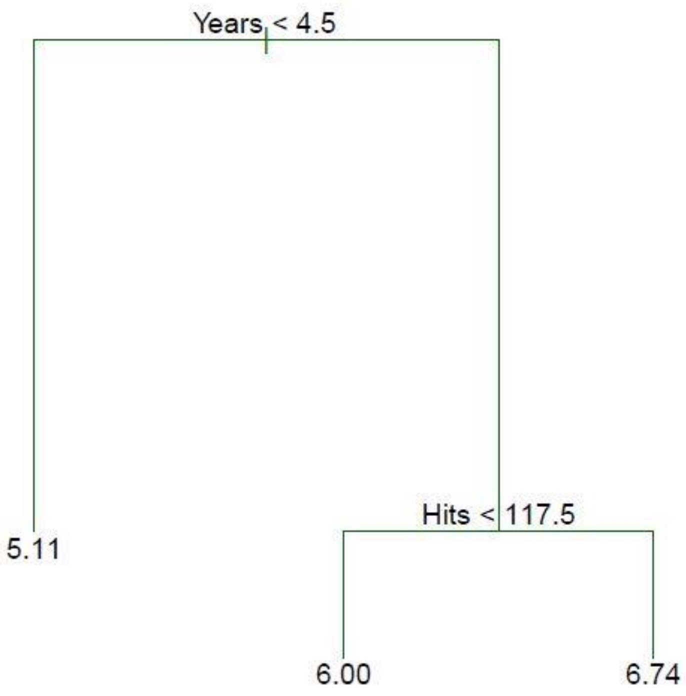
Metis Intro to Data Science

Fall 2018

Decision Trees and Ensembling

Decision Trees

- Goal: Segmenting the predictor space into a number of simple regions
- Benefits: Easily Interpretable
- Drawbacks: Variance and accuracy
- Regression trees
  - Response is continuous
  - Use the mean of the region as the predictive value
- Classification trees
  - Response is discrete (classes)
  - Use the mode of the region as the predictive value



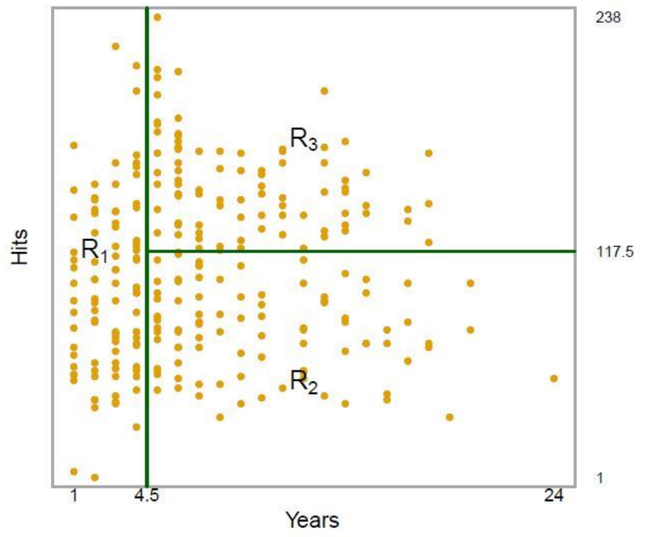
Use the Hitters data from the ISLR library for a simple example

```
data("Hitters")
# Remove incomplete cases
Hitters <- na.omit(Hitters)
kable(head(Hitters,3))
```

	AtBat	Hits	HmRun	Runs	RBI	Walks	Years	CAtBat	CHits	CHmRun
-Alan Ashby	315	81	7	24	38	39	14	3449	835	69
-Alvin Davis	479	130	18	66	72	76	3	1624	457	63
-Andre Dawson	496	141	20	65	78	37	11	5628	1575	225

Decision Trees

- From ISLR: hitters data set
- Interpretable anatomy



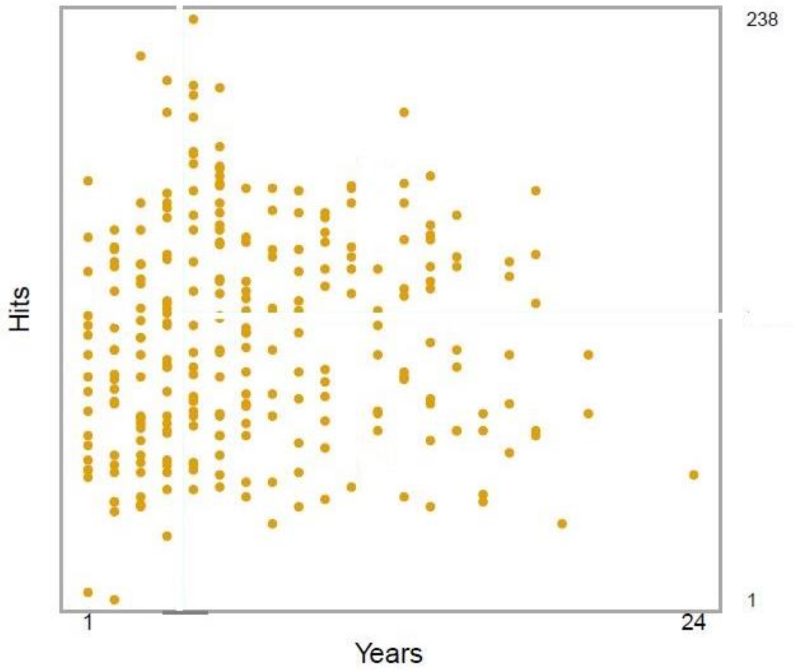
$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2$$

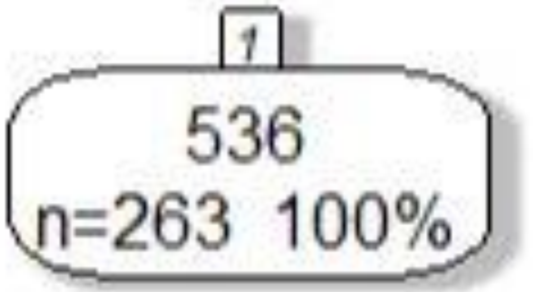
Decision Trees – Choosing regions

- Divide the predictor space into J distinct and non overlapping regions
- For every observation that falls into R_j we make the same prediction. The mean of the response values from the training set

Decision Trees – Choosing regions

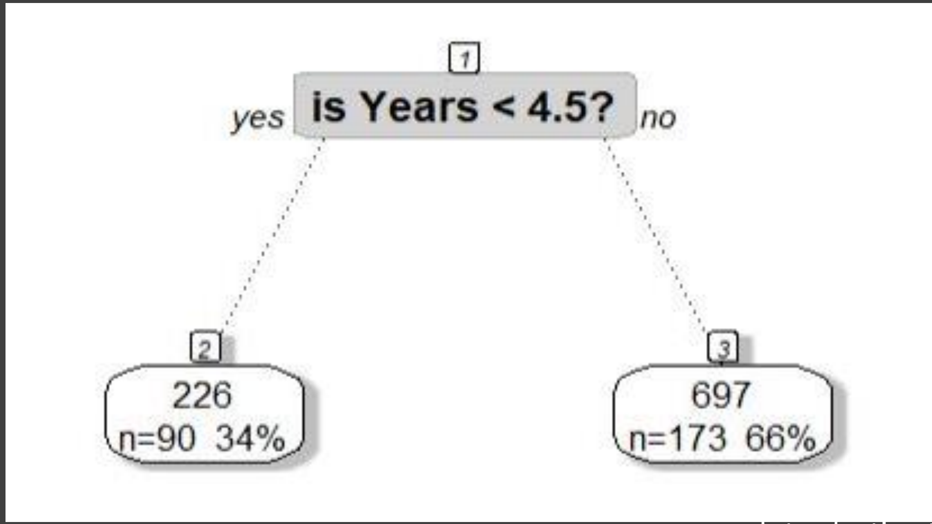
- Recursive Binary Splitting – greedy approach
- Start from the top of the tree where all the observations are in one single region
- Greedy because only the best split is made at that particular step, rather than looking ahead and picking a split that will lead to a better tree in some future step

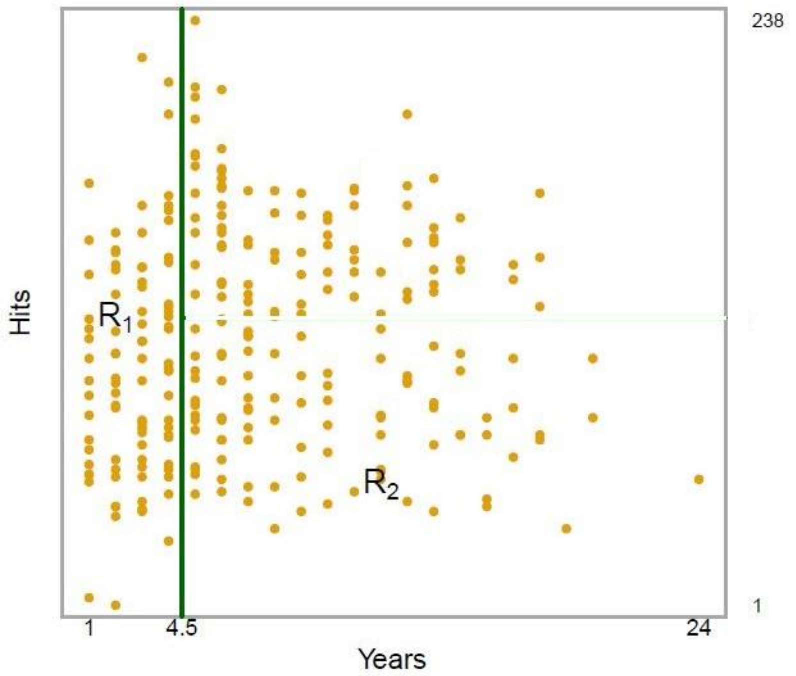




Decision Trees – Choosing regions

- Successively split the predictor space
- Each split will be indicated by 2 new branches down the tree





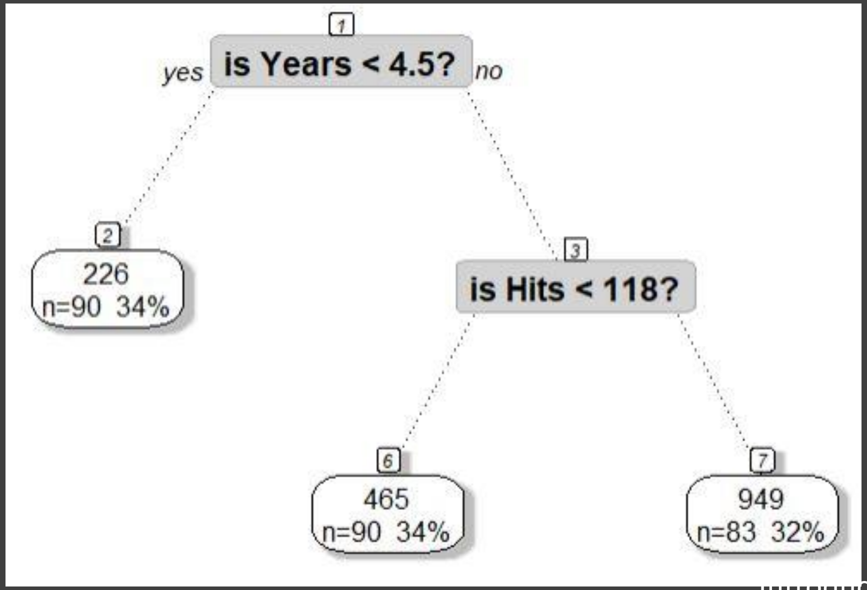
$$R_1(j,s) = \{X|X_j < s\} \text{ and } R_2(j,s) = \{X|X_j \geq s\},$$

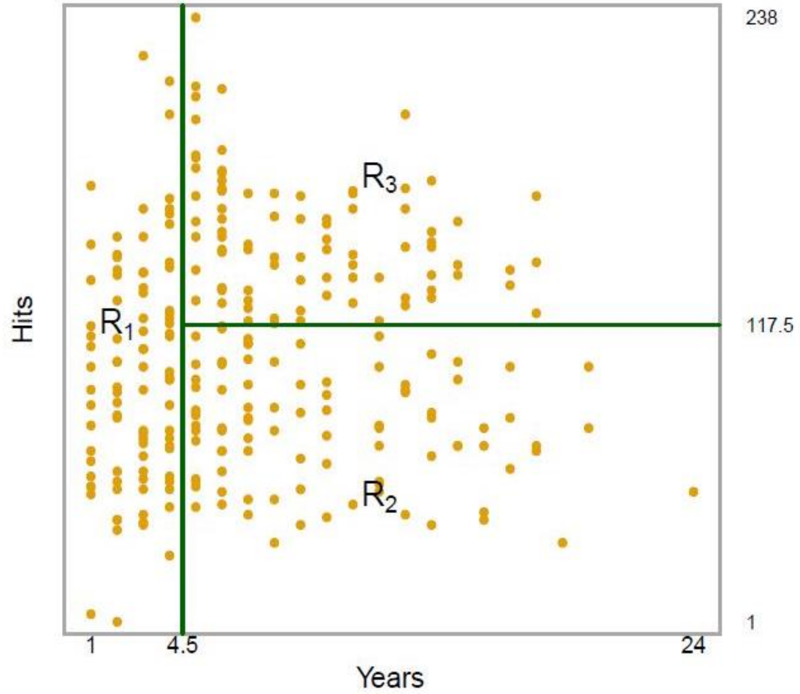
seek the value of j and s that minimize the equation

$$\sum_{i:\ x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i:\ x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2,$$

Decision Trees – Choosing regions

- Repeat the process looking for the best predictor and best cut point
- Minimize the Regional Sum of Squares





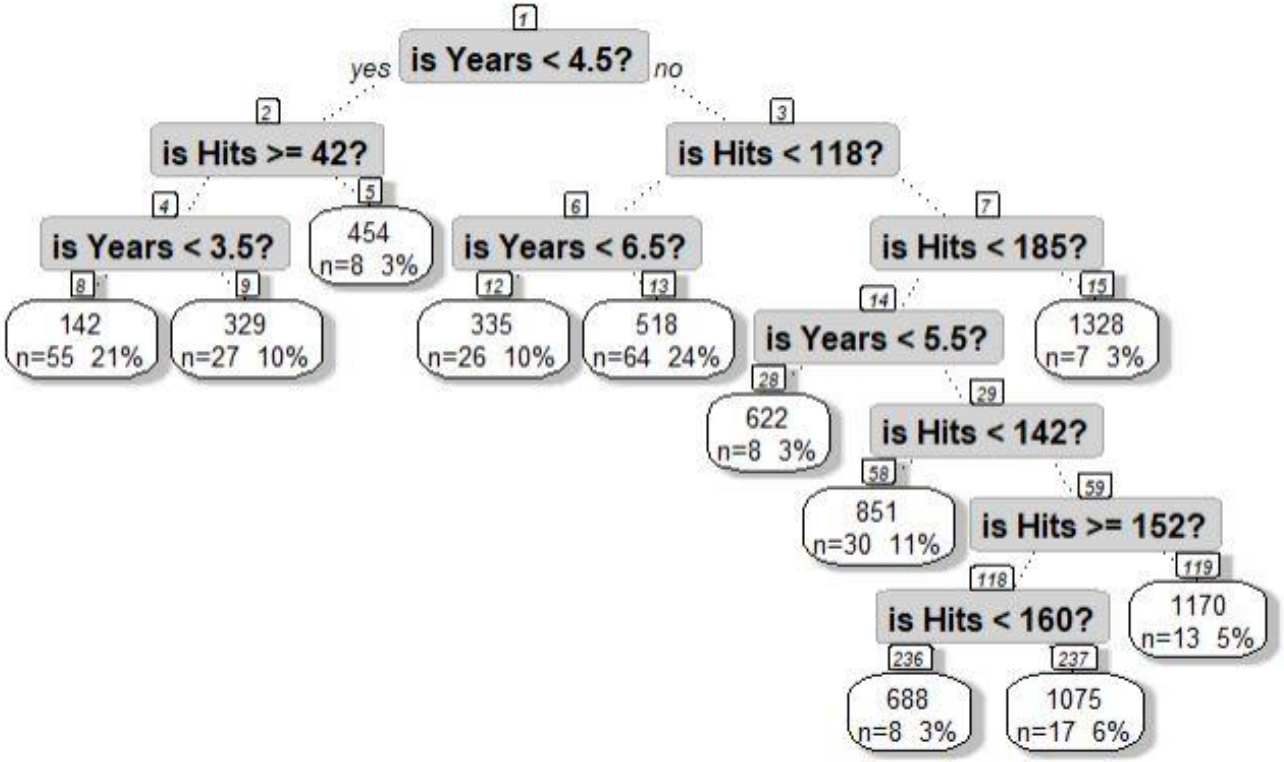
$$R_1(j,s) = \{X|X_j < s\} \text{ and } R_2(j,s) = \{X|X_j \geq s\},$$

seek the value of j and s that minimize the equation

$$\sum_{i:\, x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i:\, x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2,$$

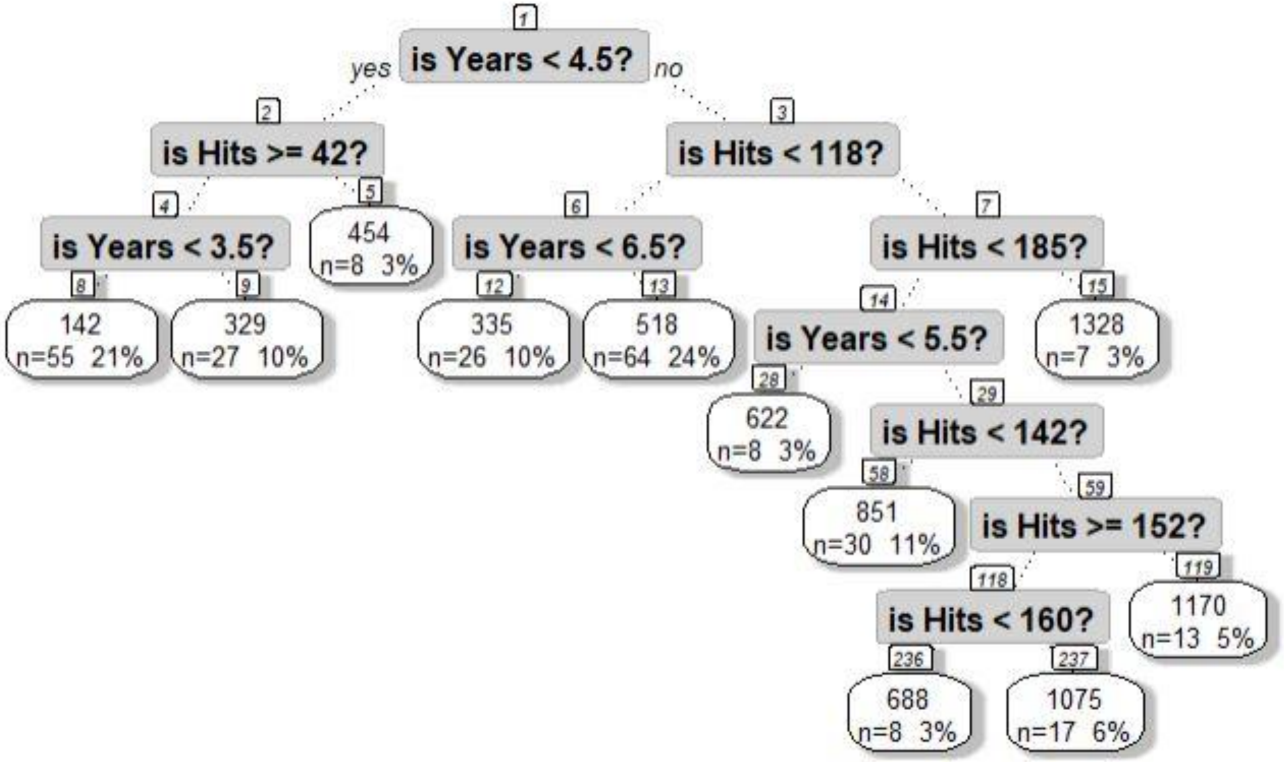
Decision Trees – Choosing regions

- Process continues until it reaches a stopping criterion
    - Example: Limit the number of observations in a node to 5
- We can now predict a new test observation by returning the mean of the training observations in the given region



Decision Trees – Pruning

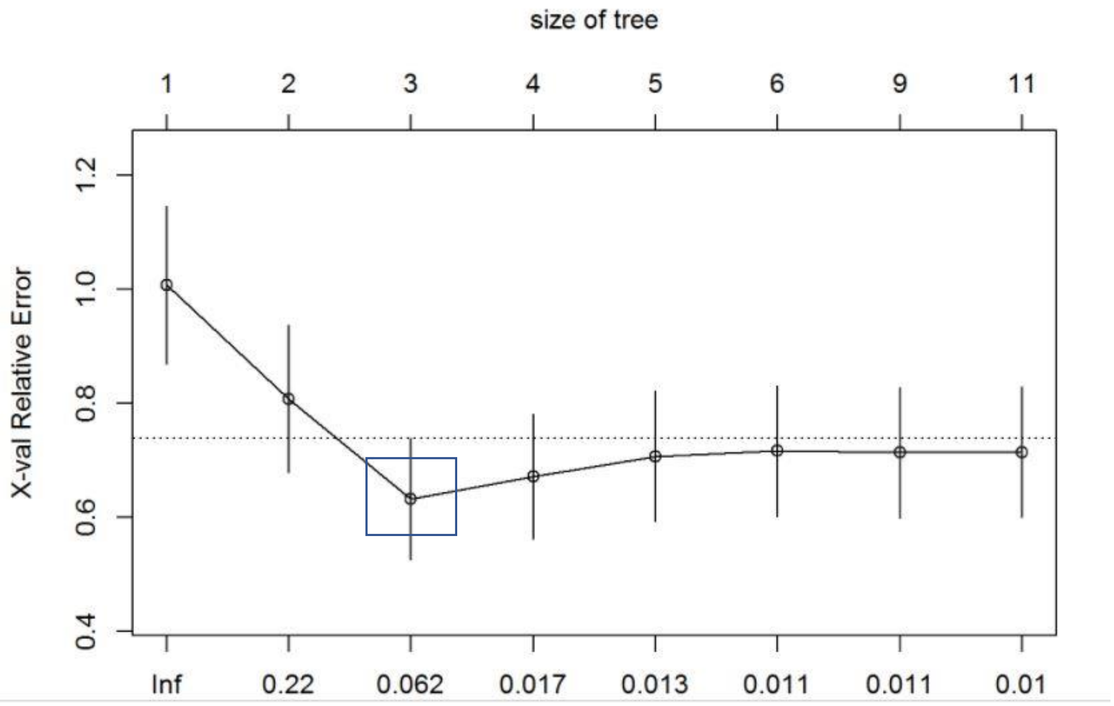
- Full tree may over fit the training dataset, leading to poor prediction on test.
- Pruning will lower the variance and increase the bias
- Consider sub-trees and look for the one with the lowest test error using cross validation



Decision Trees – Pruning

- For each additional node (sub tree)
  - Run cross validation
  - Return the error
  - Select the tree with the lowest error

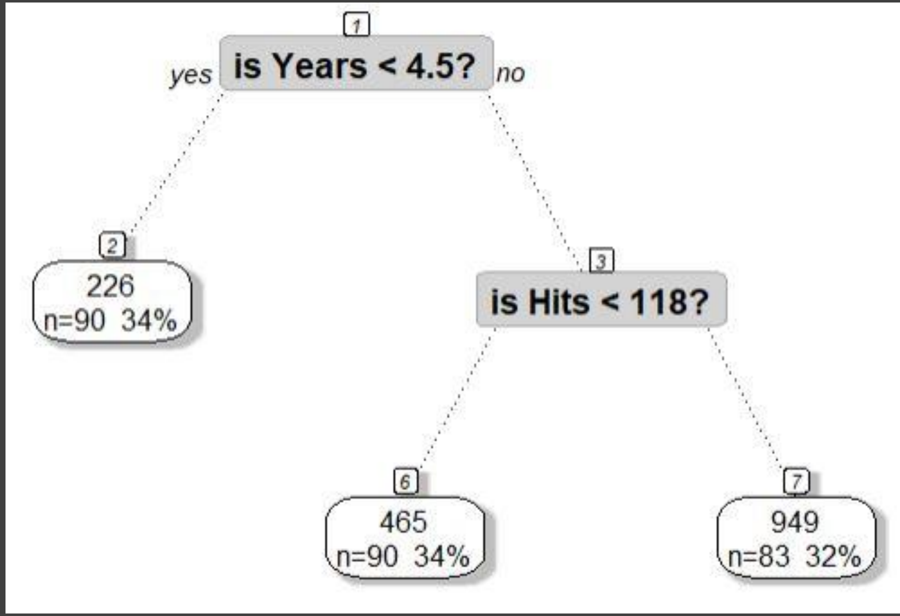
```
##          CP nsplit rel error  xerror    xstd
## 1 0.246750      0   1.00000 1.00686 0.13924
## 2 0.189906      1   0.75325 0.80766 0.12971
## 3 0.020522      2   0.56334 0.63206 0.10662
## 4 0.014281      3   0.54282 0.67086 0.10992
## 5 0.011625      4   0.52854 0.70686 0.11418
## 6 0.010870      5   0.51692 0.71573 0.11457
## 7 0.010267      8   0.48430 0.71287 0.11489
## 8 0.010000     10   0.46377 0.71403 0.11488
```

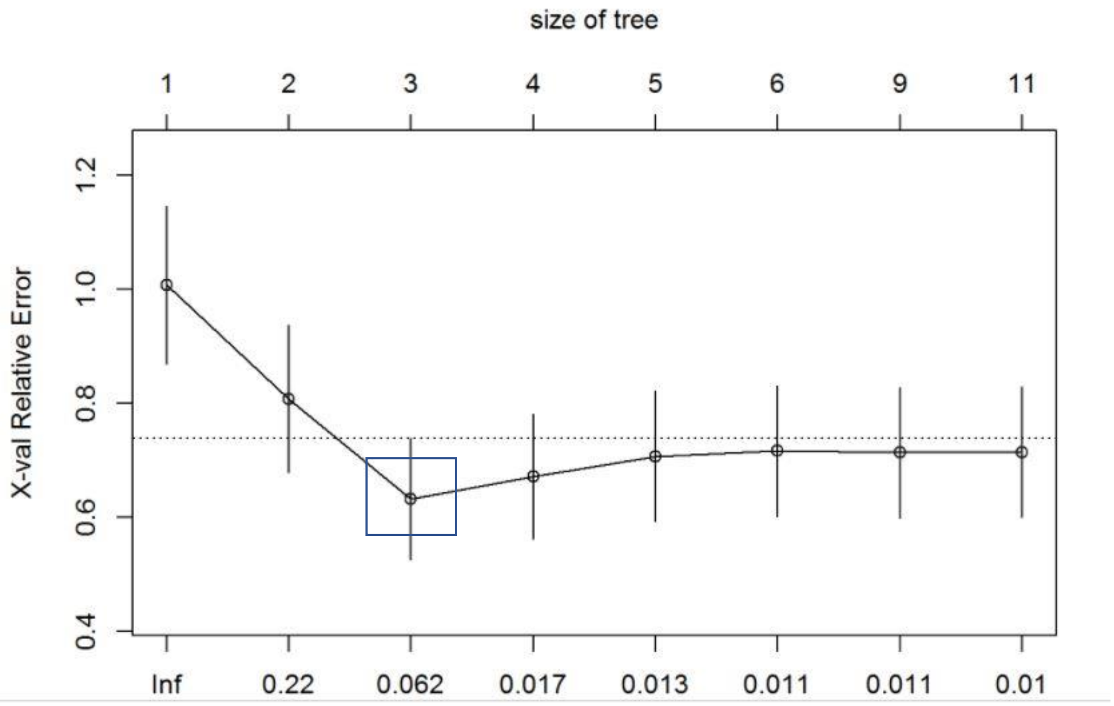


Decision Trees – Pruning

Pruned tree

```
##          CP nsplit rel error  xerror    xstd
## 1 0.246750      0   1.00000 1.00686 0.13924
## 2 0.189906      1   0.75325 0.80766 0.12971
## 3 0.020522      2   0.56334 0.63206 0.10662
## 4 0.014281      3   0.54282 0.67086 0.10992
## 5 0.011625      4   0.52854 0.70686 0.11418
## 6 0.010870      5   0.51692 0.71573 0.11457
## 7 0.010267      8   0.48430 0.71287 0.11489
## 8 0.010000     10   0.46377 0.71403 0.11488
```





Decision Trees – Classification

- Used to predict a qualitative response vs a quantitative response
- Using the mode of the region's observations instead of the mean
- Also interested in the class proportions of observations per region

$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$$

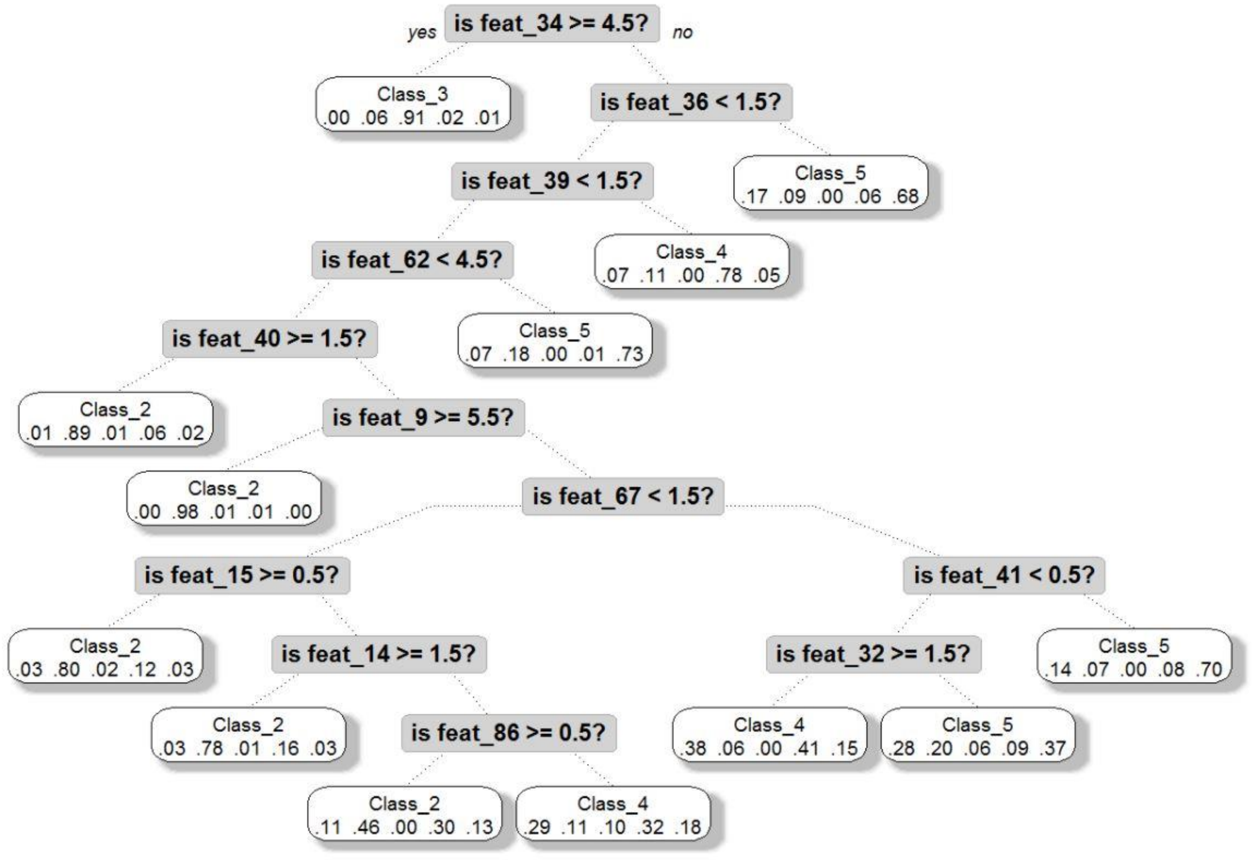
$$D = -\sum_{k=1}^{K} \hat{p}_{mk} \log \hat{p}_{mk}$$

Decision Trees – Classification

- Process is the same as regression trees, except splitting criterion
- Gini Index – measure of node purity – how often an element is labeled correctly
  - P_{mk} represents the proportion of observations in the *m*th region from the *k*th class
  - Small value indicates that a node predominantly contains observations from a single class
- Cross Entropy – alternative measure of purity

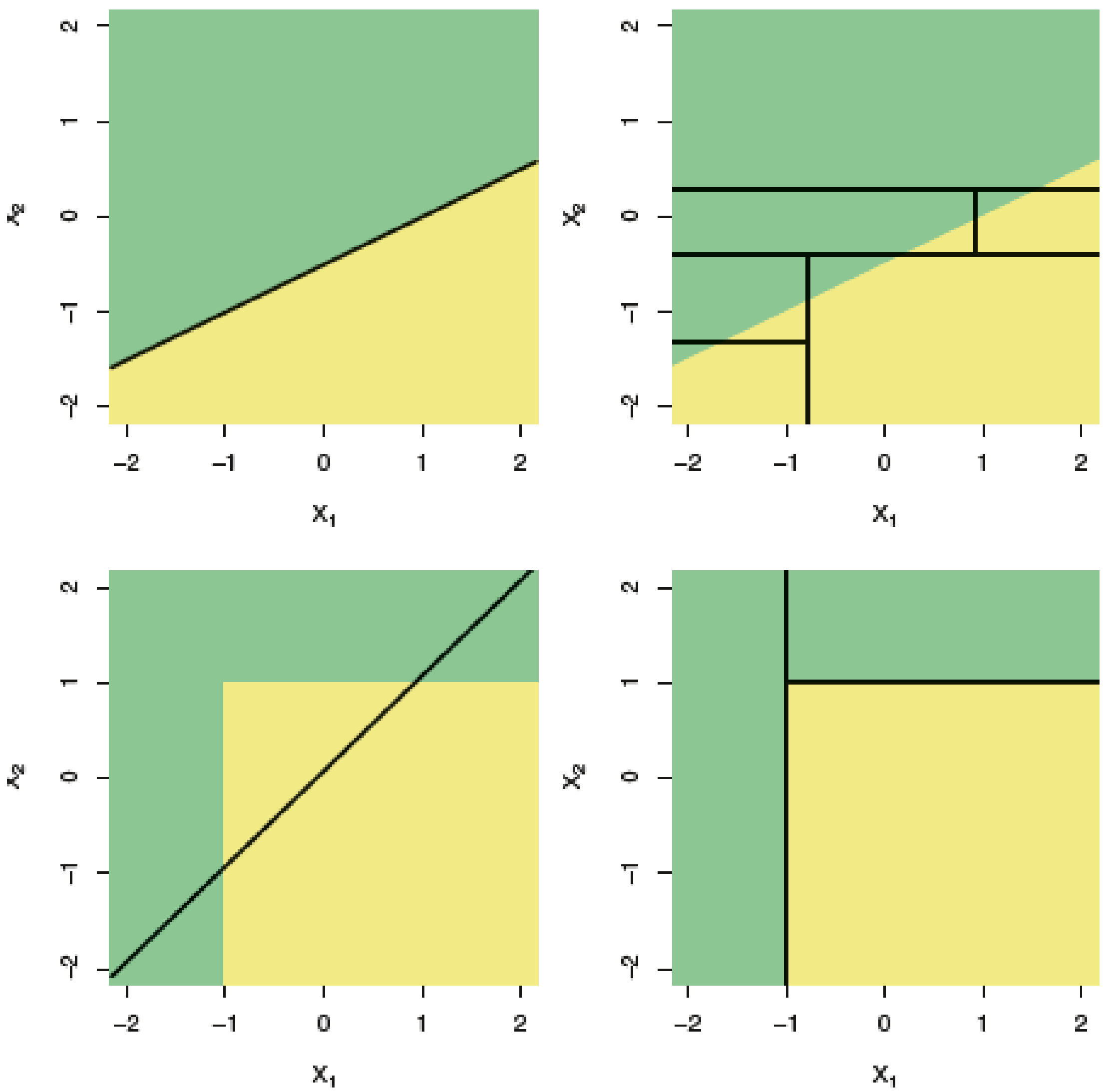
Decision Trees – Classification

- Fully grown Classification Tree
- Nodes show
  - Proportion of classes
  - Mode/prediction of node



Trees vs Linear Model Depends on how the separation of the data is composed

- Top row: Linear classifier provides a better fit than trees for a linear space
- Bottom row: Trees provide a better fit for non linear space

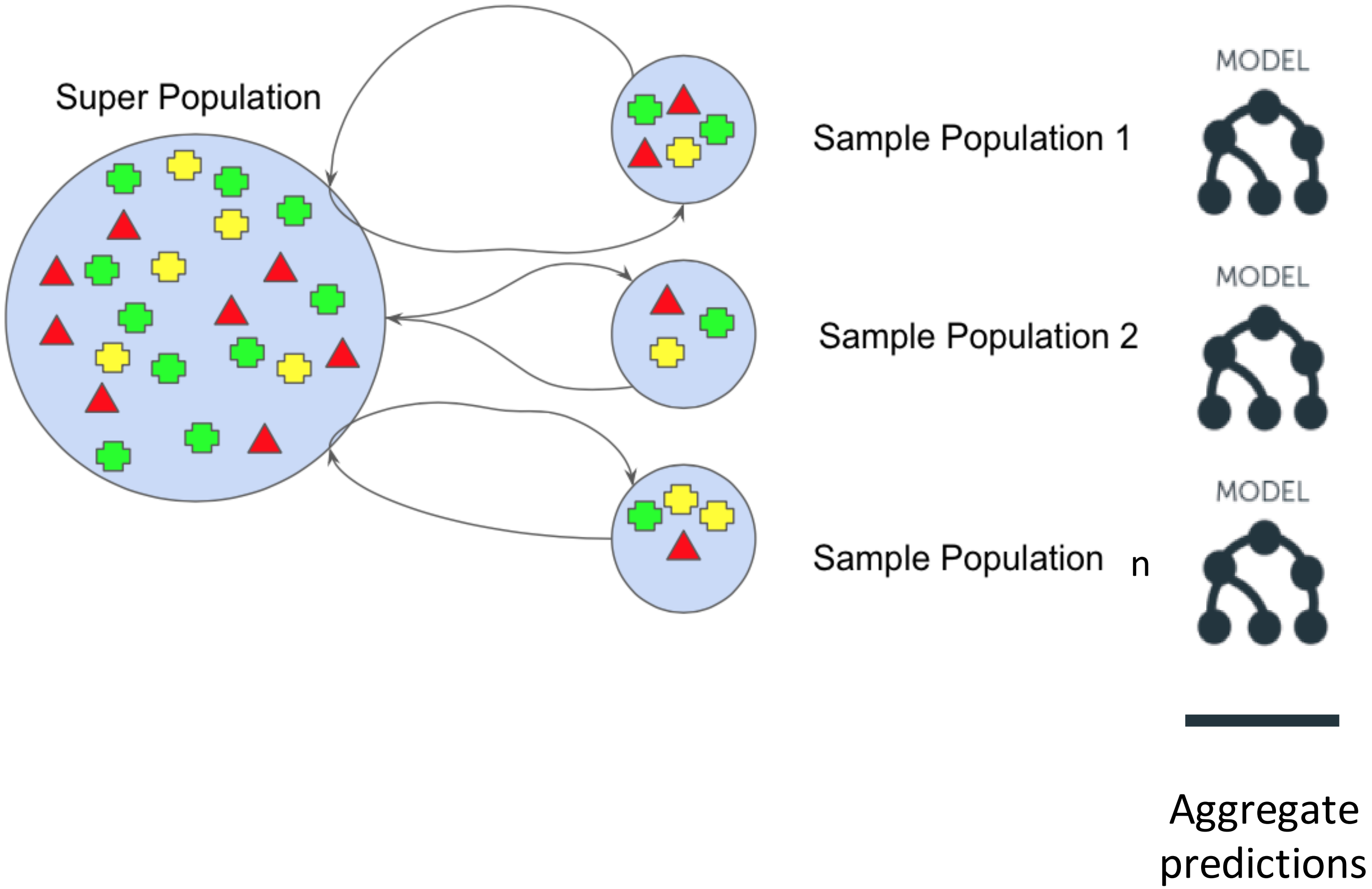


Ensembling – Improving Trees

- Biggest problem with building a decision tree is high variance.
- Solution is ensembling
- We can use multiple trees to get more accurate predictions and lower the variance

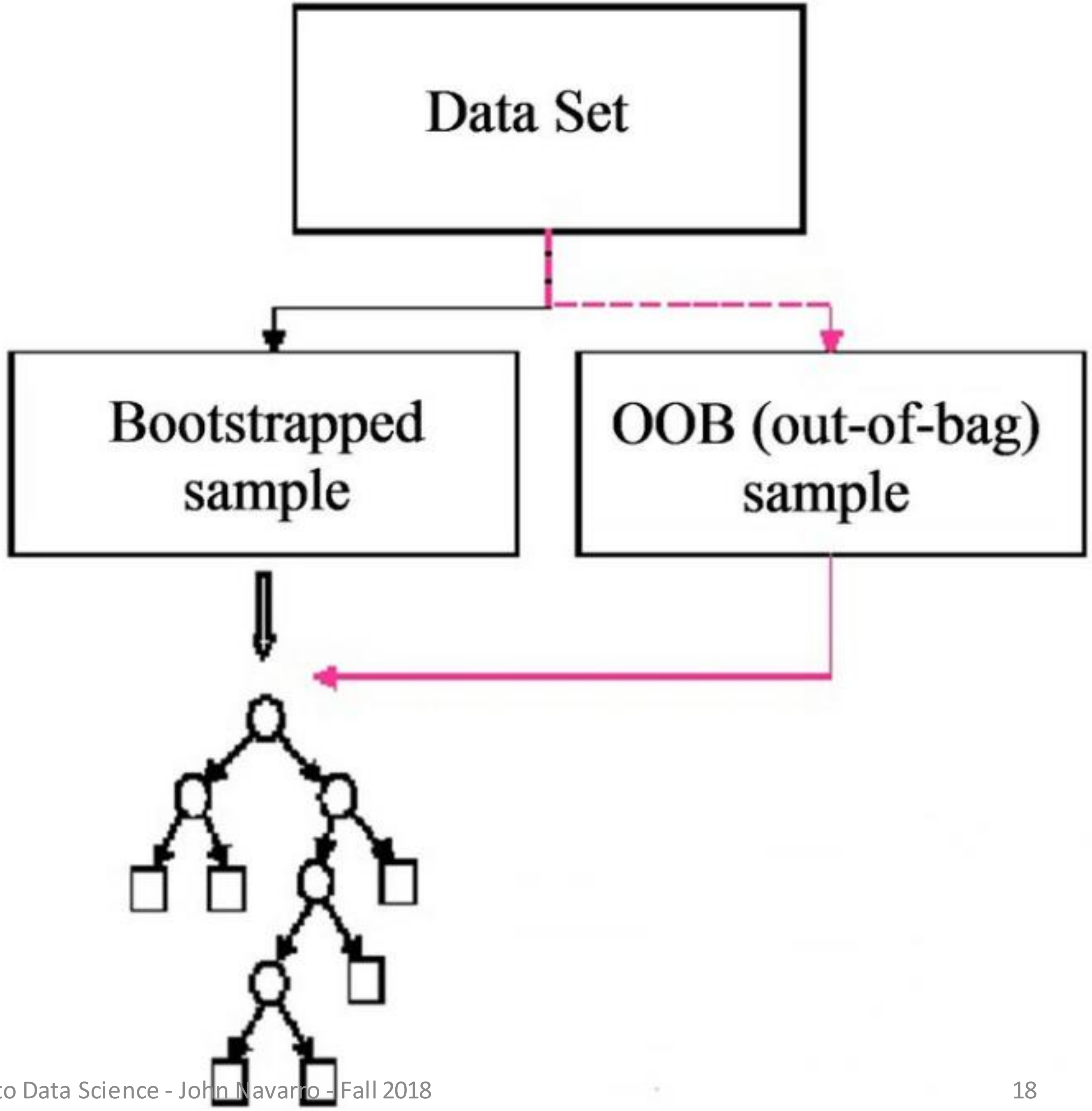
Ensemble 1 – Bagging Bootstrap Aggregation

- Step 1
  - bootstrap the data & create data set 1
  - build dec tree 1
- Step 2
  - bootstrap the data & create data set 2
  - build dec tree 2
- Step n
  - bootstrap the data & create data set n
  - build dec tree n
- Final Step
  - Aggregate predictions
  - Regression – mean
  - Classification - mode



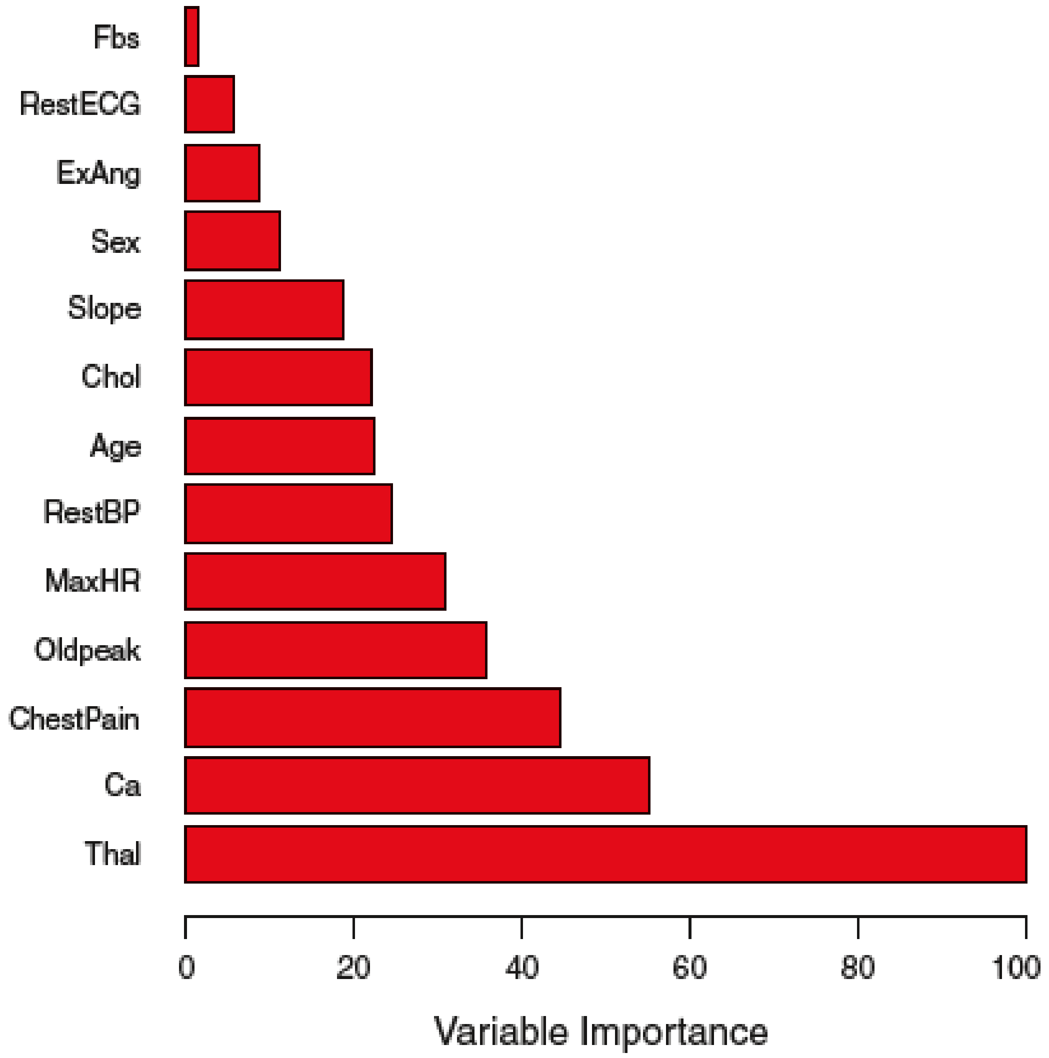
Ensemble 1 – Bagging Bootstrap Aggregation

- OOB Out of Bag observations
- Bootstrap method uses 2/3 of the data, 1/3 is for the test set



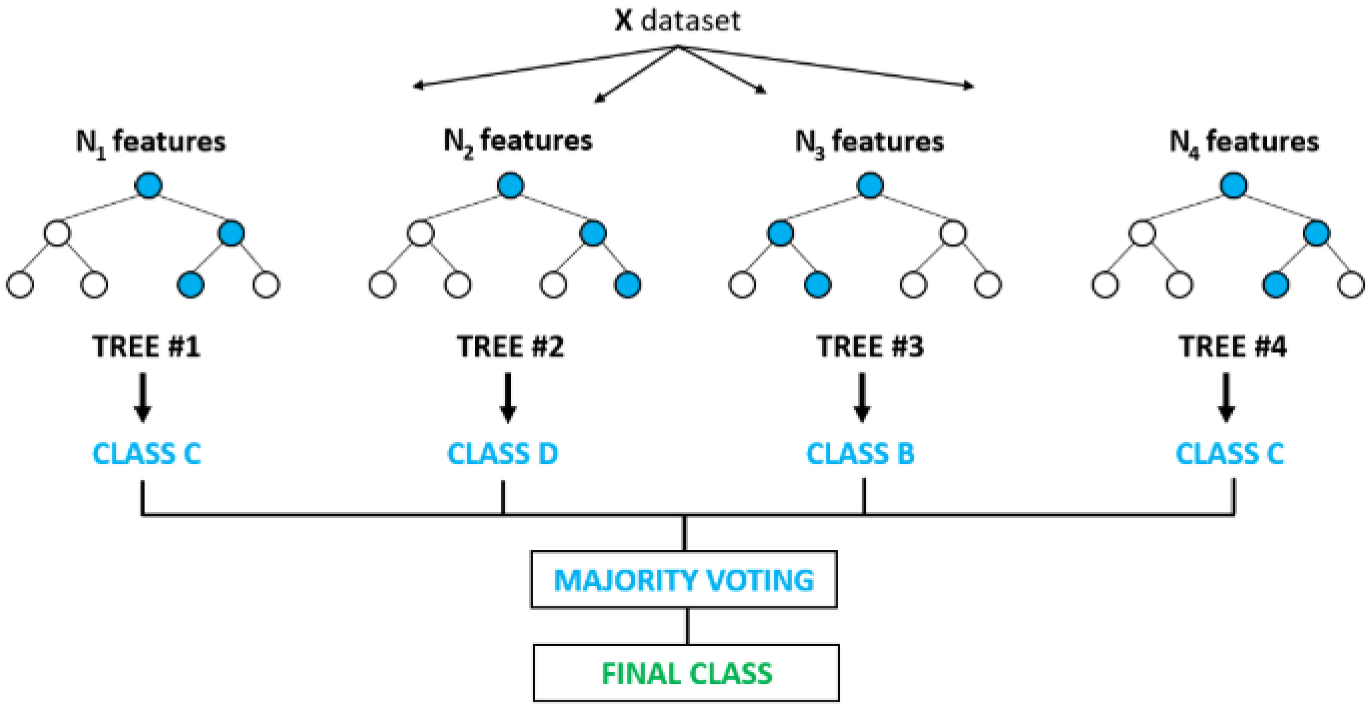
Ensemble Interpretation
Feature Importance

- Interpretation of the features is lost because we have many trees
- Different trees and different features combine to give the aggregated prediction
- Remove one feature and measure how much error changes
- Importance is relative to the most important predictor



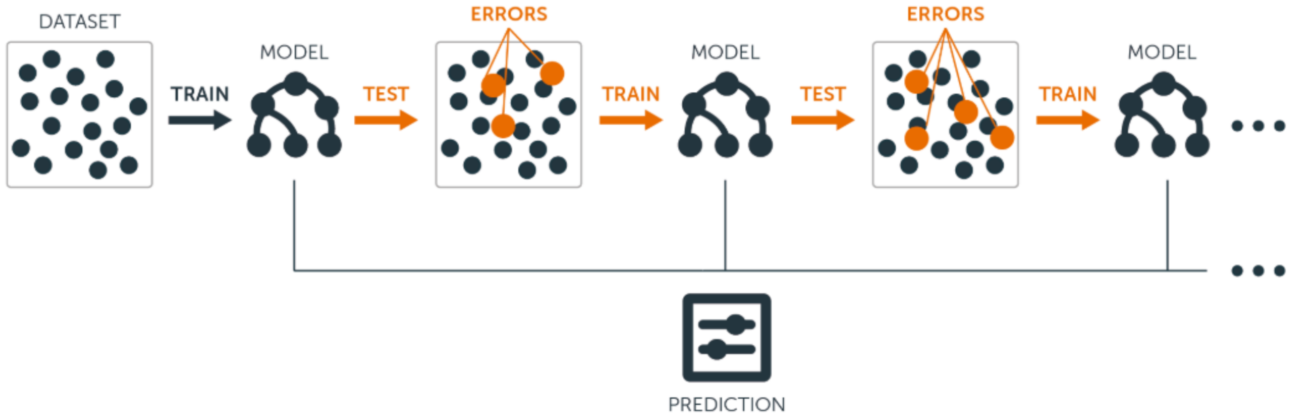
Ensemble 2 – Random Forest

- Random Forest is very similar to Bagging
- Difference is in how we make our splits (which features we consider)
- Every time we make a split, we take a random sample of subset of N features
- Regression subset: $^{N}/_{3}$
- Classification subset: $\sqrt{N}$



Aggregate predictions

Ensemble 3 – Gradient Boosted Trees



- Difference is trees sequential and dependent
- Residual output of the first tree is the input to the next tree
- Typically use short trees (stumps)
- Slow learner progresses to become powerful
- Learning rate parameter
  - $\lambda = 0.01\ or\ 0.001$
- Regression or Classification tasks

Questions?

Acknowledgments

Material for this class was taken from

- James, Introduction to Statistical Learning