metric_option — Classification and regression metrics

Description Syntax Options References Also see

Description

The h2oml *gbm* and h2oml *rf* estimation commands allow you to specify which metric is to be used for tuning and for early stopping. In addition, h2omlestat gridsummary allows you to specify a metric for reporting; h2omlestat confmatrix allows you to specify a metric for selecting an optimal threshold for classifying predictions; and h2omlgraph scorehistory allows you to specify a metric for the y axis of the graph. In each case, you may specify the metric via a metric() option or suboption. The allowed list of metrics for each command is documented here. Available metrics vary depending on whether regression, binary classification, or multiclass classification is performed.

Syntax

In h2oml *gbm and* h2oml *rf*

command ... [, ... tune(metric(*metric*) ...)]

or

command ... [, ... stop(#, metric(*metric*) ...)]

In h2omlestat gridsummary

h2omlestat gridsummary ... [, ... metric(*metric*) ...]

In h2omlestat confmatrix

h2omlestat confmatrix ... [, ... metric(*metric_conf*) ...]

In h2omlgraph scorehistory

h2omlgraph scorehistory ... [, ... metric(*metric_score*) ...]

command is one of h2oml gbregress, h2oml gbbinclass, h2oml gbmulticlass, h2oml rfregress, h2oml rfbinclass, or h2oml rfmulticlass.

metric	Description
reg_metric	metric for regression (h2oml gbregress and h2oml rfregress)
binclass_metric	metric for binary classification (h2oml gbbinclass and h2oml rfbinclass)
multiclass_metric	metric for multiclass classification (h2oml gbmulticlass and h2oml rfmulticlass)

reg_metric	Description
* deviance	deviance
* mse	mean squared error
* rmse	root mean squared error
* rmsle	root mean squared logarithmic error
* mae	mean absolute error
r2	coefficient of determination

* indicates metrics allowed for stopping.

binclass_metric	Description
* logloss	logarithmic loss
f1	F_1 score
f2	F_2 score
fhalf	$F_{0.5}$ score
accuracy	number of correct predictions as a ratio of all predictions made
precision	proportion of correct predictions in predictions of positive class
recall	proportion of correct predictions of positive class
specificity	proportion of correct predictions in the negative class
* misclassification	number of observations incorrectly classified divided by the total number of observations
* meanclasserror	mean of per-class error rates
maxclasserror	maximum of per-class error rates
meanclassaccuracy	mean of per-class accuracy
misclasscount	total count of misclassification per class
* auc	area under the ROC curve
* aucpr	area under the precision–recall curve
* mse	mean squared error
* rmse	root mean squared error
misclasserror	synonym for misclassification
meanpcerr	synonym for meanclasserror
maxpcerr	synonym for maxclasserror
meanpcacc	synonym for meanclassaccuracy
misclasscnt	synonym for misclasscount

* indicates metrics allowed for stopping.

multiclass_metric	Description
* logloss	logarithmic loss metric
accuracy	number of correct predictions as a ratio of all predictions made
* misclassification	number of observations incorrectly classified divided by the total number of observations
* meanclasserror	mean of per-class error rates
maxclasserror	maximum of per-class error rates
meanclassaccuracy	mean of per-class accuracy
misclasscount	total count of misclassification per class
* mse	mean squared error
* rmse	root mean squared error
meanpcerr	synonym for meanclasserror
maxpcerr	synonym for maxclasserror
meanpcacc	synonym for meanclassaccuracy
misclasscnt	synonym for misclasscount

* indicates metrics allowed for stopping.

metric_conf	Description
f1	F_1 score
f2	F_2 score
fhalf	$F_{0.5}$ score
accuracy	number of correct predictions as a ratio of all predictions made
precision	proportion of correct predictions in predictions of positive class
recall	proportion of correct predictions of positive class
specificity	proportion of correct predictions in the negative class
minclassaccuracy	minimum of per-class accuracy
meanclassaccuracy	mean of per-class accuracy
tn	true negative; the number of correct predictions of the negative class
fn	false negative; the number of incorrect predictions of the negative class
tp	true positive; the number of correct predictions of the positive class
fp	false positive; the number of incorrect predictions of the positive class

tnr	true-negative rate; synonym for specificity
fnr	false-negative rate; the proportion of incorrect predictions in negative class
tpr	true-positive rate; synonym for recall
fpr	false-positive rate; the proportion of incorrect predictions in positive class
mcc	Matthews correlation coefficient
meanpcacc	synonym for meanclassaccuracy
tneg	synonym for tn
fneg	synonym for fn
tpos	synonym for tp
fpos	synonym for fp
tnegrate	synonym for tnr
fnegrate	synonym for fnr
tposrate	synonym for tpr
fposrate	synonym for fpr
mccorr	synonym for mcc

metric_score	Description
reg_metric_score	metric for regression (h2oml gbregress and h2oml rfregress)
binclass_metric_score	metric for binary classification (h2oml gbbinclass and h2oml rfbinclass)
multiclass_metric_score	metric for multiclass classification (h2oml gbmulticlass and h2oml rfmulticlass)

reg_metric_score	Description
deviance	deviance
rmse	root mean squared error
mae	mean absolute error

binclass_metric_score	Description
logloss	logarithmic loss
misclassification	number of observations incorrectly classified divided by the total number of observations
auc	area under the ROC curve
aucpr	area under the precision–recall curve
rmse	root mean squared error
misclasserror	synonym for misclassification

multiclass_metric_score	Description
logloss	logarithmic loss
misclassification	number of observations incorrectly classified divided by the total number of observations
rmse	root mean squared error
misclasserror	synonym for misclassification

Options

Options are presented under the following headings:

Metrics are divided into those for regression and those for classification (binary and multiclass).

Metrics for regression

In the metric formulas, the ith observation is denoted by y_i, the predicted value by $\hat{y}$, the mean by $\overline{y}$, and the total number of observations by n.

`deviance` requests the deviance, which is a measurement of goodness-of-fit of the model.

With `h2oml rfregress` or with `h2oml gbregress` and the Gaussian loss, the deviance, D, is defined as

$$D = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

which is equivalent to the mean squared error (MSE).

With `h2oml gbregress` and the Tweedie loss, the deviance is defined as

$$D = \sum_{i=1}^{n} \left[\frac{\{\max(y,0)\}^{2-p}}{(1-p)(2-p)} - \frac{y(\hat{y})^{1-p}}{1-p} + \frac{(\hat{y})^{2-p}}{2-p} \right]$$

where p is the parameter in Tweedie and specified as `power()` in `h2oml` *gbm*.

With `h2oml gbregress` and the Poisson loss, the deviance is defined as

$$D = -2 \sum_{i=1}^{n} \left\{ y_i \log(\frac{y_i}{\hat{y}_i}) - (y_i - \hat{y}_i) \right\}$$

With `h2oml gbregress` and the Laplace loss, the deviance is defined as

$$D = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$$

which is equivalent to the mean absolute error (MAE).

`mse` requests the MSE, which is the average of the squared errors. MSE can be represented as a sum of the variance and the square of the bias. It imposes larger penalties on larger errors. Thus, it is sensitive to outliers. The formula is

$$\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

`rmse` requests the root mean squared error (RMSE). Unlike the MSE, the units of RMSE are the same as the units of the response variable, which provides a useful interpretation when the size of the error is of interest. The formula is

$$\sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}$$

`rmsle` requests the root mean squared logarithmic error (RMSLE), which is the ratio between the logarithm actual values and the logarithm of predicted values. The RMSLE is recommended when underprediction of the model is worse than the overprediction. The formula is

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left\{\ln\left(\frac{y_i+1}{\widehat{y}_i-1}\right)\right\}^2}$$

`mae` requests the MAE, which is the average of the absolute value of the error. The units of MAE are the same as the units of the response, and it is robust to outliers. A smaller MAE indicates a better performance. The formula is

$$\frac{1}{n}\sum_{i=1}^{n}|y_i-\widehat{y}_i|$$

`r2` requests the R^2, also known as the coefficient of determination. R^2 is the proportion of the variance of a response that is explained by the predictors. Because the estimated variance depends on the given dataset, we do not advise the comparison of R^2 across different datasets. The best R^2 score is 1, and it can be negative because a model can predict arbitrarily poorly. The estimated R^2 is defined as

$$1-\frac{\sum_{i=1}^{n}(y_i-\widehat{y}_i)^2}{\sum_{i=1}^{n}(y_i-\overline{y})^2}$$

Metrics for classification

For binary classification, suppose that y_i takes two possible values $\{0,1\}$, where 0 and 1 correspond to negative and positive classes, respectively. The predicted probability for the positive class and observation i is denoted by $\widehat{p}_i$ and the predicted class by $\widehat{y}_i$.

For multiclass classification, the number of classes is denoted by K and $y_{ik}=1$ if the observation i belongs to the class k and 0 otherwise. The predicted probability for the observation i and class k is denoted by $\widehat{p}_{ik}$.

`logloss` requests log loss (logarithmic loss). The goal of the log loss is to estimate the closeness of the model's predicted probabilities to the actual values of the response variable. That is, log loss indicates the ability of the model to assign higher predicted probabilities to observations in the positive class and smaller probabilities to observations in the negative class. Log loss may take any nonnegative value. For binary classification, it is defined as

$$-\frac{1}{n}\sum_{i=1}^{n}y_i\ln(\widehat{p}_i)+(1-y_i)\ln(1-\widehat{p}_i)$$

For multiclass classification, it is defined as

$$-\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K}y_{ik}\ln(\widehat{p}_{ik})$$

`f1`, `f2`, and `fhalf` are F_β scores and are functions of recall and precision. The F_β scores are defined as

$$F_\beta=(1+\beta^2)\frac{\text{precision}\times\text{recall}}{\beta^2(\text{precision}+\text{recall})}$$

where $\beta > 0$ is chosen such that recall is considered β times as important as precision. Here precision and recall are defined as in the descriptions of the `precision` and `recall` options.

`f1` requests F_1.

`f2` requests F_2, which is the harmonic mean of precision and recall.

`fhalf` requests $F_{0.5}$.

`accuracy` requests the accuracy, which is the ratio of the number of correct predictions to the total number of all predictions made. The accuracy metric is not recommended for imbalanced data (Bradley 1997; Huang and Ling 2005). For example, for a sample with 100 observations such that 96 belong to positive and 4 to negative classes, the accuracy score for a model that predicts the positive class for all observations is 0.96, which is misleading. The formula is

$$\frac{\text{tp} + \text{tn}}{\text{tp} + \text{tn} + \text{fp} + \text{fn}}$$

where tn and tp are the numbers of true negatives and true positives (correct predictions) and where fn and fp are the numbers of false negatives and false positives (incorrect predictions).

For multiclass classification, `accuracy_k` denotes the estimated accuracy for the class k.

`precision` requests the precision, which is the proportion of observations correctly predicted to be in the positive class out of all observations predicted to be in the positive class. Precision is a biased metric; it fails to account for the performance in negative classes (Powers 2011). The formula is

$$\frac{\text{tp}}{\text{tp} + \text{fp}}$$

`recall` requests the recall, also known as the sensitivity or the true-positive rate. It is the proportion of observations correctly predicted to be in the positive class out of all observations that actually belong to the positive class. Recall is a biased metric; it fails to account for the performance in negative classes (Powers 2011). The formula is

$$\frac{\text{tp}}{\text{tp} + \text{fn}}$$

`specificity` requests the specificity, also known as the true-negative rate. It is the proportion of correct predictions in the negative class. The formula is

$$\frac{\text{tn}}{\text{tn} + \text{fn}}$$

`misclassification` requests the misclassification, which is the proportion of the predictions that are false. It is equal to

$$1 - \text{accuracy}$$

For multiclass classification, the misclassification error for the class k is defined as

$$1 - \text{accuracy}_k$$

`misclasserror` is a synonym for `misclassification`.

`meanclasserror` requests the mean of the per-class misclassification errors. The misclassification error in class k is estimated by $1 - \text{accuracy}_k$, where accuracy_k is the accuracy for the class k. Then for K classes, the `meanclasserror` is

$$\frac{1}{K}\sum_{k=1}^{K}(1 - \text{accuracy}_k)$$

`meanpcerr` is a synonym for `meanclasserror`.

`maxclasserror` requests the maximum per-class misclassification error. For K classes, it is defined as

$$\max_{k=1,\ldots,K}\{1 - \text{accuracy}_k\}$$

`maxpcerr` is a synonym for `maxclasserror`.

`minclassaccuracy` requests the minimum per-class accuracy. For K classes, it is defined as

$$\min_{k=1,\ldots,K}\{\text{accuracy}_k\}$$

`meanclassaccuracy` requests the mean of the per-class accuracies. For K classes, it is defined as

$$\frac{1}{K}\sum_{k=1}^{K}\text{accuracy}_k$$

`meanpcacc` is a synonym for `meanclassaccuracy`.

`misclasscount` requests the total number of observations that a model has incorrectly classified. For the binary classification, it is defined as

$$\sum_{i=1}^{n} 1(y_i \neq \hat{y}_i)$$

where $1(\cdot)$ is an indicator function and $\hat{y}_i$ is the predicted class.

For the multiclass classification, it is defined as

$$\sum_{i=1}^{n}\sum_{k=1}^{K} 1(y_{ik} \neq \hat{y}_{ik})$$

`misclasscnt` is a synonym for `misclasscount`.

`auc` requests the area under the curve (AUC), which measures the ability of the classification model to distinguish between true positives and false positives. A higher value indicates a better classifier. A classifier with an AUC score of 0.5 is no better than a random guess. H2O uses the trapezoidal rule to approximate the area under the receiver operating characteristic (ROC) curve. The ROC curve plots the recall against the false-positive rate. For imbalanced data, AUC is preferred more than accuracy (Bradley 1997) but less recommended than the area under the precision–recall curve (AUCPR) or the Matthews correlation coefficient (MCC).

For multiclass classification with the number of classes equal to K, there exist several variations of the AUC score.

The one-versus-one AUC (OVO AUC) calculates the AUC score for all pairwise combinations of classes. The computation of this metric requires fitting one binary classification per class pair. Thus, there are $K \times (K-1)/2$ binary classifiers.

The one-versus-rest AUC (OVR AUC) calculates the AUC score for one class with the rest of the classes. The computation of this metric requires fitting one binary classifier per class, where a given class is regarded as the "positive" class and the remaining classes are regarded as the "negative" class.

The macro average OVR AUC is a uniform weighted average of all OVR AUCs.

$$\frac{1}{K}\sum_{k=1}^{K}\text{AUC}(k, K_{-k})$$

where K is the number of classes and $\text{AUC}(j, K_{-j})$ is the AUC with class j as the positive class and the rest of classes K_{-j} as the negative class.

The weighted average OVR AUC calculates the prevalence weighted average of all OVR AUCs, where the prevalence of class k, $p(k)$, is the number of observations in class k.

$$\frac{1}{\sum_{k=1}^{K} p(k)}\sum_{k=1}^{K} p(k)\text{AUC}(k, K_{-k})$$

The macro average OVO AUC is a uniformly weighted average of all OVO AUCs

$$\frac{2}{K}\sum_{k=1}^{K}\sum_{j\neq k}^{K}\frac{1}{2}\{\text{AUC}(k, j) + \text{AUC}(j, k)\}$$

The weighted average OVO AUC is a prevalence weighted average of all OVO AUCs.

$$\frac{2}{\sum_{k=1}^{K}\sum_{j\neq k}^{K} p(k\cup j)}\sum_{k=1}^{K}\sum_{j\neq k}^{K} p(k\cup j)\frac{1}{2}\{\text{AUC}(k, j) + \text{AUC}(j, k)\}$$

`aucpr` requests the AUCPR. It is a weighted average of precision, where the weights are determined by recall at the threshold. By construction, AUCPR is more sensitive to true-positive, false-positive, and false-negative rates than AUC. Thus, it is more suitable for highly imbalanced data.

For multiclass classification, AUCPR metrics are defined similarly to the corresponding AUC metrics.

`tn` requests the true-negative metric, tn, which is the number of correct predictions of the negative class.

`tneg` is a synonym for `tn`.

`fn` requests the false-negative metric, fn, which is the number of incorrect predictions of the negative class.

`fneg` is a synonym for `fn`.

`tp` requests the true-positive metric, tp, which is the number of correct predictions of the positive class.

`tpos` is a synonym for `tp`.

`fp` requests the false-positive metric, fp, which is the number of incorrect predictions of the positive class.

`fpos` is a synonym for `fp`.

`tnr` requests the true-negative rate, which is the same as specificity.

`tnegrate` is a synonym for `tnr`.

`fnr` requests the false-negative rate, which is the proportion of incorrect predictions in the positive class. The formula is

$$\frac{\text{fn}}{\text{tp} + \text{fn}}$$

`fnegrate` is a synonym for `fnr`.

`tpr` requests the true-positive rate, which is the same as recall.

`tposrate` is a synonym for `tpr`.

`fpr` requests the false-positive rate, which is the proportion of incorrect predictions in the negative class. The formula is

$$\frac{\text{fp}}{\text{tn} + \text{fp}}$$

`fposrate` is a synonym for `fpr`.

`mcc` requests the MCC, which measures how well a binary classifier detects true and false positives, and true and false negatives. The MCC provides correlation between the actual and predicted values.

$$\frac{\text{tp} \times \text{tn} - \text{fp} \times \text{fn}}{\sqrt{(\text{tp} + \text{fp})(\text{tp} + \text{fn})(\text{tn} + \text{fp})(\text{tn} + \text{fn})}}$$

`mccorr` is a synonym for `mcc`.

Additional classification metrics

Below, we provide definitions for additional metrics that are reported by H2OML commands for classification but that need not be specified via the `metric()` option.

Gini coefficient. Often referred to as the Gini index, this estimates the "purity" of a dataset in classification problems. For a binary classification, the Gini coefficient is calculated as

$$\text{Gini} = 1 - (p_1^2 + p_2^2)$$

where p_1 and p_2 are the proportions of class 1 and 2, respectively.

R² for classification. This represents the degree to which the predicted probability and the actual class move together. The best R^2 score is 1, and it can be negative because a model can predict arbitrarily poorly. For binary classification, the estimated R^2 is defined as

$$1 - \frac{\sum_{i=1}^{n} (y_i - \widehat{p}_i)^2}{\sum_{i=1}^{n} (y_i - \overline{p}_i)^2}$$

For multiclass classification, it is defined as

$$1 - \frac{\sum_{i=1}^{n} \sum_{k=1}^{K} (y_{ik} - \widehat{p}_{ik})^2}{\sum_{i=1}^{n} \sum_{k=1}^{K} (y_i - \overline{p}_{ik})^2}$$

MSE for classification. This is the average of the squared errors, where error is the difference between the predicted probability and the actual class. For binary classification, the formula is

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{p}_i)^2$$

For multiclass classification, it is

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K}(y_{ik} - \widehat{p}_{ik})^2$$

RMSE for classification. This is the square root of MSE.

References

Bradley, A. P. 1997. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition* 30: 1145–1159. https://doi.org/10.1016/S0031-3203(96)00142-2.

Huang, J., and C. X. Ling. 2005. Using AUC and accuracy in evaluating learning algorithms. *IEEE Transactions on Knowledge and Data Engineering* 17: 299–310. https://doi.org/10.1109/TKDE.2005.50.

Powers, D. M. W. 2011. Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. *Journal of Machine Learning Technologies* 2: 37–63.

Also see

[H2OML] **h2oml** — Introduction to commands for Stata integration with H2O machine learning

[H2OML] ***h2oml gbm*** — Gradient boosting machine for regression and classification

[H2OML] ***h2oml rf*** — Random forest for regression and classification

[H2OML] **h2omlestat gridsummary** — Display grid-search summary

[H2OML] **h2omlestat confmatrix** — Display confusion matrix

[H2OML] **h2omlgraph scorehistory** — Produce score history plot

For suggested citations, see the FAQ on citing Stata documentation.