Decision Trees I

Introduction to Decision Trees

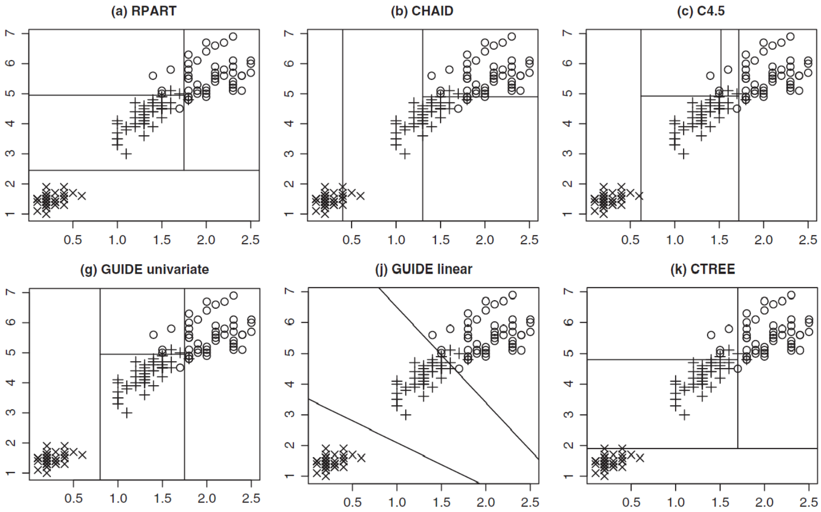
Introduction

Decision Trees

- Data-driven approach for relating X and Y
- Popular and (somewhat) easy to interpret
- Important building block (base learner) for ensemble methods
- Many different tree building algorithms exist (Zhang & Singer 2010, Loh 2014)
  - Focus on interaction detection, prediction, parameter instability...

Introduction

Figure: Decision Tree Algorithms



Loh (2014)

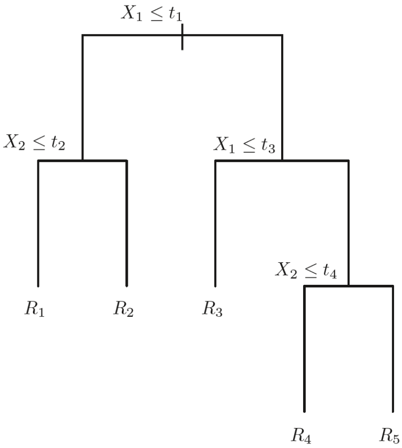
Classification and Regression Trees (CART)

Classification and Regression Trees (CART)

- Approach for partitioning the predictor space into smaller subregions via "recursive binary splitting"
- Results in a "top-down" tree structure with...
  - Internal nodes within the tree
  - Terminal nodes as endpoints
- Can be applied to regression and classification problems

Classification and Regression Trees (CART)

Figure: A small tree



James et al. (2013)

Tree growing

Growing a **regression tree**

Define pairs of regions for all $X_1, X_2, ..., X_p$ predictors and cutpoints c

$$\tau_L(j,c) = \{X|X_j < c\} \text{ and } \tau_R(j,c) = \{X|X_j \geq c\}$$

Find split s which maximizes the reduction in RSS

$$\Delta RSS(s,\tau) = RSS(\tau) - RSS(\tau_L) - RSS(\tau_R)$$

$$RSS(\tau) = \sum_{i \in \tau} (y_i - \hat{y})^2$$

with $\hat{y}$ being the mean of y in node τ

Tree growing

Growing a **regression tree**

Define pairs of regions for all $X_1, X_2, ..., X_p$ predictors and cutpoints c

$$\tau_L(j, c) = \{X | X_j < c\} \text{ and } \tau_R(j, c) = \{X | X_j \geq c\}$$

Find split s which maximizes the reduction in RSS

$$\Delta RSS(s, \tau) = RSS(\tau) - RSS(\tau_L) - RSS(\tau_R)$$

$$RSS(\tau) = \sum_{i \in \tau} (y_i - \hat{y})^2$$

with $\hat{y}$ being the mean of y in node τ

Tree growing

Growing a **classification tree**

Define pairs of regions for all $X_1, X_2, ..., X_p$ predictors and cutpoints c

$$\tau_L(j, c) = \{X|X_j < c\} \text{ and } \tau_R(j, c) = \{X|X_j \geq c\}$$

Find split s which maximizes the reduction in node impurity

$$\Delta I(s, \tau) = I(\tau) - p(\tau_L)I(\tau_L) - p(\tau_R)I(\tau_R)$$

Impurity measures

$$I_{Gini}(\tau) = \sum_{k=1}^{K} \hat{p}_k(1 - \hat{p}_k)$$

$$I_{entropy}(\tau) = -\sum_{k=1}^{K} \hat{p}_k \log \hat{p}_k$$

with $\hat{p}_k$ being the proportion of observations from class k in node τ

Tree growing

Growing a **classification tree**

Define pairs of regions for all $X_1, X_2, ..., X_p$ predictors and cutpoints c

$$\tau_L(j,c) = \{X|X_j < c\} \text{ and } \tau_R(j,c) = \{X|X_j \geq c\}$$

Find split s which maximizes the reduction in node impurity

$$\Delta I(s,\tau) = I(\tau) - p(\tau_L)I(\tau_L) - p(\tau_R)I(\tau_R)$$

Impurity measures

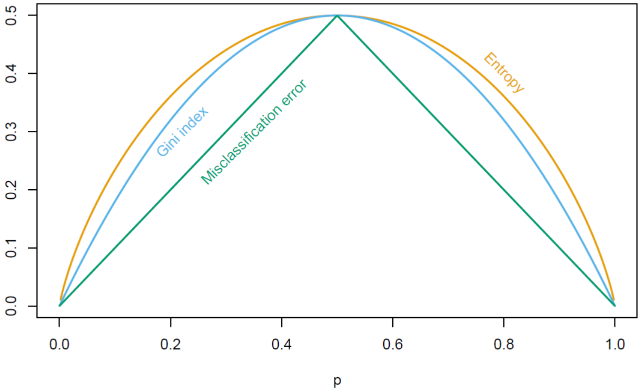
$$I_{Gini}(\tau) = \sum_{k=1}^{K} \hat{p}_k(1-\hat{p}_k)$$

$$I_{entropy}(\tau) = -\sum_{k=1}^{K} \hat{p}_k \log \hat{p}_k$$

with $\hat{p}_k$ being the proportion of observations from class k in node τ

Tree growing

Figure: Misclassification error, Gini index & entropy (scaled)



Hastie et al. (2009)

Tree growing

Algorithm 1: Tree growing process

```
Define stopping criteria;
Assign training data to root node;
if stopping criterion is reached then
    end splitting;
else
    find the optimal split point;
    split node into two subnodes at this split point;
    for each node of the current tree do
        continue tree growing process;
    end
end
```

Tree structure

A given tree

$$\mathcal{T} = \sum_{m=1}^{M} \gamma_m \cdot 1_{(i \in \tau_m)}$$

consists of a set of $m = 1, 2, \ldots, M$ nodes which can be used for prediction by...

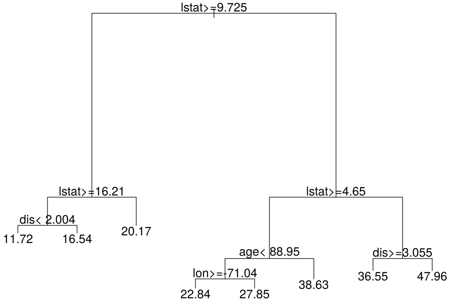
- Regression
  - ...using the mean of y for training observations in τ_m
- Classification
  - ...going with the majority class in τ_m

$\rightarrow$ Prediction surface: Block-wise relationship between features and outcome

Tree structure

Figure: CART examples

(a) Regression tree



(b) Classification tree