Machine Learning with R at LRZ: Introduction to mlr

Resampling

We will continue our example for spam classification

a) Instead of manually splitting train and test set create a holdout set directly in mlr. Use the set to evaluate the performance of an algorithm of your choice on the spam data. Use 80% of the data for training and create stratified splits.

```
library(mlr)
data(spam, package = "kernlab")
spam.task = makeClassifTask(data = spam, target = "type")
lrn = makeLearner("classif.rpart", predict.type = "prob")
```

b) Now create a 10-fold crossvalidation and evaluate AUC and training time

Benchmarking

We would like to create a small benchmark study to see how much complexity is required to achieve an AUC of at least 98%.

a) Create the following learning algorithms to compare their performance

- Featureless baseline learner
- Linear Discriminant Analysis
- Logistic Regression
- Classification Tree
- Random Forest

b) Benchmark the five learning algorithms with a 5-fold crossvalidation (ensure identical folds for all learners). Measure the AUC as well as the runtime.

c) Vizualize the results. Which learner would you use in practice and as a spam detector?

Tuning

Tune `mtry`, `nodesize` and `sampsize` of the random forest to get the best possible tuning error.

a) Define reasonable bounds for the parameter space. (Hint: Have a look at the number of rows and columns of the spam data)

b) Use a random search to optimize over the parameter space.