mlr3calibration

Adrian Glos

2024-08-30

Installation

You can install the mlr3calibration package from CRAN or from GitHub with the following code:

```
# install.packages("mlr3calibration")
# remotes::install_github("mlr-org/mlr3calibration")
```

Calibration

To use the mlr3calibration package, we first need a binary classification task

```
# Load a binary classification task
set.seed(1)
data("Sonar", package = "mlbench")
task = as_task_classif(Sonar, target = "Class", positive = "M")
splits = partition(task)
task_train = task$clone()$filter(splits$train)
task_test = task$clone()$filter(splits$test)
```

To calibrate a learner you need an uncalibrated learner, a resampling strategy and a calibration method (platt, beta or isotonic). To prevent overfitting, calibration can be performed using cross-validation. The dataset is divided into k folds, and each fold is used for calibration while the other (k-1) folds are used for training the classifier (UMFORMULIEREN). If, for example, a holdout resampling strategy is selected, then no cross-validated calibration is performed, but the base learner is trained on the training split and the calibrator on the holdout.

```
# Initialize the uncalibrated learner
learner_uncal <- lrn("classif.ranger", predict_type = "prob")

# Initialize the calibrated learner
rsmp <- rsmp("cv", folds = 5)
learner_cal <- as_learner(po("calibration_cv", learner = learner_uncal,
                             method = "platt", rsmp = rsmp))

# Set ID's for the learners
learner_uncal$id <- "Uncalibrated Learner"
learner_cal$id <- "Calibrated Learner"
```

The calibrated learner can be trained in the same way as the base learner.

```
# Train the learners
learner_uncal$train(task_train)
learner_cal$train(task_train)
```

Calibration Measures

```
# Predict the Learners
preds_uncal <- learner_uncal$predict(task_test)
preds_cal <- learner_cal$predict(task_test)

# Calculate the ECE
ece_uncal <-  preds_uncal$score(msr("classif.ece"))
ece_cal <-  preds_cal$score(msr("classif.ece"))

# Uncalibrated ECE
ece_uncal
```

```
## classif.ece
##   0.1232608
```

```
# Calibrated ECE
ece_cal
```

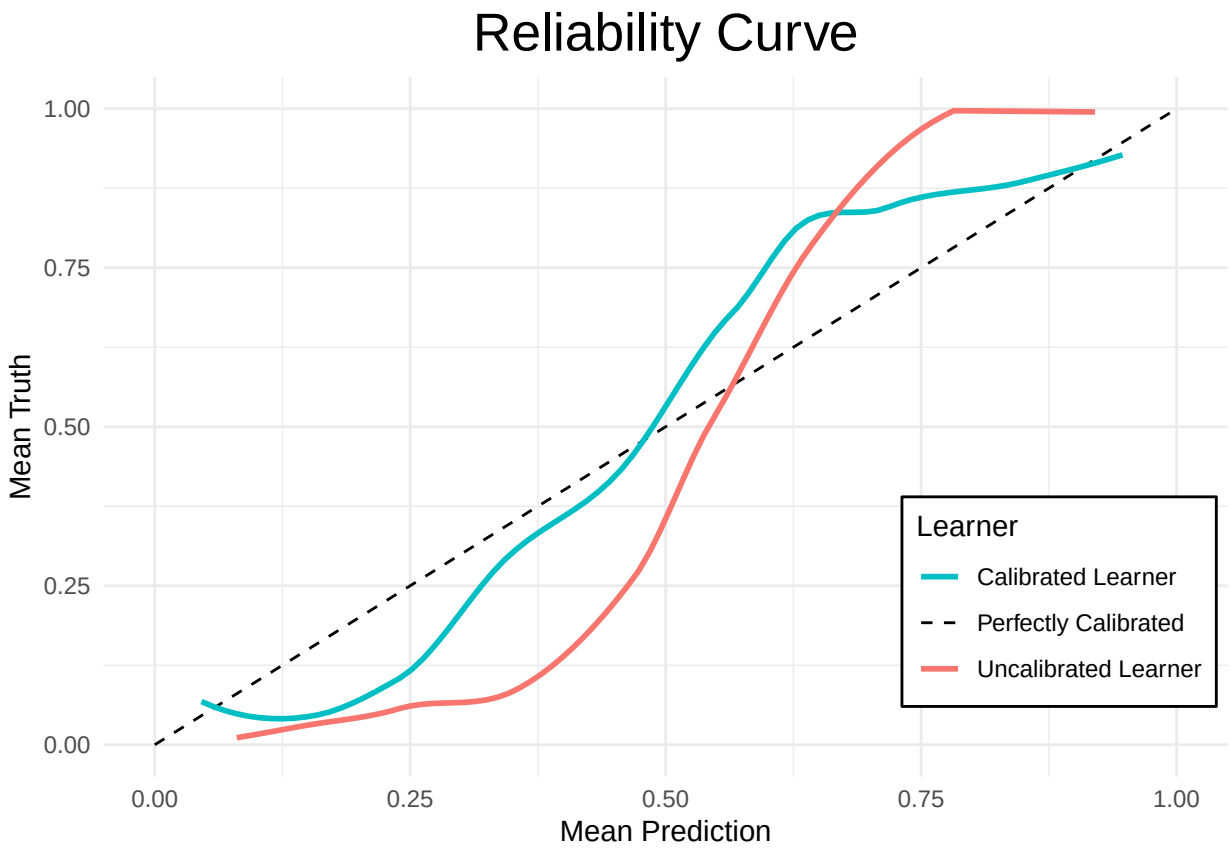
```
## classif.ece
##  0.03209833
```

Raliability Curve

You can also plot the Reliabilitiy Curve to visualize the calibration of the learners.

```
# List the Learners you want to plot
lrns = list(learner_uncal, learner_cal)

# Plot the reliability curve
calibrationplot(lrns, task_test, smooth = TRUE)
```


Reliability Curve
Mean Truth
1.00
0.75
0.50
0.25
0.00
0.00
0.25
0.50
0.75
1.00
Mean Prediction
Learner
Calibrated Learner
Perfectly Calibrated
Uncalibrated Learner