ENSEMBLE LEARNING

TRU CAO

HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY
AND JOHN VON NEUMANN INSTITUTE

OUTLINE

- Overview
- Bagging
- Boosting
- Other ways

OVERVIEW

- There are different approaches to combination of machine learning models:
  - Training different models and then making predictions using the average of the predictions made by each model.
  - Training multiple models in sequence in which the error function used to train a particular model depends on the performance of the previously trained models.
  - Selecting one model to make the prediction in which the selection is a function of the input.

BAGGING

- Bootstrap data sets:
  - Original data set: $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$.
  - Creation of a new data set $\mathbf{X}_B$: draw N points at random from $\mathbf{X}$, with replacement, so that some points in $\mathbf{X}$ may be replicated in $\mathbf{X}_B$ (where as other points may be absent from $\mathbf{X}_B$).

BAGGING

- Training the same model on M multiple bootstrap data sets.
- Let $y_m(\mathbf{x})$ is a predictive model trained on data set m.

BAGGING

- Training the same model on M multiple bootstrap data sets.
- Let $y_m(\mathbf{x})$ is a predictive model trained on data set m.
- The committee prediction:

$$y_{COM}(\mathbf{x}) = \frac{1}{M} \Sigma_{m=1..M}\, y_m(\mathbf{x})$$

BAGGING

- Let h(**x**) be the true function and e_m(**x**) is the error of model m:

$$y_m(\mathbf{x}) = h(\mathbf{x}) + \epsilon_m(\mathbf{x}).$$

- The expected squared error of model m:

$$\mathbb{E}_{\mathbf{x}}\left[\{y_m(\mathbf{x}) - h(\mathbf{x})\}^2\right] = \mathbb{E}_{\mathbf{x}}\left[\epsilon_m(\mathbf{x})^2\right]$$

BAGGING

- The average expected squared error:

$$E_{\mathrm{AV}} = \frac{1}{M}\sum_{m=1}^{M}\mathbb{E}_{\mathbf{x}}\left[\epsilon_m(\mathbf{x})^2\right]$$

- The committee expected squared error:

$$\begin{aligned} E_{\mathrm{COM}} &= \mathbb{E}_{\mathbf{x}}\left[\left\{\frac{1}{M}\sum_{m=1}^{M} y_m(\mathbf{x}) - h(\mathbf{x})\right\}^2\right] \\ &= \mathbb{E}_{\mathbf{x}}\left[\left\{\frac{1}{M}\sum_{m=1}^{M} \epsilon_m(\mathbf{x})\right\}^2\right] \end{aligned}$$

BAGGING

- Assume the errors have zero mean and are uncorrelated:

$$\begin{aligned} \mathbb{E}_{\mathbf{x}}\left[\epsilon_m(\mathbf{x})\right] &= 0 \\ \mathbb{E}_{\mathbf{x}}\left[\epsilon_m(\mathbf{x})\epsilon_l(\mathbf{x})\right] &= 0, \qquad m \neq l \end{aligned}$$

- The committee squared error is reduced by M times:

$$E_{\mathrm{COM}} = \frac{1}{M} E_{\mathrm{AV}}$$

BOOSTING

- AdaBoost (Adaptive Boosting):
  - Multiple base classifiers are trained in sequence.
  - Each base classifier is trained using a weighted form of the same data set that depends on the performance of the previously trained base classifiers (previously misclassified data points are given more weights).
  - All the trained base classifier are combined for prediction through a weighted majority voting scheme.

BOOSTING

AdaBoost

1. Initialize the data weighting coefficients $\{w_n\}$ by setting $w_n^{(1)} = 1/N$ for $n = 1, \ldots, N$.
2. For $m = 1, \ldots, M$:
   (a) Fit a classifier $y_m(\mathbf{x})$ to the training data by minimizing the weighted error function

$$J_m = \sum_{n=1}^{N} w_n^{(m)} I(y_m(\mathbf{x}_n) \neq t_n) \qquad (14.15)$$

   where $I(y_m(\mathbf{x}_n) \neq t_n)$ is the indicator function and equals 1 when $y_m(\mathbf{x}_n) \neq t_n$ and 0 otherwise.

Bishop, C.M. (2006) Pattern Recognition and Machine Learning.

BOOSTING

(b) Evaluate the quantities

$$\epsilon_m = \frac{\sum_{n=1}^{N} w_n^{(m)} I(y_m(\mathbf{x}_n) \neq t_n)}{\sum_{n=1}^{N} w_n^{(m)}} \tag{14.16}$$

and then use these to evaluate

$$\alpha_m = \ln \left\{ \frac{1 - \epsilon_m}{\epsilon_m} \right\}. \tag{14.17}$$

(c) Update the data weighting coefficients

$$w_n^{(m+1)} = w_n^{(m)} \exp \{- \alpha_m I(y_m(\mathbf{x}_n) \neq t_n)\} \tag{14.18}$$

BOOSTING

3. Make predictions using the final model, which is given by

$$Y_M(\mathbf{x}) = \operatorname{sign}\left(\sum_{m=1}^{M} \alpha_m y_m(\mathbf{x})\right). \qquad (14.19)$$

OTHER WAYS

- Only produce an output when more than half of the base classifier agree.
- The probability of the ensemble getting the correct answer is a binomial distribution:

$$\sum_{k=T/2+1}^{T} \binom{T}{k} p^k (1-p)^{T-k},$$

where p is the success rate of each base classifier, and T is the number of base classifiers.

OTHER WAYS

- The power of ensemble learning: if $p > 0.5$ then the correctness probability approaches 1 as $T \rightarrow \infty$.