

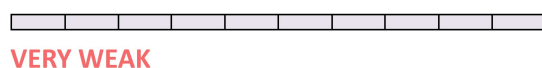


ROBERT v 1.0.4 2023/10/30 18:14:12

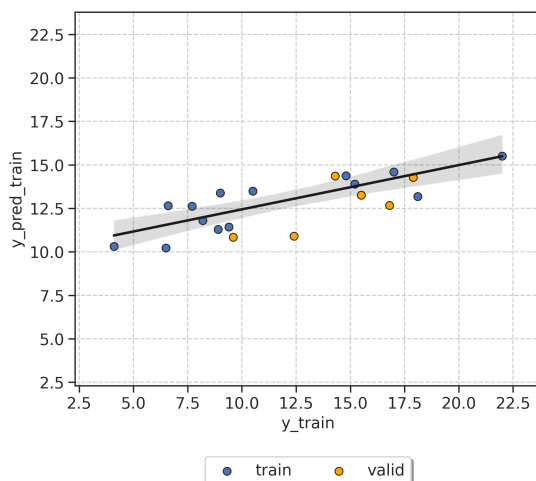
**How to cite:** ROBERT v 1.0.4, Dalmau, D.; Alegre-Requena, J. V., 2023. <https://github.com/jvalegre/robert>**ROBERT SCORE***This score is designed to analyze the predictive ability of the models using different metrics.***No PFI (all descriptors):**

ML model: RF

Proportion Train:Validation = 70:30

**The model has a score of 0/10**

- The valid. set shows an  $R^2$  of 0.6
- The valid. set has 16.7% of outliers
- Using 20:91 points(train+valid.):descriptors
- The valid. set passes 0 VERIFY tests

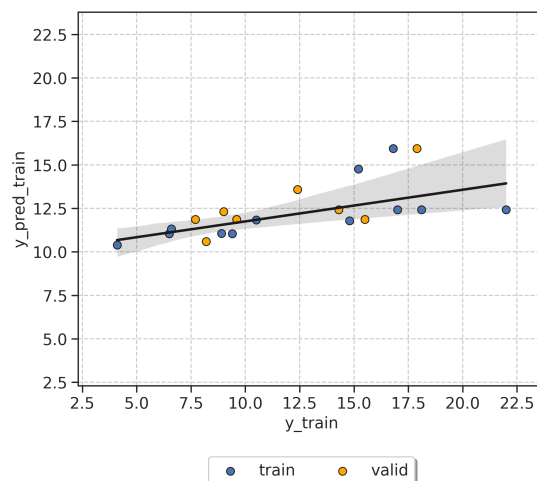
Train :  $R^2 = 0.69$ , MAE = 3.7, RMSE = 4.1Valid. :  $R^2 = 0.6$ , MAE = 2.1, RMSE = 2.5**PFI (only important descriptors):**

ML model: RF

Proportion Train:Validation = 60:40

**The model has a score of 6/10**

- The valid. set shows an  $R^2$  of 0.52
- The valid. set has 0.0% of outliers
- Using 20:3 points(train+valid.):descriptors
- The valid. set passes 3 VERIFY tests

Train :  $R^2 = 0.39$ , MAE = 3.7, RMSE = 4.5Valid. :  $R^2 = 0.52$ , MAE = 2.6, RMSE = 2.8**Score thresholds** (detailed in <https://robert.readthedocs.io/en/latest/Score/score.html>) **$R^2$** 

- $R^2 > 0.85$
- $0.85 > R^2 > 0.70$
- $R^2 < 0.70$

**Outliers**

- $< 7.5\%$  of outliers
- $7.5\% < \text{outliers} < 15\%$
- $> 15\%$  of outliers

**Points:descriptors**

- $> 10:1$  p:d ratio
- $10:1 > \text{p:d ratio} > 3:1$
- p:d ratio  $< 3:1$

**VERIFY tests**

- Up to •••• (tests pass)
- (all tests failed)

### Some tips to improve the score

- △ The model uses only 20 datapoints, adding meaningful datapoints might help to improve the model.
- △ One of your models have more than 7.5% of outliers (5% is expected for a normal distribution with the t-value of 2 that ROBERT uses), using a more homogeneous distribution of results might help.
- △ 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 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 (vaska\_short.csv)

*Warning! SMILES workflows might not be reproduced exactly due to subtle changes in xTB descriptors ( $\pm 0.1\%$ ):*

To obtain the same results, download the descriptor database (AQME-ROBERT\_vaska\_short.csv) and run:

- Run: `python -m robert --csv_name "AQME-ROBERT_vaska_short.csv" --y "barrier" --names "code_name"`

### **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==1.0.4`
- Install scikit-learn-intelex: `pip install scikit-learn-intelex==2023.2.1`

*(if scikit-learn-intelex is not installed, slightly different results might be obtained)*

- Install AQME and its dependencies: `conda install -c conda-forge aqme`
- Adjust AQME version: `pip install aqme==1.5.2`
- Install xTB: `conda install -c conda-forge xtb`
- Adjust xTB version (if possible): `conda install -c conda-forge xtb=6.6.1`

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

```
python -m robert --csv_name "vaska_short.csv" --y "barrier" --aqme --qdescp_keywords "--qdescp_atoms ['Ir']"
```

### **4. Execution time, Python version and OS:**

Originally run in Python 3.10.12 using Linux #1 SMP Tue Dec 21 19:02:23 UTC 2021

Total execution time: 1427.37 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: RandomForestRegressor  
 random\_state: 43  
 names: code\_name  
 n\_estimators: 20  
 max\_depth: 5  
 max\_features: 0.5  
 min\_samples\_split: 2  
 min\_samples\_leaf: 1  
 min\_weight\_fraction\_leaf: 0  
 ccp\_alpha: 0  
 oob\_score: False  
 max\_samples: 0.25

#### PFI (only important descriptors):

sklearn model: RandomForestRegressor  
 random\_state: 0  
 names: code\_name  
 n\_estimators: 5  
 max\_depth: 20  
 max\_features: 1.0  
 min\_samples\_split: 2  
 min\_samples\_leaf: 1  
 min\_weight\_fraction\_leaf: 0  
 ccp\_alpha: 0  
 oob\_score: True  
 max\_samples: 0.25

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

#### No PFI (all descriptors):

split: KN  
 type: reg  
 error\_type: rmse

#### PFI (only important descriptors):

split: KN  
 type: reg  
 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 correl. 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

**AQME**

*This module performs RDKit conformer generation from SMILES, followed by the creation of 200+ molecular and atomic descriptors using RDKit, xTB and DBSTEP (saved as AQME-ROBERT\_FILENAME.csv).*

The complete output (AQME\_data.dat) and raw data are stored in the AQME folder.

Time AQME: 1347.85 seconds

---

**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: 1.52 seconds

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

---

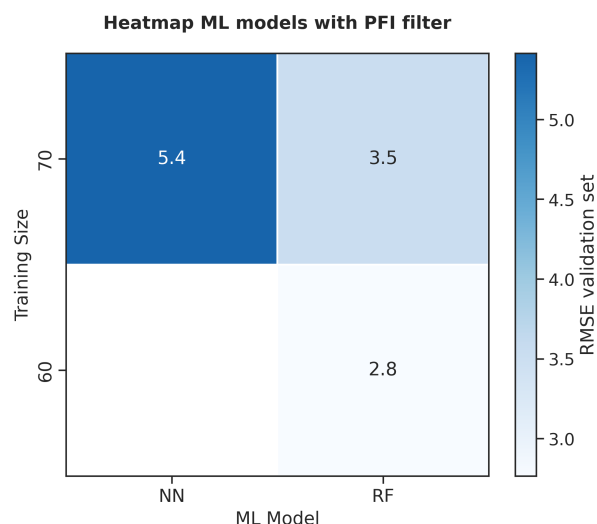
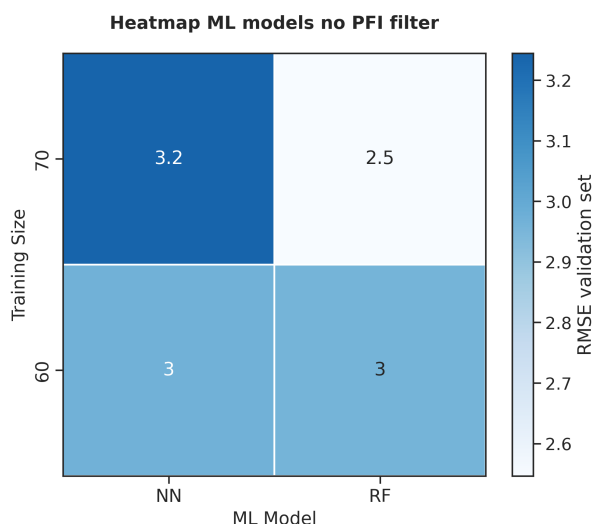
**GENERATE**

*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: 52.43 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: 1.06 seconds

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

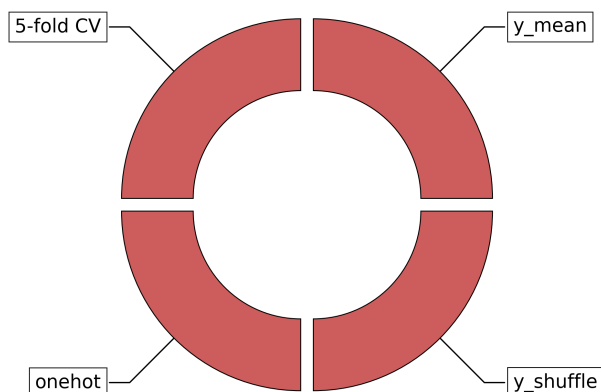
#### No PFI (all descriptors):

Original RMSE (valid. set)  $2.5 + 25\%$  thres. = 3.2  
 x 5-fold CV: FAILED, RMSE = 4.6, higher than thres.  
 x y\_mean: FAILED, RMSE = 2.8, lower than thres.  
 x y\_shuffle: FAILED, RMSE = 3.1, lower than thres.  
 x onehot: FAILED, RMSE = 2.5, lower than thres.

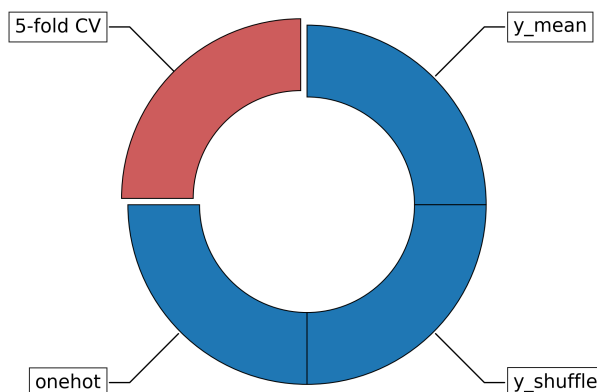
#### PFI (only important descriptors):

Original RMSE (valid. set)  $2.8 + 25\%$  thres. = 3.5  
 x 5-fold CV: FAILED, RMSE = 5.0, higher than thres.  
 o y\_mean: PASSED, RMSE = 3.5, higher than thres.  
 o y\_shuffle: PASSED, RMSE = 4.3, higher than thres.  
 o onehot: PASSED, RMSE = 3.5, higher than thres.

**VERIFY tests of RF\_70\_No\_PFI**



**VERIFY tests of RF\_60\_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: 24.51 seconds

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

#### No PFI (all descriptors):

##### Prediction metrics and descriptors

- Points Train:Validation = 14:6
- Proportion Train:Validation = 70:30
- Number of descriptors = 91
- Proportion (train+valid.) points:descriptors = 20:91
- Train :  $R^2 = 0.69$ , MAE = 3.7, RMSE = 4.1
- Valid. :  $R^2 = 0.6$ , MAE = 2.1, RMSE = 2.5

##### Outliers (max. 10 shown)

Train: 0 outliers out of 14 datapoints (0.0%)

Validation: 1 outliers out of 6 datapoints (16.7%)

- ir\_tbp\_1\_dft-pyz\_1\_dft-sime\_1\_dft-co\_1\_dft-cch\_1\_sm
- i1\_1\_s\_1 (-2.0 SDs)

#### PFI (only important descriptors):

##### Prediction metrics and descriptors

- Points Train:Validation = 12:8
- Proportion Train:Validation = 60:40
- Number of descriptors = 3
- Proportion (train+valid.) points:descriptors = 20:3
- Train :  $R^2 = 0.39$ , MAE = 3.7, RMSE = 4.5
- Valid. :  $R^2 = 0.52$ , MAE = 2.6, RMSE = 2.8

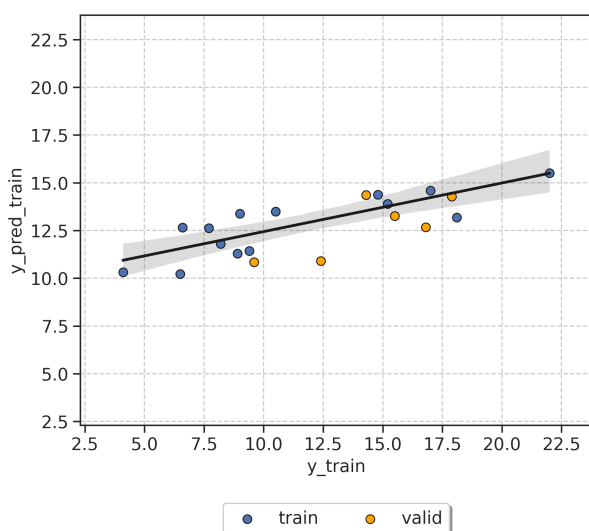
##### Outliers (max. 10 shown)

Train: 1 outliers out of 12 datapoints (8.3%)

- ir\_tbp\_1\_dft-pyz\_1\_dft-pyz\_1\_dft-co\_1\_dft-icn\_1\_smi
- 1\_1\_s\_1 (2.3 SDs)

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

Predictions\_train\_valid of RF\_70\_No\_PFI



Predictions\_train\_valid of RF\_60\_PFI

