ROBERT v 1.2.1 2025/08/05 19:00:12

How to cite: Dalmau, D.; Alegre Requena, J. V. WIREs Comput Mol Sci. 2024, DOI: 10.1002/WCMS.1733

Section A. ROBERT Score

This score is designed to evaluate the models using different metrics.

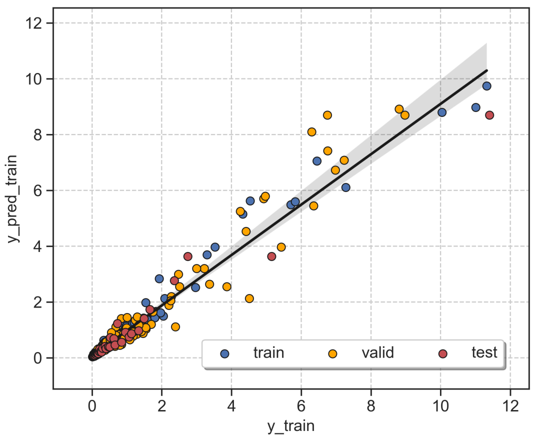
No PFI (standard descriptor filter):

Model = RF · Train:Validation:Test = 54:36:10

Points(train+valid.):descriptors = 330:15

Score = 10 / 10

STRONG



Train : R^2 = 0.98, MAE = 0.12, RMSE = 0.28

Valid. : R^2 = 0.95, MAE = 0.25, RMSE = 0.45

Test : R^2 = 0.96, MAE = 0.24, RMSE = 0.56

Severe warnings

- No severe warnings detected

Moderate warnings

- Uneven y distribution (Section C)
- Potential "faulty" outliers (Section E)

Overall assessment

- The model seems reliable

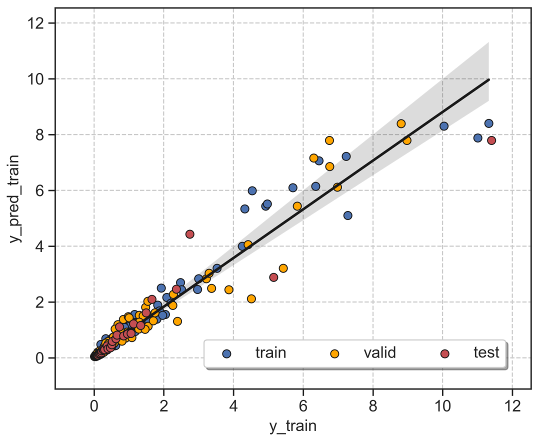
PFI (only most important descriptors):

Model = RF · Train:Validation:Test = 63:27:10

Points(train+valid.):descriptors = 330:4

Score = 10 / 10

STRONG



Train : R^2 = 0.95, MAE = 0.14, RMSE = 0.38

Valid. : R^2 = 0.94, MAE = 0.24, RMSE = 0.47

Test : R^2 = 0.9, MAE = 0.28, RMSE = 0.77

Severe warnings

- No severe warnings detected

Moderate warnings

- Uneven y distribution (Section C)
- Potential "faulty" outliers (Section E)

Overall assessment

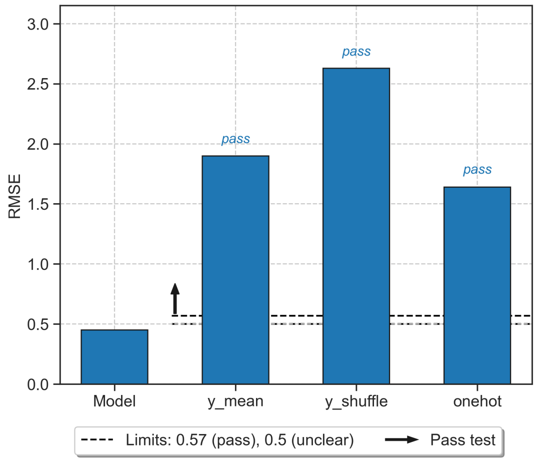
- The model seems reliable

Section B. Advanced Score Analysis

This section explains each component that comprises the ROBERT score.

1. Model vs "flawed" models (3 / 3)

The model predicts right for the right reasons.
Pass: +1, Unclear: 0, Fail: -1. *Details here.*



2. Predictive ability of the model (2 / 2)

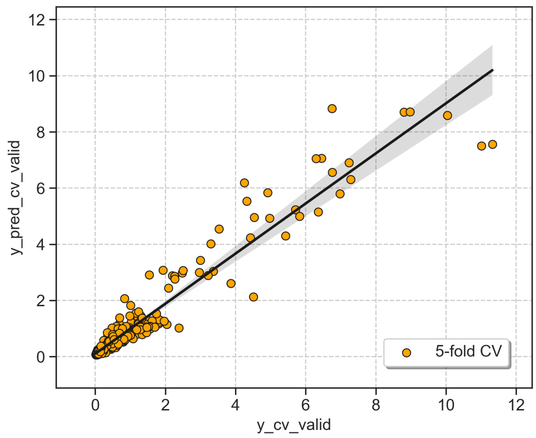
Good predictive ability with R^2 (test) = 0.96.
R^2 0.70-0.85: +1, R^2 >0.85: +2.

3. Cross-validation (5-fold CV) of the model

Overfitting analysis on the model with 3a and 3b:

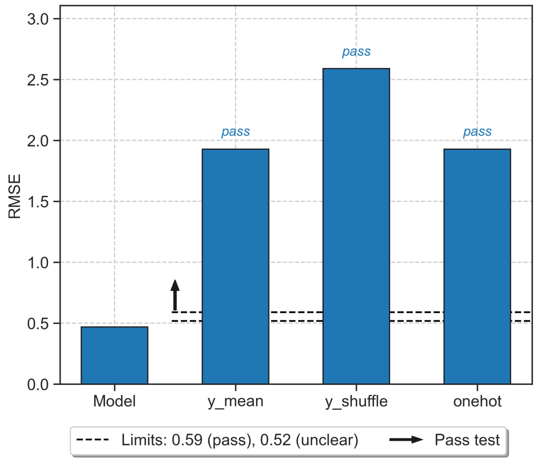
<u>3a. CV predictions train + valid.</u> (2 / 2)

Good predictive ability with R^2 (5-fold CV) = 0.93.
R^2 0.70-0.85: +1, R^2 >0.85: +2.



1. Model vs "flawed" models (3 / 3)

The model predicts right for the right reasons.
Pass: +1, Unclear: 0, Fail: -1. *Details here.*



2. Predictive ability of the model (2 / 2)

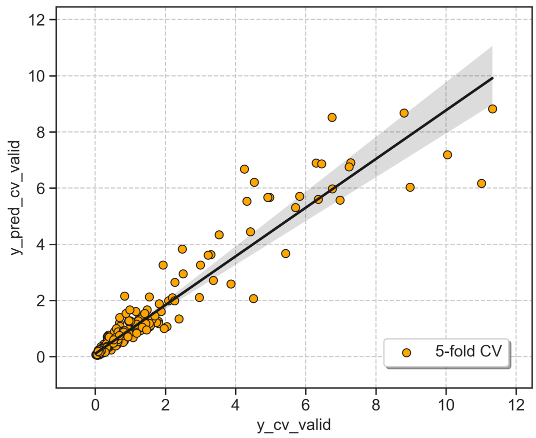
Good predictive ability with R^2 (test) = 0.9.
R^2 0.70-0.85: +1, R^2 >0.85: +2.

3. Cross-validation (5-fold CV) of the model

Overfitting analysis on the model with 3a and 3b:

<u>3a. CV predictions train + valid.</u> (2 / 2)

Good predictive ability with R^2 (5-fold CV) = 0.91.
R^2 0.70-0.85: +1, R^2 >0.85: +2.

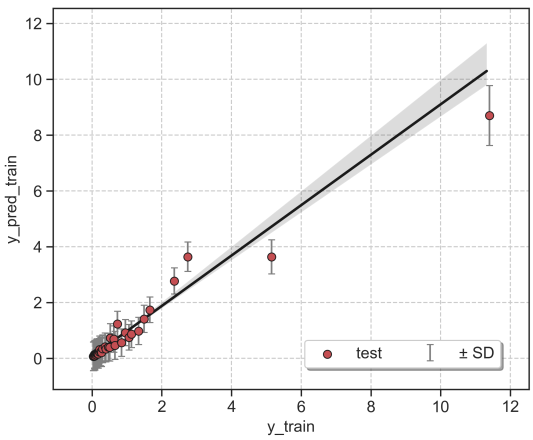


3b. Avg. standard deviation (SD) (2 / 2)

Low variation, 4*SD (test) = 2.0 (18% y-range).

4*SD 25-50% y-range: +1, 4*SD < 25% y-range: +2.

Details here.

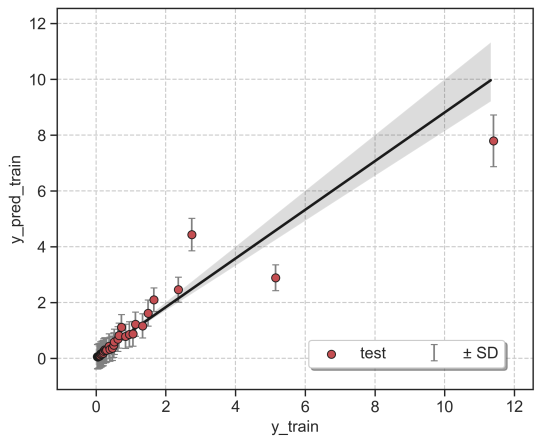


3b. Avg. standard deviation (SD) (2 / 2)

Low variation, 4*SD (test) = 1.8 (16% y-range).

4*SD 25-50% y-range: +1, 4*SD < 25% y-range: +2.

Details here.



4. Points(train+valid.):descriptors (1 / 1)

Decent number of descps. (ratio 330:15).

5 or more points per descriptor: +1.

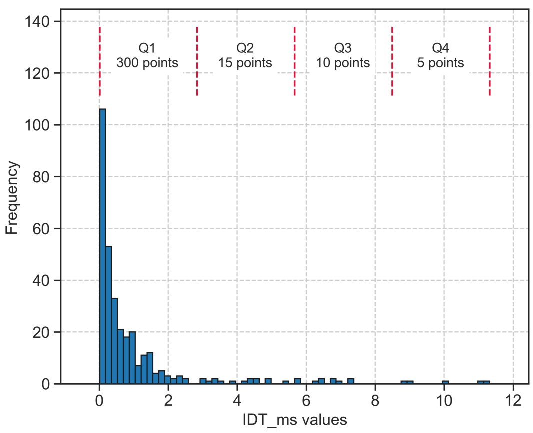
4. Points(train+valid.):descriptors (1 / 1)

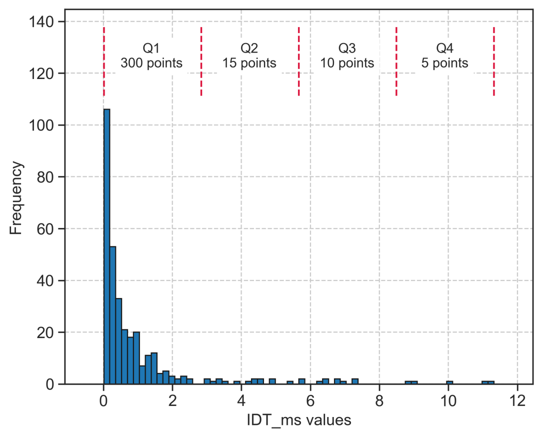
Decent number of descps. (ratio 330:4).

5 or more points per descriptor: +1.

Section C. Distribution of y Values

This section shows the distribution of y values within the training and validation sets.





y distribution analysis

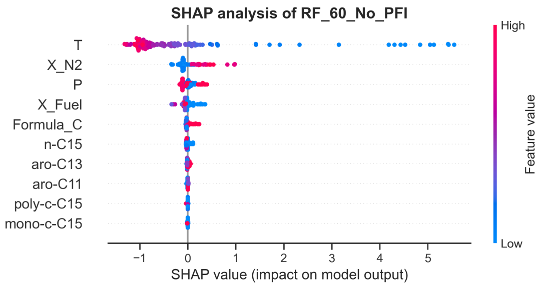
x WARNING! Your data is not uniform (Q4 has 5 points while Q1 has 300)

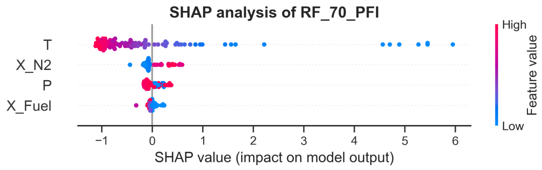
y distribution analysis

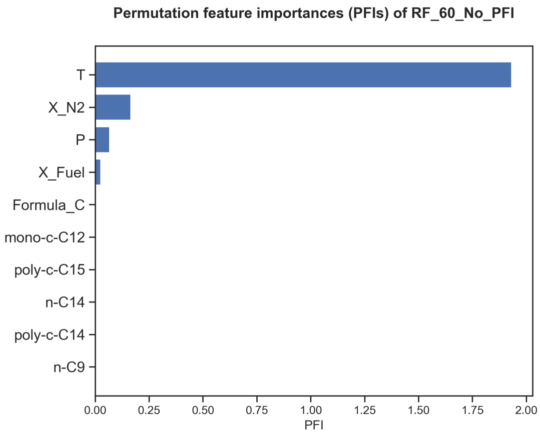
x WARNING! Your data is not uniform (Q4 has 5 points while Q1 has 300)

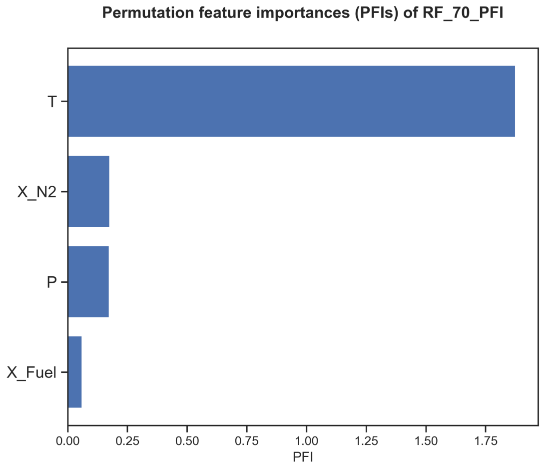
Section D. Feature Importances

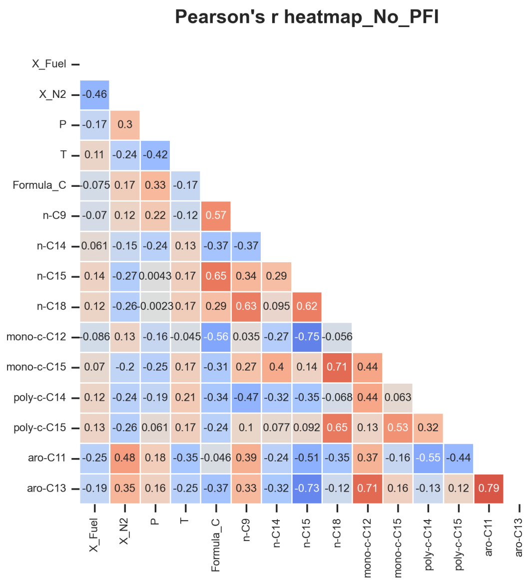
This section presents feature importances measured using the validation set.

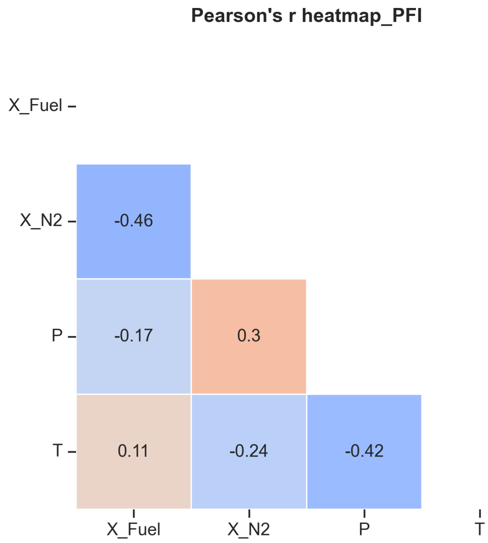












Correlation analysis

o Correlations between variables are acceptable

Correlation analysis

o Correlations between variables are acceptable

Section E. Outlier Analysis

This section detects outliers using the standard deviation (SD) of errors from the training set.

No PFI (standard descriptor filter):

Outliers (max. 10 shown)
Train: 7 outliers out of 198 datapoints (3.5%)
- 308 (3.8 SDs)
- 316 (2.8 SDs)
- 319 (3.1 SDs)
- 324 (4.4 SDs)
- 332 (7.5 SDs)
- 348 (5.7 SDs)
- 349 (4.1 SDs)

Validation: 12 outliers out of 132 datapoints (9.1%)
- 11 (4.6 SDs)
- 298 (2.1 SDs)
- 299 (3.4 SDs)
- 307 (3.1 SDs)
- 315 (2.6 SDs)
- 318 (2.7 SDs)
- 326 (7.2 SDs)
- 327 (6.6 SDs)
- 340 (5.3 SDs)
- 342 (4.7 SDs)

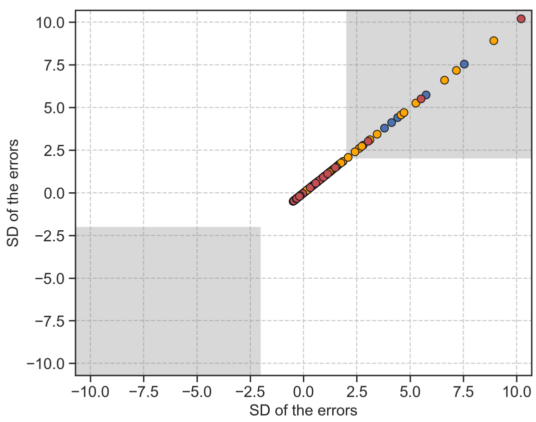
Test: 3 outliers out of 36 datapoints (8.3%)
- 27 (3.0 SDs)
- 351 (5.5 SDs)
- 341 (1e+01 SDs)

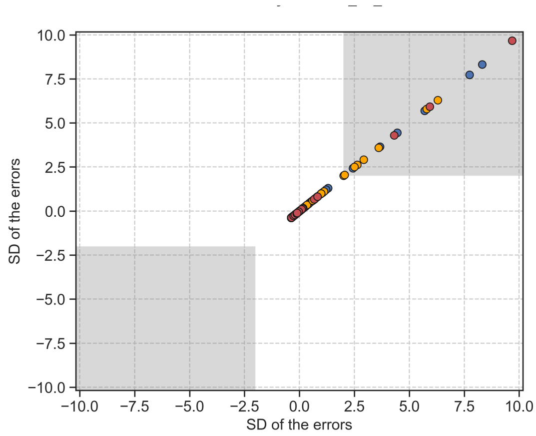
PFI (only most important descriptors):

Outliers (max. 10 shown)
Train: 6 outliers out of 230 datapoints (2.6%)
- 308 (3.7 SDs)
- 316 (2.4 SDs)
- 324 (4.4 SDs)
- 332 (8.3 SDs)
- 348 (7.7 SDs)
- 349 (5.7 SDs)

Validation: 9 outliers out of 100 datapoints (9.0%)
- 11 (2.6 SDs)
- 289 (2.0 SDs)
- 326 (2.5 SDs)
- 327 (2.0 SDs)
- 333 (2.9 SDs)
- 340 (5.8 SDs)
- 342 (3.6 SDs)
- 350 (2.1 SDs)
- 361 (6.3 SDs)

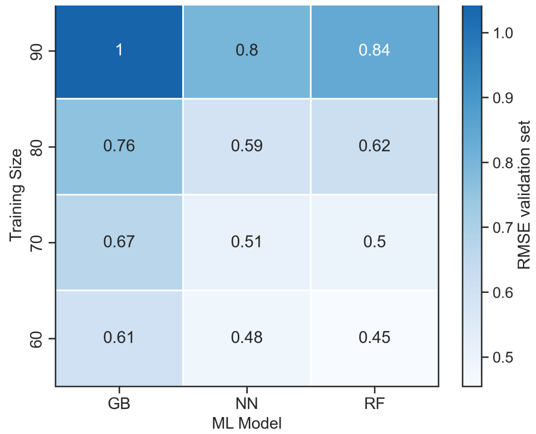
Test: 3 outliers out of 36 datapoints (8.3%)
- 27 (4.3 SDs)
- 351 (5.9 SDs)
- 341 (9.7 SDs)

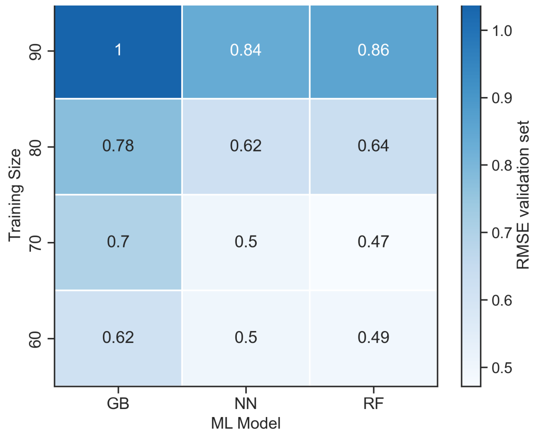




Section F. Model Screening

This section compares different combinations of hyperoptimized algorithms and partition sizes.





Section G. 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 (Ignition_2D_GC_ML.csv)

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

- Install ROBERT and its dependencies: conda install -y -c conda-forge robert
- Adjust ROBERT version: pip install robert==1.2.1
- Install scikit-learn-intelex: pip install scikit-learn-intelex==2024.7.0

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

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

python -m robert --names "Point" --y "IDT_ms" --model "[RF,GB,NN]" --csv_name "Ignition_2D_GC_ML.csv"

4. Execution time, Python version and OS:

Originally run in Python 3.12.2 using Linux #1 SMP Fri Apr 20 16:44:24 UTC 2018

Total execution time: 191.11 seconds *(the number of processors should be specified by the user)*

Section H. 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 (standard descriptor filter):

sklearn model: RandomForestRegressor
random_state: 233
names: Point
n_estimators: 40
max_depth: 60
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: 0.75

PFI (only most important descriptors):

sklearn model: RandomForestRegressor
random_state: 0
names: Point
n_estimators: 40
max_depth: 20
max_features: 0.75
min_samples_split: 2
min_samples_leaf: 1
min_weight_fraction_leaf: 0
ccp_alpha: 0
oob_score: False
max_samples: 0.5

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

No PFI (standard descriptor filter):

split: KN
type: reg
error_type: rmse

PFI (only most important descriptors):

split: KN
type: reg
error_type: rmse

Section I. 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

Miscellaneous

General tips to improve the models and instructions to predict new values.

Some general tips to improve the score

1. Adding meaningful datapoints might help to improve the model. Also, using a uniform population of datapoints across the whole range of y values usually helps to obtain reliable predictions across the whole range. More information about the range of y values used is available in Section C.

2. Adding meaningful descriptors or replacing/deleting the least useful descriptors used might help. Feature importances are gathered in Section D.

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.