Guess the word?

Agenda:

1. Ensemble Techniques
2. Types of Ensemble
3. Bagging
4. Random Forest
5. Bias Variance
6. Scikit-Code

Ensemble Techniques.

→ group of things / items

→ multiple models used together

→ a powerful model

$M_1, M_2, M_3 \ldots M_k$ → "Base learners"

(e.g.) Survival → "M" → More Powerful than each.

Types of Ensembles.

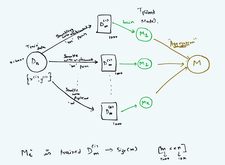
1. Bagging → RF
2. Boosting → XGBoost
3. Stacking → "Stagg"
4. Cascading → Training / Hard Attrib.

Key Aspect: $\underbrace{[M_1, M_2, M_3 \ldots M_k]}_{\text{Combiner } (M)}$ Base Models

Bagging [Bootstrapped Aggregation]

Bootstrap: Sampling → "Sampling with replacement"

Aggregation → avg, count, weighted, median ...

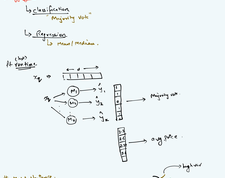


M_i is trained $D'_m \rightarrow$ $\underset{\text{train}}{\text{Sy}}(x)$ $\underset{\text{train}}{\text{Sy}}$

Aggregation

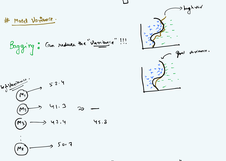
1. Classification
   "Majority Vote"
   → Regression
   → mean / median.

If Classification:



Model Variance.

Bagging → Low variance !!!



Variance:

M_1 → 5:9

M_2 → 41:3 ⇒ —

M_3 → 43:7 41:3

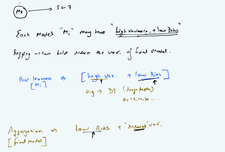
M_4 → 50:9

Each model "M" may have "high variance, or low bias"

Bagging → can take low values as input of final model.

Aggregation → low bias + reduced var. [final model]

Why Bagging reduces the variance?



Bagging → σ^2_M

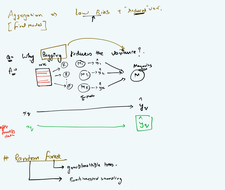
σ^2_M → $\frac{\sigma^2}{n}$

Random Forest

→ guassian/white noise.

→ Contrast/Variance smoothing

RF = DT + Row sampling + Feat sampling + Aggregation



$d' < d$
$n' < n$

(Bagging)

$(R \times C \times D)$

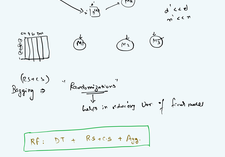
Bagging ⇒ "Randomization"

↓

taken in ordering that of final model

RF: DT + $R \times C \times D$ + Agg

← OOB.



Q: why the RF always outperform DT?

A: Var. part of the error.

Q: RF = DT.

when data is noisy.

$\boxed{\text{Bias}} + \boxed{\text{Variance}} = \boxed{\text{Error}}$

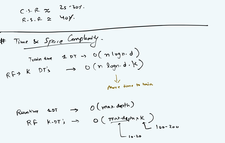
$D \rightarrow$ Bias + σ^2

Bias can be improved by hyper-tuning (K)

Bias & Variance Tradeoff

1. RF → takes ordering random.
   → Row Bias + Feat. Bias (DT) are also low bias.
   → Variance: $Var(x) = agg\,[m_1, m_2, m_3 \ldots m_k]$

   $K \uparrow$ → $\sigma \downarrow$
   $K \downarrow$ → $\sigma \uparrow$



2. $\underset{N=8}{D_N} \xrightarrow[C=3]{R=4} D'$ $d' < d$ $n' < n$

d' → Col. sampling ratio ; n' → row. sampling ratio

$C:R \downarrow$ ⇒ overfitting ↓
$R:S.R \downarrow$ ⇒ overfitting ↓

$C:R \approx$ 2:3 best.
$R:S.R \approx$ 4:5.

Time & Space Complexity.

Tree fit: $O(n \log n \cdot d)$

RF → K DTs → $O(n \log n \cdot d \cdot K)$

Row sort is main

Prediction: $O(\text{max depth})$

RF: $K \cdot O(\text{max depth} \cdot K)$ → $\underset{\text{Tree-Tree}}{\text{Tree-Leaf}}$

$\Rightarrow$ $\underset{\text{Tree}}{\text{Leaf}}$

Space Complexity: $O(\text{max depth} \times K)$

"Multi-Core Algo"

Popular

M_1 M_2 M_3 ... M_N

$K \uparrow \Rightarrow$ complexity (RF/Bagg) ↓