NPTEL Week-7 Live Session

on Machine Learning and Deep Learning - Fundamentals and Applications (noc24_ee146)

A course offered by: Prof. Manas Kamal Bhuyan, IIT Guwahati

NPTEL Quiz Solution: Week-6

(Ensemble classifiers)

Bagging.

Boosting

Stacking.

Powered by:

FREE ONLINE EDUCATION swayam

शिक्षित भारत, उन्नत भारत

भारतीय प्रौद्योगिकी संस्थान गुवाहाटी • Indian Institute of Technology Guwahati •

PMRF

Prime Minister's Research Fellows

Ministry of Education

Government of India

सत्यमेव जयते

भारतीय प्रौद्योगिकी संस्थान पटना

INDIAN INSTITUTE OF TECHNOLOGY PATNA

विद्यार्थी लभते विद्याम

By

Arka Roy

NPTEL PMRF TA

Prime Minister's Research Fellow

Department of Electrical Engineering, IIT Patna

Web: https://sites.google.com/view/arka-roy/home

1) In AdaBoost, the final prediction is determined by

- A ~~simple~~ average of weak learners' predictions
- The ~~prediction~~ of the last weak
- A weighted sum of weak learners' predictions
- A random selection of weak learners' predictions

$\hat{y} \in \hat{y}_1 \,||\, \hat{y}_2 \,||\, \hat{y}_3$ ✗

linear combination.

$w_1 h_1 + w_2 h_2 + \cdots$

Refers to the prediction from weak learners.

2) Boosting algorithms assign: → Down weighting → For Correctly classified samples.
→ upweighting. → For mis classified samples

- ~~Equal weights to all training instances~~
- ✓ Weights based on the difficulty of each instance
- ~~Weights inversely proportional to the class distribution~~
- ~~Random weights to each base model~~

For mis classified samples:-

W_t, ↓ initial weight

$$\underbrace{w_{t+1} = W_t \cdot \exp(\alpha_t)}_{\text{upweighting.}}$$

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

For Correctly classified samples:-

$$\underbrace{W_{t+1} = W_t \cdot \exp(-\alpha_t)}_{\text{Down weighting.}}$$

ϵ_t :- misclassification error rate

3) Consider a binary classification problem where a single weak classifier has an accuracy of 60%. By combining multiple classifiers through majority voting, the overall accuracy improves. If 5 independent classifiers are combined, what is the probability that the ensemble's accuracy exceeds 60%?

- 50%
- 60.25%
- 90.63%
- 68.26%

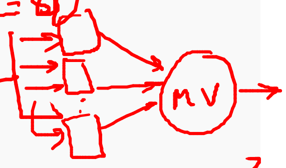
The problem is binary classification problem, $y = \{+1, -1\}$ class.

Acc = 60% when only single model is used.

Now 5 classifiers have been employed.

P(Accuracy > 60% while using 5 classifiers with MV) = $P(X=3) + P(X=4) + P(X=5)$

Bagging.



Bagging.

* Lets say you are using 5 classifiers, then for what condition you can say the Accuracy of majority voting is 60%.

(x, SPAM)

$\hat{y}_i = \{ NS, SP, SP, SP, NP \}$ $N_s = 5$.

MV

$\hat{y}_{final} = \text{SPAM}$

$P = \frac{3}{5} \times \frac{20}{100}$

60

3) Consider a binary classification problem where a single weak classifier has an accuracy of 60%. By combining multiple classifiers through majority voting, ~~the overall accuracy improves~~. If 5 independent classifiers are combined, what is the probability that the ensemble's accuracy exceeds 60%?

- 50%
- 60.25%
- 90.63%
- ~~68.26%~~

$$P(60\% \text{ or above after MV}) = \underbrace{P(X=3)}_{60\%} + \underbrace{P(X=4) + P(X=5)}_{>60\%}$$

$$= \binom{5}{3}(0.6)^3(0.4)^2 + \binom{5}{4}(0.6)^4(0.4)^1 + \binom{5}{5}(0.6)^5(0.4)^0$$

$$= \frac{5!}{3!\,2!} + \frac{5!}{4!\,1!} + \frac{5!}{5!\,0!}$$

$$= 0.3456 + 0.2592 + 0.07776$$

$$= 0.6825$$

$y = \{+1, -1\} \Rightarrow$ follows Bernoulli distribution.

$\{yes, no\} \rightarrow$ Discrete case.

$$P(X=k) = \binom{N}{k} p^k (1-p)^{N-k}$$

p → 60%. $(1-p)$ → 40%.

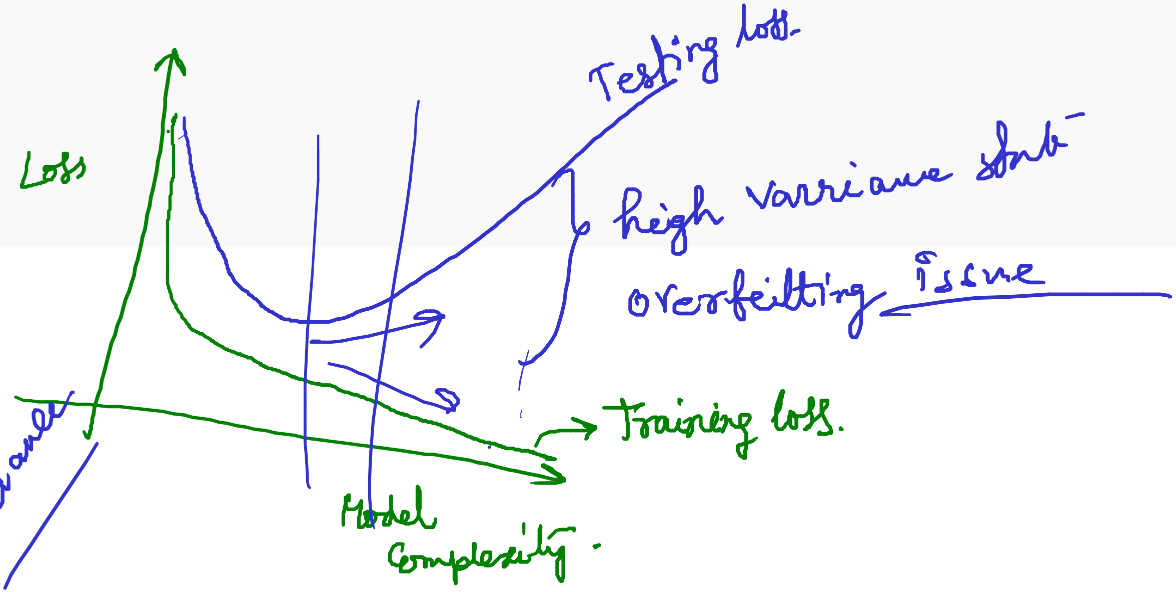
4) _________ is useful when high variance models are available.

overfitting.

- AdaBoost
- Stacking
- Boosting
- Bagging

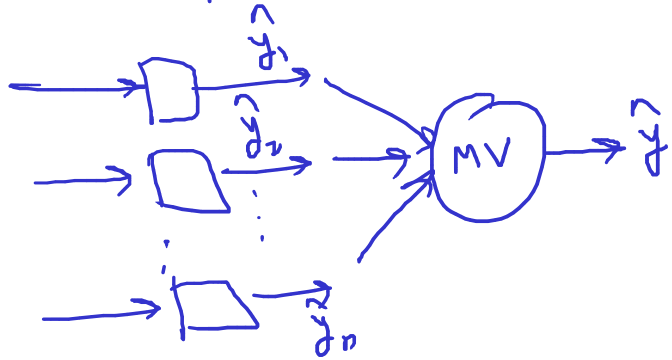
Bagging helps to reduce overfitting ~ high variance

overfitting



5) Bagging techniques combine the predictions of base models by:

- ○ Taking the average of their predictions
- ○ Taking the maximum or majority vote of their predictions
- ○ Weighting the predictions based on their performance (Boosting)
- ○ Ignoring the predictions of the base models



6) In the ~~third~~ Second. iteration of AdaBoost, the weight assigned to a misclassified data point is 0.4. If the initial weight for all data points is 1, what was the misclassification rate of this data point at the end of the second iteration?

- 70%
- 90%
- 81%
- 86%

$w_2 = 0.4$ ✓

$w_1 = 1$ ✓

$\varepsilon = ?$

$\alpha = \frac{1}{2}\ln\left(\frac{1-\varepsilon}{\varepsilon}\right)$

$\boxed{w_{t+1} = w_t \cdot e^{\alpha}}$

$w_{t+1} = w_t \cdot e^{\frac{1}{2}\ln\left(\frac{1-\varepsilon}{\varepsilon}\right)}$

$= w_t\, e^{\ln\left(\frac{1-\varepsilon}{\varepsilon}\right)^{1/2}}$

$\boxed{w_{t+1} = w_t \cdot \left(\frac{1-\varepsilon}{\varepsilon}\right)^{1/2}}$

$\Rightarrow 0.4 = \left(\frac{1-\varepsilon}{\varepsilon}\right)^{1/2}$

$\Rightarrow 0.16\varepsilon = 1-\varepsilon$

$\Rightarrow 1.16\varepsilon = 1$

$\Rightarrow \varepsilon = \frac{1}{1.16} = 0.8620$

$\varepsilon\% = 86.20\%$

7) Which of the following does bagging help to reduce in machine learning models? (

- [x] Model variance
- [ ] Bias → Boosting Technique helps in reducing the model's bias.
- [x] Overfitting
- [ ] Model complexity

bagging helps to reduce overfitting

overfitting ≈ high variance

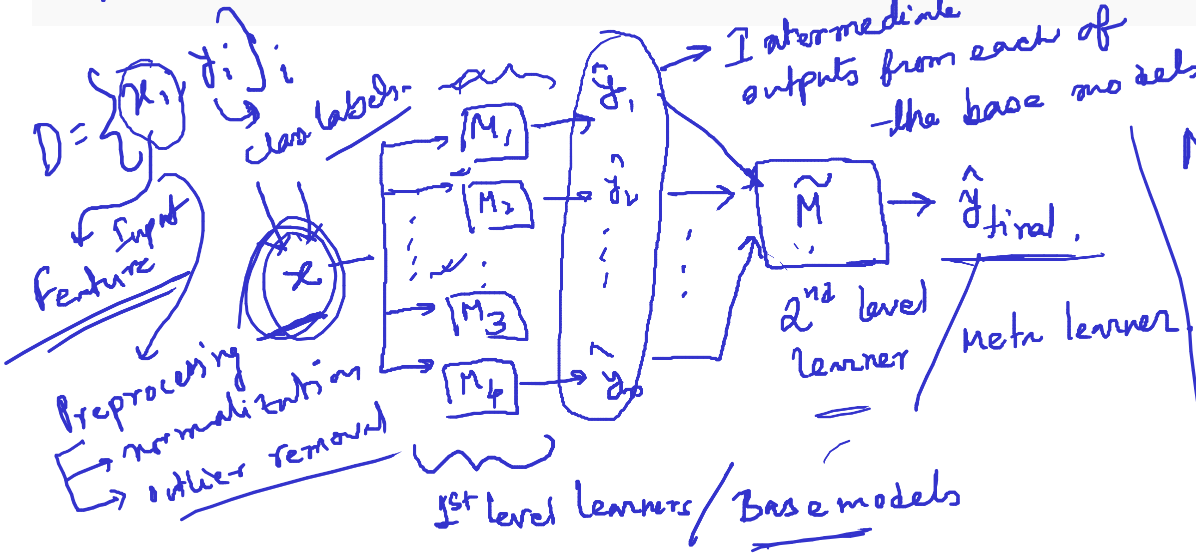
8) Random Forest can handle

- Large datasets
- Linear and nonlinear data
- Classification and regression tasks
- All of the above

10) In stacking, what is a meta-learner?

- ~~A model that performs data preprocessing~~
- ~~A base model trained on the original dataset~~
- A model trained on the predictions of base models ✓
- ~~A model used to aggregate the predictions of base models~~

$$\hat{y} = \frac{1}{N}(\hat{y}_1 + \hat{y}_2 + \dots + \hat{y}_n)$$

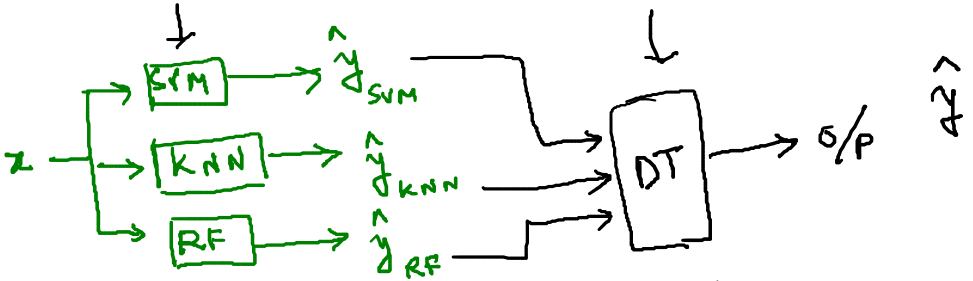


Meta learner ($\tilde{M}$) learns from the previous outputs yielded from the base models!!

Stacking:- $\{D^n = \{x_i, y_i\}\}_{i=1}^{N}$

x_i → Feature

y_i → class labels (categorical variables)



First level learners

2nd level learner

Measure the performance between the predicted class label ($\hat{y}$) and the original class label (y).

* 2nd level learner learns from the output of first level learner.

Bagging :- $D = \{x_i, y_i\}_{i=1}^{N}$

$y = \{$ cancer, ~~...~~ healthy $\}$

$\tilde{x} \rightarrow$ Randomly sampled from $D = \{x, y\}_i$ with replacement ($\tilde{x}$) & we can do the sampling 'N' no of times if we have 'N' such classifiers.

$\tilde{x} \rightarrow$ SVM, linear $\rightarrow \hat{y}_1 = C$

$\tilde{x} \rightarrow$ SVM, quadratic $\rightarrow \hat{y}_2 = H$

$\tilde{x} \rightarrow$ SVM, cubic $\rightarrow \hat{y}_3 = H$

$\tilde{x} \rightarrow$ RF $\rightarrow \hat{y}_4 = H$

$\tilde{x} \rightarrow$ KNN, k = 3 $\rightarrow \hat{y}_5 = C$

$\hat{y}_1, \ldots, \hat{y}_5 \rightarrow$ Majority voting $\rightarrow \hat{y} \mid \tilde{x} \in H$

Decision making.

* $\tilde{x} \in x$ by random sampling with replacement

* you have N such classifiers working parally.

* Decision is taken based on the majority voting of the outputs recieved from 'n' no of classifiers.

$\text{Mode}(\hat{y}_1, \hat{y}_2, \hat{y}_3, \hat{y}_4, \hat{y}_5 \ldots)$

$= \text{Mode}(C, \underbrace{H, H, H}, C, \ldots)$

$= \text{Mode}(H) = 3$

$\text{Mode}(C) = 2$

* Relying on single classifier may lead to overfitting. however Bagging helps to mitigate this issue by considering majority voting.

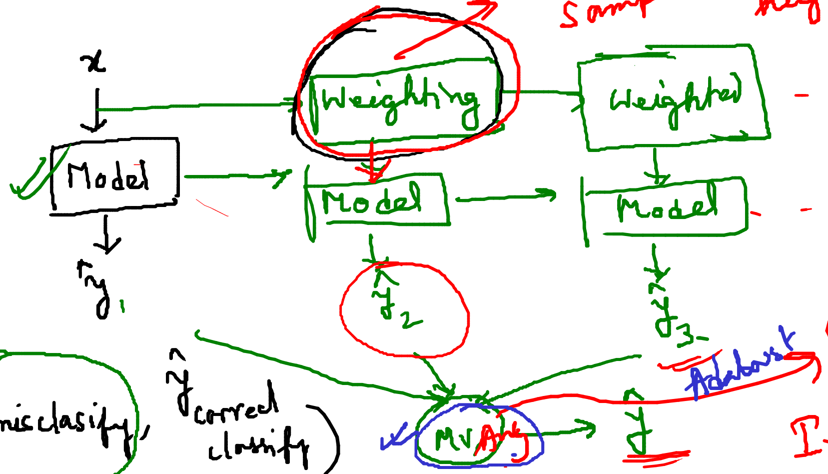
Boosting:- $D = \{x_i, y_i\}_{i=1}^{N}$

Misclassified samples will be highlighted more

α (weighting factor).

$\alpha = \frac{1}{2}\ln\left(\frac{1-e}{e}\right)$

e = error rate for a misclassified data



Weighted linear combination of all the individual outputs

Implementation Algorithm

$\hat{y} \cup (\hat{y}_{misclasify}, \hat{y}_{correct\ classify})$

* Find the misclassified samples.
* Put some sort of weighting factor to up weight the misclassified samples and downweight the correctly classified samples and again train the model.
* If you find error again follow step 2