VSB TECHNICAL UNIVERSITY OF OSTRAVA | FACULTY OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE | DEPARTMENT OF COMPUTER SCIENCE

Data Analysis 3

Ensemble methods

Jan Platoš

November 29, 2020

Department of Computer Science
Faculty of Electrical Engineering and Computer Science
VŠB - Technical University of Ostrava

Table of Contents

- Ensemble methods
  - Bagging
    - Random Forrest
    - Extra Trees
  - Boosting
    - AdaBoost
    - Gradient Boosted Decision Trees
    - Light Gradient Boosting Machine
  - Bucket of models
  - Stacking

Ensemble methods

Ensemble methods

- The main idea is that different classifiers may make different predictions on test instances with the same train data.
- This is caused by the specific characteristics of the classifiers, their sensitivity to the random artifacts in the data, etc.
- The basic approach is to apply basic ensemble learners multiple times by using different models or the same model on different subsets of data.
- Two basic approaches exists:
  - Data-centered ensembles
  - Model-centered ensembles

Ensemble methods

- Data-centered ensembles
  - Single classification model is used.
  - The dataset is derived into set of subsets.
  - The method of dataset derivation differs - sampling, incorrectly classified data from previous set, manipulation with features, manipulation with class labels, etc.
- Model-centered ensembles
  - Many different algorithms are used in each ensemble iteration.
  - The dataset used by each model is the same as the original dataset.
  - The motivation is that different classifiers works better on particular part of data.
  - This approach is valid as long as the specific errors are not reflected by the majority of the ensembles.

Ensemble methods - Bias

- Every classifier makes its own modeling assumptions about the nature of the decision boundary between classes:
  - The classifier may incorrectly classify data even with large training dataset.
  - The modeled decision boundary does not match the real boundary.
  - Therefore, the classifier has an inherent error - **inherent bias**.
- When a classifier has **high bias**, it will make **consistently incorrect predictions** over particular choices of test instances near the incorrectly modeled decision-boundary.

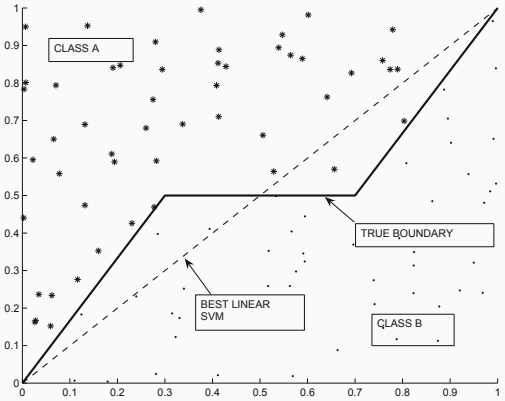

Figure 1: Bias on Linear SVM example

Ensemble methods - Variance

- Random variations in the choices of the training data will lead to different models.
  - Test instances such as X are **inconsistently classified** by decision trees which were created by different choices of training data sets.
  - This is a manifestation of model **variance**.
  - Model variance is closely related to **over-fitting**.
- When a classifier has an over-fitting tendency, it will make inconsistent predictions for the same test instance over different training data sets.

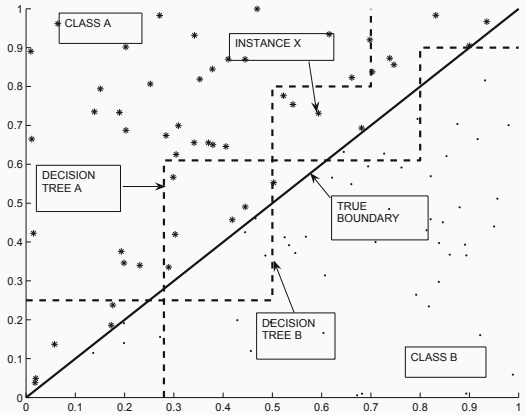

Figure 2: Variance on Decision Tree example

Ensemble methods- Bagging

- Also known as bootstrapped aggregation.
- It is focused on variance reduction of the prediction.
- With the variance of the prediction equals to σ^2, the variance of the average of k independent and identically distributed (i.i.d.) prediction is reduced to $\frac{\sigma^2}{k}$.
- The i.i.d. predictors are approximated with bootstrapping (sampling with replacement).

Ensemble methods- Bagging

- The k different sets are constructed from the original dataset.
- Each set is used for model training.
- The predicted class is the dominant class over all classifiers.
- This approach decreases the variance, but may increase the bias.
- More detailed models need to be used to reduce bias as well, otherwise, slightly degradation in accuracy may be achieved.
- The i.i.d. is usually not fully satisfied.
- The performance limit of the bagging is done by the pairwise correlation between models ρ

$$\rho \cdot \sigma^2 + \frac{(1-\rho) \cdot \sigma^2}{k}$$

Ensemble methods - Bagging - Random Forrest

- Random forests can be viewed as a generalization of the basic bagging method, as applied to decision trees.
- The main drawback of using decision-trees directly with bagging is that the split choices at the top levels of the tree are statistically likely to remain approximately invariant to bootstrapped sampling.
- Therefore, the trees are more correlated, which limits the amount of error reduction obtained from bagging.
- The idea is to use a randomized decision tree model with less correlation between the different ensemble components.
- The final results are often more accurate than a direct application of bagging on decision trees.

Ensemble methods - Bagging - Random Forrest

- The *random-split-selection* introduces randomness into split criterion.
- The coefficient $q \leq d$ is used to regulate the randomness.
- The split-point selection is preceded by the random selection of q features.
- Smaller number of q reduces the correlation between different trees but decreases the accuracy.
- Moreover, this improves the construction process because only subset of features need to be investigated.

Ensemble methods - Bagging - Random Forrest

- The good trade-off between correlation reduction and accuracy was investigated as

$$q = \log_2(d) + 1$$

- Low-dimension data does not benefit from this approach due to large q with respect to the d.
- The trees are grown without pruning to reduce bias of the prediction.
- Random trees are resistant to noise and outliers and usually better than pure bagging.

Ensemble methods - Bagging - Extra Trees

- Slightly different approach is used by the Extra Trees - Extremely Randomized Trees.
- The main changes are focused to increase the variance.
- The data are not sampled using bootstrapping - all data are used for each tree.
- First, the subset of randomly selected features of size q is randomly selected.
- The split of each feature is chosen randomly.
- The best split is selected from the sampled ones.
- Due to two random sampling, trees are really random and less computationally expensive.

Ensemble methods - Boosting

- In boosting, a weight is associated with each input instance.
- Different classifiers are trained with these weights.
- The weights are modified iteratively based on classification performance.
- Each classifier is constructed using the same algorithm.
- The relative weights are increased on incorrectly classified instances, according to the hypothesis that the misclassification is caused by classifier bias.
- The overall bias is then decreased.

Ensemble methods - Boosting

- The predicted class is determined by the weighted aggregation of the particular prediction of each model.
- The primary purpose is to reduce bias of the classification.
- This approach is more sensitive to the noised datasets.
- A typical example is *AdaBoost* algorithm.

Ensemble methods - Boosting - AdaBoost (Adaptive Boosting)

- In binary classification, where labels are from $\{-1; 1\}$.
- The weights are initialized to $\frac{1}{n}$ for each of the n instances.
- The weights are in each iteration updated according the correctness of the prediction.
  - $W_{t+1}(i) = W_t(i)e^{\alpha_t}$ for incorrect classification.
  - $W_{t+1}(i) = W_t(i)e^{-\alpha_t}$ for correct classification.
- The α_t is defined as a function:

$$\frac{1}{2} \ln \left(\frac{1 - \epsilon_t}{\epsilon_t} \right)$$

- where ϵ_t is the fraction of incorrectly classified instances at t-th iterations.

Ensemble methods - Boosting - AdaBoost (Adaptive Boosting)

- The termination criterion are defined as:
  - $\epsilon_t = 0$ - all instances are correctly classified.
  - $\epsilon_t > 0.5$ - the classification is worse than random.
  - User-defined number of iterations is reached.
- The classification of test instance is done using aggregation over all models:

$$y_{pred} = \sum_t p_t \alpha_t$$

- where $p_t \in \{-1; 1\}$ is the prediction in the t-th iteration
- and the α_t is defined as a function:

$$\frac{1}{2} \ln \left(\frac{1 - \epsilon_t}{\epsilon_t} \right)$$

Ensemble methods - Boosting - Gradient Boosted Decision Trees

- Uses a Decision Trees as a weak learners.
- A loss function is used to detect the residuals, e.g. mean squared error (MSE) for a regression task and logarithmic loss (log loss) for a classification tasks.
- The existing trees are left unchanged when a new tree is added.
- The new tree is trained on the previous model residual.

Ensemble methods - Boosting - Gradient Boosted Decision Trees

- The increasing number of trees may lead in overfitting (this is a difference against the random forest).
- The learning rate affect the speed of learning (small value more robust model).
- Small learning rate requires more trees, more tree leads to overfitting....
- May lead to more precise models than the random forests.
- It is highly sensitive to learning rate and number of learners parameters.
- It is also very sensitive to outliers and noise.

Ensemble methods - Boosting - Light Gradient Boosting Machine

- Another gradient boosting algorithm that utilizes decision trees.
- The trees used grows leaf-wise - the leaf with maximal error is grown to achieve better results.
- The features are selected according the it nature - sparse features are combined.
- Designed to process large dataset with many features.
- Contains more than 100 parameters that may be tuned.

Ensemble methods - Bucket of models

- An method that combines several different algorithms together and removes the necessity of *a priori* selection of the particular classification algorithm.
- The dataset is divided into two subsets A and B (a hold-out principle).
- Each algorithm is trained on the A set and evaluated on B set.
- The best algorithm is selected as a winner and then it is retrained on the complete dataset.
- A cross-validation may be used instead of hold-out principle.
- Different algorithm may be represented by the same algorithm with different parameters.
- Due to *winner-take-all* principle, the best found classifier is selected.
- This approach reduces both bias and variance but it is limited by the parameters on the winner.

Ensemble methods - Stacking

- Stacking is a two-level classification approach.
- Several algorithm are used for classification.
- The dataset is divided into two subsets A and B (a hold-out principle).
- First level:
  - Training of the k different classifier (ensemble components) on the set A.
  - These components are generated using:
    - bagging,
    - k-rounds boosting,
    - k different decision tress,
    - k heterogeneous classifiers.

Ensemble methods - Stacking

- Second level:
  - Determine the k outputs of each trained classifier on a set B.
  - Create a new set of k features from these outputs.
  - The class label is known from the ground-truth data.
  - Train a classifier on this new representation of the set B.

Ensemble methods - Stacking

- Sometimes, the original features of B are combined with k generated features from the first level.
- The class predictions may be replaced with class probabilities.
- A m-way cross-validation may be used on the first level, where only $(m-1)$ folds are used for training and the second level classifier is trained on whole dataset.
- This approach is very flexible and reduces both bias and variance.
- Other ensemble approaches may be viewed as special cases of Stacking (i.e. majority voting in second level, etc.).

Questions?