Methods of machine learning

Exercise sheet III

May 14th, 2025

In this exercise class we apply boosting and bagging of decision trees to the classical *wines data set*. To this end, we use AdaBoost and random forests.

1. **Task.**
   a) Load the data and split it into training and test sets (80-20 split).
   b) Train a full decision tree classifier, check its training and test error and display the learned classifier.
   c) Now apply the AdaBoost algorithm to this kind of learner (using `AdaBoostClassifier` from `sklearn.ensemble`) with $T = 10$ iterations. Check again training and test error and output the weights α_t, $t = 1, \dots, T$. What do you notice?

2. **Task.**
   a) Now modify the decision tree learner to have at most $n = 2$ leaves. Again, train a corresponding decision tree classifier, check its training and test error and display the learned classifier.
   b) Again apply AdaBoost $T = 10$ iterations. Check again training and test error and output the weights α_t, $t = 1, \dots, T$. What do you notice now?
   c) Vary now the number of iterations $T \in \{1, \dots, 25\}$ and plot the resulting training and test error versus T. Which choice of T is best?

3. **Task.**
   a) Train a random forest calssifier using `RandomForestClassifier` with 20 trees and estimate its generalization error.
   b) Display the confusion matrix for the predictions. And check precision, recall and f1-score of the predictions using `classification_report`.
   c) Train a random forest calssifier using `RandomForestClassifier` with 20 trees again and print both test set accuracy and OOB accuracy.
   d) Train a random forest calssifier using `RandomForestClassifier` with 5 trees and print OOB accuracy. What do you notice?

4. **Task.**
   a) Study the importance of all features with the random forest calssifier with 20 trees and reorder the features according the their estimated importance.
   b) Increase the number of features starting with the most important feature and report the cross validation error of the previous random forest calssifier versus the number of sorted features.

5. **Task.**
   a) Perform a grid search to find the best combination of parameters for n_estimators, max_features, max_depth and criterion. Use `GridSearchCV` from `sklearn.model_selection` and the accuracy for evaluation. Train a random forest calssifier with the found best combination, estimate its generalization error and print the computation time.
   b) Perform a random search to find the best combination of parameters for n_estimators, max_features, max_depth and criterion. Use `RandomizedSearchCV` from `sklearn.model_selection` and the accuracy for evaluation. Train a random forest calssifier withthe found best combination, estimate its generalization error and print the computation time.