
km documentation

Release 0.1.0

Krzysztof Joachimiak

Jul 19, 2019

Contents

1	Installation	3
2	Usage	5
2.1	kaggle-metrics API	5
2.1.1	Classification	5
2.1.2	Regression	6
2.1.3	Order-based	6
2.1.4	Other	6
	Python Module Index	15
	Index	17

Metrics for Kaggle competitions.

CHAPTER 1

Installation

You can install this module directly from GitHub repo with command:
or as a PyPI package


```

from xgboost import XGBRegressor
import kaggle_metrics as km

X_train, y_train, X_test, y_test = get_data()

# Train
clf = XGBRegressor()
clf.fit(X_train, y_train)

# Get predictions
y_pred = clf.predict(X_test)

# Evaluate with kaggle-metrics
km.rmse(y_test, y_pred)

```

Contents:

2.1 kaggle-metrics API

2.1.1 Classification

<code>log_loss(y_true, y_pred)</code>	Logarithmic loss
<code>mean_consequential_error(y_true, y_pred)</code>	Mean consequential error
<code>hamming_loss(y_true, y_pred)</code>	Hamming loss
<code>mean_utility(y_true, y_pred, weights)</code>	Mean utility
<code>matthews_correlation_coefficient(y_true, y_pred)</code>	Matthews Correlation Coefficient
<code>roc_auc(y_true, y_pred[, jump])</code>	Area under ROC (Receiver Operating Characteristics) curve

Continued on next page

Table 1 – continued from previous page

<code>gini(y_true, y_pred)</code>	Gini
-----------------------------------	------

2.1.2 Regression

<code>root_mean_squared_error(y_true, y_pred)</code>	Root mean squared error.
<code>root_mean_squared_logarithmic_error(y_true, y_pred, ...)</code>	Root mean squared logarithmic error.
<code>mean_absolute_error(y_true, y_pred)</code>	Mean absolute error.
<code>weighted_mean_absolute_error(y_true, y_pred, ...)</code>	Weighted mean absolute error.
<code>mean_absolute_percentage_error(y_true, y_pred)</code>	Mean absolute percentage error
<code>mean_percentage_error(y_true, y_pred)</code>	Mean percentage error
<code>mean_absolute_percentage_deviation(y_true, y_pred, ...)</code>	Mean absolute percentage error

2.1.3 Order-based

<code>average_precision_at_k(true_positive)</code>	Average precision at position k
<code>average_precision(true_positive)</code>	Average precision
<code>mean_average_precision(true_positive)</code>	Mean average precision

2.1.4 Other

<code>intersection_over_union(y_true, y_pred)</code>	Intersection over union
--	-------------------------

`kaggle_metrics.log_loss(y_true, y_pred)`
 Logarithmic loss

Parameters

- **y_true** (`numpy.ndarray`) – Targets
- **y_pred** (`numpy.ndarray`) – Class probability

Returns score – Logarithmic loss score

Return type float

References

`kaggle_metrics.mean_consequential_error(y_true, y_pred)`
 Mean consequential error

Parameters

- **y_true** (`numpy.ndarray`) – Targets
- **y_pred** (`numpy.ndarray`) – Class predictions (0 or 1 values only)

Returns score – Mean consequential error score

Return type float

References

Notes

The higher the better.

`kaggle_metrics.hamming_loss(y_true, y_pred)`

Hamming loss

Parameters

- **y_true** (*numpy.ndarray*) – Targets
- **y_pred** (*numpy.ndarray*) – Class predictions (0 or 1 values only)

Returns **score** – Hamming loss score

Return type float

References

Notes

The smaller the better.

`kaggle_metrics.mean_utility(y_true, y_pred, weights)`

Mean utility

Parameters

- **y_true** (*numpy.ndarray*) – Targets
- **y_pred** (*numpy.ndarray*) – Class predictions (0 or 1 values only)

Returns **score** – Mean utility score

Return type float

References

Notes

The higher the better.

`kaggle_metrics.matthews_correlation_coefficient(y_true, y_pred)`

Matthews Correlation Coefficient

Parameters

- **y_true** (*numpy.ndarray*) – Targets
- **y_pred** (*numpy.ndarray*) – Class predictions (0 or 1 values only)

Returns **score** – Matthews Correlation Coefficient score

Return type float

References

`kaggle_metrics.roc_auc(y_true, y_pred, jump=0.01)`
Area under ROC (Receiver Operating Characteristics) curve

Parameters

- **y_true** (*numpy.ndarray*) – Targets
- **y_pred** (*numpy.ndarray*) – Class probability

References

Returns roc_auc_score – ROC AUC score

Return type float

`kaggle_metrics.gini(y_true, y_pred)`
Gini

Parameters

- **y_true** (*numpy.ndarray*) – Targets
- **y_pred** (*numpy.ndarray*) – Class probability

Returns gini_score – Gini score

Return type float

References

`kaggle_metrics.mce(y_true, y_pred)`
Mean consequential error

Parameters

- **y_true** (*numpy.ndarray*) – Targets
- **y_pred** (*numpy.ndarray*) – Class predictions (0 or 1 values only)

Returns score – Mean consequential error score

Return type float

References

Notes

The higher the better.

`kaggle_metrics.mcc(y_true, y_pred)`
Matthews Correlation Coefficient

Parameters

- **y_true** (*numpy.ndarray*) – Targets
- **y_pred** (*numpy.ndarray*) – Class predictions (0 or 1 values only)

Returns score – Matthews Correlation Coefficient score

Return type float

References

`kaggle_metrics.root_mean_squared_error(y_true, y_pred)`

Root mean squared error.

Parameters

- **y_true** (*ndarray*) – Ground truth
- **y_pred** (*ndarray*) – Array of predictions

Returns `rmsle` – Root mean squared error

Return type float

References

`kaggle_metrics.root_mean_squared_logarithmic_error(y_true, y_pred)`

Root mean squared logarithmic error.

Parameters

- **y_true** (*ndarray*) – Ground truth
- **y_pred** (*ndarray*) – Array of predictions

Returns `rmsle` – Root mean squared logarithmic error

Return type float

References

`kaggle_metrics.mean_absolute_error(y_true, y_pred)`

Mean absolute error.

Parameters

- **y_true** (*ndarray*) – Ground truth
- **y_pred** (*ndarray*) – Array of predictions

Returns `rmsle` – Mean absolute error

Return type float

References

`kaggle_metrics.weighted_mean_absolute_error(y_true, y_pred, weights)`

Weighted mean absolute error.

Parameters

- **y_true** (*ndarray*) – Ground truth
- **y_pred** (*ndarray*) – Array of predictions

Returns `rmsle` – Weighted mean absolute error

Return type float

References

`kaggle_metrics.mean_absolute_percentage_error` (*y_true*, *y_pred*)
Mean absolute percentage error

Parameters

- **y_true** (*ndarray*) –
- **truth** (*Ground*) –
- **y_pred** (*ndarray*) –
- **of predictions** (*Array*) –

Returns `mean_absolute_percentage_error` – Mean absolute percentage error

Return type float

References

`kaggle_metrics.mean_percentage_error` (*y_true*, *y_pred*)
Mean percentage error

Parameters

- **y_true** (*ndarray*) – Ground truth
- **y_pred** (*ndarray*) – Array of predictions

Returns `mean_percentage_error` – Mean percentage error

Return type float

References

`kaggle_metrics.mean_absolute_percentage_deviation` (*y_true*, *y_pred*)
Mean absolute percentage error

Parameters

- **y_true** (*ndarray*) –
- **truth** (*Ground*) –
- **y_pred** (*ndarray*) –
- **of predictions** (*Array*) –

Returns `mean_absolute_percentage_error` – Mean absolute percentage error

Return type float

References

`kaggle_metrics.rmse` (*y_true*, *y_pred*)
Root mean squared error.

Parameters

- **y_true** (*ndarray*) – Ground truth
- **y_pred** (*ndarray*) – Array of predictions

Returns `rmsle` – Root mean squared error

Return type float

References

`kaggle_metrics.rmsle(y_true, y_pred)`

Root mean squared logarithmic error.

Parameters

- `y_true` (*ndarray*) – Ground truth
- `y_pred` (*ndarray*) – Array of predictions

Returns `rmsle` – Root mean squared logarithmic error

Return type float

References

`kaggle_metrics.mae(y_true, y_pred)`

Mean absolute error.

Parameters

- `y_true` (*ndarray*) – Ground truth
- `y_pred` (*ndarray*) – Array of predictions

Returns `rmsle` – Mean absolute error

Return type float

References

`kaggle_metrics.wmae(y_true, y_pred, weights)`

Weighted mean absolute error.

Parameters

- `y_true` (*ndarray*) – Ground truth
- `y_pred` (*ndarray*) – Array of predictions

Returns `rmsle` – Weighted mean absolute error

Return type float

References

`kaggle_metrics.mape(y_true, y_pred)`

Mean absolute error.

Parameters

- `y_true` (*ndarray*) – Ground truth
- `y_pred` (*ndarray*) – Array of predictions

Returns `rmsle` – Mean absolute error

Return type float

References

`kaggle_metrics.mpe(y_true, y_pred)`
Mean percentage error

Parameters

- **y_true** (*ndarray*) – Ground truth
- **y_pred** (*ndarray*) – Array of predictions

Returns `mean_percentage_error` – Mean percentage error

Return type float

References

`kaggle_metrics.mapd(y_true, y_pred)`
Mean absolute percentage error

Parameters

- **y_true** (*ndarray*) –
- **truth** (*Ground*) –
- **y_pred** (*ndarray*) –
- **of predictions** (*Array*) –

Returns `mean_absolute_percentage_error` – Mean absolute percentage error

Return type float

References

`kaggle_metrics.average_precision_at_k(true_positive)`
Average precision at position k

Parameters `true_positive` (*numpy.ndarray*) – True positive for ordered values in query

Returns `score` – A vector of average precision score for every k-th point

Return type `numpy.ndarray`

References

`kaggle_metrics.average_precision(true_positive)`
Average precision

Parameters `true_positive` (*numpy.ndarray*) – True positive for ordered values in query

Returns `score` – A vector of average precision score

Return type `numpy.ndarray`

References

`kaggle_metrics.mean_average_precision` (*true_positive*)

Mean average precision

Parameters `true_positive` (*numpy.ndarray*) – True positive values for n queries (n_queries, answers)

Returns `score` – Mean average precision score

Return type float

References

`kaggle_metrics.ap` (*true_positive*)

Average precision

Parameters `true_positive` (*numpy.ndarray*) – True positive for ordered values in query

Returns `score` – A vector of average precision score

Return type `numpy.ndarray`

References

`kaggle_metrics.ap_at_k` (*true_positive*)

Average precision at position k

Parameters `true_positive` (*numpy.ndarray*) – True positive for ordered values in query

Returns `score` – A vector of average precision score for every k-th point

Return type `numpy.ndarray`

References

`kaggle_metrics.map` (*true_positive*)

Mean average precision

Parameters `true_positive` (*numpy.ndarray*) – True positive values for n queries (n_queries, answers)

Returns `score` – Mean average precision score

Return type float

References

`kaggle_metrics.intersection_over_union` (*y_true, y_pred*)

Intersection over union

Parameters

- `y_true` (*numpy.ndarray*) – Ground truth
- `y_pred` (*numpy.ndarray*) – Prediction

Returns `iou_score` – Intersection over union score

Return type float

- genindex

modindex search

k

kaggle_metrics, 6

A

`ap()` (in module *kaggle_metrics*), 13
`ap_at_k()` (in module *kaggle_metrics*), 13
`average_precision()` (in module *kaggle_metrics*), 12
`average_precision_at_k()` (in module *kaggle_metrics*), 12

G

`gini()` (in module *kaggle_metrics*), 8

H

`hamming_loss()` (in module *kaggle_metrics*), 7

I

`intersection_over_union()` (in module *kaggle_metrics*), 13

K

`kaggle_metrics` (module), 6

L

`log_loss()` (in module *kaggle_metrics*), 6

M

`mae()` (in module *kaggle_metrics*), 11
`map()` (in module *kaggle_metrics*), 13
`mapd()` (in module *kaggle_metrics*), 12
`mape()` (in module *kaggle_metrics*), 11
`matthews_correlation_coefficient()` (in module *kaggle_metrics*), 7
`mcc()` (in module *kaggle_metrics*), 8
`mce()` (in module *kaggle_metrics*), 8
`mean_absolute_error()` (in module *kaggle_metrics*), 9
`mean_absolute_percentage_deviation()` (in module *kaggle_metrics*), 10
`mean_absolute_percentage_error()` (in module *kaggle_metrics*), 10

`mean_average_precision()` (in module *kaggle_metrics*), 13
`mean_consequential_error()` (in module *kaggle_metrics*), 6
`mean_percentage_error()` (in module *kaggle_metrics*), 10
`mean_utility()` (in module *kaggle_metrics*), 7
`mpe()` (in module *kaggle_metrics*), 12

R

`rmse()` (in module *kaggle_metrics*), 10
`rmsle()` (in module *kaggle_metrics*), 11
`roc_auc()` (in module *kaggle_metrics*), 8
`root_mean_squared_error()` (in module *kaggle_metrics*), 9
`root_mean_squared_logarithmic_error()` (in module *kaggle_metrics*), 9

W

`weighted_mean_absolute_error()` (in module *kaggle_metrics*), 9
`wmae()` (in module *kaggle_metrics*), 11