Slide4

Friday, June 4, 2021 9:36 PM

TutorialSlide4

University of Glasgow

STATS5099: Data Mining

Tree-based methods

Xiaochen Yang
xiaochen.yang@glasgow.ac.uk

University of Glasgow

This week's content

- Classification trees
- Bagging and random forest

University of Glasgow

Classification trees

- Non-parametric classifier

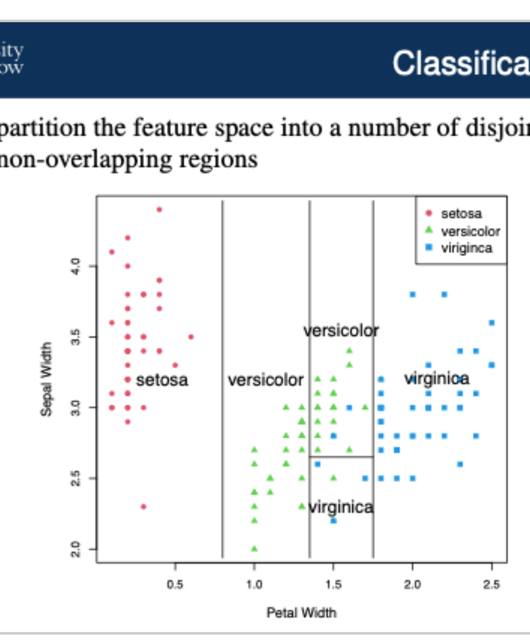
1. partition the feature space into a number of disjoint and non-overlapping regions
2. predict the class of a given observation as the most commonly occurring class of training observations in the region to which it belongs

---

University of Glasgow

Classification trees

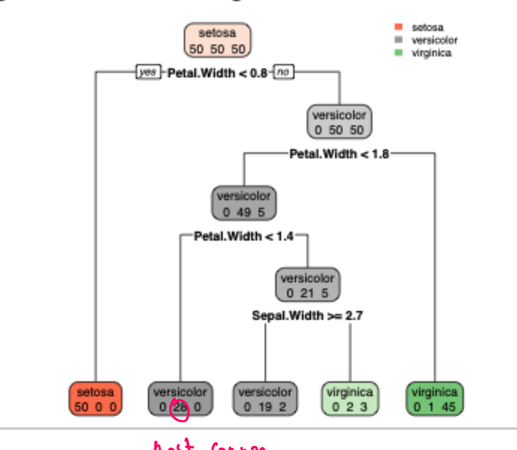
1. partition the feature space into a number of disjoint and non-overlapping regions



University of Glasgow

Classification trees

2. predict the class of a given observation as the most commonly occurring class of training observations in the region to which it belongs



most common

University of Glasgow

Step 1: partition the feature space

Build a tree (recursive binary split)

- top-down approach: begin at the the top of the tree, and successively splits the feature space
- binary split at each step
- best split: feature and cut-off point combination that leads to the largest possible reduction in error rate, Gini index or entropy

① ② ③ where to split

University of Glasgow

Step 1: partition the feature space

Build a tree (recursive binary split)

- top-down approach: begin at the the top of the tree, and successively splits the feature space
- binary split at each step
- best split: feature and cut-off point combination that leads to the largest possible reduction in error rate, Gini index or entropy

Avoid overfitting

- minimum number of observations in any terminal node
- maximum depth
- ...
- **Pruning**

University of Glasgow

Step 1: partition the feature space

Build a tree (recursive binary split)

- top-down approach: begin at the the top of the tree, and successively splits the feature space
- binary split at each step
- best split: feature and cut-off point combination that leads to the largest possible reduction in error rate, Gini index or entropy

Prune a tree

- Apply cost complexity pruning to the large tree in order to obtain a sequence of best subtrees, as a function of α.

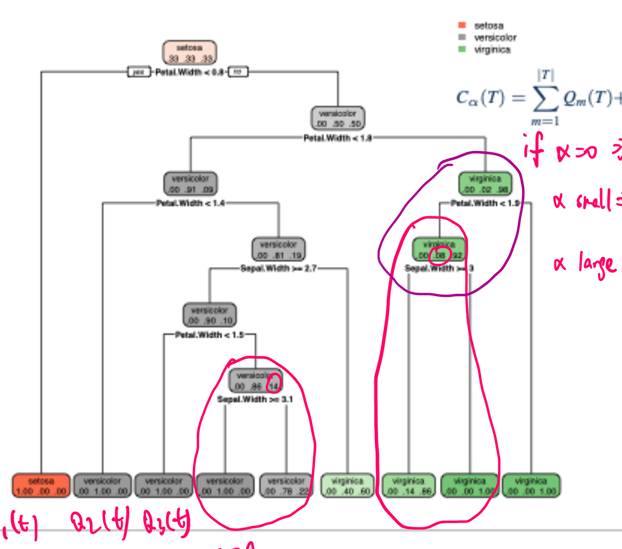
find α, s.t. (balance)

$$C_\alpha(T) = \sum_{m=1}^{|T|} \underbrace{Q_m(T)}_{\text{Gini index / mis clas rate} \downarrow} + \alpha \underbrace{|T|}_{\text{\# of terminals} \downarrow}$$

- Use the validation set or K-fold cross-validation to choose α.

University of Glasgow

Tree pruning



$C_\alpha(T) = \sum_{m=1}^{|T|} Q_m(T) + \alpha|T|$

if $\alpha = 0$ ⇒ not prune, since mis clas is low

α small ⇒ combine second ble 8<14

α large ⇒ cut purple ble 2<14
cut tree size

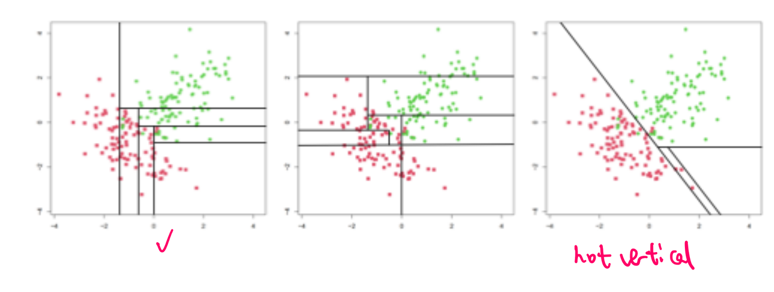
$Q_1(t)$ $Q_2(t)$ $Q_3(t)$

T=9

University of Glasgow

Conceptual question

Which of the following plot(s) can be generated by using recursive binary split?



✓ not vertical

University of Glasgow

Classification trees: summary

Pros:

- easy to explain, highly interpretable
- easily handle categorical variables and missing data

Cons:

- predictive accuracy is not very competitive
- very **non-robust**: a small change in the data can cause a large change in the final estimated tree

---

University of Glasgow

Bagging and random forests

- **ensemble** methods: combine the predictions from multiple models to make more accurate predictions than any individual model
- underlying statistical idea: averaging a set of observations reduces variance

  Given a set of n independent observations $Z_1, \ldots, Z_n$, each with mean μ and variance σ^2, the mean of $\bar{Z}$ is still μ and its variance is reduced to σ^2/n.
- bagging: average a set of *models*

University of Glasgow

Bagging and random forests

Bagging (bootstrap aggregation)

1. generate B different bootstrapped training data sets
2. build a prediction model $f_b(x)$ on the bth bootstrapped training set (all features)
3. average across all the models:

$$f_{\text{bag}}(x) = \frac{1}{B} \sum_{b=1}^{B} f_b(x)$$

(without pruning)
use large tree to bagging

University of Glasgow

Bagging and random forests

Random forest

1. generate B different bootstrapped training data sets
2. build a prediction model $f_b(x)$ on the bth bootstrapped training set *using a random sample of features* (models are not correlated)
3. average across all the models:

$$f_{\text{bag}}(x) = \frac{1}{B} \sum_{b=1}^{B} f_b(x)$$

- underlying statistical idea: averaging a set of *independent* observations reduces variance

  make sure all models are indepent

  Given a set of n *independent* observations $Z_1, \ldots, Z_n$, each with mean μ and variance σ^2, the mean of $\bar{Z}$ is still μ and its variance is reduced to σ^2/n.

University of Glasgow

Summary: considerations when choosing classifiers

- parametric *vs* nonparametric
- linear *vs* nonlinear decision boundary
- model complexity (tend to under- or over-fit?)
- computational cost (training, test)
- data: missing data, categorical features, correlated features ⇒ random forest

---

University of Glasgow

Tree-based methods in R

- classification tree:

  clas tree visualize

  `rpart`, `rpart.plot`, `printcp`, `prune`

  output:

  `variable.importance`
- bagging and random forest:

  `randomForest`, `varImpPlot` variable importance plot

  arguments:

  `mtry`: number of variables randomly sampled as candidates at each split

  bagging mtry = p
  random forest < p

  `ntree`: number of trees to grow