ROBERT v 1.0.3 2023/08/12 13:47:29

How to cite: ROBERT v 1.0.3, Dalmau, D.; Alegre-Requena, J. V., 2023. https://github.com/jvalegre/robert

ROBERT SCORE

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

No PFI (all descriptors):

ML model: NN
Proportion Train:Validation = 78:22

MODERATE

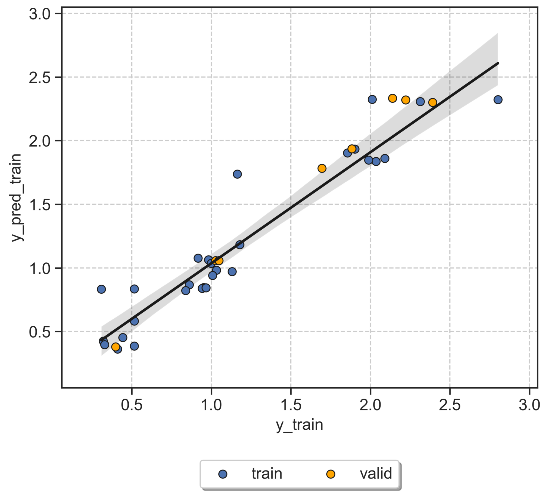
The model has a score of 8/10

●● The valid. set shows an R^2 of 0.99

●● The valid. set has 0.0% of outliers

● Using 37:12 points(train+valid.):descriptors

●●● The valid. set passes 3 VERIFY tests



Train : R^2 = 0.9, MAE = 0.14, RMSE = 0.21
Valid. : R^2 = 0.99, MAE = 0.073, RMSE = 0.092

PFI (only important descriptors):

ML model: NN
Proportion Train:Validation = 78:22

MODERATE

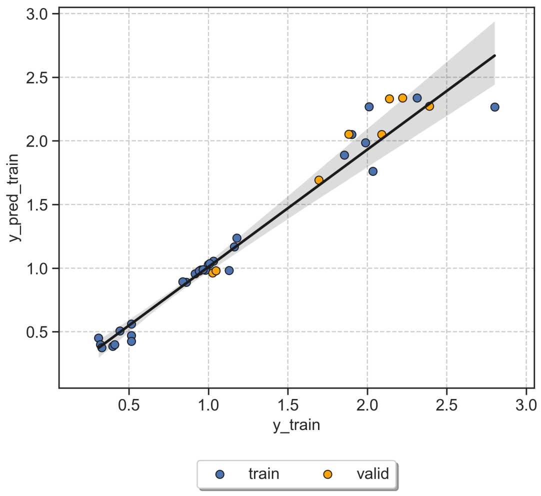
The model has a score of 8/10

●● The valid. set shows an R^2 of 0.96

●● The valid. set has 0.0% of outliers

● Using 37:5 points(train+valid.):descriptors

●●● The valid. set passes 3 VERIFY tests



Train : R^2 = 0.96, MAE = 0.08, RMSE = 0.14
Valid. : R^2 = 0.96, MAE = 0.096, RMSE = 0.11

<u>Score thresholds</u> *(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

⚠ The model uses only 37 datapoints, adding meaningful datapoints might help to improve the model.

⚠ Adding meaningful descriptors or replacing/deleting the least useful descriptors used might help. Feature importances are gathered in the SHAP and PFI sections of the /PREDICT/PREDICT_data.dat file.

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 stored in the last column of two CSV files called MODEL_SIZE_test(_No)_PFI.csv, which are stored 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!)*:

- Report with results (ROBERT_report.pdf)
- CSV database (Robert_example.csv)

2. Install the following Python modules:

- ROBERT: conda install -c conda-forge robert=1.0.3 (or pip install robert==1.0.3)
- scikit-learn-intelex: pip install scikit-learn-intelex==2023.2.1
- To generate the ROBERT_report.pdf summary, the following libraries might be necessary:

  WeasyPrint: pip install weasyprint==59.0

  GLib: conda install -c conda-forge glib

  Pango: conda install -c conda-forge pango

  GTK3: conda install -c conda-forge gtk3

3. Run ROBERT with this command line in the folder with the CSV database (originally run in Python 3.10.12):

```
python -m robert --ignore "[Name]" --names "Name" --y "Target_values" --csv_name "Robert_example.csv"
```

4. Provide number and model of processors used to achieve:

Total execution time: 70.35 seconds

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):	**PFI (only important descriptors):**
sklearn model: MLPRegressor	sklearn model: MLPRegressor
random_state: 19	random_state: 70
batch_size: 32	batch_size: 4
hidden_layer_sizes: [8, 8, 8]	hidden_layer_sizes: [16, 16]
learning_rate_init: 0.001	learning_rate_init: 0.01
max_iter: 200	max_iter: 50
validation_fraction: 0.3	validation_fraction: 0.2
alpha: 0.0001	alpha: 0.0001
shuffle: True	shuffle: True
tol: 0.0001	tol: 0.0001
early_stopping: False	early_stopping: False
beta_1: 0.9	beta_1: 0.9
beta_2: 0.999	beta_2: 0.999
epsilon: 1e-08	epsilon: 1e-08

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

No PFI (all descriptors):	**PFI (only important descriptors):**
split: KN	split: KN
type: reg	type: reg
error_type: rmse	error_type: rmse

ABBREVIATIONS

Reference section for the abbreviations used.

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

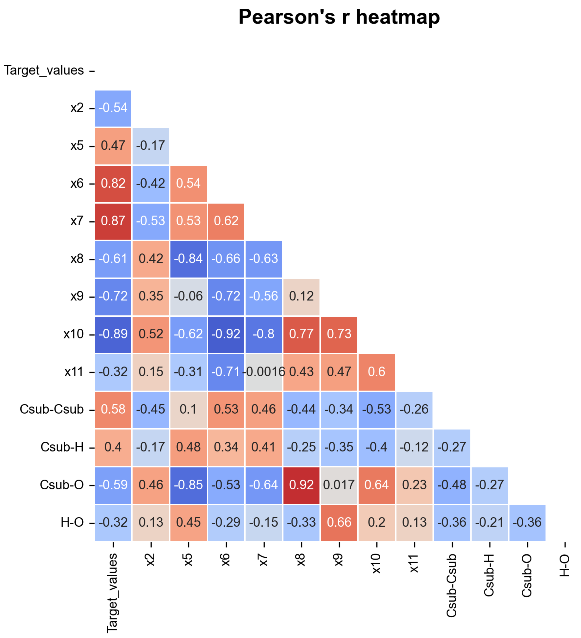
CURATE

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: 0.47 seconds

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



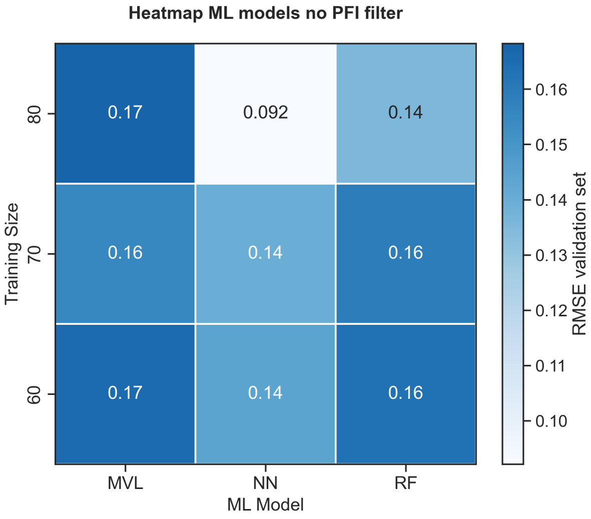
GENERATE

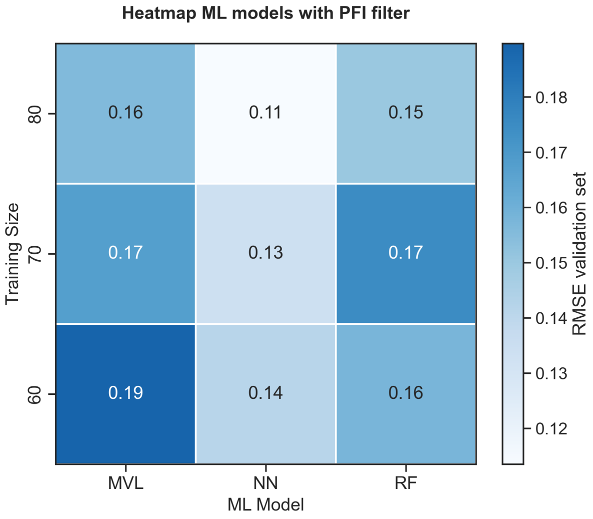
This module carries out a screening of ML models and selects the most accurate one. It includes a comparison of multiple hyperoptimized models and training sizes.

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

Time GENERATE: 62.25 seconds

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





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: 1.71 seconds

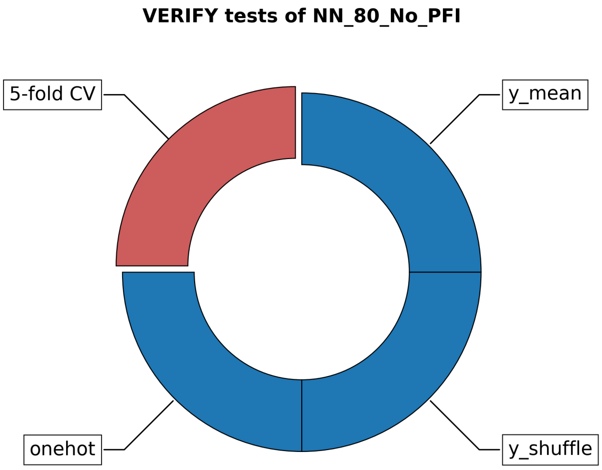
------- Images and summary generated by the VERIFY module -------

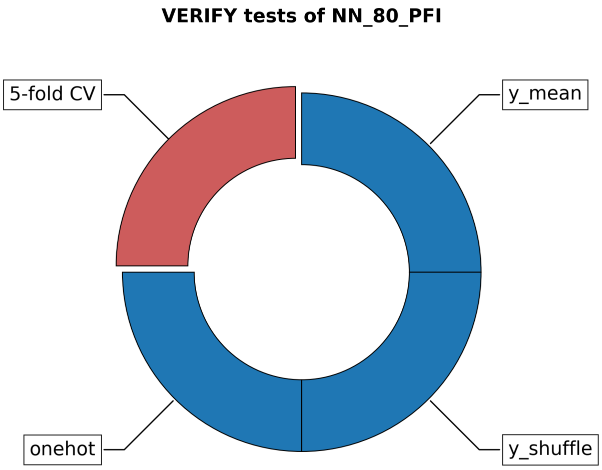
No PFI (all descriptors):

Original RMSE (valid. set) 0.092 + 25% thres. = 0.12
x 5-fold CV: FAILED, RMSE = 0.28, higher than thres.
o y_mean: PASSED, RMSE = 0.66, higher than thres.
o y_shuffle: PASSED, RMSE = 1.1, higher than thres.
o onehot: PASSED, RMSE = 0.2, higher than thres.

PFI (only important descriptors):

Original RMSE (valid. set) 0.11 + 25% thres. = 0.14
x 5-fold CV: FAILED, RMSE = 0.2, higher than thres.
o y_mean: PASSED, RMSE = 0.49, higher than thres.
o y_shuffle: PASSED, RMSE = 0.65, higher than thres.
o onehot: PASSED, RMSE = 0.2, higher than thres.





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: 5.92 seconds

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

No PFI (all descriptors):

Prediction metrics and descriptors

- Points Train:Validation = 29:8
- Proportion Train:Validation = 78:22
- Number of descriptors = 12
- Proportion (train+valid.) points:descriptors = 37:12
- Train : R^2 = 0.9, MAE = 0.14, RMSE = 0.21
- Valid. : R^2 = 0.99, MAE = 0.073, RMSE = 0.092

Outliers (max. 10 shown)

Train: 3 outliers out of 29 datapoints (10.3%)

- 6 (2.5 SDs)
- 19 (2.8 SDs)
- 21 (2.2 SDs)

Validation: 0 outliers out of 8 datapoints (0.0%)

PFI (only important descriptors):

Prediction metrics and descriptors

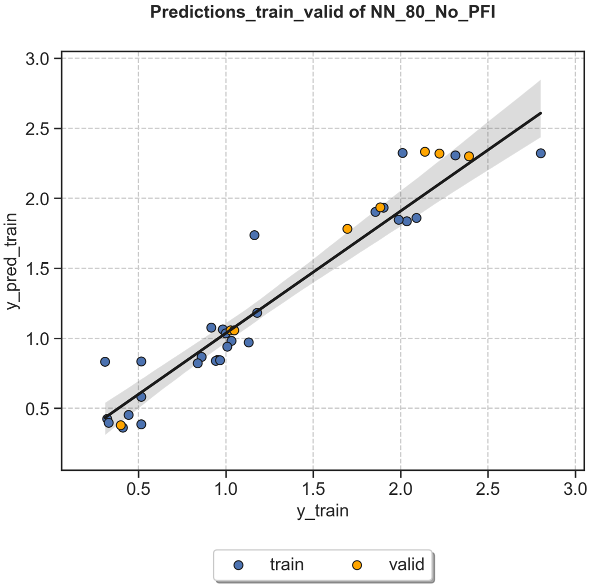
- Points Train:Validation = 29:8
- Proportion Train:Validation = 78:22
- Number of descriptors = 5
- Proportion (train+valid.) points:descriptors = 37:5
- Train : R^2 = 0.96, MAE = 0.08, RMSE = 0.14
- Valid. : R^2 = 0.96, MAE = 0.096, RMSE = 0.11

Outliers (max. 10 shown)

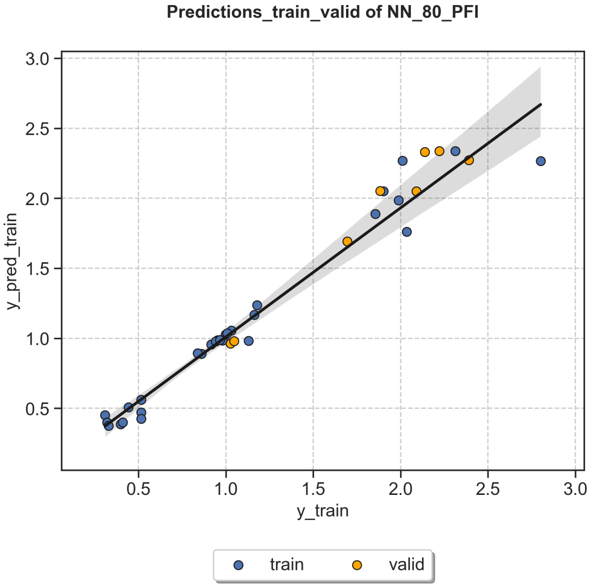
Train: 1 outliers out of 29 datapoints (3.4%)

- 21 (4.2 SDs)

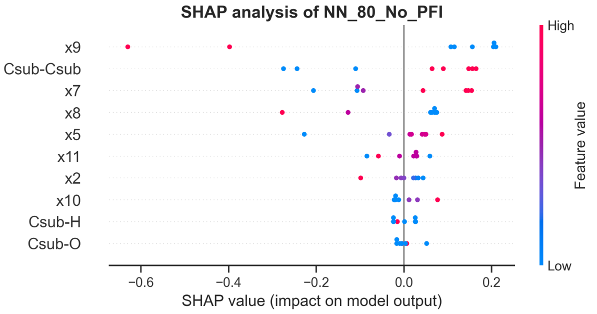
Validation: 0 outliers out of 8 datapoints (0.0%)



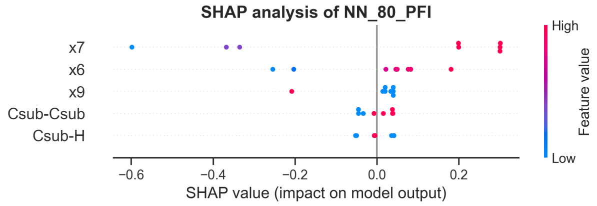
Predictions_train_valid of NN_80_No_PFI



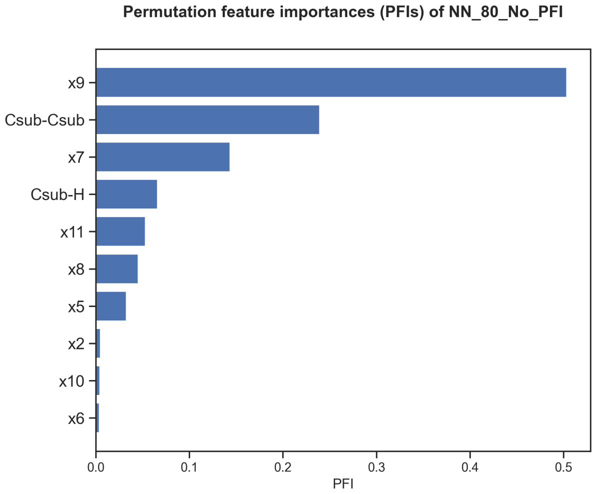
Predictions_train_valid of NN_80_PFI



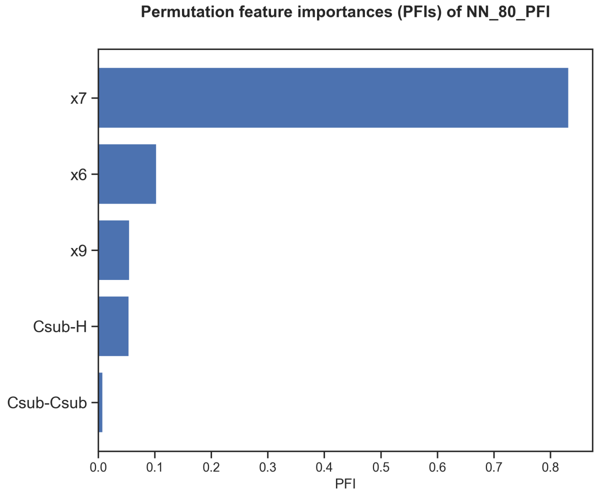
SHAP analysis of NN_80_No_PFI



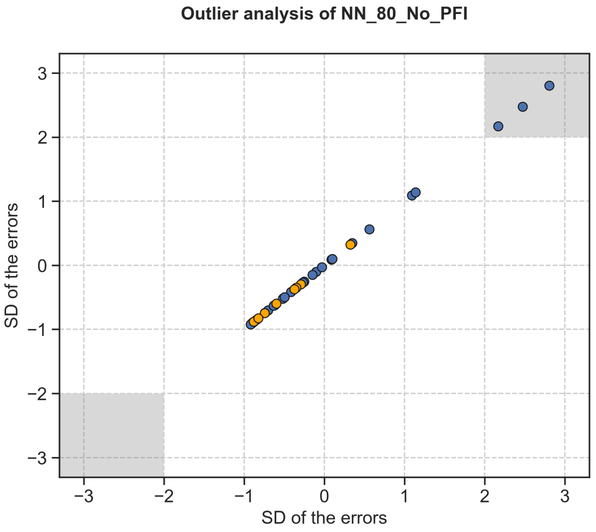
SHAP analysis of NN_80_PFI



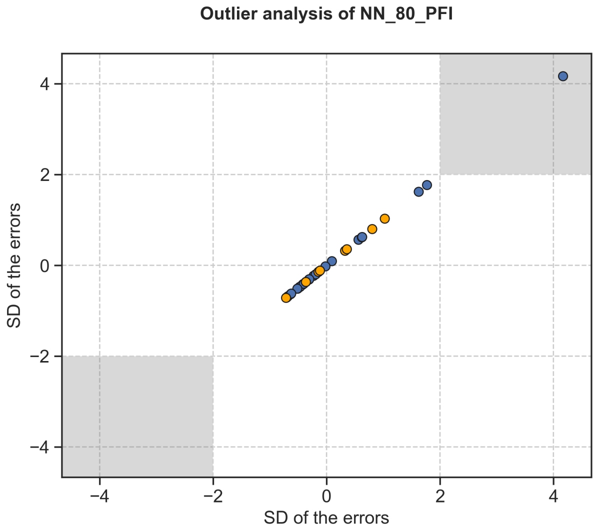
Permutation feature importances (PFIs) of NN_80_No_PFI



Permutation feature importances (PFIs) of NN_80_PFI



Outlier analysis of NN_80_No_PFI



Outlier analysis of NN_80_PFI