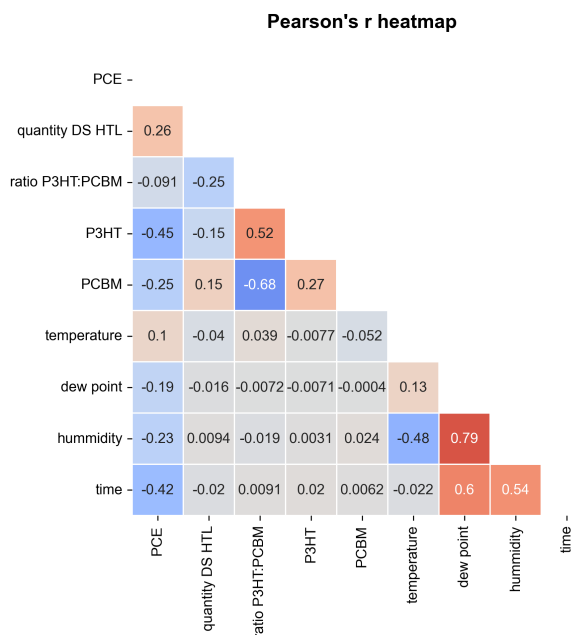
GENERATE

This module carries out a screening of ML models and selects the most accurate one. It includes a comparison of multiple hyperoptimized models and training sizes.

The complete output (GENERATE_data.dat) and heatmaps are stored in the GENERATE folder.

Time GENERATE: 0.52 seconds

------- Images generated by the GENERATE module -------

Heatmap ML models no PFI filter

Heatmap ML models with PFI filter
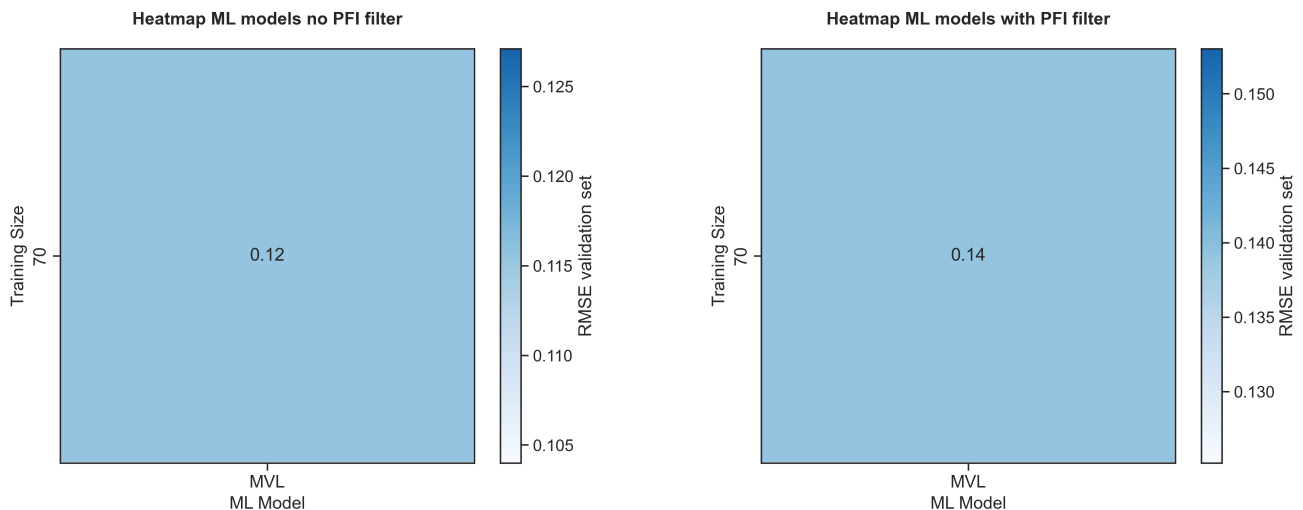
VERIFY

Determination of predictive ability of models using four tests: 5-fold CV, y-mean (error against the mean y baseline), y-shuffle (predict with shuffled y values), and one-hot (predict using one-hot encoding instead of the X values).

The complete output (VERIFY_data.dat) and donut plot are stored in the VERIFY folder.

Time VERIFY: 0.4 seconds

------- **Images and summary generated by the VERIFY module** -------

No PFI (all descriptors):

Original RMSE (valid. set) 0.12 + 25% thres. = 0.14
   o 5-fold CV: PASSED, RMSE = 0.13, lower than thres.
   o y_mean: PASSED, RMSE = 0.24, higher than thres.
   o y_shuffle: PASSED, RMSE = 0.35, higher than thres.
   o onehot: PASSED, RMSE = 0.24, higher than thres.

PFI (only important descriptors):
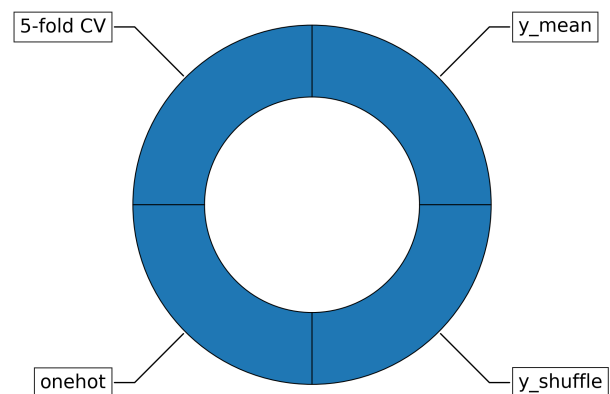
Original RMSE (valid. set) 0.14 + 25% thres. = 0.17
   o 5-fold CV: PASSED, RMSE = 0.13, lower than thres.
   o y_mean: PASSED, RMSE = 0.27, higher than thres.
   o y_shuffle: PASSED, RMSE = 0.4, higher than thres.
   o onehot: PASSED, RMSE = 0.26, higher than thres.

VERIFY tests of MVL_70_No_PFI
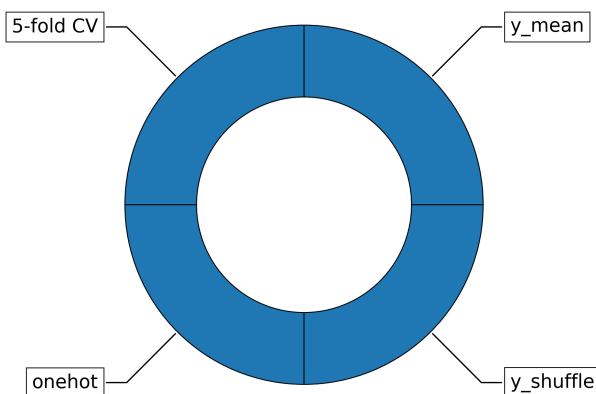
VERIFY tests of MVL_70_PFI

🖥 PREDICT

This module predicts and plots the results of training and validation sets from GENERATE, as well as from external test sets (if any). Feature importances from SHAP and PFI, and outlier analysis are also represented.

The complete output (PREDICT_data.dat) and heatmaps are stored in the PREDICT folder.

Time PREDICT: 2.87 seconds

------- Images and summary generated by the PREDICT module -------

No PFI (all descriptors):

Prediction metrics and descriptors
- Points Train:Validation = 116:50
- Proportion Train:Validation = 70:30
- Number of descriptors = 8
- Proportion (train+valid.) points:descriptors = 166:8
- Train : R^2 = 0.73, MAE = 0.11, RMSE = 0.13
- Valid. : R^2 = 0.78, MAE = 0.087, RMSE = 0.12

Outliers (max. 10 shown)
Train: 5 outliers out of 116 datapoints (4.3%)
- 1 (2.1 SDs)
- 3 (2.4 SDs)
- 11 (2.2 SDs)
- 58 (2.4 SDs)
- 62 (3.1 SDs)
Validation: 2 outliers out of 50 datapoints (4.0%)
- 29 (2.0 SDs)
- 54 (2.6 SDs)

PFI (only important descriptors):

Prediction metrics and descriptors
- Points Train:Validation = 116:50
- Proportion Train:Validation = 70:30
- Number of descriptors = 6
- Proportion (train+valid.) points:descriptors = 166:6
- Train : R^2 = 0.74, MAE = 0.098, RMSE = 0.13
- Valid. : R^2 = 0.74, MAE = 0.1, RMSE = 0.14

Outliers (max. 10 shown)
Train: 5 outliers out of 116 datapoints (4.3%)
- 3 (3.1 SDs)
- 8 (2.6 SDs)
- 11 (2.4 SDs)
- 33 (2.0 SDs)
- 58 (2.8 SDs)
Validation: 5 outliers out of 50 datapoints (10.0%)
- 1 (3.4 SDs)
- 5 (3.0 SDs)
- 29 (2.1 SDs)
- 54 (3.0 SDs)
- 62 (2.9 SDs)

Predictions_train_valid of MVL_70_No_PFI

Predictions_train_valid of MVL_70_PFI

SHAP analysis of MVL_70_No_PFI

SHAP analysis of MVL_70_PFI

Permutation feature importances (PFIs) of MVL_70_No_PFI

Permutation feature importances (PFIs) of MVL_70_PFI

Outlier analysis of MVL_70_No_PFI

Outlier analysis of MVL_70_PFI