A typical task in evaluating the results of machine learning models is making a ROC curve, this plot can inform the analyst how well a model can discriminate one class from a second. We developed MLeval (<https://cran.r-project.org/web/packages/MLeval/index.html>), a evaluation package for R, to make ROC curves, PR curves, PR gain curves, and calibration curves. These plots are all using ggplot2 and it also yields performance metrics such as, Matthew’s correlation coefficient, specificity, sensitivity, and includes confidence intervals.

MLeval is aimed to make life as simple as possible. It can be run directly on a data frame of predicted probabilities and ground truth probabilities (labels), or on the Caret ‘train’ function output which performs cross validation to avoid overfitting. It also makes it easy to compare different models together. Let’s see an example.

## load libraries required for analysis

library(MLeval)

library(caret)

Run Caret on the Sonar data with 3 different models, then evaluate by passing the results objects as a list into ‘evalm’.

## load data and run Caret

data(Sonar)

ctrl <- trainControl(method="cv", summaryFunction=twoClassSummary, classProbs=T,

savePredictions = T)

fit1 <- train(Class ~ .,data=Sonar,method="rf",trControl=ctrl)

ctrl <- trainControl(method="cv", summaryFunction=twoClassSummary, classProbs=T,

savePredictions = T)

fit2 <- train(Class ~ .,data=Sonar,method="gbm",trControl=ctrl)

ctrl <- trainControl(method="cv", summaryFunction=twoClassSummary, classProbs=T,

savePredictions = T)

fit3 <- train(Class ~ .,data=Sonar,method="nb",trControl=ctrl)

## run MLeval

res <- evalm(list(fit1,fit2,fit3),gnames=c('rf','gbm','nb'))

The results (metrics and plots) can be accessed through the list object 'evalm' produces. We can see below that random forest and gbm perform the same, whereas naive bayes does not do as well falling behind the others in the two discrimination tests (ROC and PRG). However, in the calibration curves we can see all models are quite well calibrated, showing that being good at calibration does not always imply good discrimination.

The PRG curve standardises precision to the baseline, whereas the PR curve has a variable baseline, making it unsuitable to compare between data with different class distributions. This plot will change depending on which class is defined as positive, and is a deficiency of precision recall for non extremely imbalanced tasks. Credit card fraud is an example of where positives << negatives and it becomes more appropiate.

In the first two plots the analysis performed is the same, the probabilities are ranked from high to low then a sensitivity analysis is performed of the probability cut-off parameter to define a positive. For each iteration true positive rate vs true negative rate are calculated and plotted in the case of the ROC, for PRG it is precision gain vs recall gain. In the last plot, we plot predicted vs real probabilities (in bins) and the aim is for them to match as closely as possible (grey diagonal line = perfect). See our vignette for more information (<https://cran.r-project.org/web/packages/MLeval/vignettes/introduction.pdf>). Code is attached with this repository.





