Ensembles (03/08/2021)

Admin

- HW #3 due today (03/08)
- Project proposal due 03/11
- No live online class on Wed

We are aiming to increase the performance of our classifier by combining multiple models (possibly at the risk of increased complexity)

So, why should we learn multiple classifiers and combine them together?

- Statistical reasons (low bias and high variance)
- Computational reasons
- Representational reasons
- Too much data or too little data

Condorcet Jury Thm (1785)

Suppose we have an odd number of classifiers, T, for a two class problem. The probability of a correct classification is p and the outputs of the individual classifiers are independent. Also, the outputs of the classifiers are combined with a simple majority vote. So, in order to get a correct classification we need $\lfloor T/2 \rfloor + 1$ classifiers to get a correct classification. The accuracy of the majority vote ($\lfloor T/2 \rfloor + 1$ successes over T Bernoulli trials

$$P_{ens} = \sum_{t=\lfloor T/2 \rfloor + 1}^{T} \binom{T}{t} p^t (1-p)^{T-t}$$

- If $p > 1/2$ then $P_{ens} \to 1$ as $T \to \infty$
- If $p < 1/2$ then $P_{ens} \to 0$ as $T \to \infty$

Piazza Clarification

$\alpha_i^s \rightarrow$ Lagrange multipliers from source svm

$$W_s = \sum_{i=1}^{n_s} \alpha_i^s y_i x_i^s \in \mathbb{R}^2$$

q

source data

$W_T = \cdots$

$$\hat{y} = W_s^T x + b$$

$W_T \in \mathbb{R}^2$

$(1\times 2)(2\times 1) \rightarrow 1\times 1$

q_i

$$\min \frac{1}{2}\alpha^T H \alpha + q^T \alpha$$

$$\text{s.t.} \quad A\alpha = b$$

$$0 \le \alpha \le d$$

target

$$\sum_{i=1}^{n} \alpha_i (1 - B w_s^T x_i)$$

n

source

$$\sum_{i=1}^{n} \alpha_i q_i = 1$$

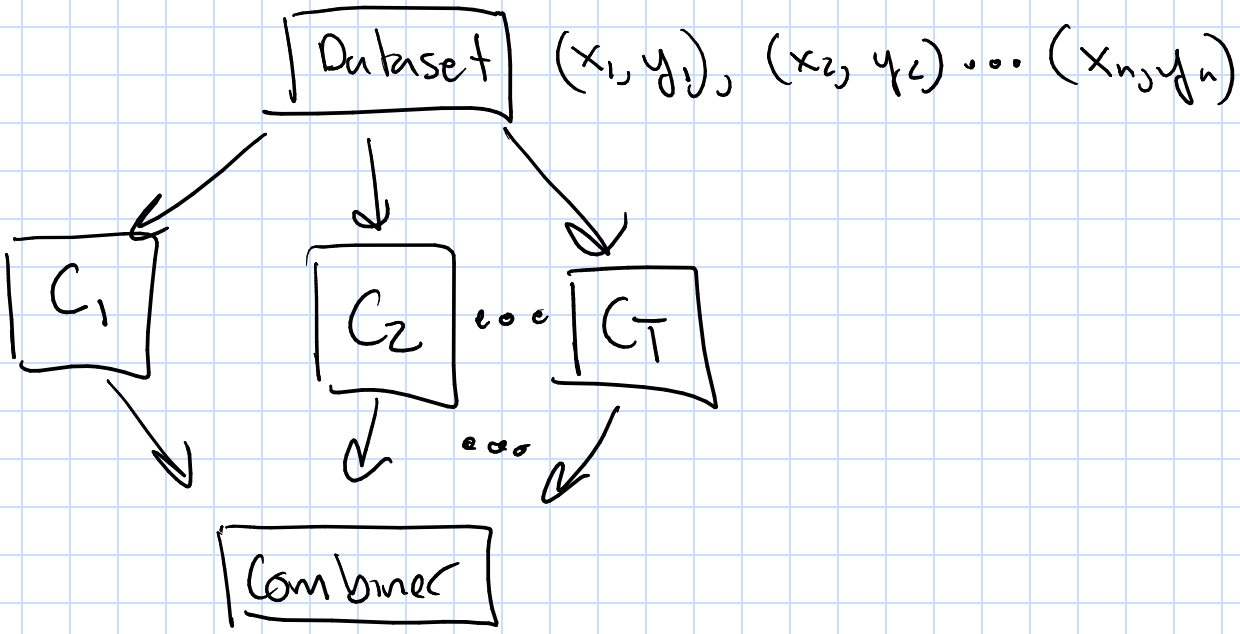
$$\max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j x_i^T x_j$$

$$\text{s.t.} \quad \sum \alpha_i y_i = 0$$

$$0 \le \alpha_i \le C$$

$$W_s = \sum_{i=1}^{n} \alpha_i y_i x_i$$

Central idea



The Bootstrap

A bootstrap dataset is one that is created from an original data set $D = \{(x_i, y_i)\}_{i=1}^{n}$ by sampling n points at random (with replacement) from D.

$$D = \begin{Bmatrix} (x_1, y_1) \\ (x_2, y_2) \\ (x_3, y_3) \end{Bmatrix} \begin{matrix} [1] \\ [2] \\ [3] \end{matrix}$$

1 1 2
2 3 2

$$D_1 = \begin{Bmatrix} (x_2, y_2) \\ (x_3, y_3) \\ (x_2, y_2) \end{Bmatrix}$$

$\approx$ 63% of the samples are unique

Bootstrap Aggregation (Bagging) Random Forest

Input: $S = \{(x_i, y_i)\}_{i=1}^{n}$, Round T, Classifier C

Training

for $t = 1, \ldots T$

① S_t is an n sample bootstrap from S

② Randomly choose $k < d$ features

③ Learn C_t from S_t

Testing [Majority Vote]

① $V_{t,j} = \begin{cases} 1 & C_t \text{ chose } w_j \\ 0 & \text{otherwise} \end{cases}$

② $V_j = \sum_{t=1}^{T} V_{t,j}$

③ Choose the w_j with the most votes

Diversity

- Instability is not a bad trait with Bagging. Instability adds diversity into the ensemble.
- Why does bagging work? Bagging takes the average of multiple classifiers to reduce the variance