

The `train` Function

One of the primary tools in the package is the `train()` function which can be used to

- evaluate, using resampling, the effect of model tuning parameters on performance
- choose the “optimal” model across these parameters
- estimate model performance from a training set

This function sets up a grid of tuning parameters for a number of classification and regression routines, fits each model and calculates a resampling based performance measure.

- `train()` function can be used to tune models by picking the complexity parameters that are associated with the optimal resampling statistics.
- For particular model, a grid of parameters (if any) is created and the model is trained on slightly different data for each candidate combination of tuning parameters.
- Across each data set, the performance of held-out samples is calculated and the mean and standard deviation is summarized for each combination.
- The combination with the optimal resampling statistic is chosen as the final model and the entire training set is used to fit a final model.

Syntax

The `train()` function has the following arguments:

x: a matrix or data frame of predictors. Currently, the function only accepts numeric values (i.e., no factors or character variables).

In some cases, the `model.matrix` function may be needed to generate a data frame or matrix of purely numeric data.

y: a numeric or factor vector of outcomes. The function determines the type of problem (classification or regression) from the type of the response given in this argument.

method: a character string specifying the type of model to be used.

metric: a character string with values of "Accuracy", "Kappa", "RMSE" or "Rsquared".

The metric value determines the objective function used to select the final model. For example, selecting "Kappa" makes the function select the tuning parameters with the largest value of the mean Kappa statistic computed from the held-out samples.