Supervised machine learning II

Computational Biomedicint, WS 2023, Prof. Dr. Michael Altenbuchinger

January 10, 2024

Learning objectives: Learn to apply random forests and support vector machines. Assess their performance.

(1) Support vector machines

In this exercise we want to predict if a patient gets acute kidney failure (AKI) after cardiac surgery. For this, we have serum metabolic fingerprints (NMR spectra) of 85 patient that underwent cardiac surgery. These specimens were acquired within a short time period after the surgery, before the acute kidney failure was visible. For this purpose, we will use support vector machines (SVMs).

- Install and load the package "e1071".
- Load the metabolic fingerprints "data_PQN.rds" and the outcome vector "responses.rds", where 1 corresponds to "AKI" and 0 to "non-AKI" (`readRDS()`).
- First, train a SVM using the full dataset (`svm()`). Ensure that you make a classification and not a regression!
- There are different ways for the high-dimensional embedding of SVMs. In the lecture, we learned about the polynomial kernel. Train a SVM using this kernel.
- Use the predict command (`predict()`) to predict probabilities for all samples using the training data.
- Use the ROCR package to calculate the AUC of the corresponding ROC curve, where we compare these predictions with the actual outcomes (`prediction()`, `performance()`). Plot the ROC curve.
- Set up a leave-one-out cross validation and record all predictions.
- Plot for these predictions the ROC curve and the precision-recall curve. Calculate the AUCs.
- Try different kernels and assess the performance.
- What do you mean: does the high-dimensional embedding yield a performance gain?

(2) Random forests

Here, we want to predict the presence of heart disease from different variables such as age, gender, and cholesterol. These data are diverse, covering categorical as well as continuous data. This makes it a nice application for random forests.

- Install and load the package “randomForest”.
- Load the data “processed_cleveland.rds” using `readRDS()`. There is also an explanation to all variables that you find in “description.txt” if you want to have more information about the data.
- The data are already prepared that you can immediately run the random forest. However, first convince yourself about the classes of the different variables. You can use `class()` on the different variables that you can retrieve as usual from the data frame. You can also check that for all variables at once using `sapply(..., class)`.
- Split up your data into a training cohort (75% of the patients) and a test cohort.
- Learn a random forest for the prediction of heart disease on the training data (`randomForest()`). Heart disease is given by the variable `num` in the dataset. Visualize your random forest fit using `plot()`.
- Plot the feature importance in your random forest fit using `barplot()`. You can find it using `str()`.
- Predict heart disease for the test data (`predict()`). You can check the performance using the true and predicted labels by the employing `table()`.
- Visualize the performance using the “ROCR” package (`prediction()`, `performance()`). Plot the ROC curve and the Precision-Recall curve. Calculate the area under the curve.