



Environment and
Climate Change Canada

Environnement et
Changement climatique Canada

RANDOM FORESTS: THE BASICS, GUIDELINES ON MODEL OPTIMIZATION

Koreen Millard, Amir
Behnamian, Sarah Banks



Canada

Learning Objectives

- Image Classification
- Classification and Regression Trees (CART)
- Random Forests:
 - Basics
 - Advantages
 - Best Practices
- How to implement Random Forests in R

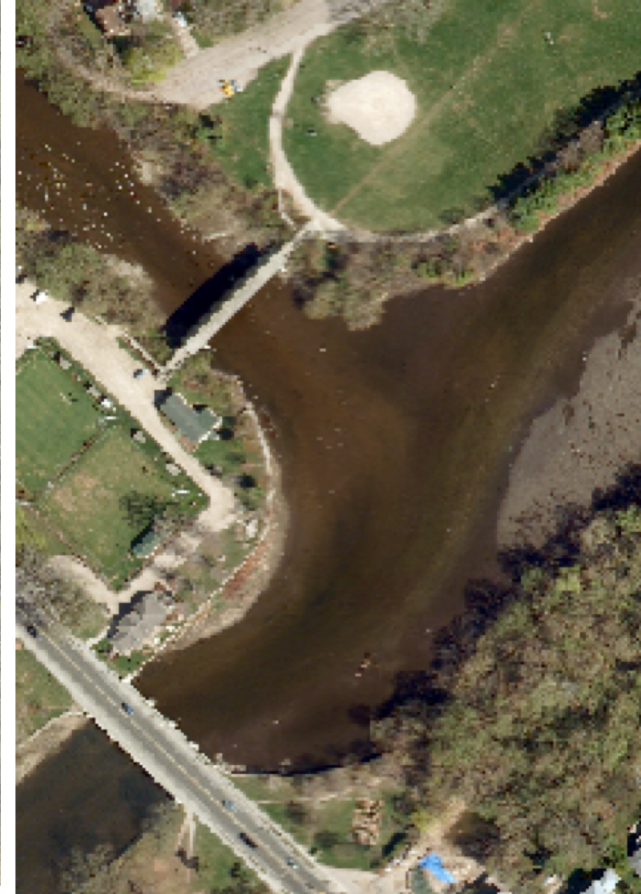
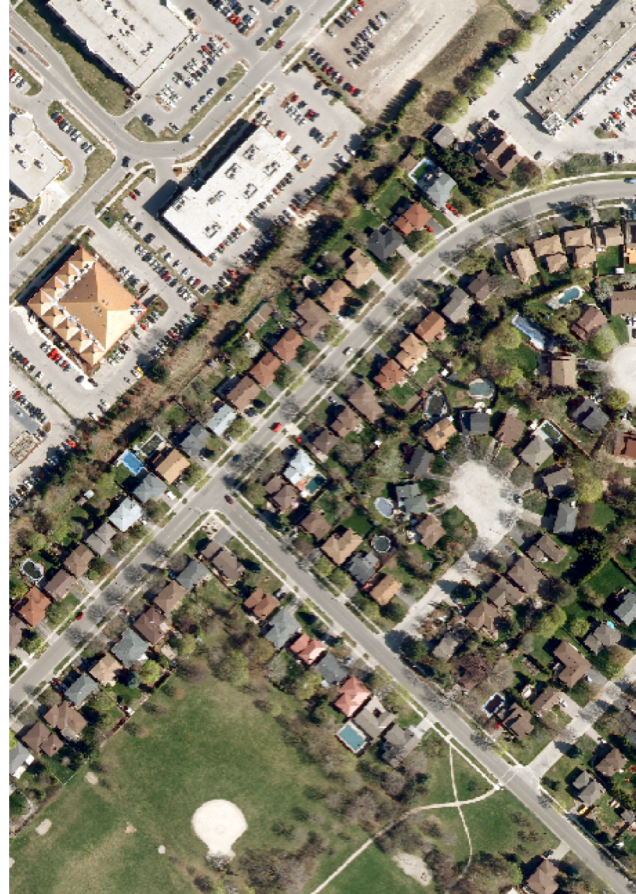
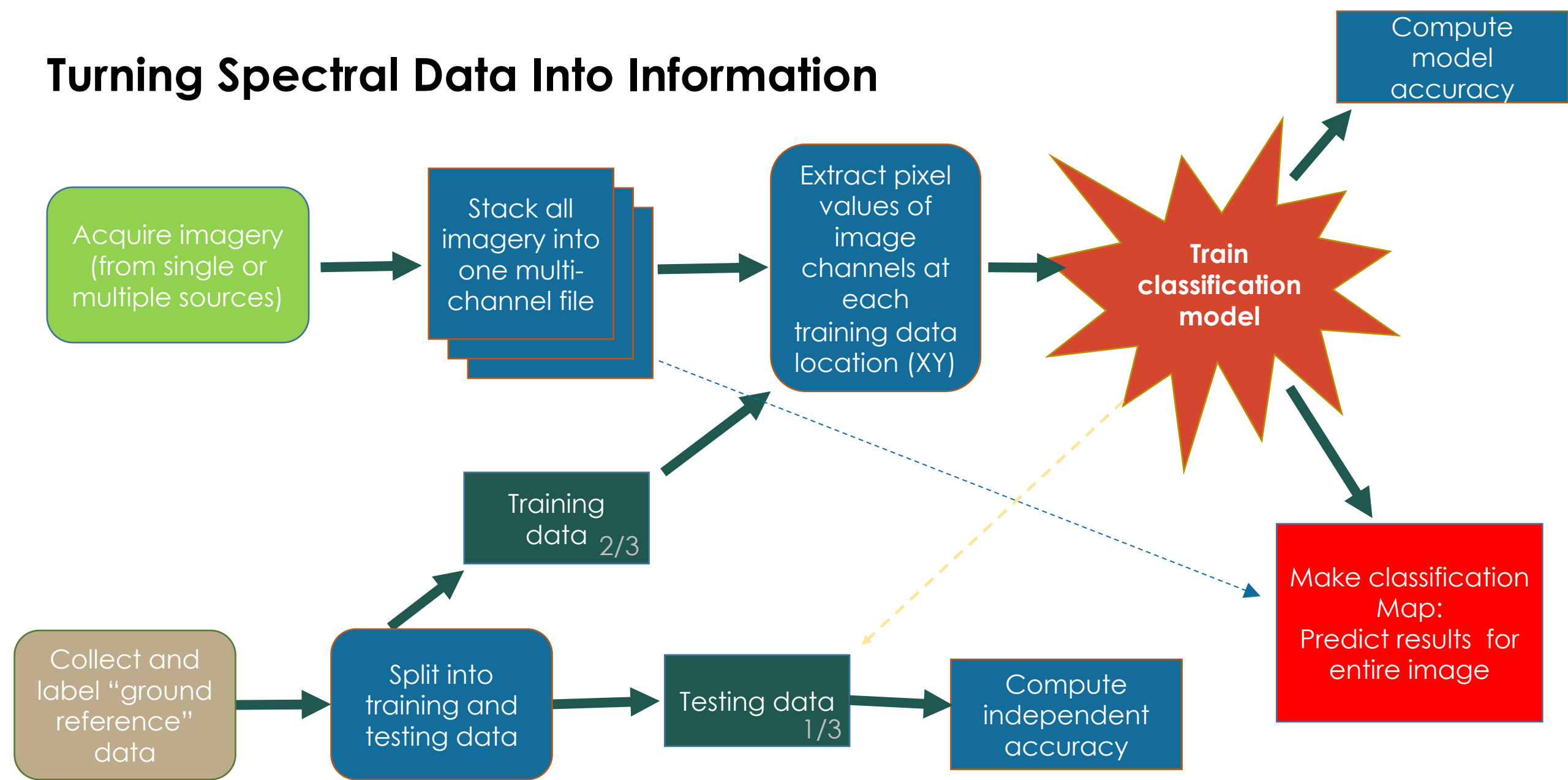


Image classification is the process of grouping **spectral classes** and assigning them **informational class names**

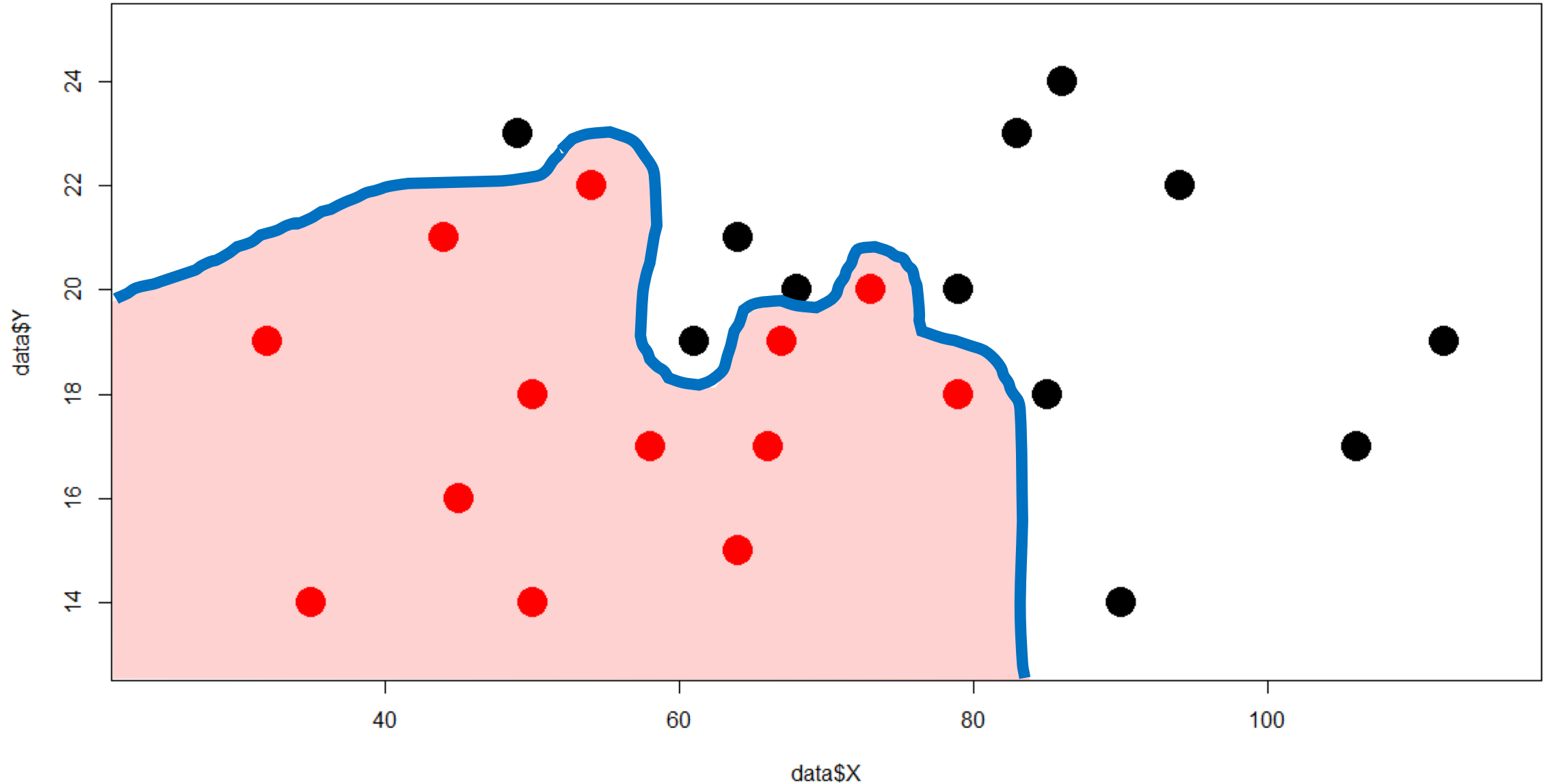
Turning Spectral Data Into Information



Classification and Regression Trees (CART): Basics

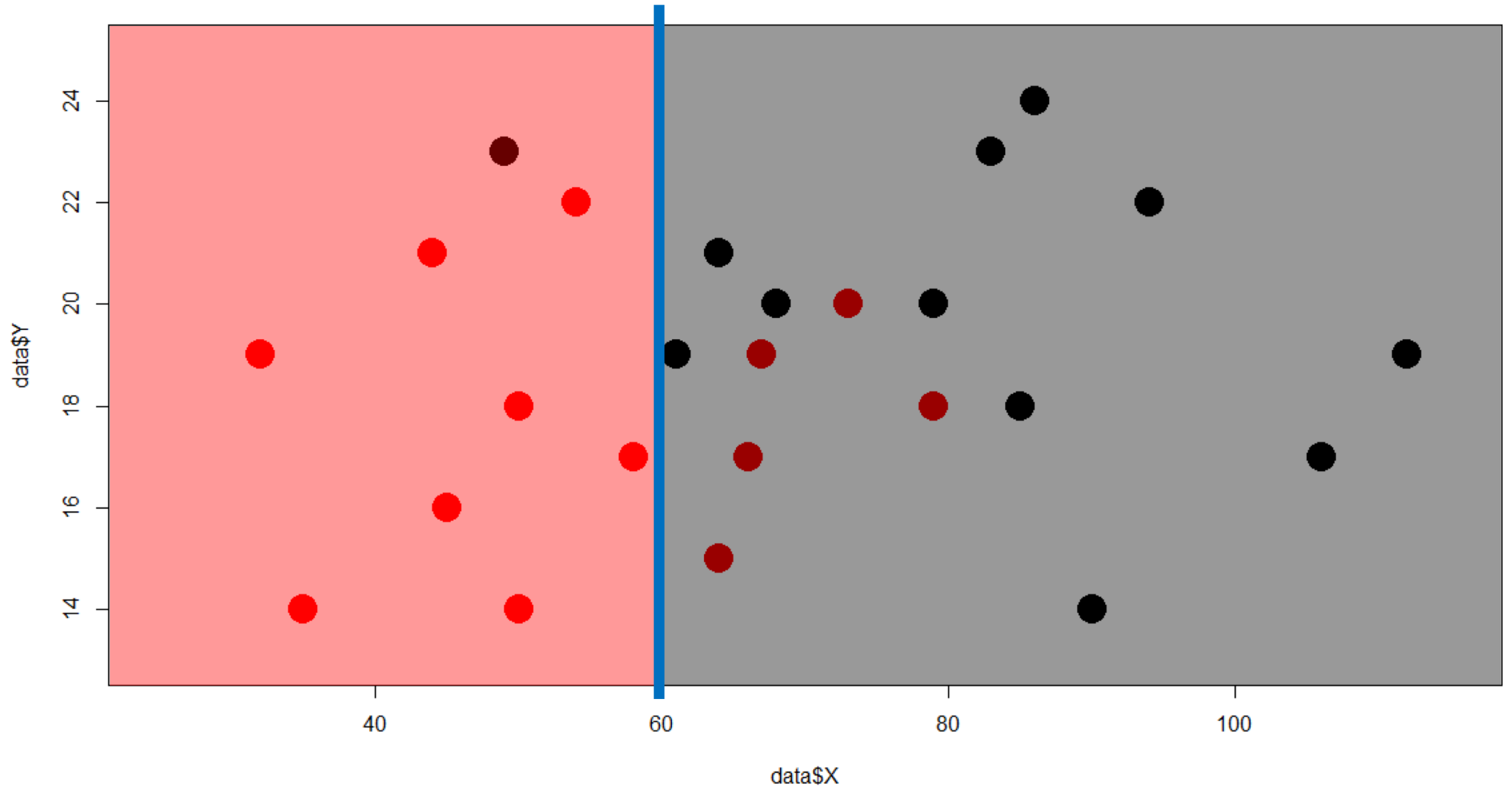
- Partitions or “splits” the data based on a set of binary rules
- Tries to make groups that are as homogeneous as possible
- Produces an easily-interpretable “tree” of decisions

Classification and Regression Trees (CART): Example



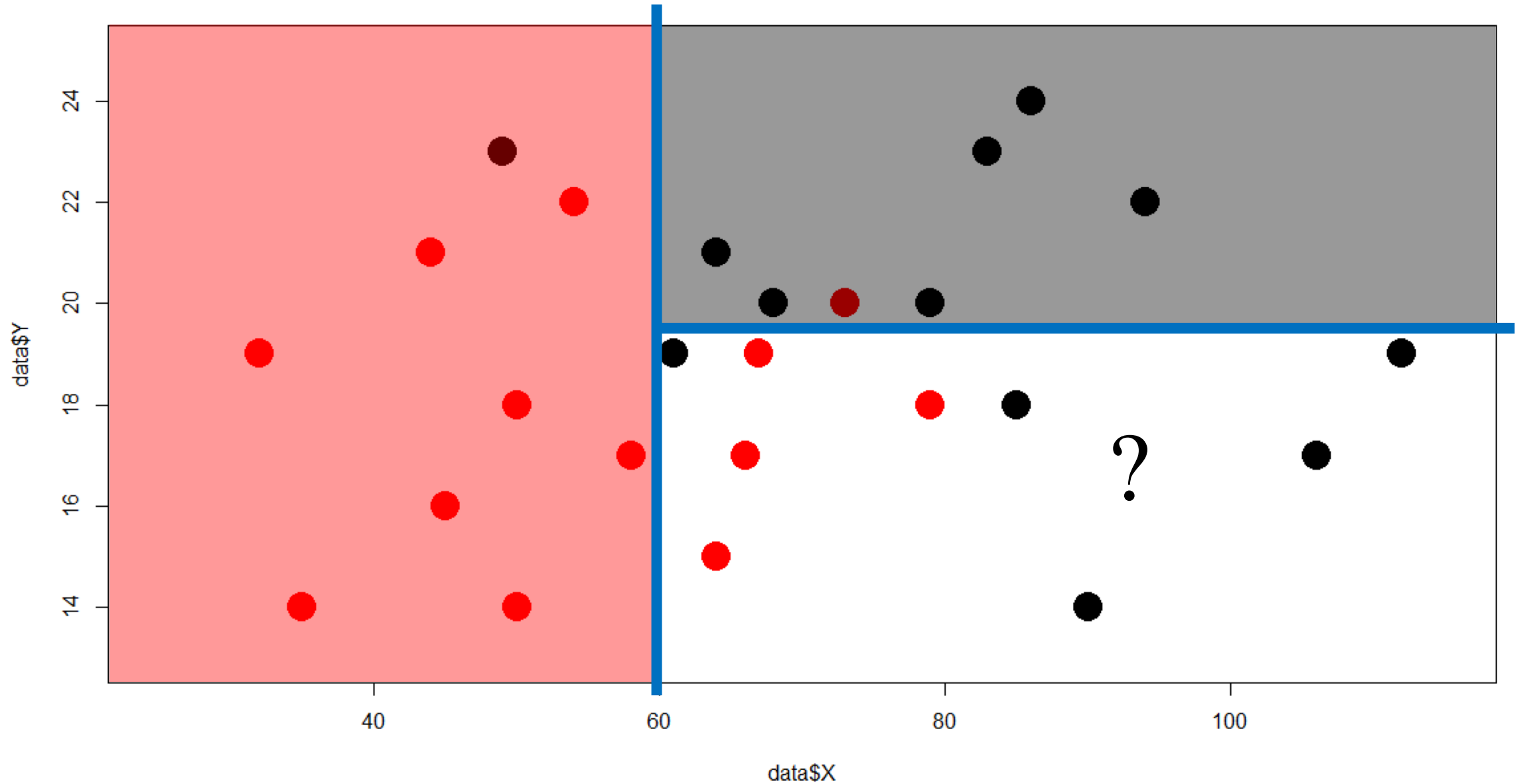
Classification and Regression Trees (CART): Example

Accuracy = 76%
1st split $X > 60$



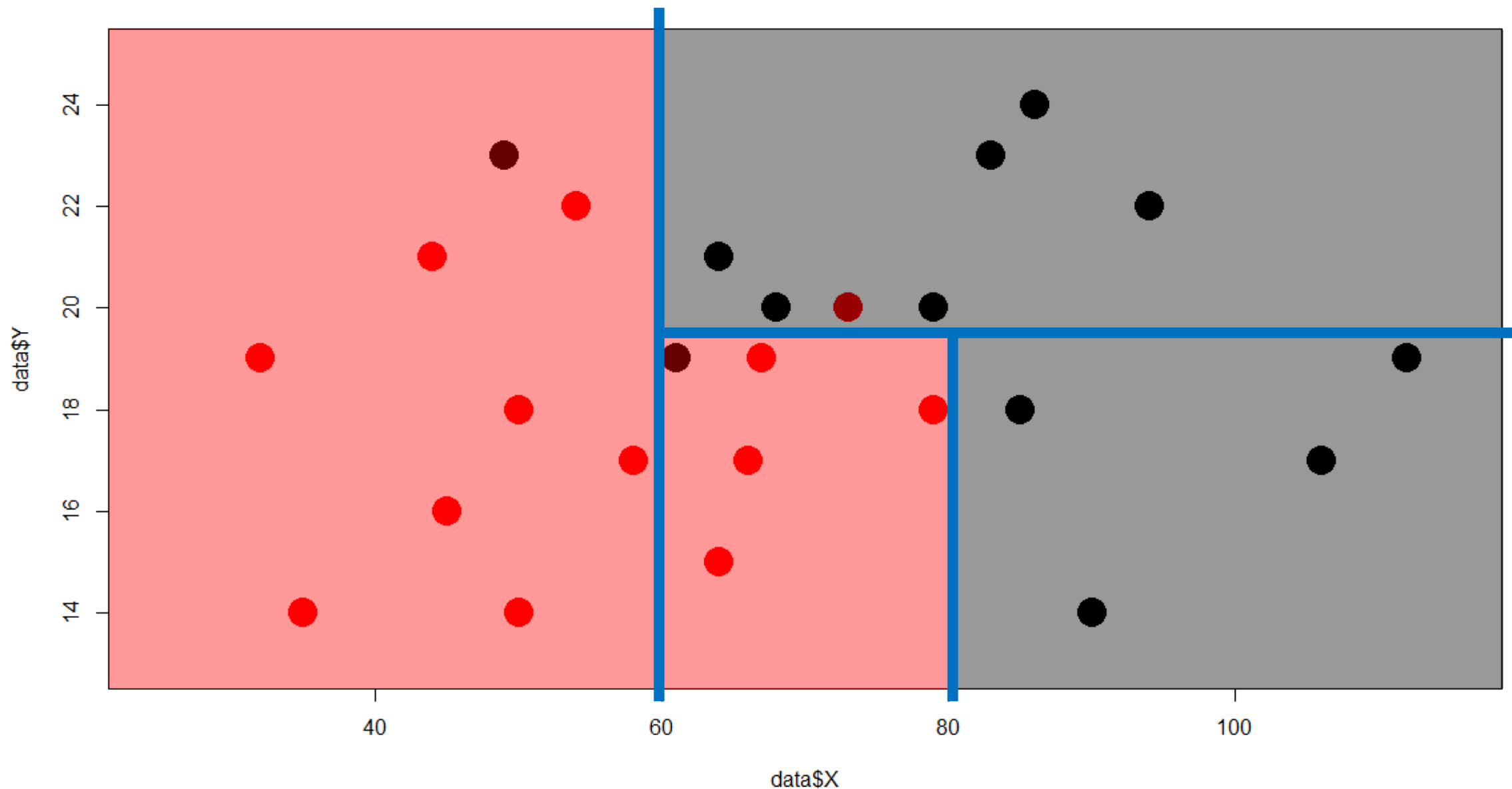
Classification and Regression Trees (CART): Example

Accuracy = 72%
2nd split $Y < 19$

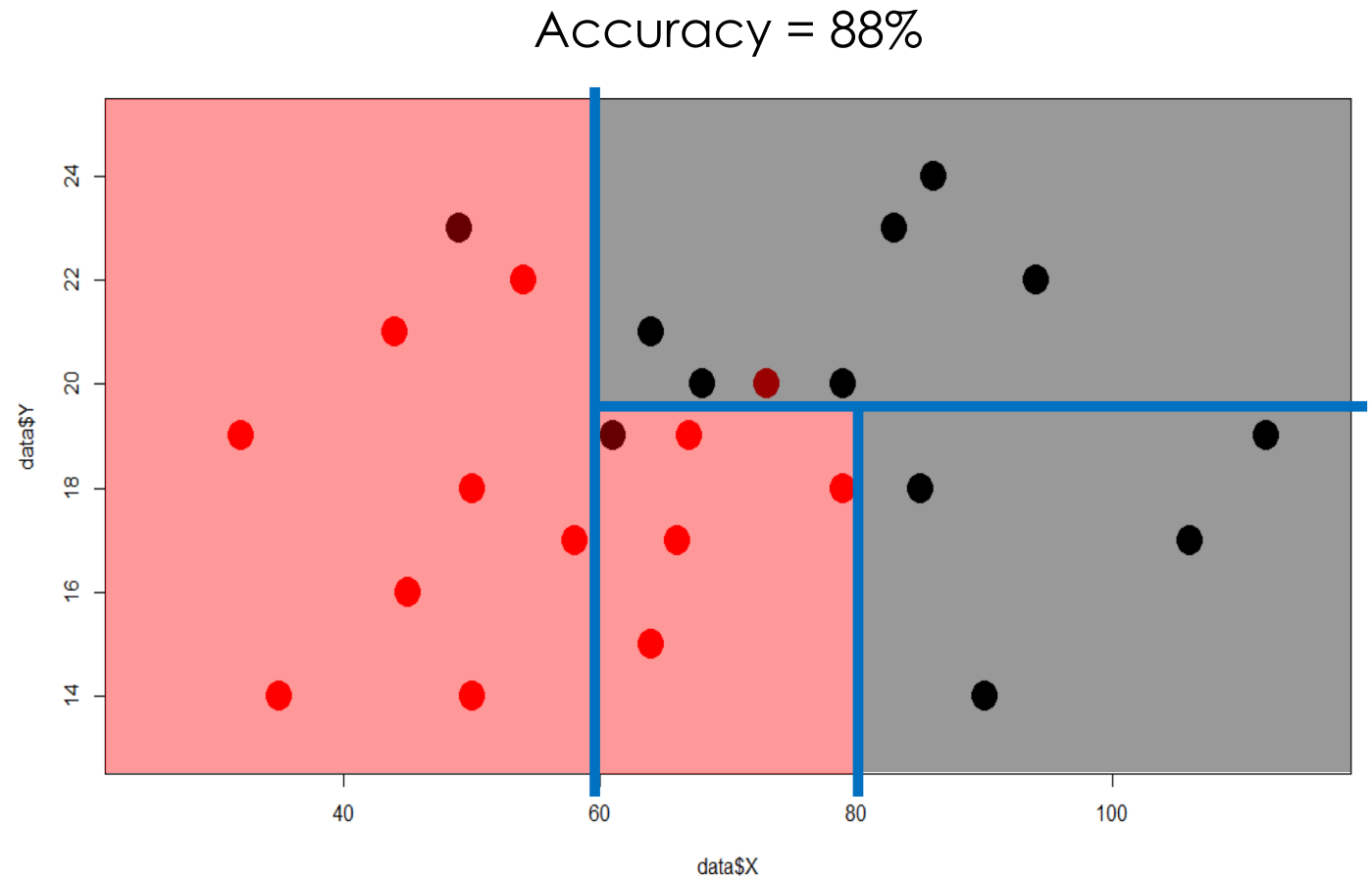
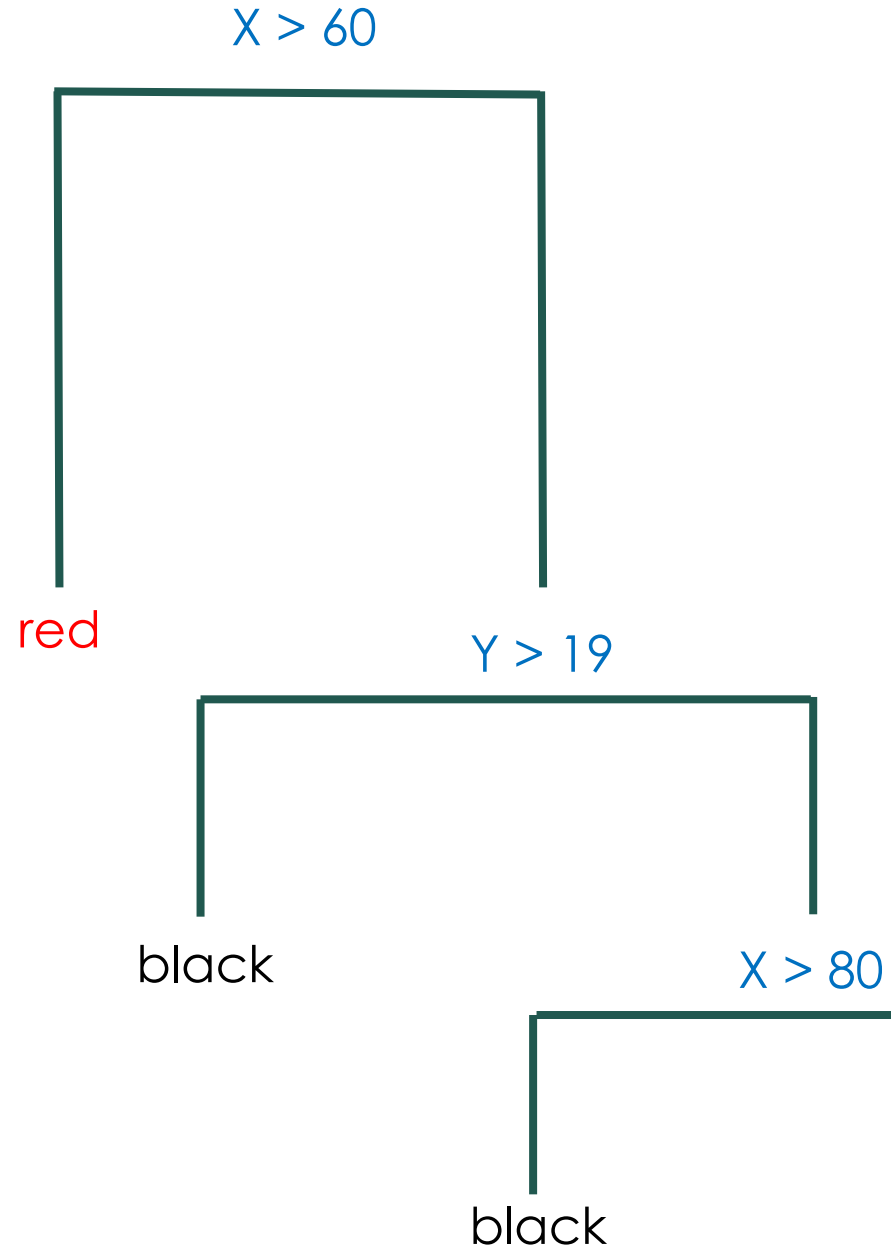


Classification and Regression Trees (CART): Example

Accuracy = 88%
3rd split $X < 80$

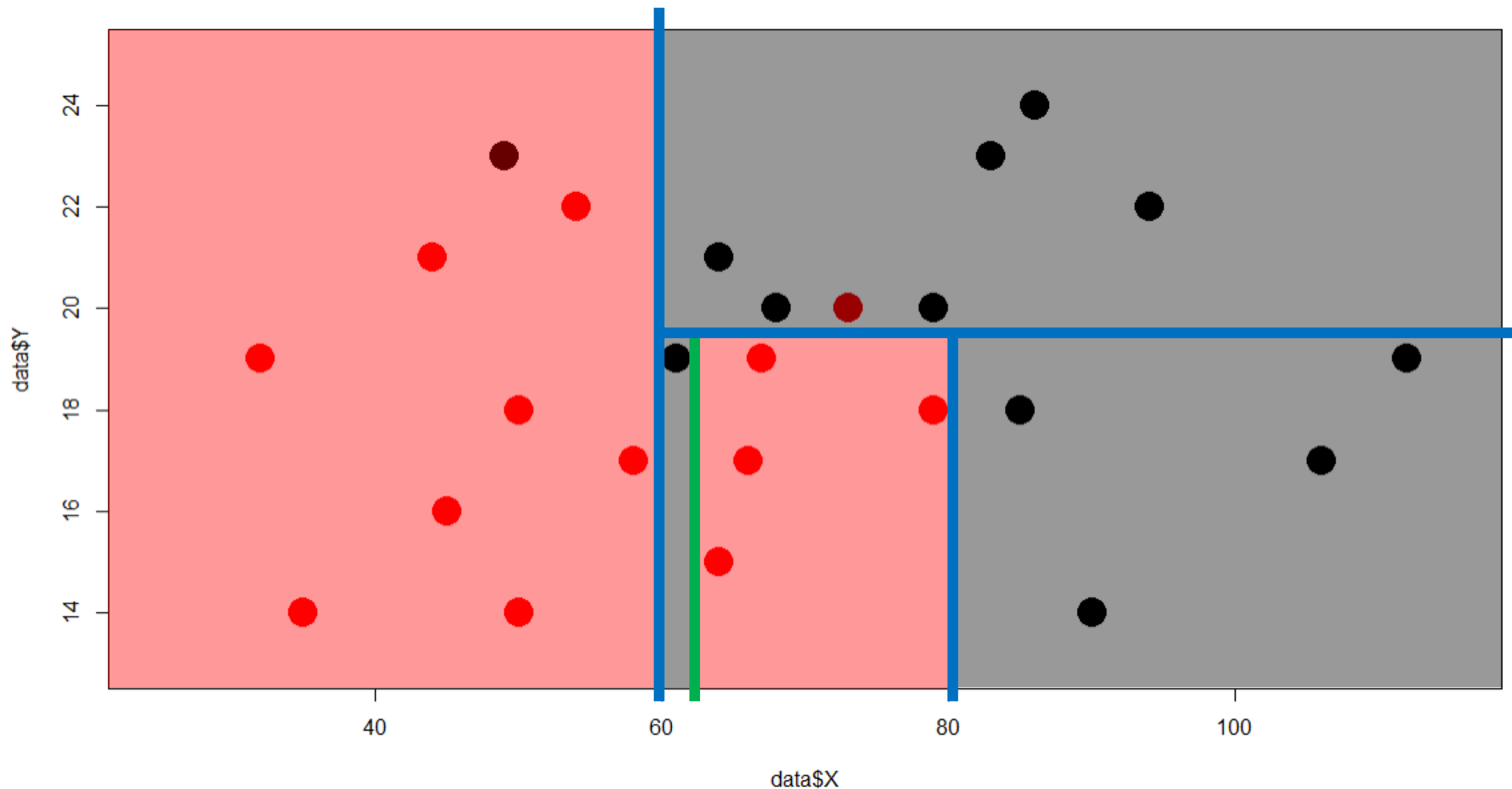


Classification and Regression Trees (CART): Example



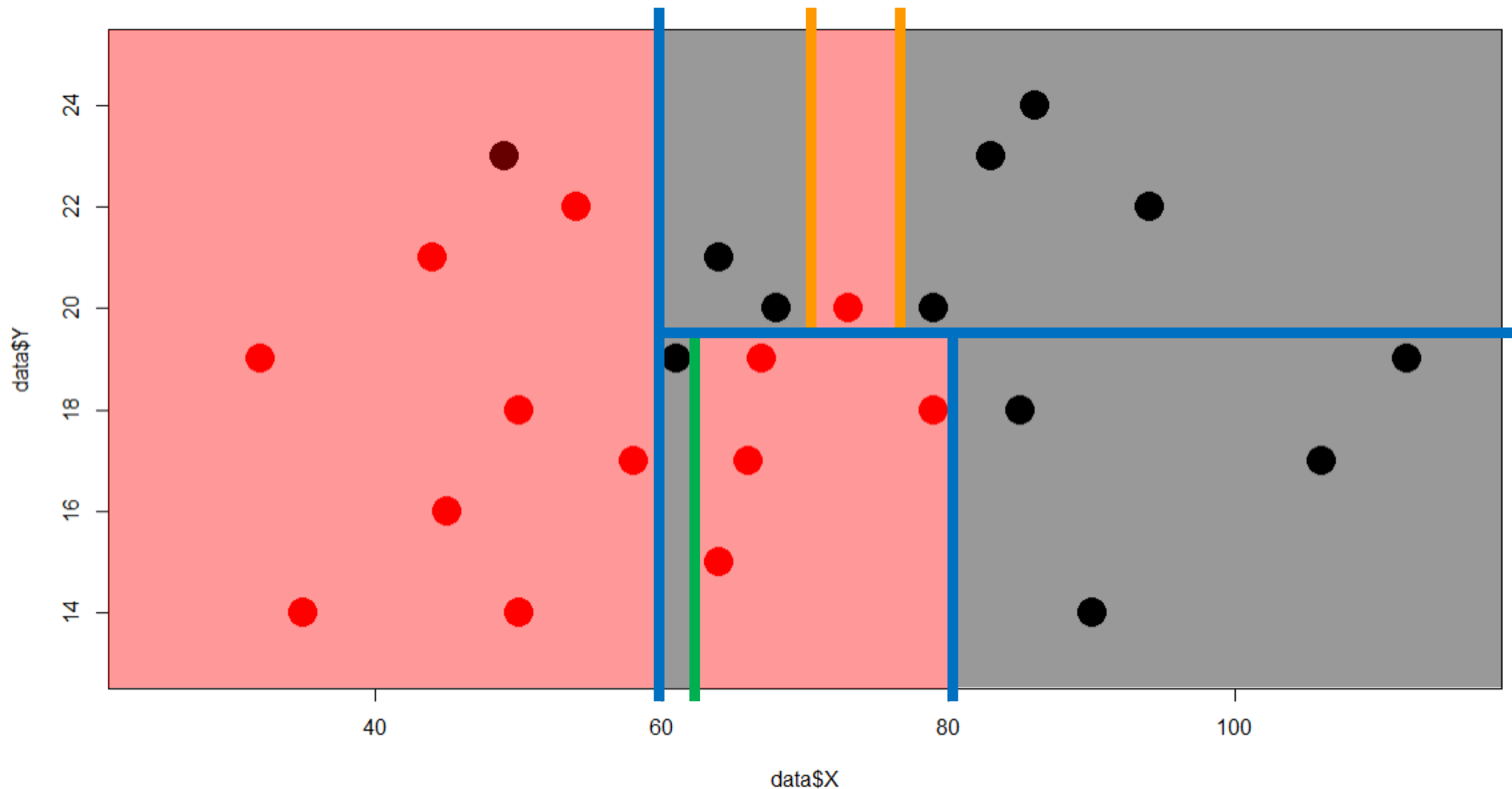
Classification and Regression Trees (CART): Example

Accuracy = 92%
4th split $60 > x < 63$



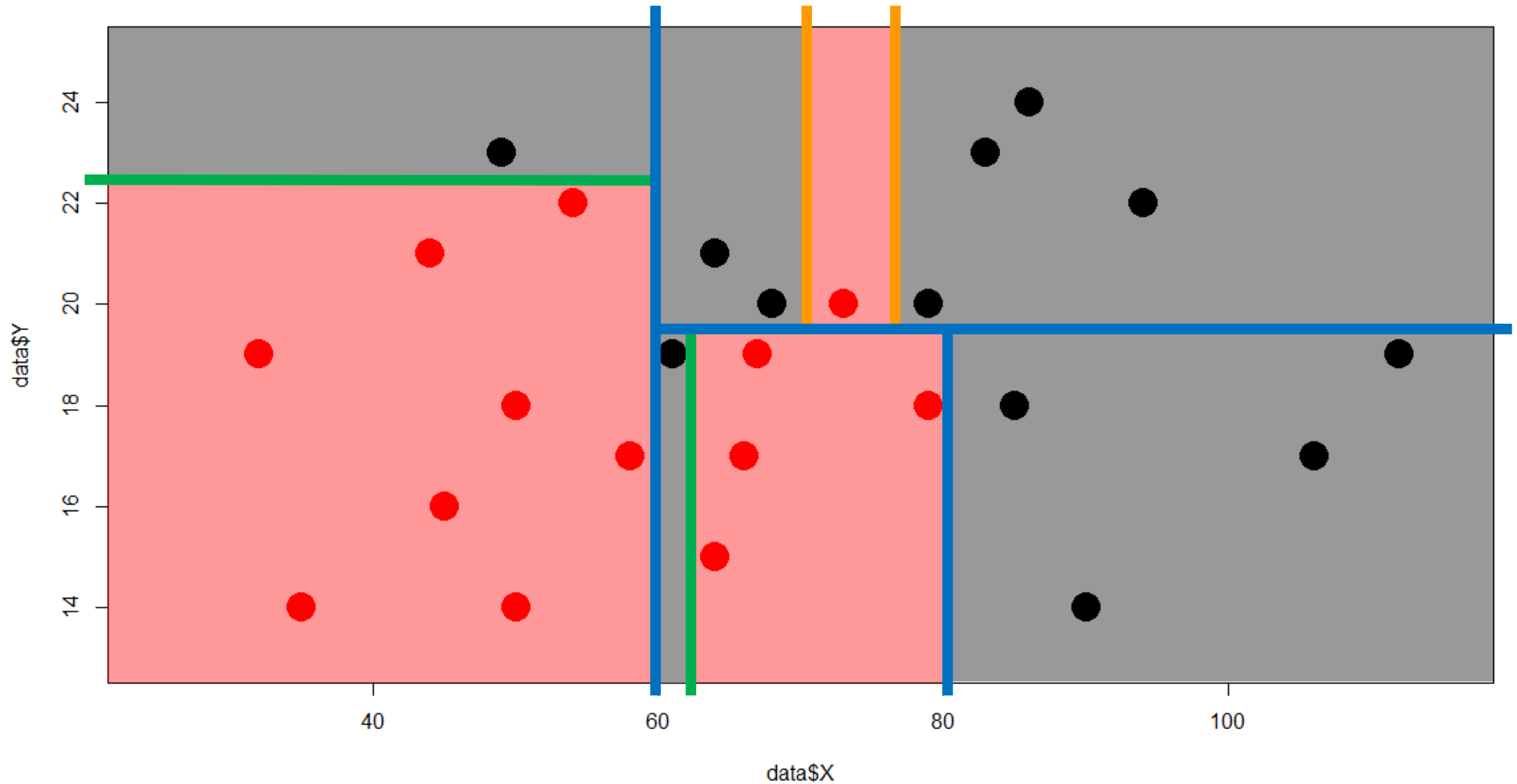
Classification and Regression Trees (CART): Example

Accuracy = 96%
5th split $69 > x < 75$



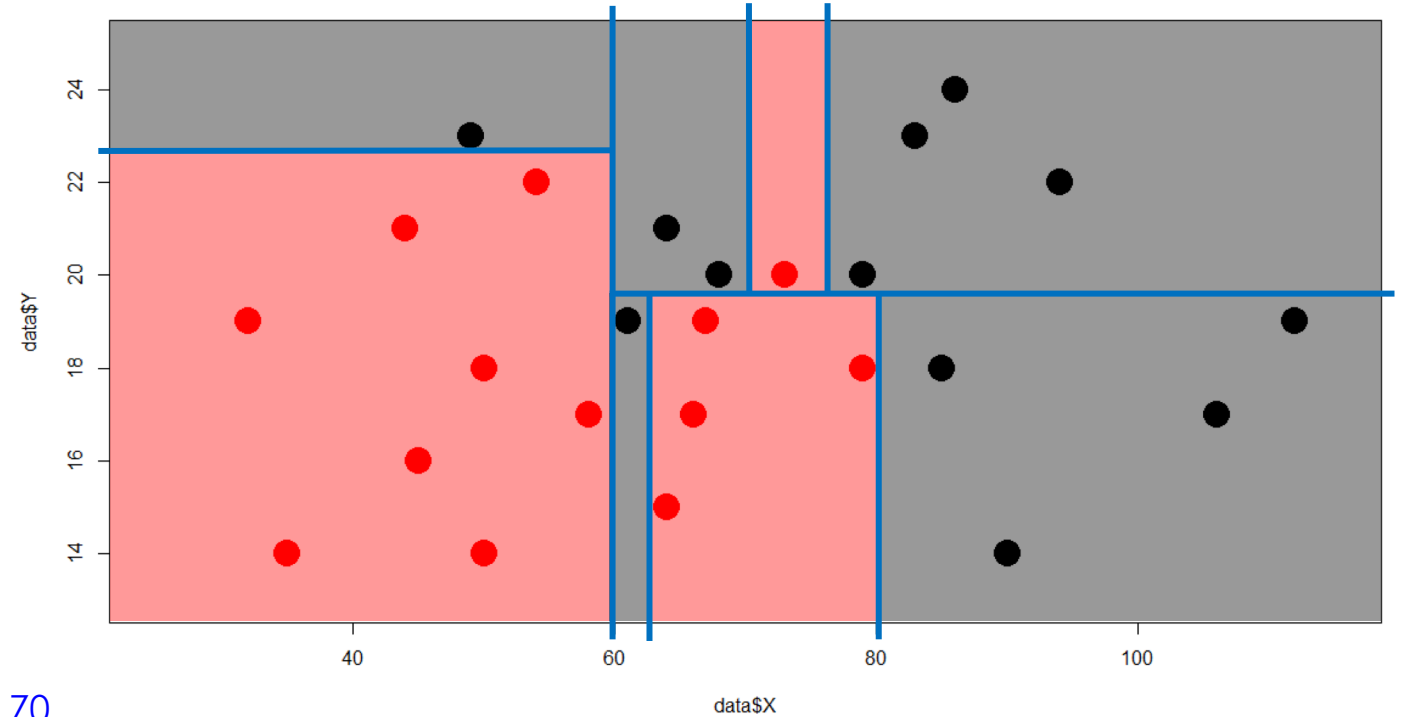
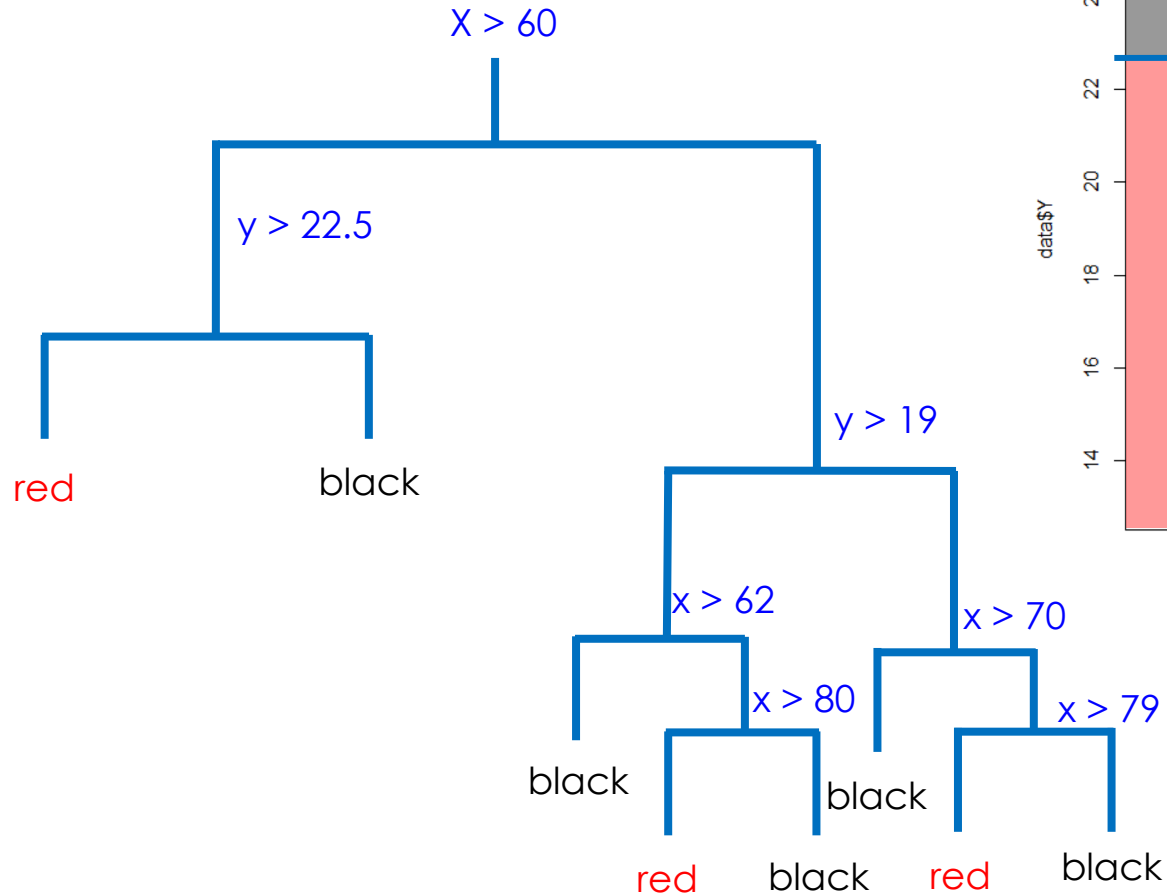
Classification and Regression Trees (CART): Example

Accuracy = 100%
6th split $Y > 22.5$



Classification and Regression Trees (CART): Example

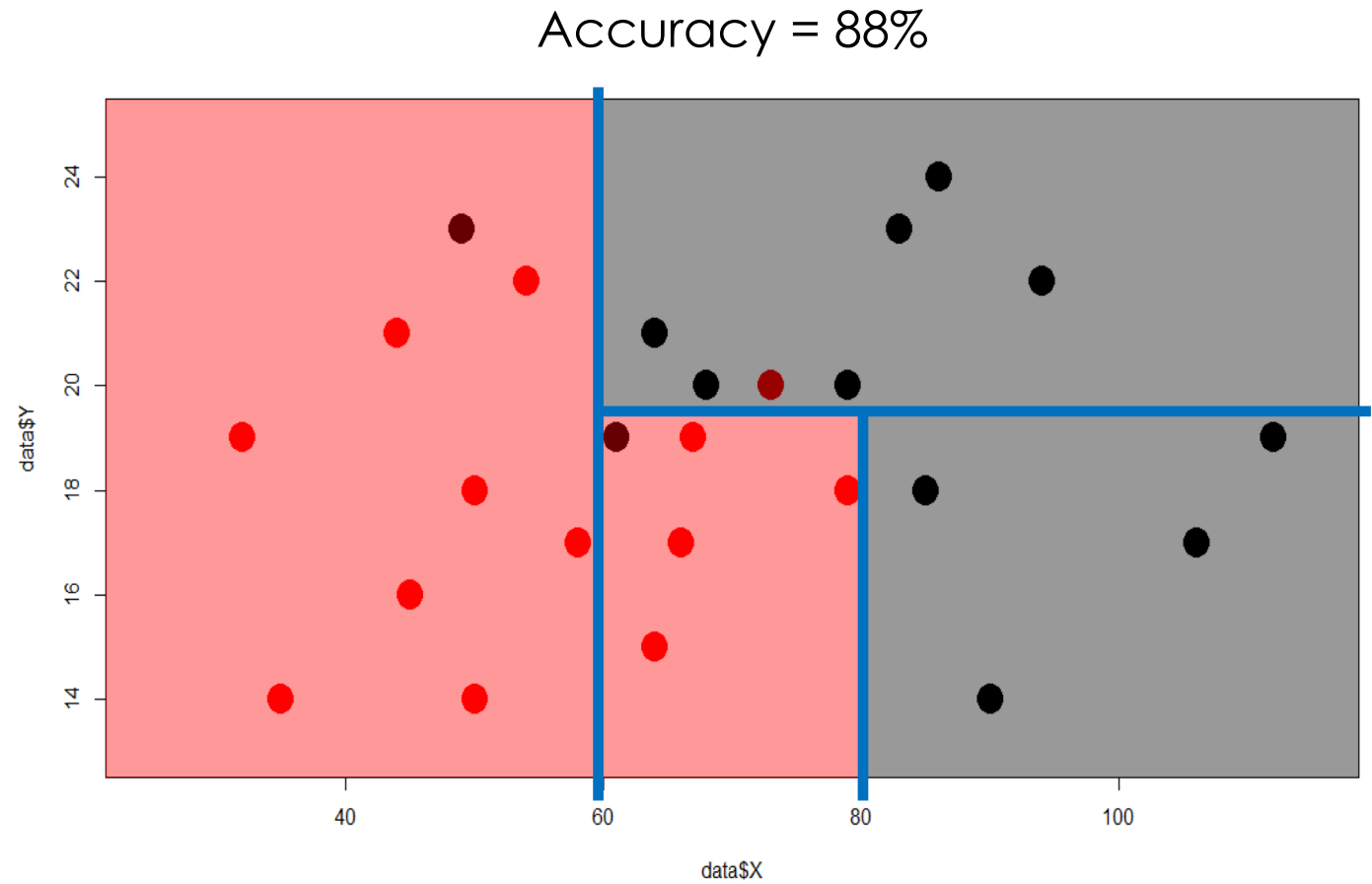
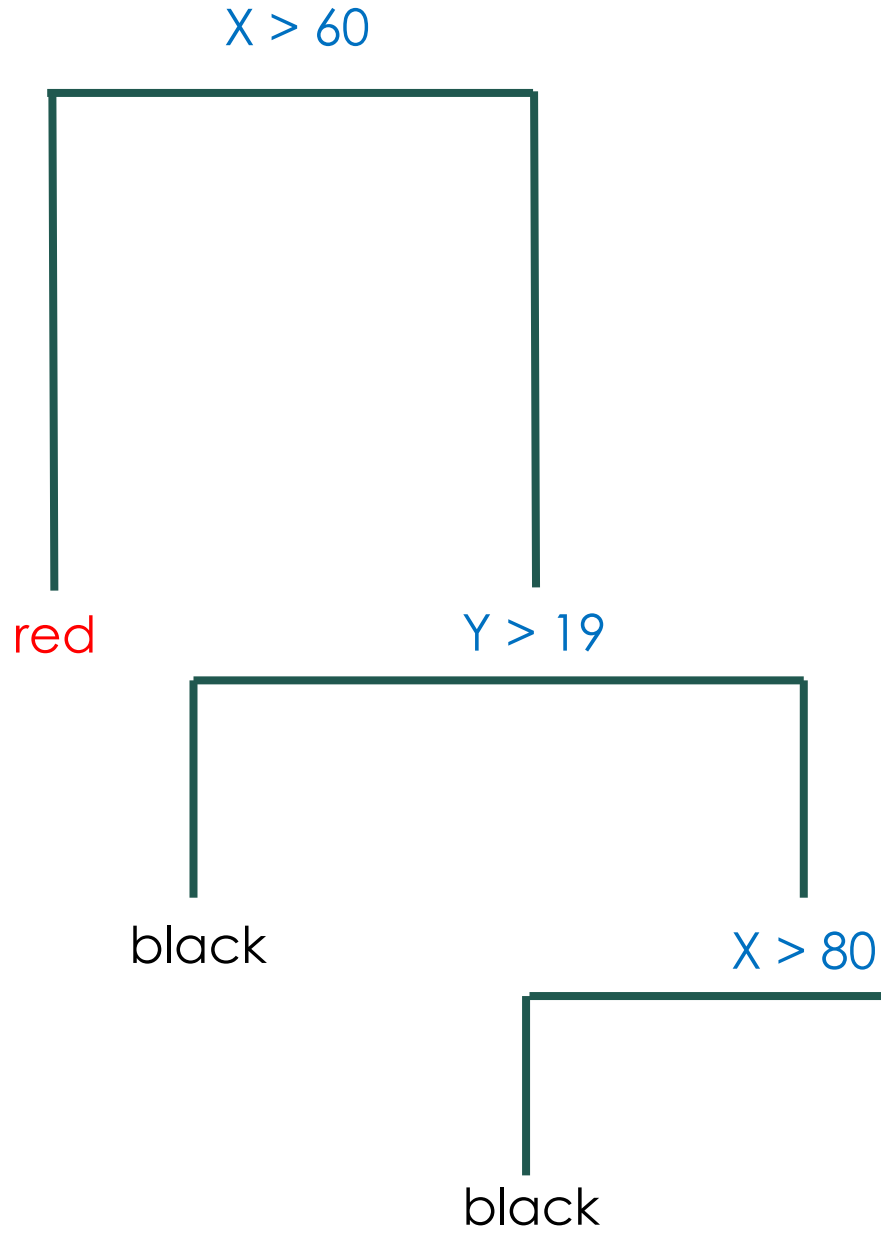
Fully Grown Tree Accuracy = 100%



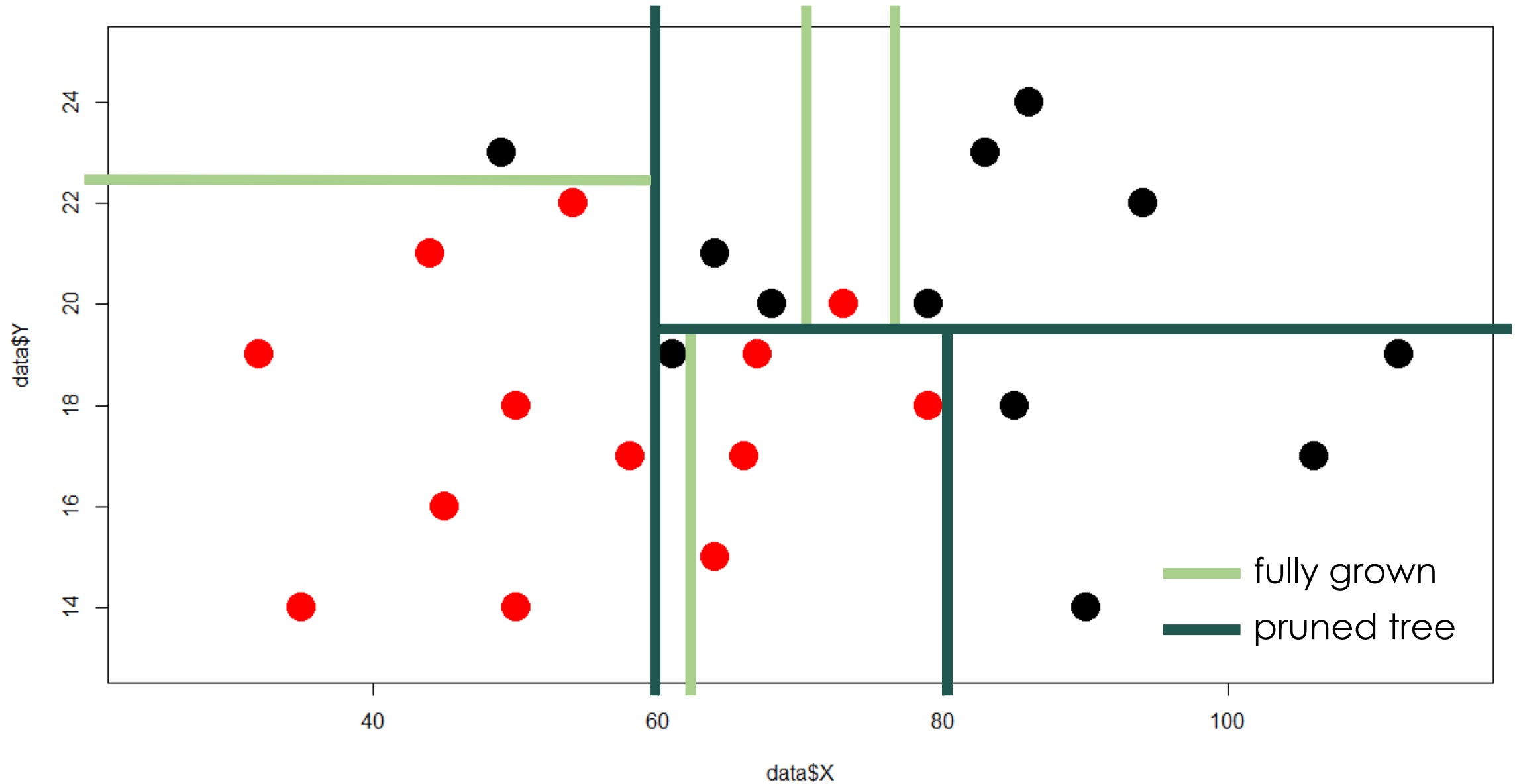
Overfitting

- When a tree is extremely well-fit to all of the samples in the dataset
- Pruning required in order to apply the results more generally
 - i.e. outside the training data
- How to check for overfitting?
 - Independent validation

Classification and Regression Trees (CART): Example



Classification and Regression Trees (CART): Example

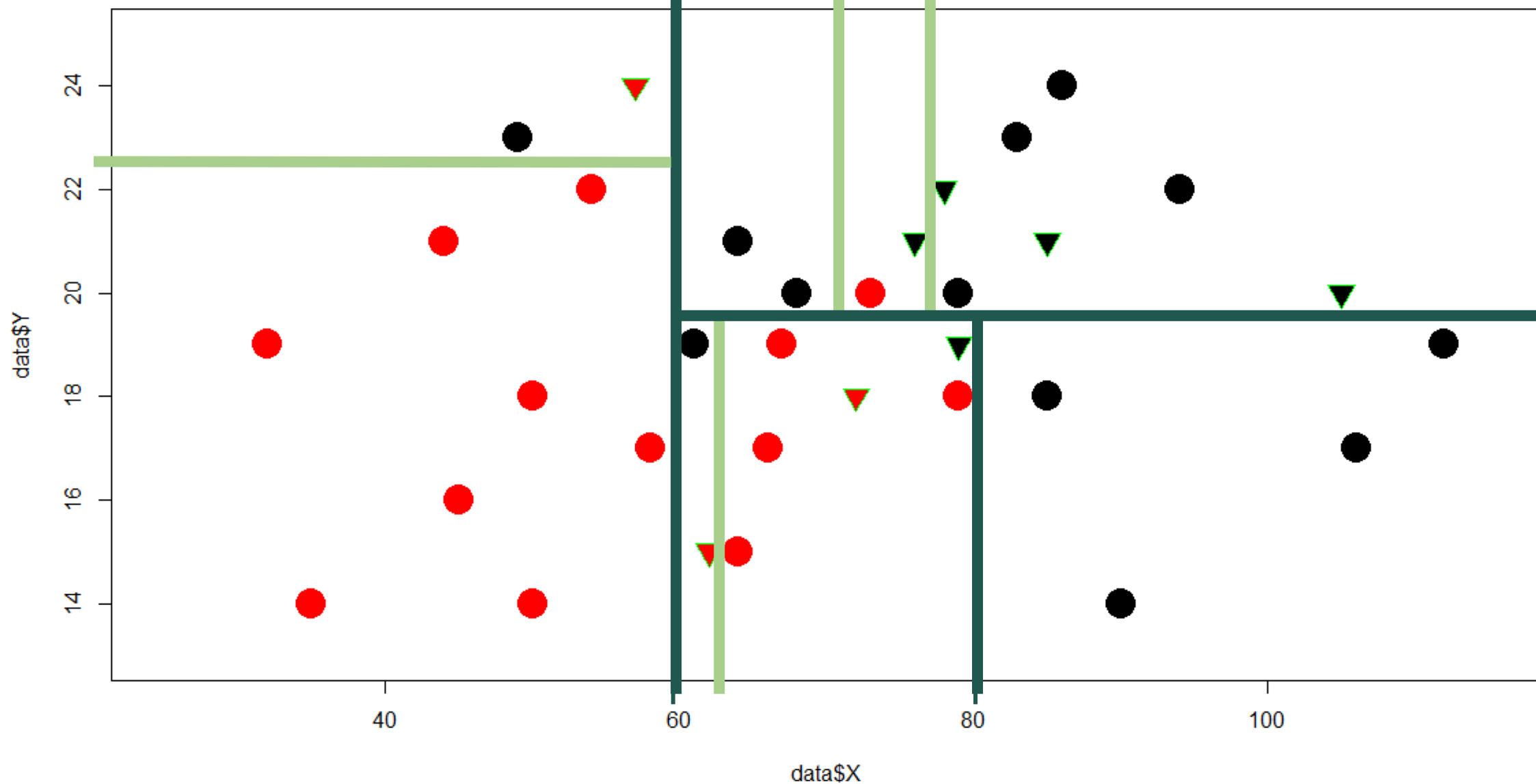


Independent Validation

Fully grown tree

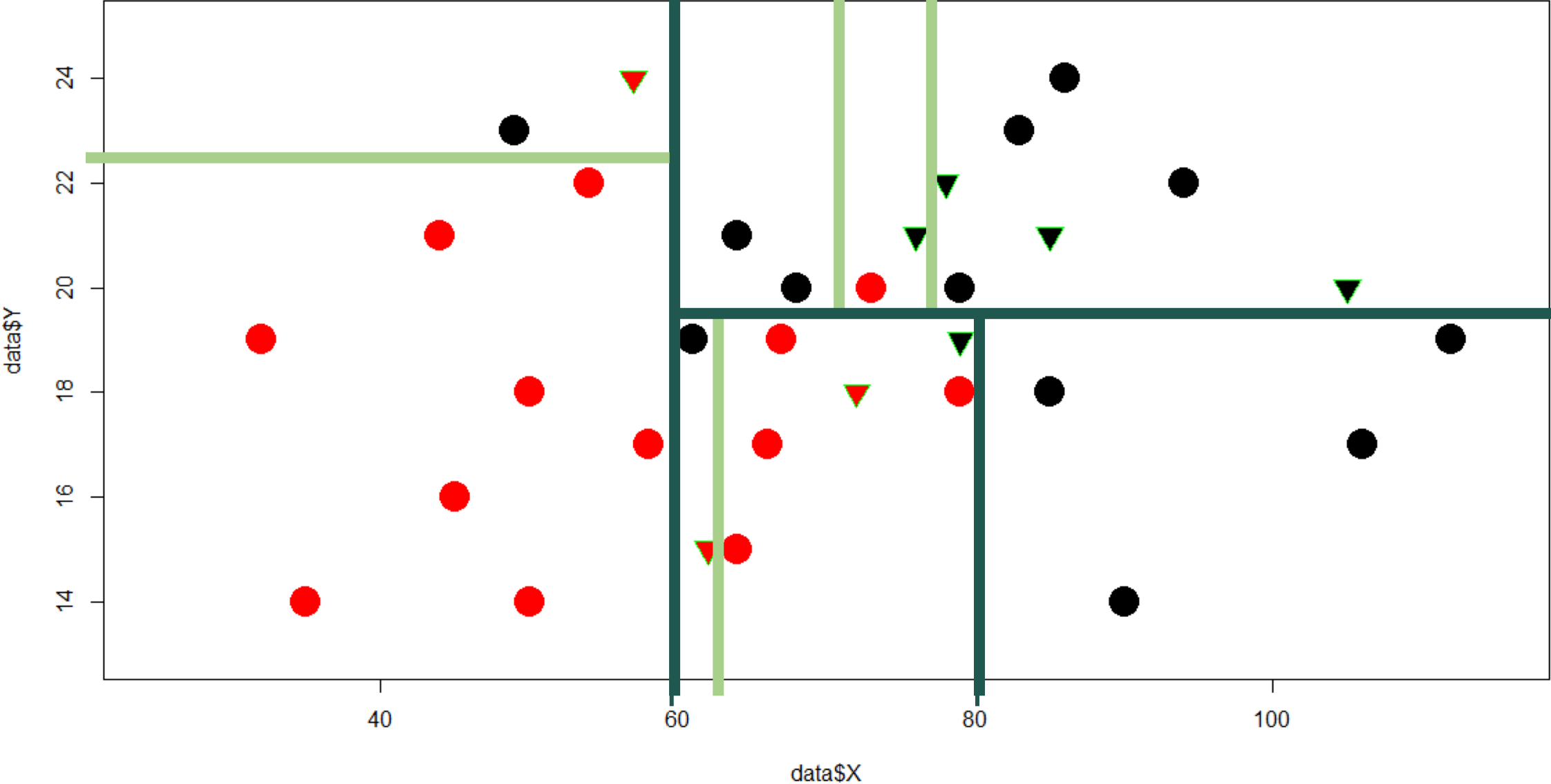
Model Accuracy = 100%

Independent Accuracy = 62.5%



Independent Validation

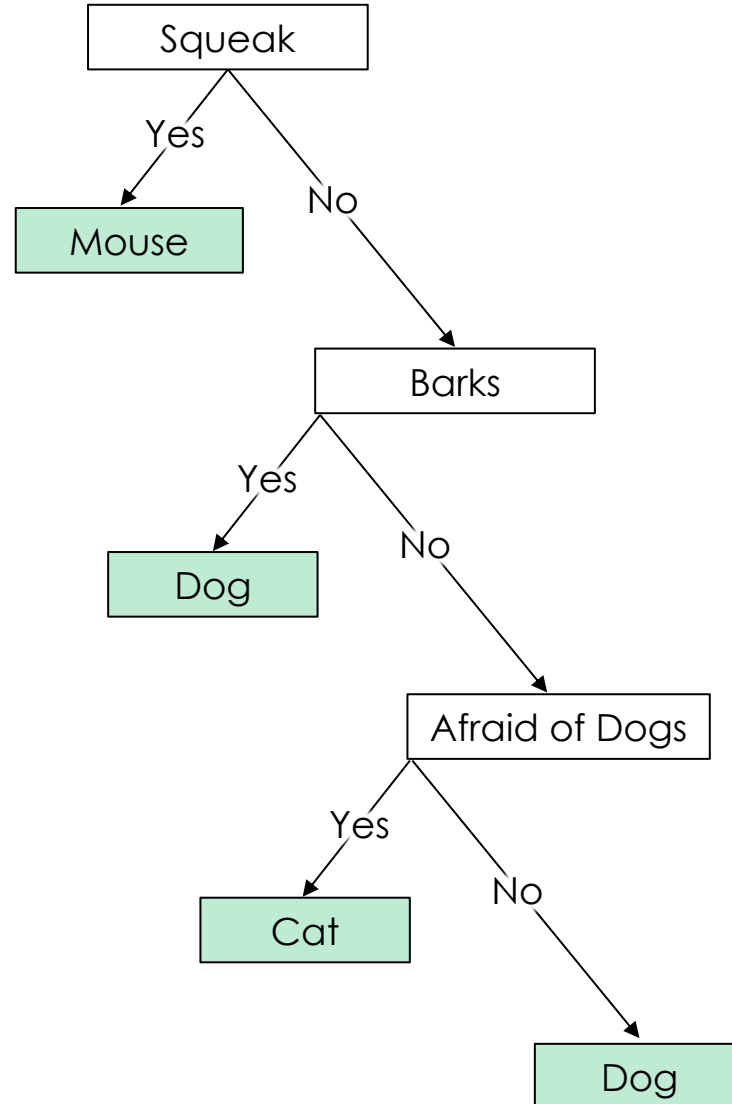
Pruned Tree
Model Accuracy = 88%
Independent Accuracy = 87.5%



Classification and Regression Trees (CART): Basics

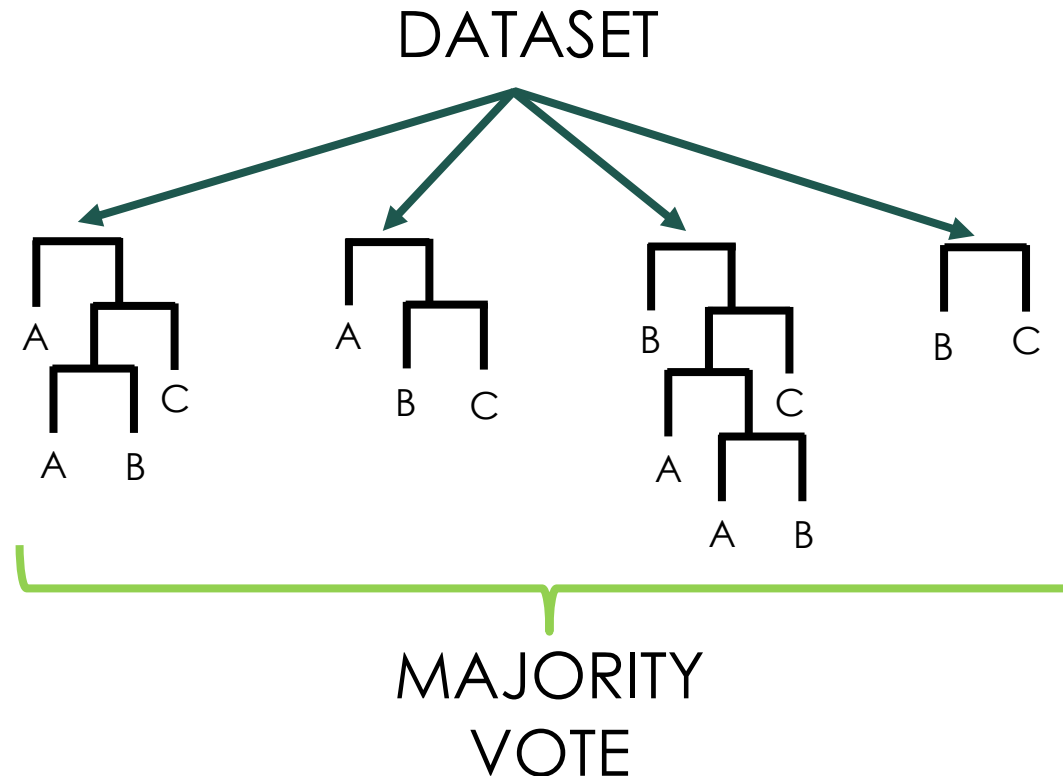
- Important Terms:

	Species	Barks	Pet	Squeeks	Meows	Collar	Afraid of Dogs	Length	Height	Likes Cheese
1	Mouse	No	Yes	Yes	No	No	Yes	0.14	0.07	Yes
2	Mouse	No	No	Yes	No	No	Yes	0.18	0.09	Yes
4	Mouse	No	Yes	Yes	No	No	Yes	0.13	0.06	Yes
4	Mouse	No	Yes	Yes	No	No	Yes	0.13	0.06	Yes
5	Mouse	No	No	Yes	No	No	Yes	0.13	0.07	Yes
6	Mouse	No	No	Yes	No	No	Yes	0.11	0.05	Yes
9	Mouse	No	No	Yes	No	No	Yes	0.15	0.08	Yes
9	Mouse	No	No	Yes	No	No	Yes	0.15	0.08	Yes
12	Cat	No	Yes	No	No	Yes	Yes	0.40	0.15	No
13	Cat	No	Yes	No	Yes	Yes	Yes	0.37	0.09	Yes
14	Cat	No	Yes	No	Yes	Yes	Yes	0.36	0.17	Yes
15	Cat	No	Yes	No	Yes	Yes	No	0.30	0.16	Yes
16	Cat	No	Yes	No	Yes	Yes	Yes	0.30	0.16	Yes
16	Cat	No	Yes	No	Yes	Yes	Yes	0.30	0.16	Yes
18	Cat	No	Yes	No	Yes	Yes	Yes	0.33	0.22	Yes
18	Cat	No	Yes	No	Yes	Yes	Yes	0.33	0.22	Yes
20	Dog	Yes	Yes	No	No	Yes	No	0.53	0.35	Yes
20	Dog	Yes	Yes	No	No	Yes	No	0.53	0.35	Yes
21	Dog	Yes	Yes	No	No	Yes	No	0.51	0.33	Yes
21	Dog	Yes	Yes	No	No	Yes	No	0.51	0.33	Yes
22	Dog	Yes	Yes	No	No	Yes	No	0.16	0.32	Yes
22	Dog	Yes	Yes	No	No	Yes	No	0.16	0.32	Yes
23	Dog	No	No	No	No	No	No	0.52	0.26	Yes
24	Dog	Yes	Yes	No	No	Yes	No	0.53	0.27	Yes
24	Dog	Yes	Yes	No	No	Yes	No	0.53	0.27	Yes
27	Dog	No	Yes	No	No	Yes	No	0.58	0.29	Yes
27	Dog	No	Yes	No	No	Yes	No	0.58	0.29	Yes



Random Forests: Basics

- A “forest” of binary decision trees (Breiman, 2001)
- Ensemble learning technique
- Works well with high-dimensional datasets (continuous or categorical)



Random Forests: Basics

- Random Forests vs. CART

	CART	Random Forests
Number of Trees	1	$n \gg 1$
Pruning	Applied	All fully grown
Variables tested for splitting	All	$m \ll M$ (all variables)
Training Set	All	$\approx 2/3$
Accuracy	Independent required	Internally estimated (OOBE)

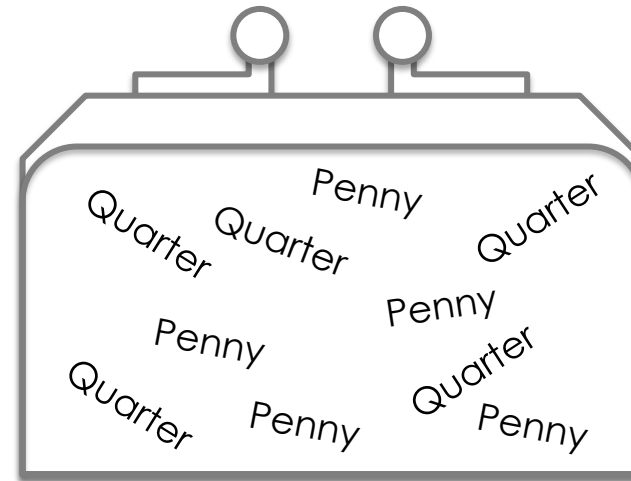
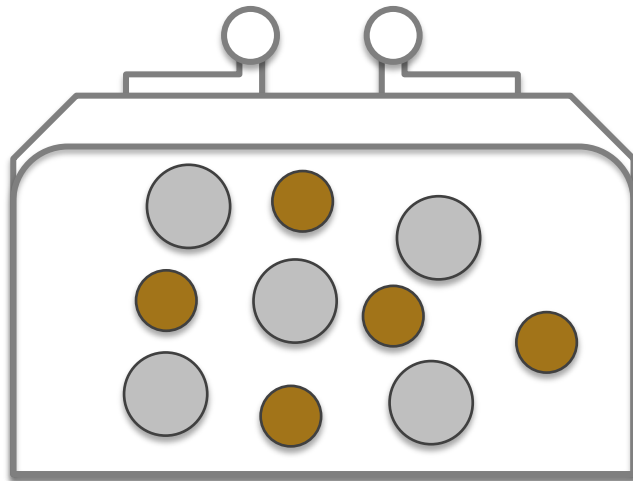
Random Forests: Basics

- `ntree` – number of trees per forest.
- Probability – number of trees that voted with the majority divided by the total number of trees.
- `mtry` – number of variables tested to determine optimal split at each node.
- Out of Bag Accuracy – internal validation; based on $\approx 1/3$ of the dataset not used during the construction of a given tree.
- Mean Decrease in Accuracy (MDA) – quantifies variable “importance” by measuring the change in accuracy after the values of the variable are randomly permuted.

Random Forests: Basics

- Gini Impurity: probability of classifying a data point incorrectly
- Randomly pick a point in dataset & randomly classify it according to class distribution
- $25\% + 25\% = 50\%$; Gini Impurity = 0.5

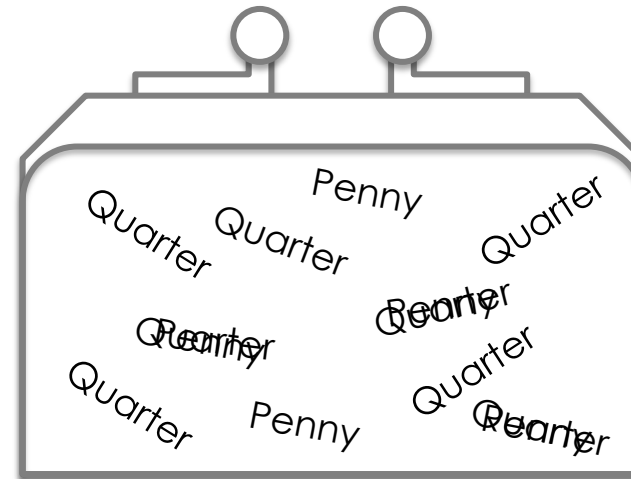
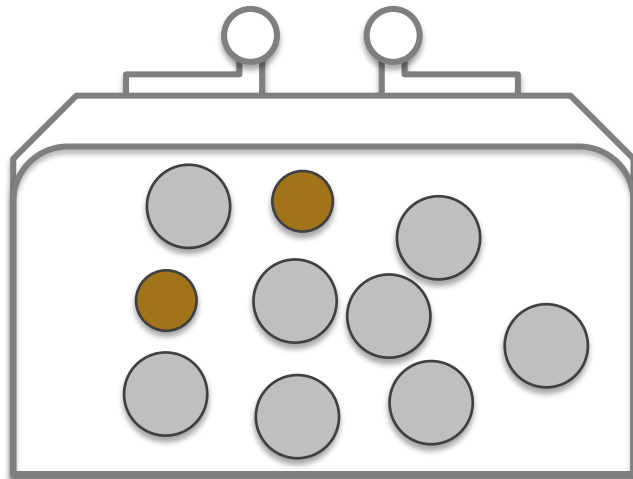
Event	Probability
Choose quarter (50%), classify penny (50%)	$50\% \times 50\% = 25\%$
Choose penny (50%), classify quarter (50%)	$50\% \times 50\% = 25\%$



Random Forests: Basics

- Gini Impurity: probability of classifying a data point incorrectly
- Randomly pick a point in dataset & randomly classify it according to class distribution
- $16\% + 16\% = 32\%$; Gini Impurity = 0.32

Event	Probability
Choose quarter (80%), classify penny (20%)	$80\% \times 20\% = 16\%$
Choose penny (20%), classify quarter (80%)	$20\% \times 80\% = 16\%$



Random Forests: Basics

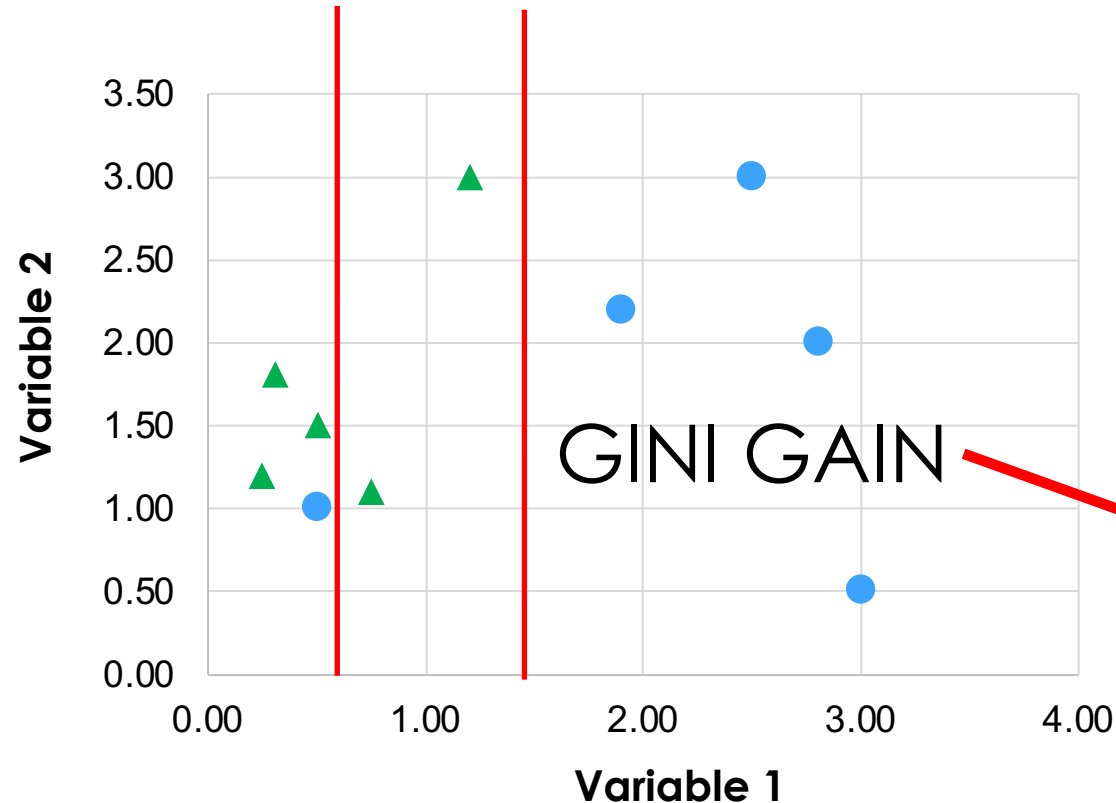
- How Gini Impurity is used for splitting:
 - We know the Gini Impurity of the whole dataset: $G = 0.5*(1-0.5)+0.5*(1-0.5) = 0.5$
 - We don't know where the perfect split is, but we can test all possible splits
 - Determine quality of split by weighting impurity of subsequent branches resulting from split

Gini Impurities

Before the split: 0.50

Right Branch = 0.44

Left Branch = 0.38



Left branch has 4 elements

Right branch has 6

$$(0.40*0.38) + (0.60*0.44) = 0.42$$

With this split the amount of impurity removed is:

$$0.5 - 0.42 = 0.08$$

Random Forests: Basics

- Random Forests Inputs:
- Labelled classes with associated predictor variables
 - Categorical or continuous

	Species	Barks	Pet	Squeeks	Meows	Collar	Afraid of Dogs	Length	Height	Likes Cheese
1	Mouse	No	Yes	Yes	No	No	Yes	0.14	0.07	Yes
2	Mouse	No	No	Yes	No	No	Yes	0.18	0.09	Yes
4	Mouse	No	Yes	Yes	No	No	Yes	0.13	0.06	Yes
4	Mouse	No	Yes	Yes	No	No	Yes	0.13	0.06	Yes
5	Mouse	No	No	Yes	No	No	Yes	0.13	0.07	Yes
6	Mouse	No	No	Yes	No	No	Yes	0.11	0.05	Yes
9	Mouse	No	No	Yes	No	No	Yes	0.15	0.08	Yes
9	Mouse	No	No	Yes	No	No	Yes	0.15	0.08	Yes
12	Cat	No	Yes	No	No	Yes	Yes	0.4	0.15	No
13	Cat	No	Yes	No	Yes	Yes	Yes	0.37	0.09	Yes
14	Cat	No	Yes	No	Yes	Yes	Yes	0.36	0.17	Yes
16	Cat	No	Yes	No	Yes	Yes	Yes	0.3	0.16	Yes
16	Cat	No	Yes	No	Yes	Yes	Yes	0.3	0.16	Yes
16	Cat	No	Yes	No	Yes	Yes	Yes	0.30	0.16	Yes
18	Cat	No	Yes	No	Yes	Yes	Yes	0.33	0.22	Yes
18	Cat	No	Yes	No	Yes	Yes	Yes	0.33	0.22	Yes
20	Dog	Yes	Yes	No	No	Yes	No	0.53	0.35	Yes
20	Dog	Yes	Yes	No	No	Yes	No	0.53	0.35	Yes
21	Dog	Yes	Yes	No	No	Yes	No	0.51	0.33	Yes
21	Dog	Yes	Yes	No	No	Yes	No	0.51	0.33	Yes
22	Dog	Yes	Yes	No	No	Yes	No	0.16	0.32	Yes
22	Dog	Yes	Yes	No	No	Yes	No	0.16	0.32	Yes
23	Dog	No	No	No	No	No	No	0.52	0.26	Yes
24	Dog	Yes	Yes	No	No	Yes	No	0.53	0.27	Yes
24	Dog	Yes	Yes	No	No	Yes	No	0.53	0.27	Yes
27	Dog	No	Yes	No	No	Yes	No	0.58	0.29	Yes
27	Dog	No	Yes	No	No	Yes	No	0.58	0.29	Yes

Random Forests: Example

- 1) Create the training data to grow the tree
 - For N number of cases, randomly sample N cases (with replacement)

Original

Species	Barks	Pet	Squeaks	Meows	Collar	Afraid of Dogs	Length	Height	Likes Cheese
1	Mouse	No	Yes	Yes	No	Yes	0.14	0.07	Yes
2	Mouse	No	No	Yes	No	Yes	0.18	0.09	Yes
3	Mouse	No	No	Yes	No	Yes	0.13	0.06	Yes
4	Mouse	No	Yes	Yes	No	Yes	0.13	0.06	Yes
5	Mouse	No	No	Yes	No	Yes	0.13	0.07	Yes
6	Mouse	No	No	Yes	No	Yes	0.11	0.05	Yes
7	Mouse	No	No	Yes	No	Yes	0.13	0.06	Yes
8	Mouse	No	No	Yes	No	Yes	0.16	0.08	Yes
9	Mouse	No	No	Yes	No	Yes	0.15	0.08	Yes
10	Cat	No	Yes	No	Yes	Yes	0.31	0.19	Yes
11	Cat	No	Yes	No	Yes	No	0.38	0.20	Yes
12	Cat	No	Yes	No	Yes	Yes	0.40	0.15	No
13	Cat	No	Yes	No	Yes	Yes	0.37	0.09	Yes
14	Cat	No	Yes	No	Yes	Yes	0.36	0.17	Yes
15	Cat	No	No	No	Yes	No	0.32	0.22	Yes
16	Cat	No	Yes	No	Yes	Yes	0.30	0.16	Yes
17	Cat	No	Yes	No	Yes	Yes	0.35	0.24	Yes
18	Cat	No	Yes	No	Yes	Yes	0.33	0.22	Yes
19	Dog	Yes	No	No	No	No	0.58	0.33	Yes
20	Dog	Yes	Yes	No	No	Yes	0.53	0.35	Yes
21	Dog	Yes	Yes	No	No	Yes	0.51	0.33	Yes
22	Dog	Yes	Yes	No	No	Yes	0.16	0.32	Yes
23	Dog	No	No	No	No	No	0.52	0.26	Yes
24	Dog	Yes	Yes	No	No	Yes	0.53	0.27	Yes
25	Dog	Yes	Yes	No	No	Yes	0.37	0.16	Yes
26	Dog	Yes	Yes	No	No	Yes	0.53	0.29	Yes
27	Dog	Yes	Yes	No	No	Yes	0.58	0.29	Yes

Training Set

	Species	Barks	Pet	Squeaks	Meows	Collar	Afraid of Dogs	Length	Height	Likes Cheese
1	Mouse	No	Yes	Yes	No	No	Yes	0.14	0.07	Yes
2	Mouse	No	No	Yes	No	No	Yes	0.18	0.09	Yes
4	Mouse	No	Yes	Yes	No	No	Yes	0.13	0.06	Yes
4	Mouse	No	Yes	Yes	No	No	Yes	0.13	0.06	Yes
5	Mouse	No	No	Yes	No	No	Yes	0.13	0.07	Yes
6	Mouse	No	No	Yes	No	No	Yes	0.11	0.05	Yes
9	Mouse	No	No	Yes	No	No	Yes	0.15	0.08	Yes
9	Mouse	No	No	Yes	No	No	Yes	0.15	0.08	Yes
12	Cat	No	Yes	No	Yes	Yes	Yes	0.4	0.15	No
13	Cat	No	Yes	No	Yes	Yes	Yes	0.37	0.09	Yes
14	Cat	No	Yes	No	Yes	Yes	Yes	0.36	0.17	Yes
16	Cat	No	Yes	No	Yes	Yes	Yes	0.3	0.16	Yes
16	Cat	No	Yes	No	Yes	Yes	Yes	0.3	0.16	Yes
16	Cat	No	Yes	No	Yes	Yes	Yes	0.3	0.16	Yes
18	Cat	No	Yes	No	Yes	Yes	Yes	0.33	0.22	Yes
18	Cat	No	Yes	No	Yes	Yes	Yes	0.33	0.22	Yes
20	Dog	Yes	Yes	No	No	Yes	No	0.53	0.35	Yes
20	Dog	Yes	Yes	No	No	Yes	No	0.53	0.35	Yes
21	Dog	Yes	Yes	No	No	Yes	No	0.51	0.33	Yes
21	Dog	Yes	Yes	No	No	Yes	No	0.51	0.33	Yes
22	Dog	Yes	Yes	No	No	Yes	No	0.16	0.32	Yes
22	Dog	Yes	Yes	No	No	Yes	No	0.16	0.32	Yes
23	Dog	No	No	No	No	No	No	0.52	0.26	Yes
24	Dog	Yes	Yes	No	No	Yes	No	0.53	0.27	Yes
24	Dog	Yes	Yes	No	No	Yes	No	0.53	0.27	Yes
27	Dog	No	Yes	No	No	Yes	No	0.58	0.29	Yes
27	Dog	No	Yes	No	No	Yes	No	0.58	0.29	Yes

Remaining
for Internal
Accuracy
Assessment

≈1/3

Random Forests: Example

- 2) For M number of variables randomly select a subset ($m \ll M$) to determine how each node is split (**mtry**)
 - For each variable evaluate **ALL** splits

Training Set

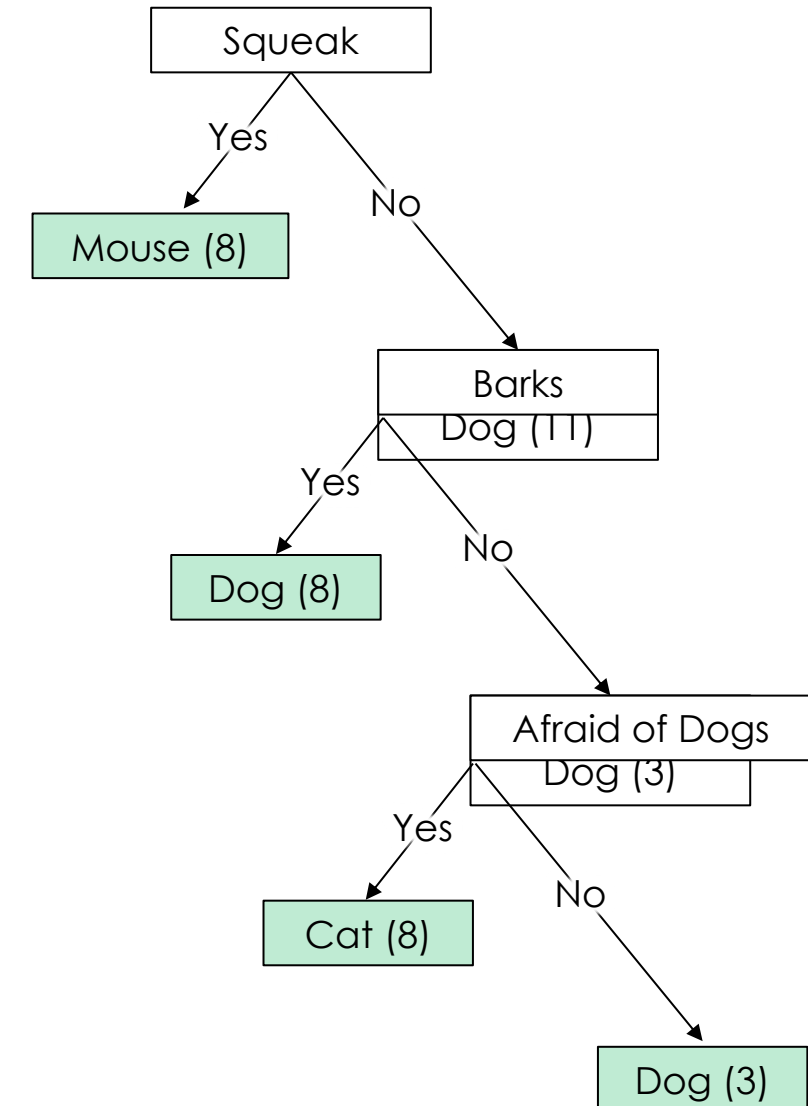
Species	Barks	Pet	Squeaks	Meows	Collar	Afraid of Dogs	Length	Height	Likes Cheese
---------	-------	-----	---------	-------	--------	----------------	--------	--------	--------------

12	Cat	No	Yes	No	No	Yes	Yes	0.4	0.15	No
13	Cat	No	Yes	No	Yes	Yes	Yes	0.37	0.09	Yes
14	Cat	No	Yes	No	Yes	Yes	Yes	0.36	0.17	Yes
16	Cat	No	Yes	No	Yes	Yes	Yes	0.3	0.16	Yes
16	Cat	No	Yes	No	Yes	Yes	Yes	0.3	0.16	Yes
16	Cat	No	Yes	No	Yes	Yes	Yes	0.30	0.16	Yes
18	Cat	No	Yes	No	Yes	Yes	Yes	0.33	0.22	Yes
18	Cat	No	Yes	No	Yes	Yes	Yes	0.33	0.22	Yes

23	Dog	No	No	No	No	No	No	0.52	0.26	Yes
27	Dog	No	Yes	No	No	Yes	No	0.58	0.29	Yes
27	Dog	No	Yes	No	No	Yes	No	0.58	0.29	Yes

Gini GAIN
to Identify Optimal Split

Variable	Evaluated Split Points	Gini
Cheese	Yes or No	0.04
Barks	Yes or No	0.26
Collar	Yes or No	0.02
Height	≤ 0.06	0.09
Height	≤ 0.07	0.17
Height	≤ 0.08	0.26
Height	≤ 0.09	0.28
Height	≤ 0.15	0.25
Height	≤ 0.16	0.26
Height	≤ 0.17	0.28
Height	≤ 0.22	0.27
Height	≤ 0.26	0.31
Height	≤ 0.27	0.22
Height	≤ 0.29	0.18
Height	≤ 0.32	0.09
Height	≤ 0.33	0.04
Height	≤ 0.35	0.00



Random Forests: Example

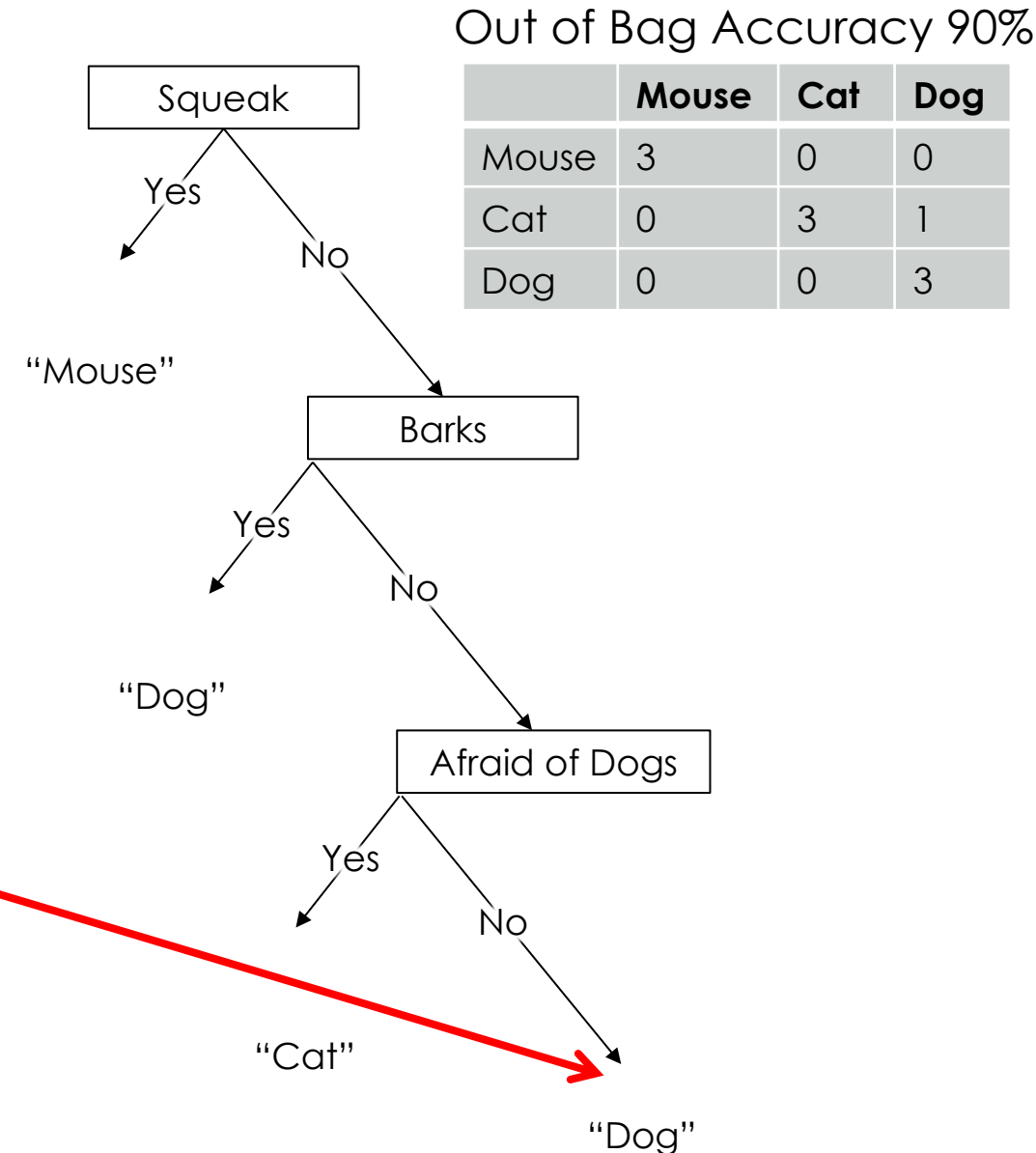
3) Run the Out of Bag Samples Down the Tree and calculate the Out of Bag Accuracy

Original

	Species	Barks	Pet	Squeaks	Meows	Collar	Afraid of Dogs	Length	Height	Likes Cheese
1	Mouse	No	Yes	Yes	No	No	Yes	0.14	0.07	Yes
2	Mouse	No	No	Yes	No	No	Yes	0.18	0.09	Yes
3	Mouse	No	No	Yes	No	No	Yes	0.13	0.06	Yes
4	Mouse	No	Yes	Yes	No	No	Yes	0.13	0.06	Yes
5	Mouse	No	No	Yes	No	No	Yes	0.13	0.07	Yes
6	Mouse	No	No	Yes	No	No	Yes	0.11	0.05	Yes
7	Mouse	No	No	Yes	No	No	Yes	0.13	0.06	Yes
8	Mouse	No	No	Yes	No	No	Yes	0.16	0.08	Yes
9	Mouse	No	No	Yes	No	No	Yes	0.15	0.08	Yes
10	Cat	No	Yes	No	Yes	Yes	Yes	0.31	0.19	Yes
11	Cat	No	Yes	No	Yes	No	Yes	0.38	0.20	Yes
12	Cat	No	Yes	No	Yes	Yes	Yes	0.40	0.15	No
13	Cat	No	Yes	No	Yes	Yes	Yes	0.37	0.09	Yes
14	Cat	No	Yes	No	Yes	Yes	Yes	0.36	0.17	Yes
15	Cat	No	No	No	Yes	No	No	0.32	0.22	Yes
16	Cat	No	Yes	No	Yes	Yes	Yes	0.30	0.16	Yes
17	Cat	No	Yes	No	Yes	Yes	Yes	0.35	0.24	Yes
18	Cat	No	Yes	No	Yes	Yes	Yes	0.33	0.22	Yes
19	Dog	Yes	No	No	No	No	No	0.58	0.33	Yes
20	Dog	Yes	Yes	No	No	Yes	No	0.53	0.35	Yes
21	Dog	Yes	Yes	No	No	Yes	No	0.51	0.33	Yes
22	Dog	Yes	Yes	No	No	Yes	No	0.16	0.32	Yes
23	Dog	No	No	No	No	No	No	0.52	0.26	Yes
24	Dog	Yes	Yes	No	No	Yes	No	0.53	0.27	Yes
25	Dog	Yes	Yes	No	No	Yes	No	0.37	0.16	Yes
26	Dog	Yes	Yes	No	No	Yes	No	0.53	0.29	Yes
27	Dog	Yes	Yes	No	No	Yes	No	0.58	0.29	Yes

Remaining for Internal Accuracy Assessment

≈1/3



Random Forests: Example

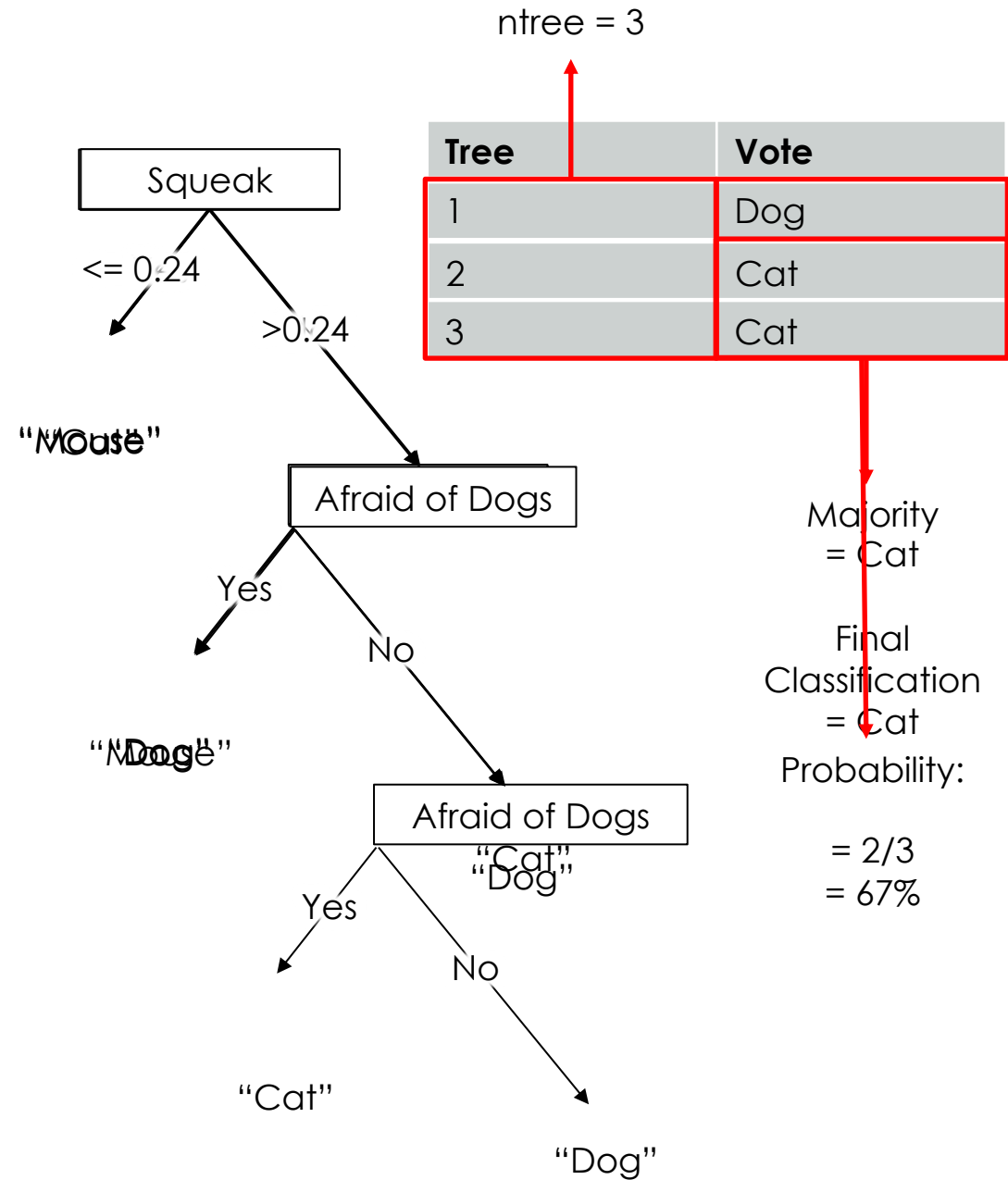
4) Take the mode prediction of all trees (**ntree**) to determine the final classification

Original

Species	Barks	Pet	Squeaks	Meows	Collar	Afraid of Dogs	Length	Height	Likes Cheese
---------	-------	-----	---------	-------	--------	----------------	--------	--------	--------------



Cat	No	Yes	No	Yes	Yes	No	0.35	0.24	Yes
-----	----	-----	----	-----	-----	----	------	------	-----

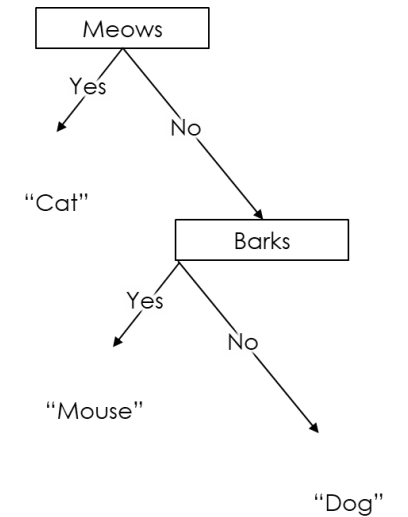
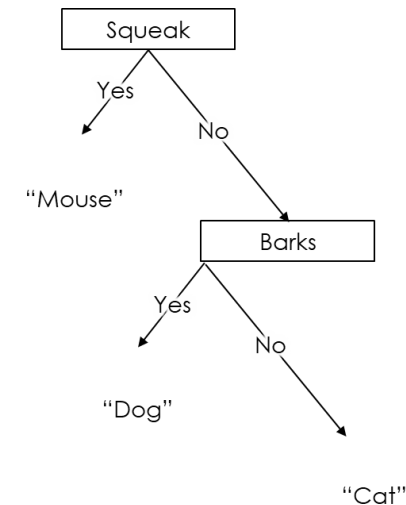
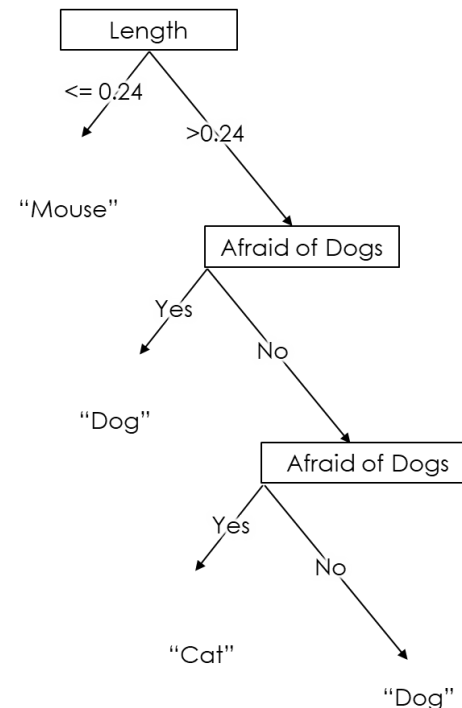


Random Forests: Example

- Ensemble Method:
 - Multiple learners, randomness → diversity

Original

	Species	Barks	Pet	Squeaks	Meows	Collar	Afraid of Dogs	Length	Height	Likes Cheese
1	Mouse	No	Yes	Yes	No	No	Yes	0.14	0.07	Yes
2	Mouse	No	No	Yes	No	No	Yes	0.18	0.09	Yes
3	Mouse	No	No	Yes	No	No	Yes	0.13	0.06	Yes
4	Mouse	No	Yes	Yes	No	No	Yes	0.13	0.06	Yes
5	Mouse	No	No	Yes	No	No	Yes	0.13	0.07	Yes
6	Mouse	No	No	Yes	No	No	Yes	0.11	0.05	Yes
7	Mouse	No	No	Yes	No	No	Yes	0.13	0.06	Yes
8	Mouse	No	No	Yes	No	No	Yes	0.16	0.08	Yes
9	Mouse	No	No	Yes	No	No	Yes	0.15	0.08	Yes
10	Cat	No	Yes	No	Yes	Yes	Yes	0.31	0.19	Yes
11	Cat	No	Yes	No	Yes	No	Yes	0.38	0.20	Yes
12	Cat	No	Yes	No	Yes	Yes	Yes	0.40	0.15	No
13	Cat	No	Yes	No	Yes	Yes	Yes	0.37	0.09	Yes
14	Cat	No	Yes	No	Yes	Yes	Yes	0.36	0.17	Yes
15	Cat	No	No	No	Yes	No	No	0.32	0.22	Yes
16	Cat	No	Yes	No	Yes	Yes	Yes	0.30	0.16	Yes
17	Cat	No	Yes	No	Yes	Yes	No	0.35	0.24	Yes
18	Cat	No	Yes	No	Yes	Yes	Yes	0.33	0.22	Yes
19	Dog	Yes	No	No	No	No	No	0.58	0.33	Yes
20	Dog	Yes	Yes	No	No	Yes	No	0.53	0.35	Yes
21	Dog	Yes	Yes	No	No	Yes	No	0.51	0.33	Yes
22	Dog	Yes	Yes	No	No	Yes	No	0.16	0.32	Yes
23	Dog	No	No	No	No	No	No	0.52	0.26	Yes
24	Dog	Yes	Yes	No	No	Yes	No	0.53	0.27	Yes
25	Dog	Yes	Yes	No	No	Yes	No	0.37	0.16	Yes
26	Dog	Yes	Yes	No	No	Yes	No	0.53	0.29	Yes
27	Dog	Yes	Yes	No	No	Yes	No	0.58	0.29	Yes



Random Forests: Example

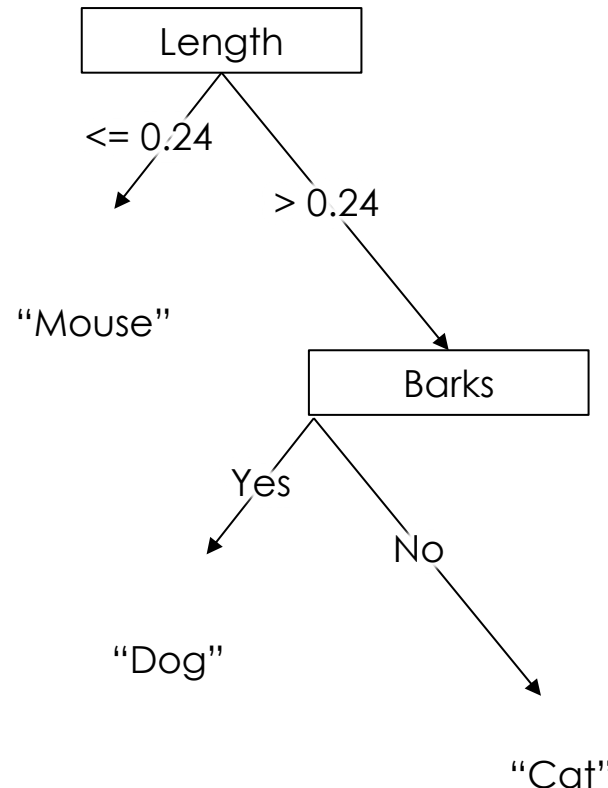
5) Calculating the Variable Importance: Mean Decrease in Accuracy (MDA)

- Values permuted across all trees
- Average decrease in accuracy; normalize by standard deviation

Original

	Species	Barks	Pet	Squeaks	Meows	Collar	Afraid of Dogs	Length	Height	Likes Cheese
1	Mouse	No	Yes	Yes	No	No	Yes		0.07	Yes
2	Mouse	No	No	Yes	No	No	Yes		0.09	Yes
3	Mouse	No	No	Yes	No	No	Yes	0.31	0.06	Yes
4	Mouse	No	Yes	Yes	No	No	Yes		0.06	Yes
5	Mouse	No	No	Yes	No	No	Yes		0.07	Yes
6	Mouse	No	No	Yes	No	No	Yes		0.05	Yes
7	Mouse	No	No	Yes	No	No	Yes	0.32	0.06	Yes
8	Mouse	No	No	Yes	No	No	Yes	0.38	0.08	Yes
9	Mouse	No	No	Yes	No	No	Yes		0.08	Yes
10	Cat	No	Yes	No	Yes	Yes	Yes	0.37	0.19	Yes
11	Cat	No	Yes	No	Yes	No	Yes	0.53	0.20	Yes
12	Cat	No	Yes	No	Yes	Yes	Yes		0.15	No
13	Cat	No	Yes	No	Yes	Yes	Yes		0.09	Yes
14	Cat	No	Yes	No	Yes	Yes	Yes		0.17	Yes
15	Cat	No	No	No	Yes	No	No	0.35	0.22	Yes
16	Cat	No	Yes	No	Yes	Yes	Yes		0.16	Yes
17	Cat	No	Yes	No	Yes	Yes	Yes	0.13	0.24	Yes
18	Cat	No	Yes	No	Yes	Yes	Yes		0.22	Yes
19	Dog	Yes	No	No	No	No	No	0.58	0.33	Yes
20	Dog	Yes	Yes	No	No	Yes	No		0.35	Yes
21	Dog	Yes	Yes	No	No	No	No		0.33	Yes
22	Dog	Yes	Yes	No	No	Yes	No		0.32	Yes
23	Dog	No	No	No	No	No	No		0.26	Yes
24	Dog	Yes	Yes	No	No	Yes	No		0.27	Yes
25	Dog	Yes	Yes	No	No	Yes	No	0.13	0.16	Yes
26	Dog	Yes	Yes	No	No	Yes	No	0.16	0.29	Yes
27	Dog	Yes	Yes	No	No	Yes	No		0.29	Yes

Original Permutated Data



Decrease in Accuracy

$$= 90 - 40$$

$$= 50\%$$

Original Out of Bag Accuracy 90%

	Mouse	Cat	Dog
Mouse	3	0	0
Cat	0	3	1
Dog	0	0	3

Out of Bag Accuracy After Permutation 40%

	Mouse	Cat	Dog
Mouse	0	3	0
Cat	1	3	0
Dog	2	0	1

Random Forests: Advantages

- Limit overfitting, without substantial increases in error, no need for pruning
- Computationally efficient
- Internal accuracy assessment
- Variable importance measures

RANDOM FOREST TUTORIAL

Random Forests: Best Practices for Remote Sensing Applications

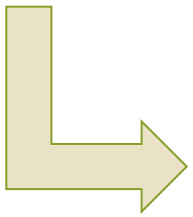
- Training Data
 - Spatial autocorrelation
 - Class weights
 - Representative of class distribution
- Parameters/Outputs
 - Ensemble Averaging
 - Inputs:
 - Correlated Variables

TRAINING DATA:

Spatial Autocorrelation

Random Forests: Best Practices for Remote Sensing Applications

- Spatial autocorrelation
- Myth: there is no need for independent validation when using Random Forests classification algorithm
- Why does this myth exist?
 - Out of bag error randomly selects a sample of data to “hold back” and use for cross validation.



Independent validation can be skipped when this OOB sample is *independent* of the data used to create the tree

- When is OOB data independent of data used to build tree?

Random Forests: Best Practices for Remote Sensing Applications

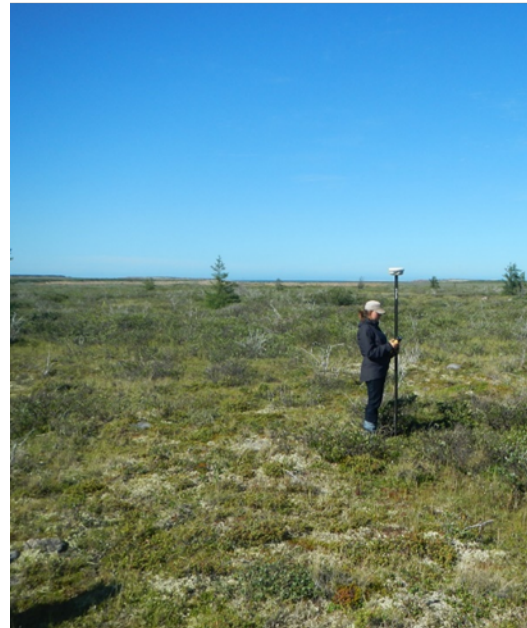
- The first law of geography:

“everything is related to everything else,
but near things are more related than distant things”

-- Waldo Tobler, 1969

- Helps understand how similar closer objects are to other nearby objects

Random Forests: Collecting Training and Testing Data



Collecting Training and Testing Data

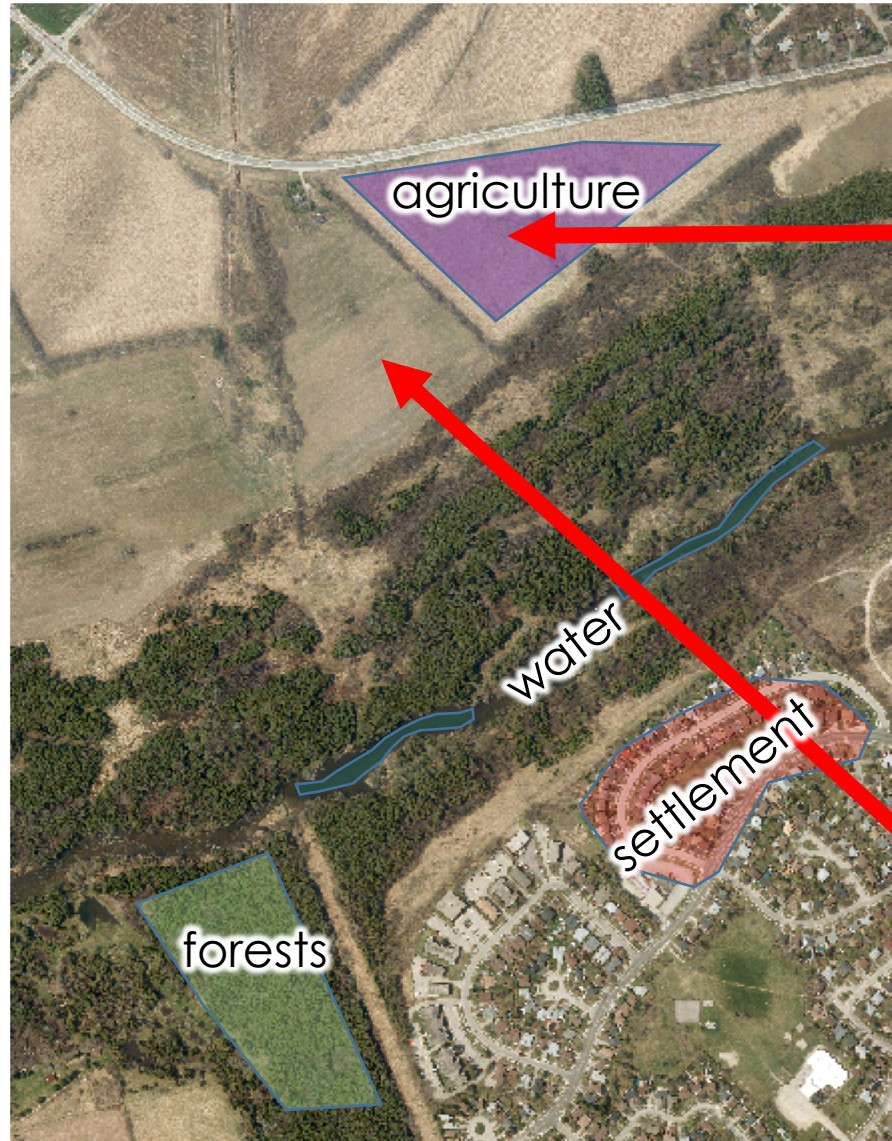
Training Polygons

Pros:

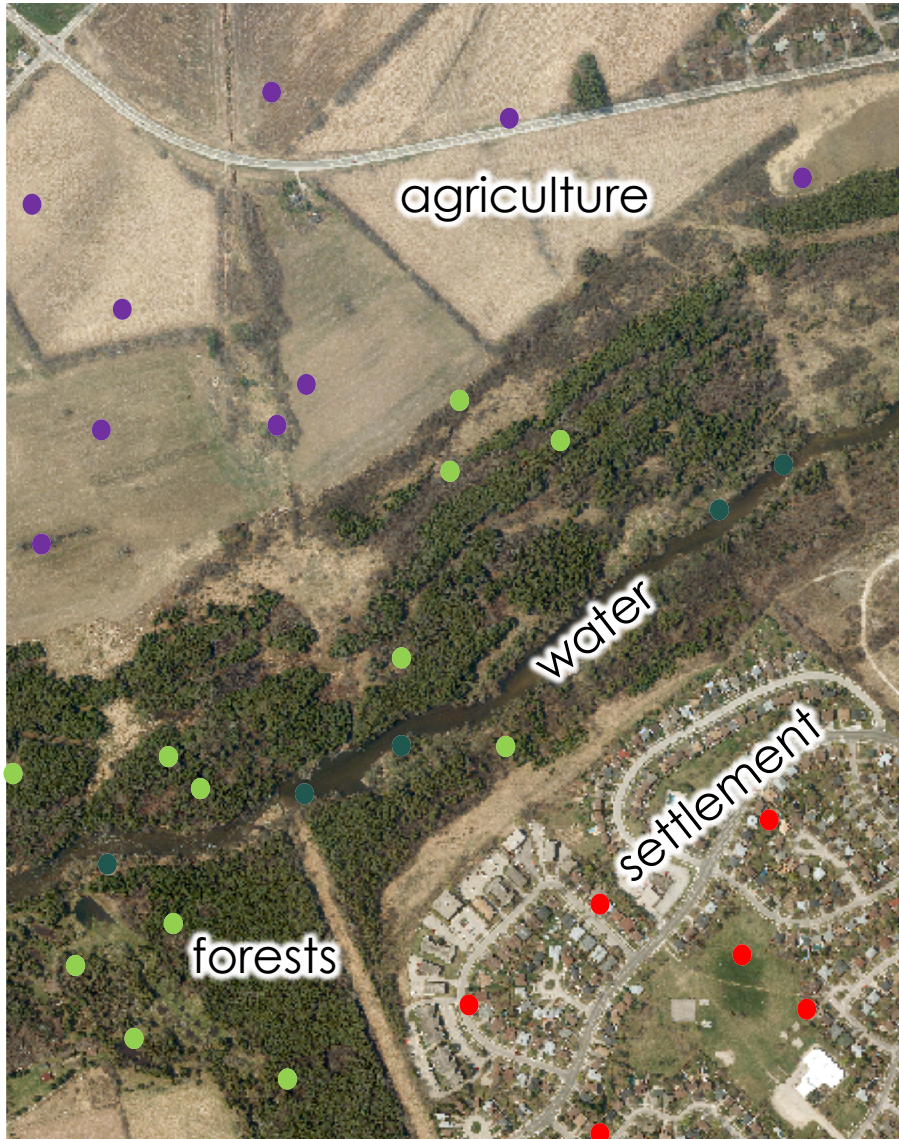
- Quick and easy to collect

Cons

- Might not capture the full variability of the class
- Selective – may only capture areas that are “easy” to classify
- Pixels within the polygon are spatially autocorrelated



Collecting Training and Testing Data



Random Sample of Points

Pros:

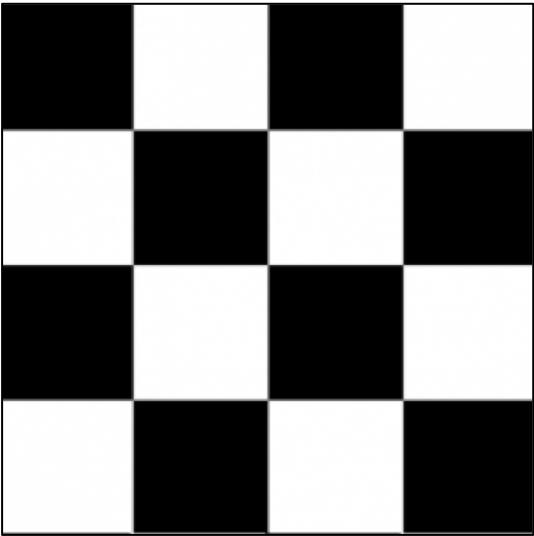
- provides best chance of capturing full variability of all classes
- Provides best representation of the proportions of each class on the landscape

Cons

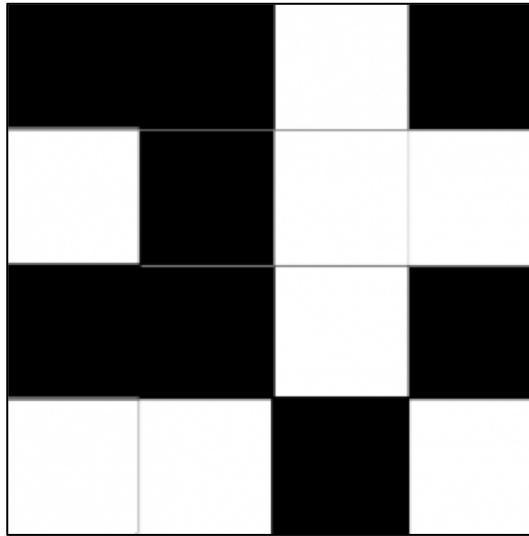
- Time consuming to collect
- Points may be placed on “fuzzy” areas/boundaries
- Need to collect many points, as each point represents a single data value

Spatial Autocorrelation

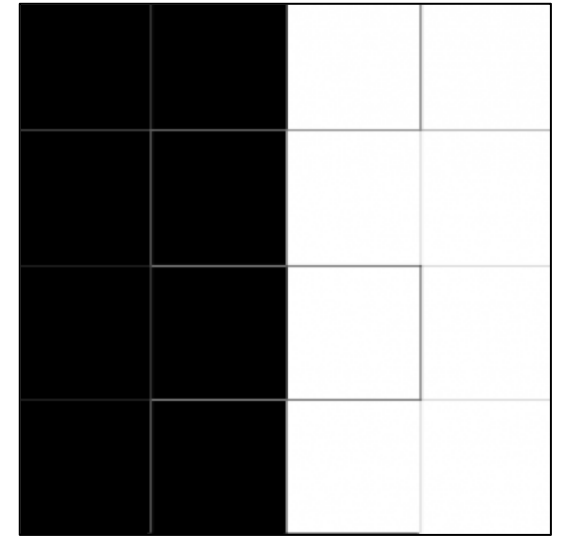
- How to detect/measure Spatial Autocorrelation
- **Moran's Index** (or simply Moran's I) measures spatial autocorrelation
 - -1 is perfect clustering of dissimilar values (perfect dispersion)
 - 0 is no autocorrelation (perfect randomness)
 - +1 indicates perfect clustering of similar values (the opposite of dispersion).



-1



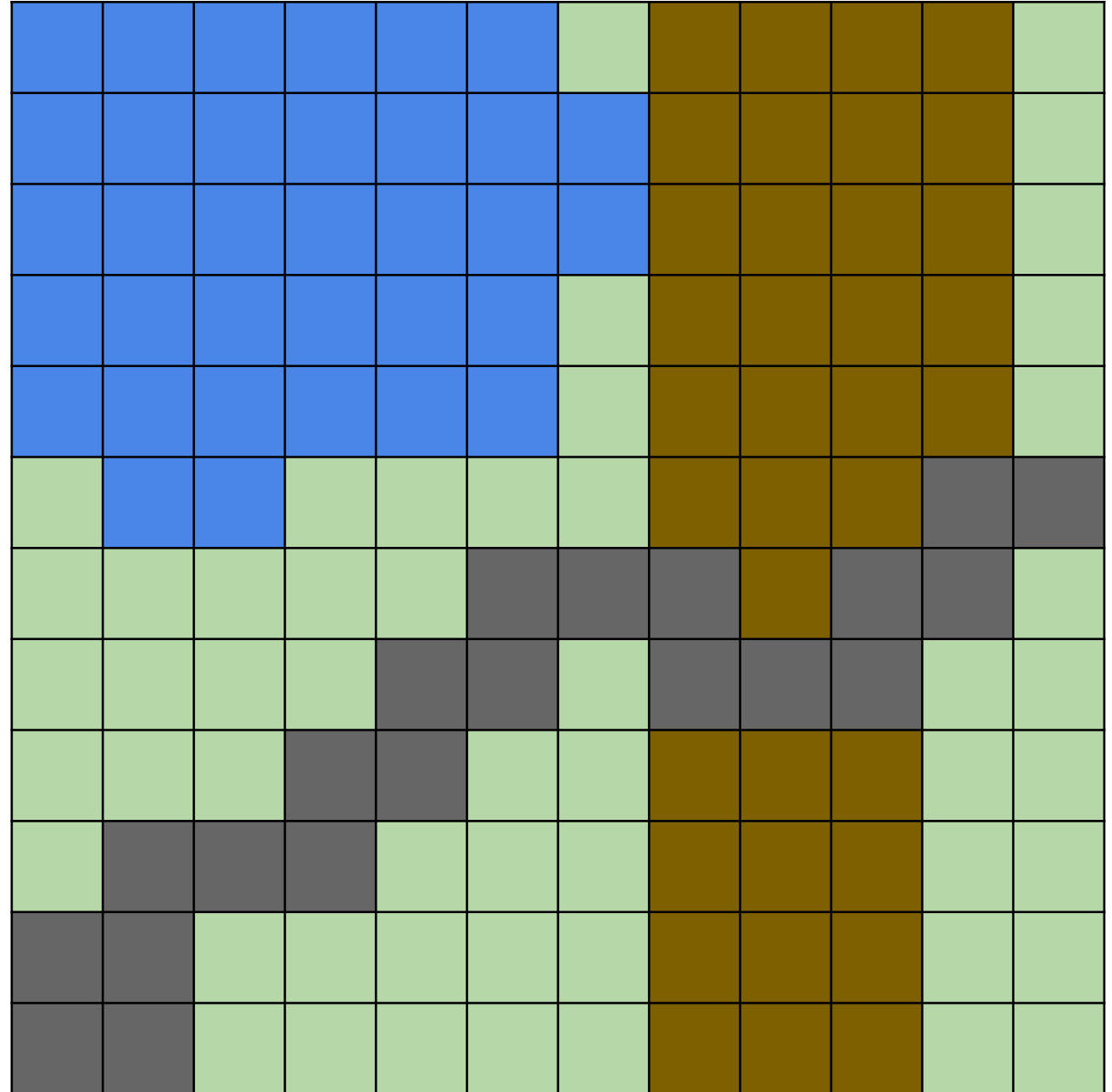
0



+1

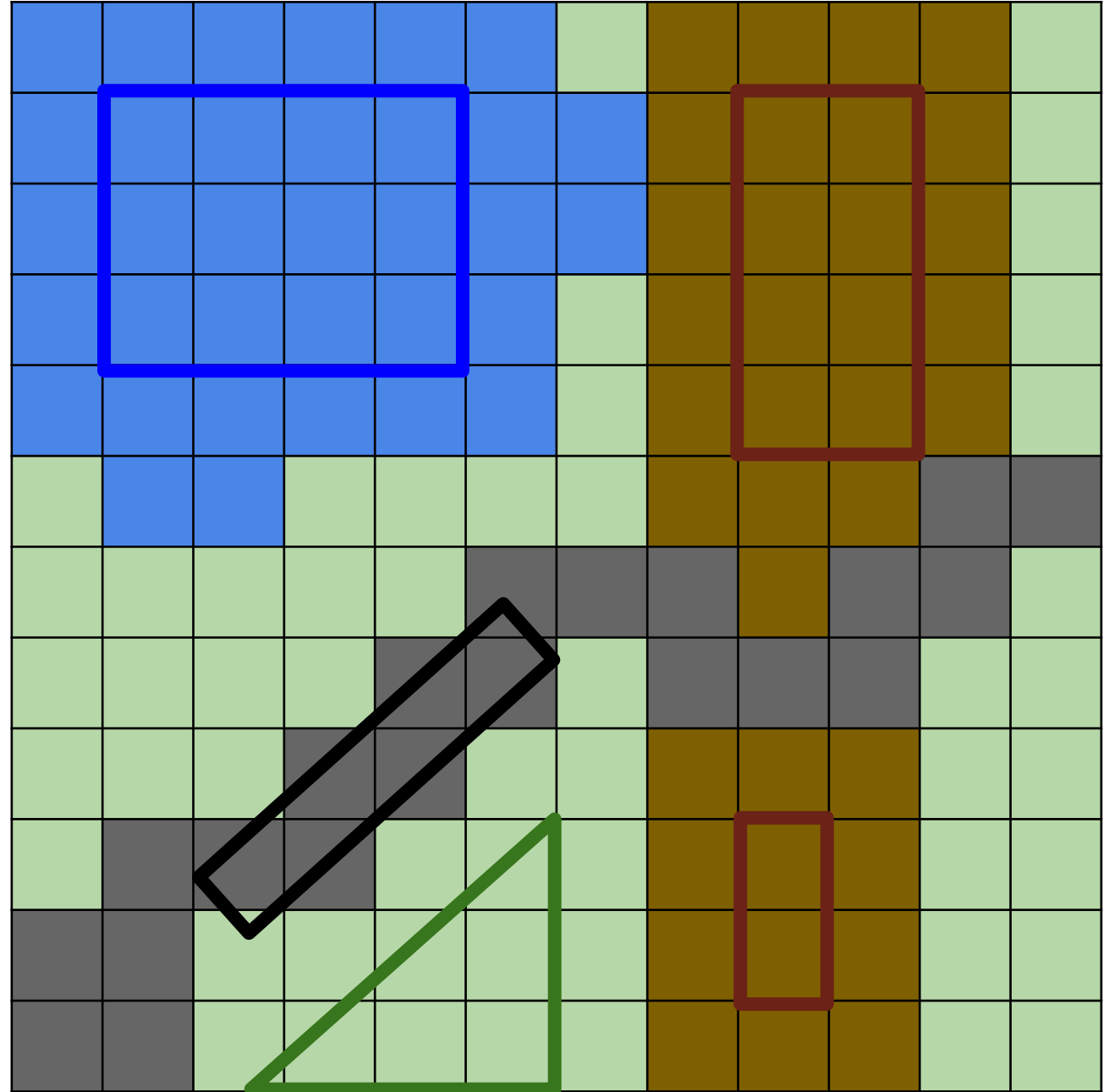
Why Spatial Autocorrelation Is Important in RF

- Out of bag error uses a *random sample* of the “training data” as “testing data”
- 12 x 12 pixels = 144



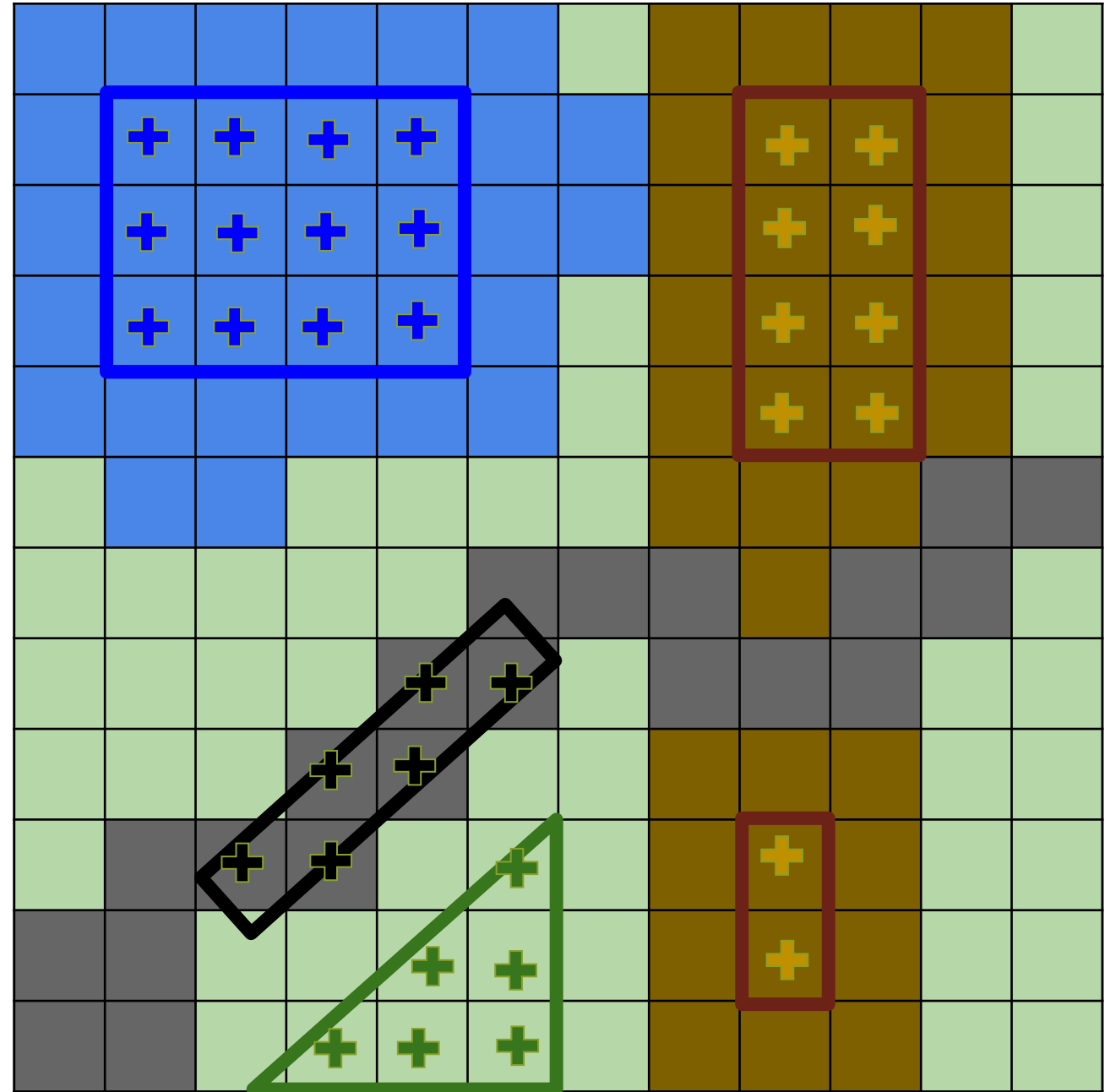
Why Spatial Autocorrelation Is Important in RF

- Out of bag error uses a *random sample* of the “training data” as “testing data”



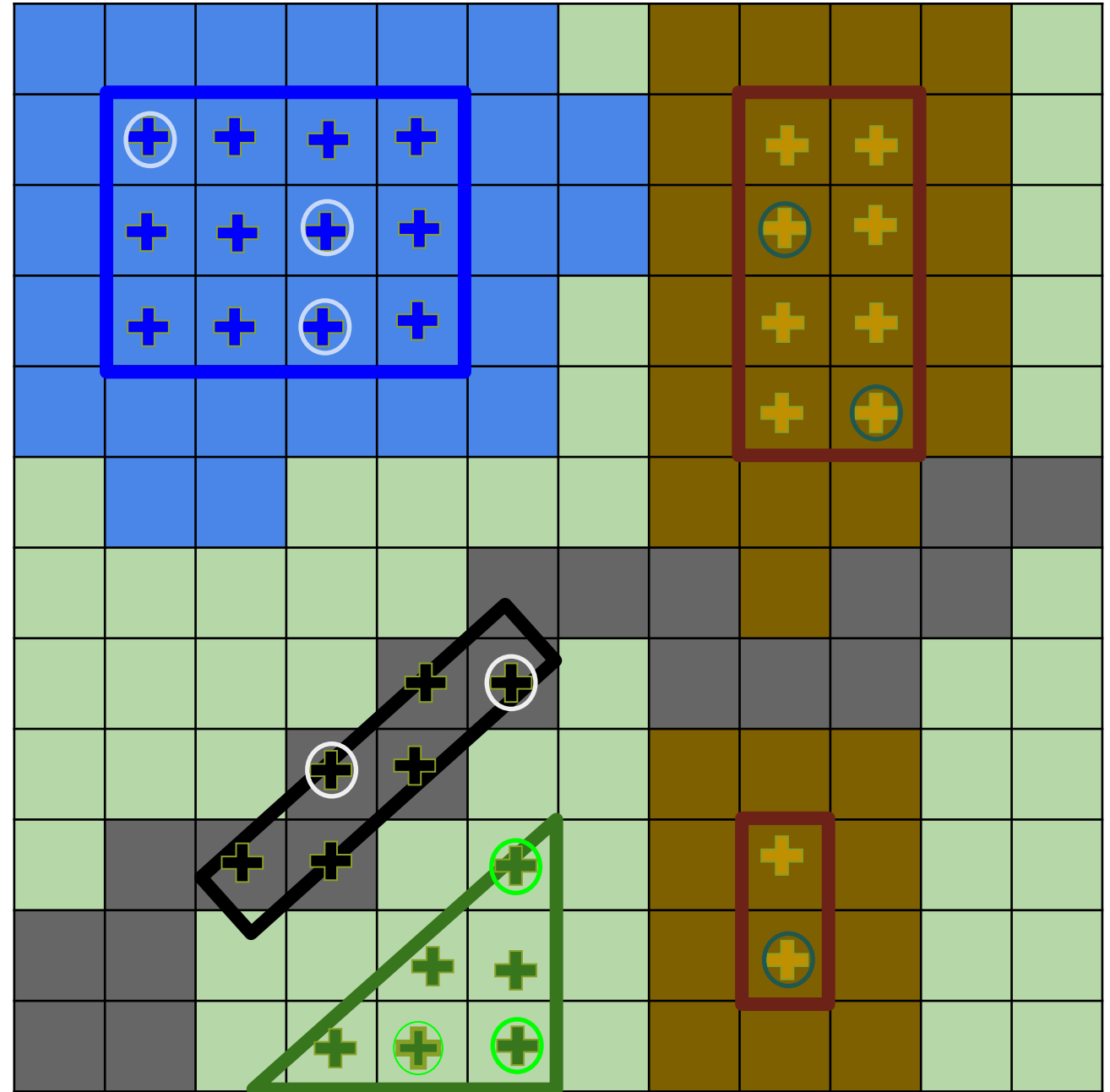
Why Spatial Autocorrelation Is Important in RF

- Out of bag error uses a *random sample* of the “training data” as “testing data”
- $N = 33$ samples



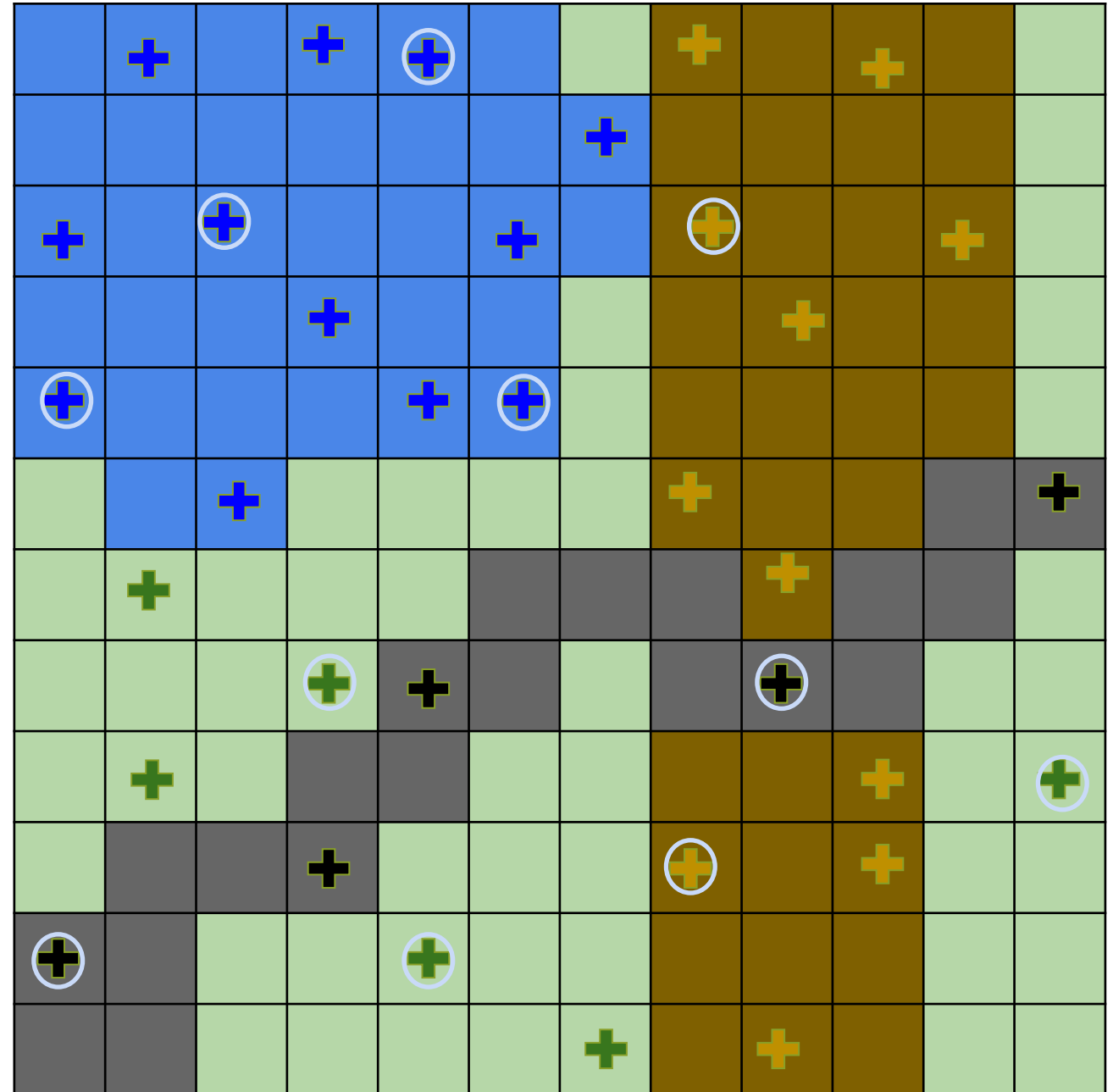
Why Spatial Autocorrelation Is Important in RF

- Out of bag error uses a *random sample* of the “training data” as “testing data”
- $N = 33$ samples
- OOB = 11 randomly selected samples



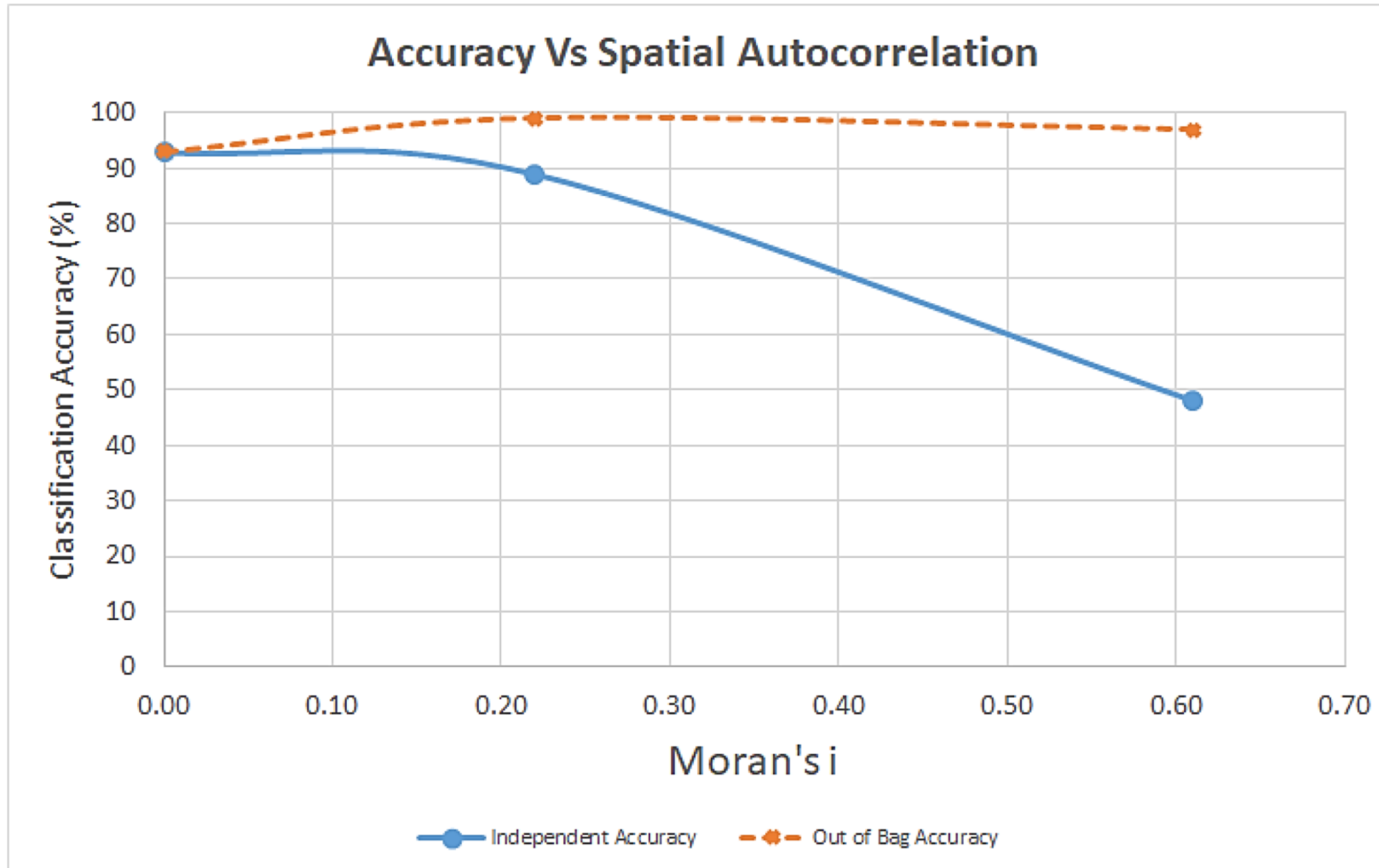
Why Spatial Autocorrelation Is Important in RF

- Out of bag error uses a *random sample* of the “training data” as “testing data”
- 12 x 12 pixels = 144



Effects of Spatial Autocorrelation on OOB

Millard and Richardson, 2015



TRAINING DATA: Imbalanced Data

How Does Balance of Training Data Affect RF Results?

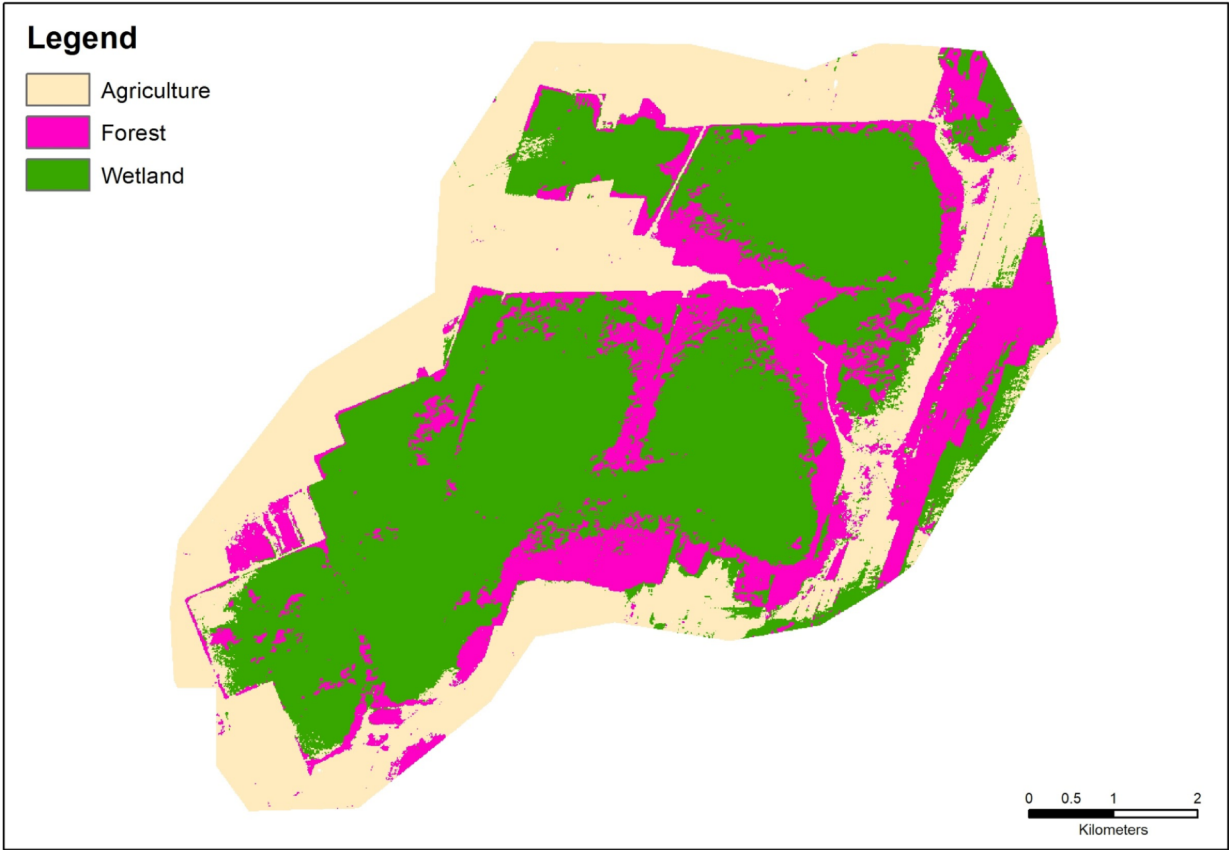
- Most classifiers aim to minimize overall error
- RF will tend to focus more on the prediction accuracy of the majority class, which often results in poor accuracy for the minority class
- In learning extremely imbalanced data, there is a significant probability that a bootstrap sample contains few or even none of the minority class, resulting in a tree with poor performance for predicting the minority class (in cases of poor separability only!)

Excel Spreadsheet

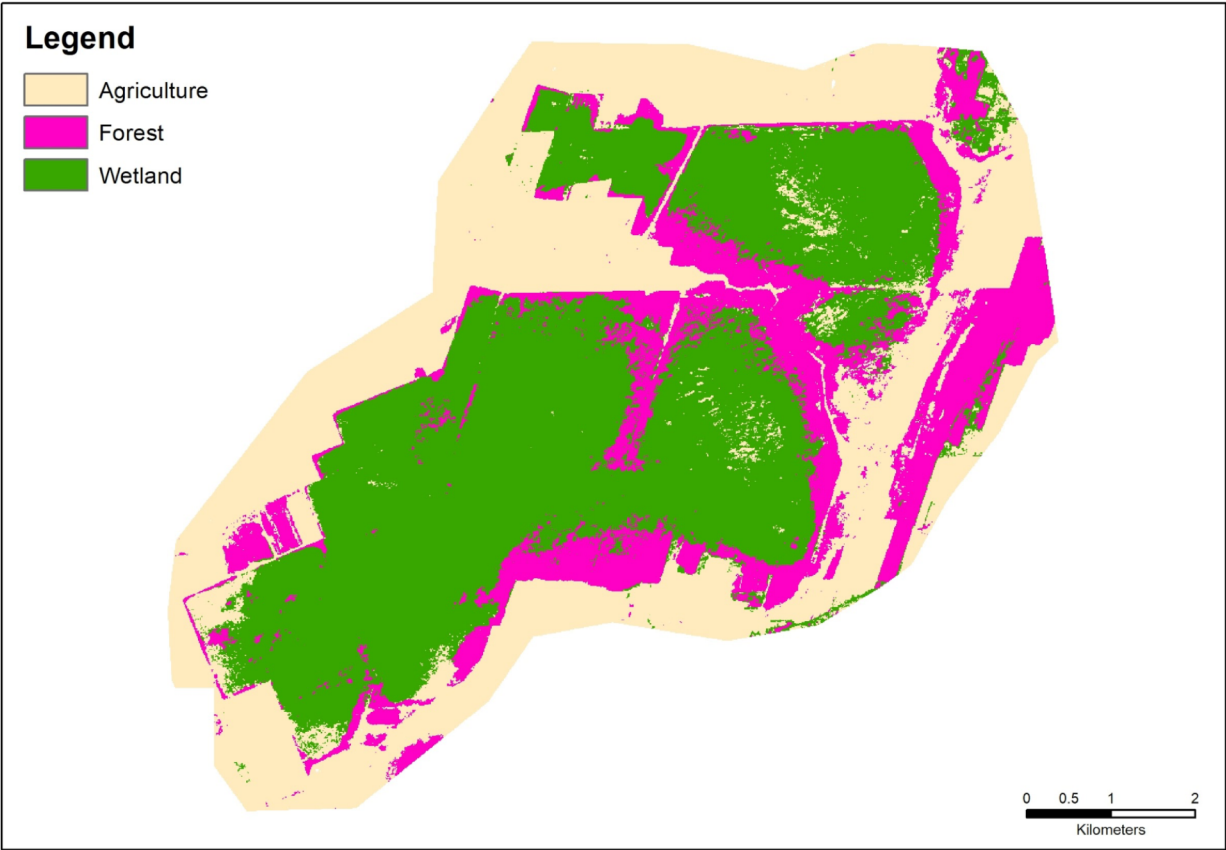
Approaches to Dealing with Imbalanced Data

- There are two common approaches to tackle the problem of extremely imbalanced data
- Cost sensitive learning
 - Assigning a high cost to misclassification of the minority class, and trying to minimize the overall cost
- Use a sampling technique
 - Either down-sampling the majority class or over-sampling the minority class, or both.

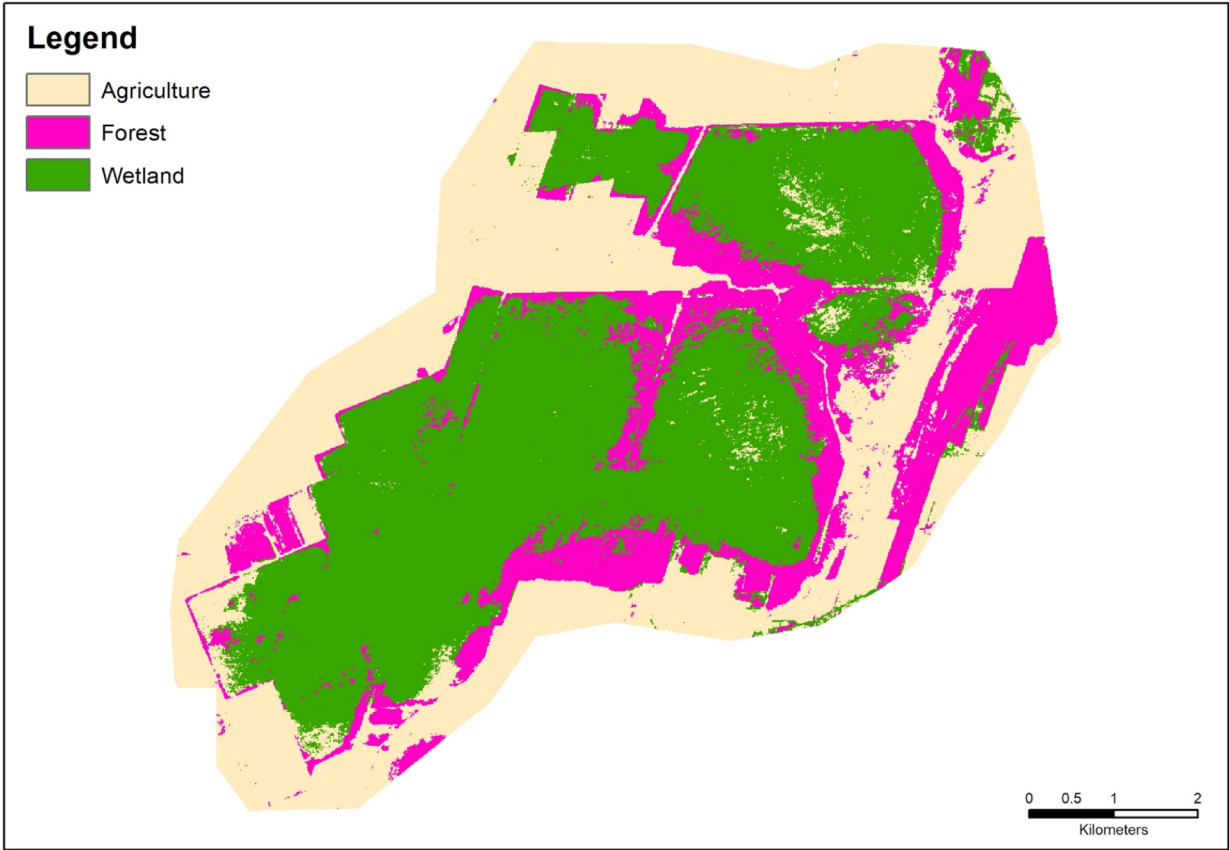
Agriculture Sample Proportion (%)	Overall OOB Error (%)	Agri OOBE rror (%)	Forests OOB Error (%)	Wetland OOB (%)
40	13.5	9	16	16
50	15.3	12	19	17
60	13.5	11	17	15
70	15	5	33	41
80	16	5	25	93
90	10	1	87	100



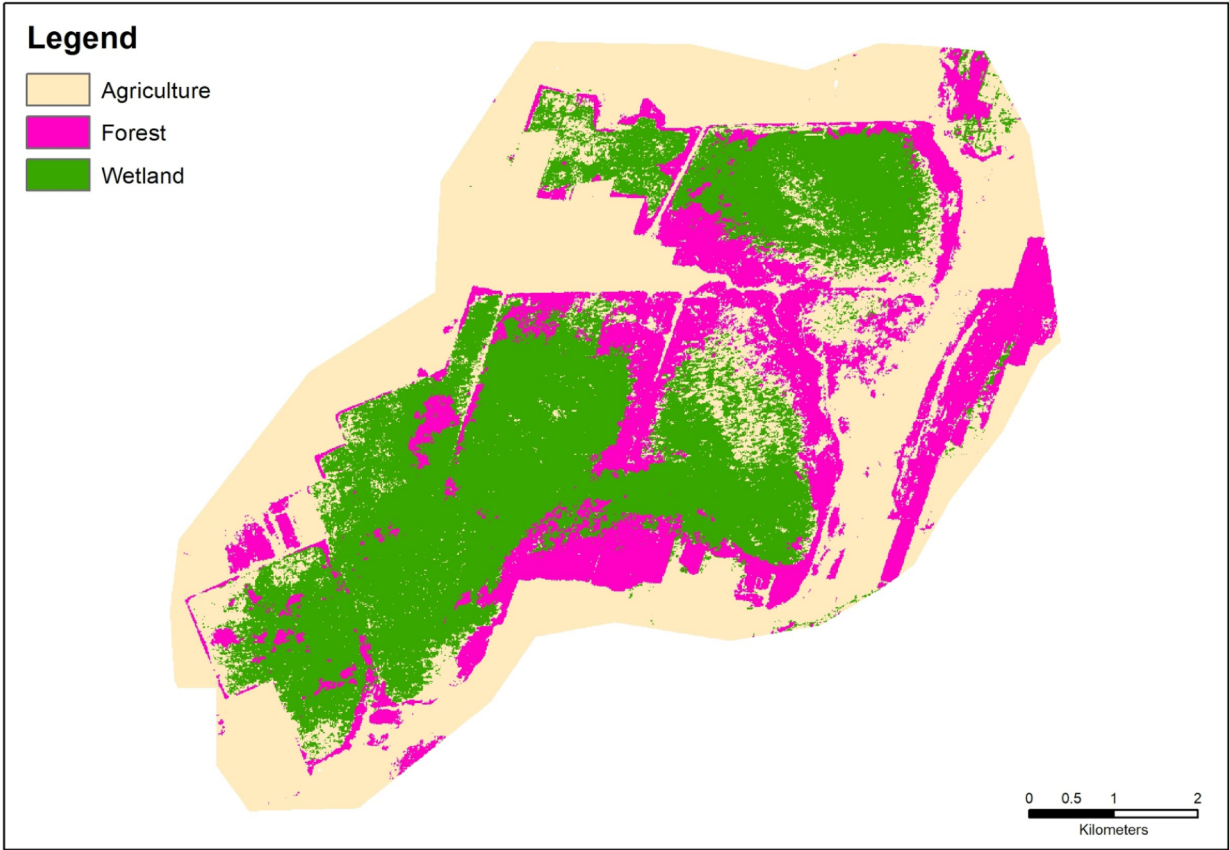
Agriculture Sample Proportion (%)	Overall OOB Error (%)	Agri OOBE rror (%)	Forests OOB Error (%)	Wetland OOB (%)
40	13.5	9	16	16
50	15.3	12	19	17
60	13.5	11	17	15
70	15	5	33	41
80	16	5	25	93
90	10	1	87	100



Agriculture Sample Proportion (%)	Overall OOB Error (%)	Agri OOBE rror (%)	Forests OOB Error (%)	Wetland OOB (%)
40	13.5	9	16	16
50	15.3	12	19	17
60	13.5	11	17	15
70	15	5	33	41
80	16	5	25	93
90	10	1	87	100



Agriculture Sample Proportion (%)	Overall OOB Error (%)	Agri OOBE rror (%)	Forests OOB Error (%)	Wetland OOB (%)
40	13.5	9	16	16
50	15.3	12	19	17
60	13.5	11	17	15
70	15	5	33	41
80	16	5	25	93
90	10	1	87	100



Agriculture Sample Proportion (%)	Overall OOB Error (%)	Agri OOBE rror (%)	Forests OOB Error (%)	Wetland OOB (%)
40	13.5	9	16	16
50	15.3	12	19	17
60	13.5	11	17	15
70	15	5	33	41
80	16	5	25	93
90	10	1	87	100



Agriculture Sample Proportion (%)	Overall OOB Error (%)	Agri OOBE rror (%)	Forests OOB Error (%)	Wetland OOB (%)
40	13.5	9	16	16
50	15.3	12	19	17
60	13.5	11	17	15
70	15	5	33	41
80	16	5	25	93
90	10	1	87	100

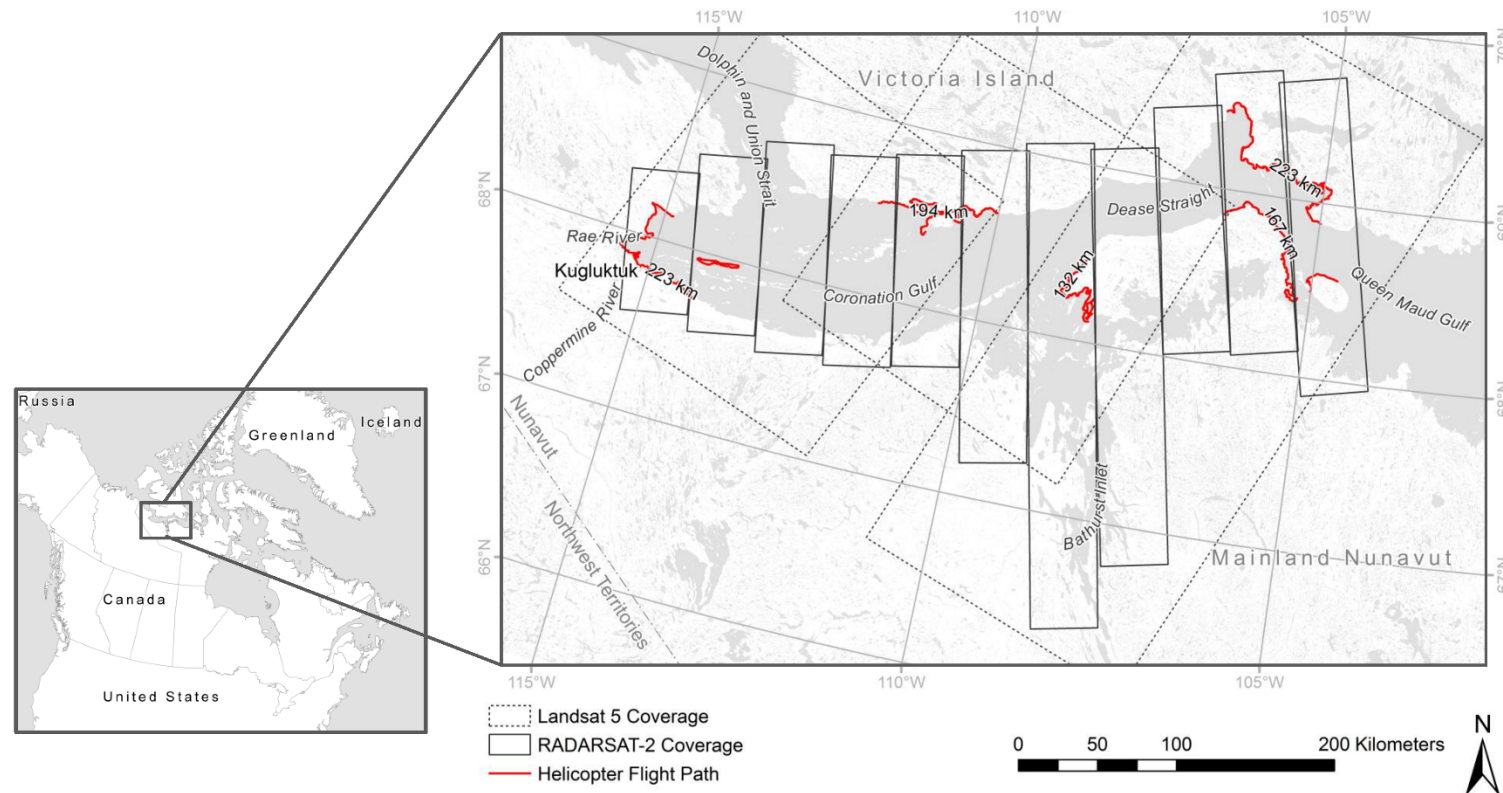


TRAINING DATA:

Class Distribution

Training Points Representative of Class Distribution

- Coronation Gulf
 - Classes: Water, Sand/Mud, Mixed Sediment, Pebble/Cobble/Boulder, Wetland, Tundra
 - RADARSAT-2, Landsat 5



Effects of Well Distributed Training Points on Accuracy and Probabilities

- Accuracies

Number Of Training Points Per-Class	Model Iteration	OOBA (%)	Independent Overall Accuracy (%)	Kappa Statistic	Water		Sand/ Mud		Mixed Sediment		Pebble/ Cobble/ Boulder		Bedrock		Wetland		Tundra	
					UA (%)	PA (%)	UA (%)	PA (%)	UA (%)	PA (%)	UA (%)	PA (%)	UA (%)	PA (%)	UA (%)	PA (%)	UA (%)	PA (%)
13	1	78	88	0.86	100	86	93	88	65	86	92	80	87	92	83	97	95	89
	2	79	87	0.85	100	84	90	87	65	86	92	80	87	92	83	97	95	89
	3	80	88	0.86	100	85	92	87	69	86	92	82	87	92	83	97	95	90
25	1	91	90	0.88	98	86	90	90	87	80	89	94	86	99	81	96	96	86
	2	89	90	0.88	98	86	92	90	87	80	88	94	86	97	81	96	96	87
	3	91	89	0.88	99	86	92	92	87	80	88	94	86	99	78	96	96	84
50	1	89	89	0.87	100	84	94	94	86	87	86	92	86	95	83	86	87	85
	2	89	88	0.86	100	85	94	93	83	87	86	91	86	92	84	85	86	86
	3	88	89	0.87	100	85	94	93	86	88	86	92	87	95	82	87	88	84
100	1	90	91	0.89	100	86	94	92	84	84	84	92	88	94	93	92	90	95
	2	90	91	0.89	100	86	94	91	83	86	87	92	88	94	93	91	89	95
	3	90	90	0.89	100	86	94	92	84	84	84	92	88	92	92	90	89	95
167	1	90	91	0.90	98	88	96	91	88	87	86	93	89	96	94	91	89	95
	2	90	92	0.90	98	88	96	92	88	88	86	93	89	95	94	91	90	95
	3	90	91	0.90	98	88	96	93	88	85	84	93	89	95	93	91	89	94

UA = User's Accuracy, PA = Producer's Accuracy

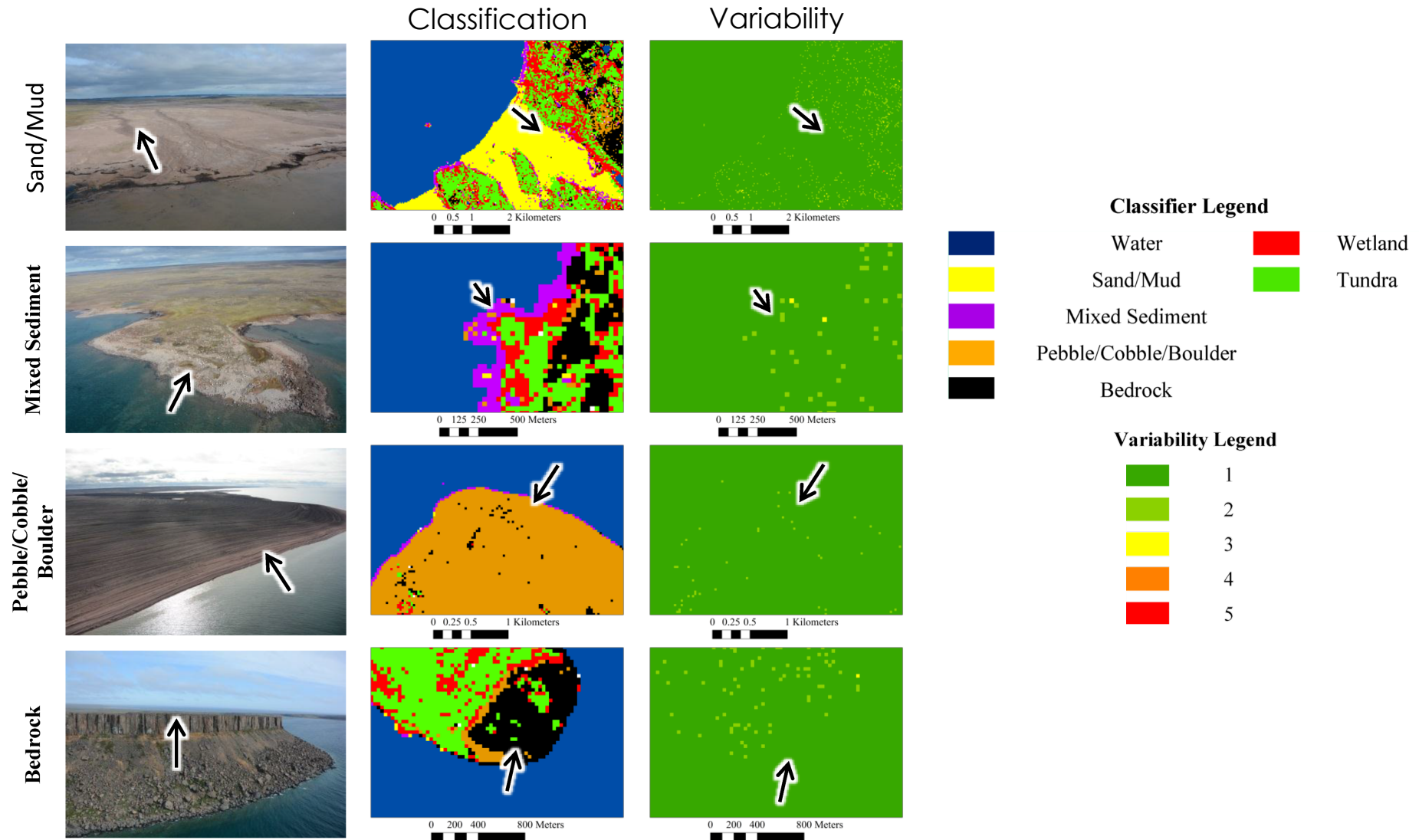
Effects of Well Distributed Training Points on Accuracy and Probabilities

- Probabilities (number voting with majority/total number of trees)

Number Of Training Points Per-Class	Model Iteration	Water	Sand/ Mud	Mixed Sediment	Pebble/ Cobble/ Boulder	Bedrock	Wetland	Tundra
13	1	0.92	0.67	0.48	0.67	0.61	0.58	0.61
	2	0.92	0.70	0.49	0.67	0.60	0.58	0.61
	3	0.92	0.68	0.49	0.66	0.61	0.58	0.61
25	1	0.96	0.72	0.57	0.78	0.66	0.71	0.70
	2	0.96	0.71	0.58	0.79	0.64	0.71	0.70
	3	0.95	0.71	0.58	0.79	0.65	0.71	0.69
50	1	0.93	0.79	0.60	0.86	0.72	0.79	0.78
	2	0.93	0.80	0.61	0.86	0.72	0.79	0.78
	3	0.93	0.79	0.60	0.86	0.72	0.79	0.78
100	1	0.97	0.86	0.68	0.88	0.82	0.78	0.79
	2	0.97	0.85	0.68	0.87	0.82	0.79	0.80
	3	0.97	0.85	0.68	0.88	0.82	0.79	0.80
167	1	0.97	0.86	0.70	0.89	0.84	0.79	0.82
	2	0.98	0.86	0.69	0.88	0.84	0.80	0.82
	3	0.97	0.86	0.69	0.89	0.84	0.80	0.83

Effects of Well Distributed Training Points on Accuracy and Probabilities

- Variability between models



PARAMETERS:

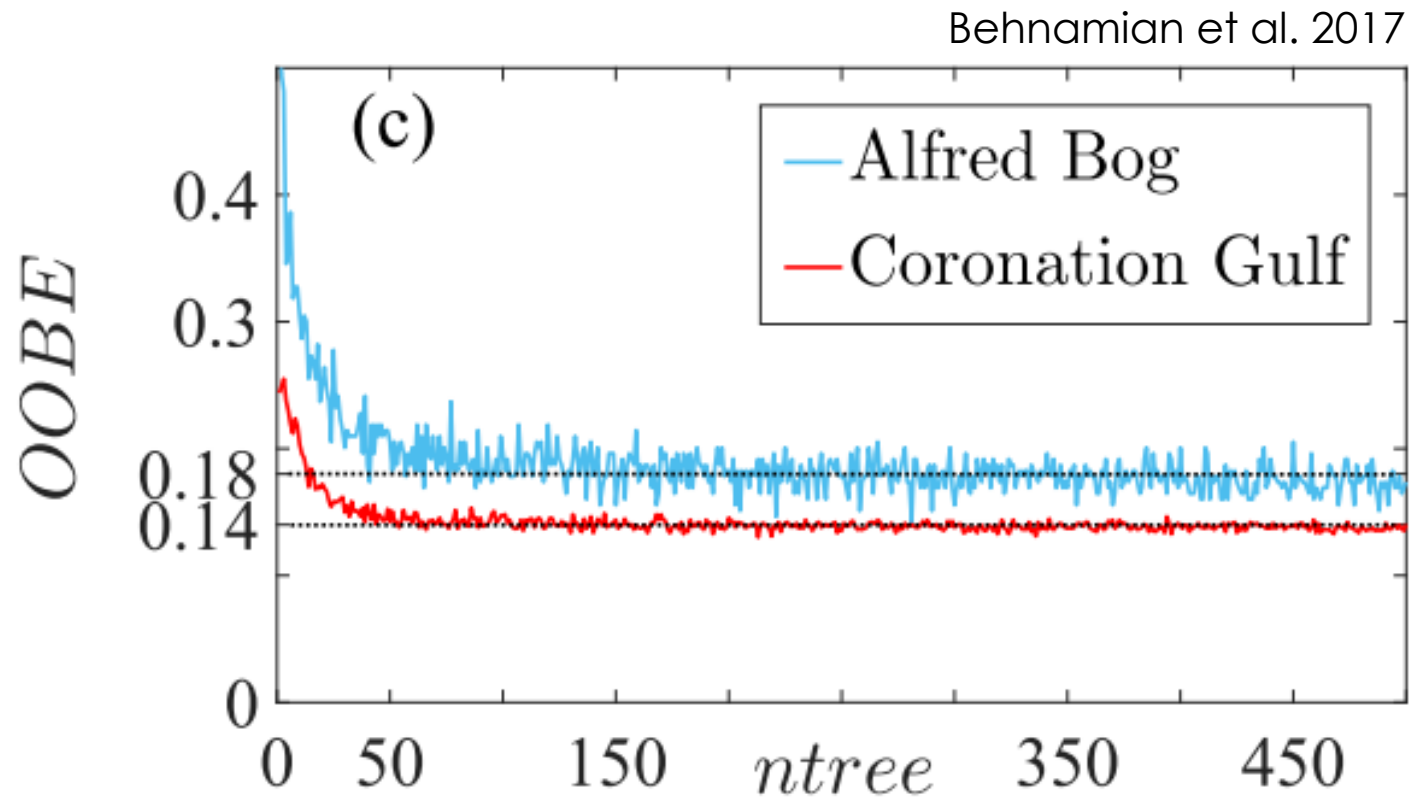
Ensemble Averaging

Achieving Stable Ensemble Averages

- Because of the random way variables and training data are selected the following can vary between runs:
 - Final prediction
 - Probabilities
 - Variable importance ranking
- Solution:
 - Average multiple runs
 - Increase number of trees

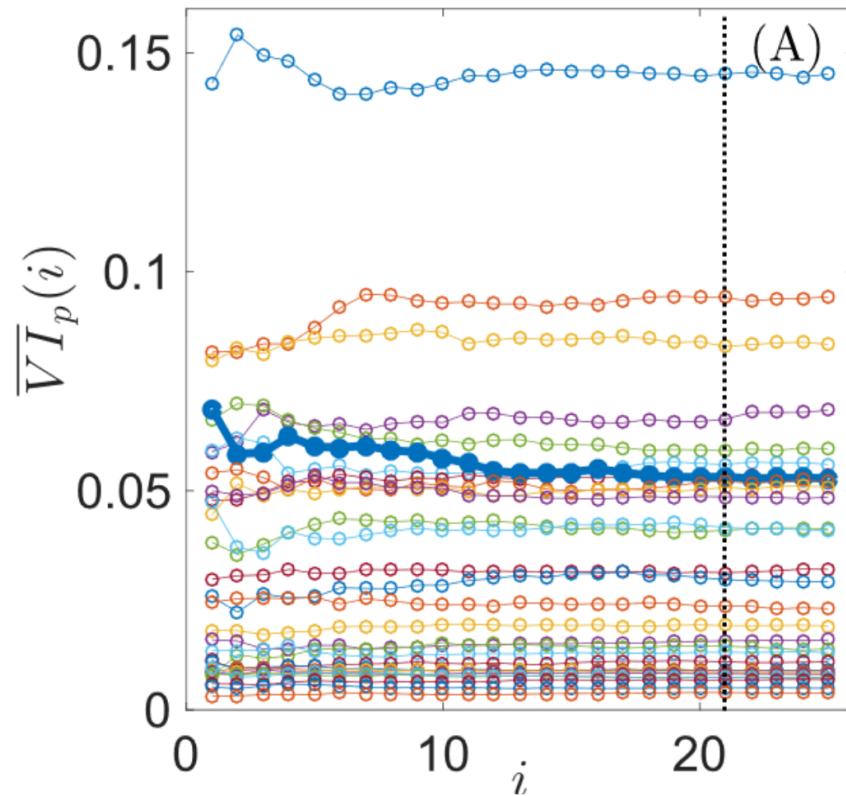
Minimum Ntree to Maximize Accuracies

- Alfred (dataset in this tutorial – low separability)
- Coronation Gulf (Banks et al. 2015– high separability)



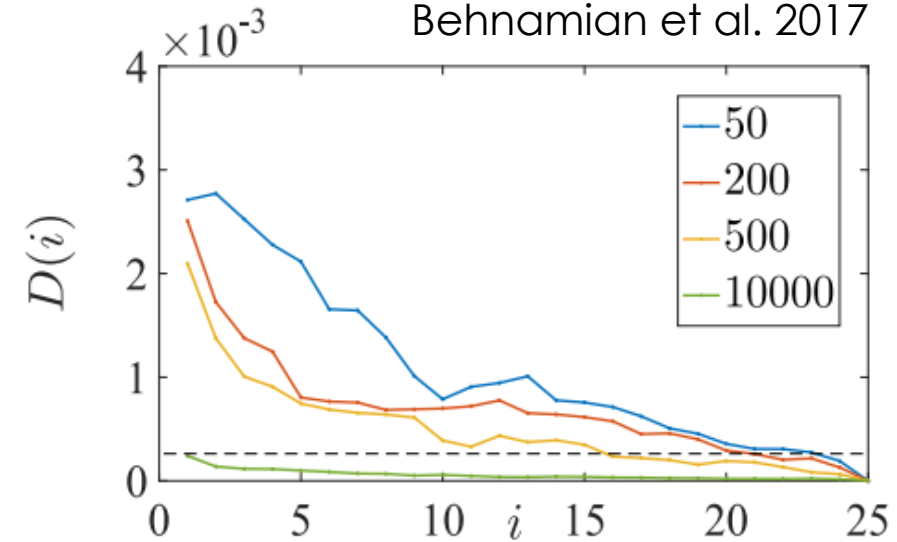
Stability of Importance Ranking and Probability Values

MDA, ntree=10,000



i = number of runs
 $\overline{VI}_p(i)$ = average of i runs

Behnamian et al. 2017



$$D(i) = \left(\frac{\sum_{p=1}^P (\overline{VI}_p(i) - \overline{VI}_p(25))^2}{P} \right)^{0.5}$$

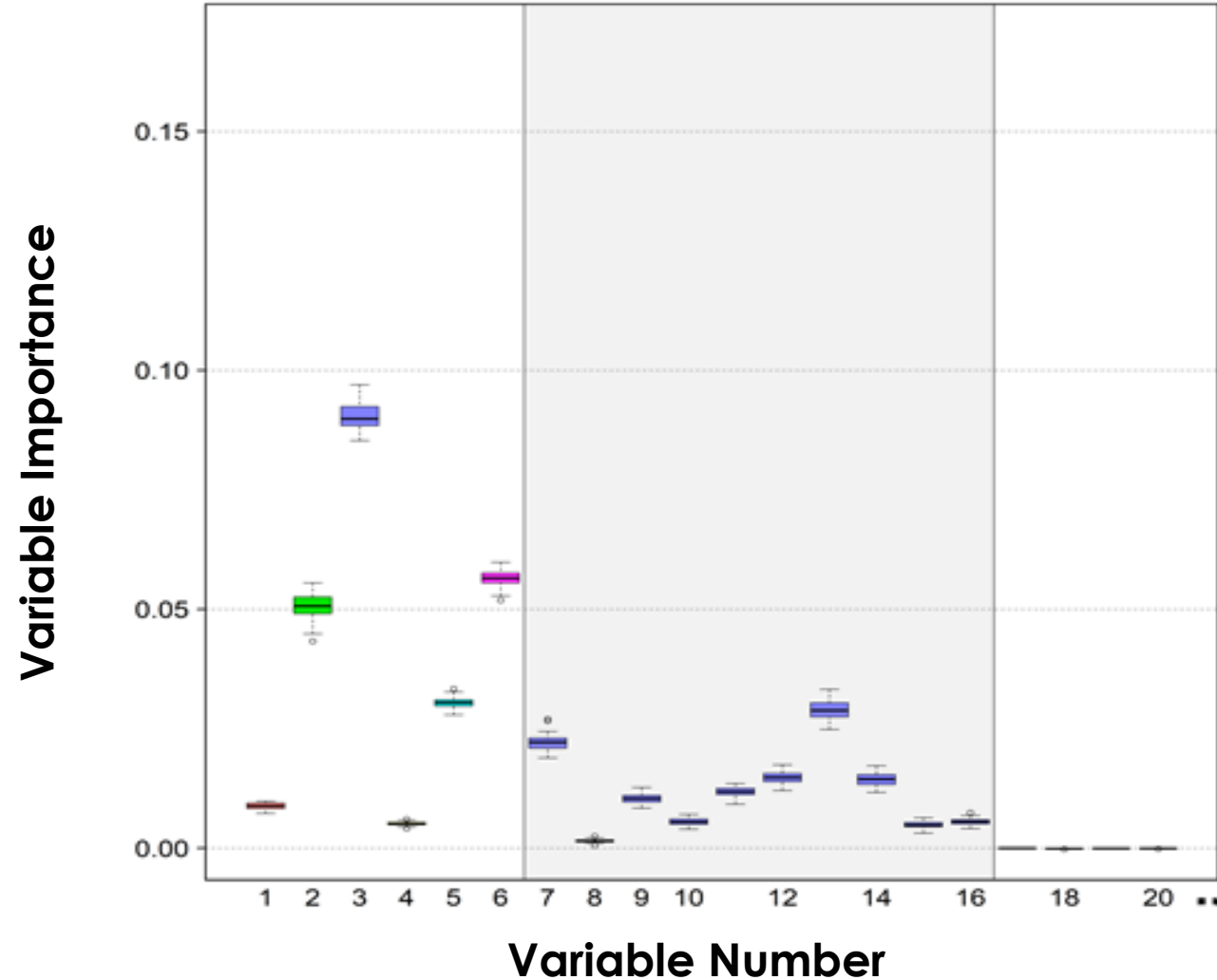
INPUTS:
Correlated Variables

Correlated Variables

- In the presence of correlated variables, Random Forests variable importance ranking measures are biased:
 - Correlated variables can compensate for information loss - random permutation
 - Variables without surrogates can have inflated importance values
- Solution:
 - Do not rely on importance rankings
 - Measures of separability may give better indication of importance

Correlated Variables: Effect on Importance

Simulated Toy Data

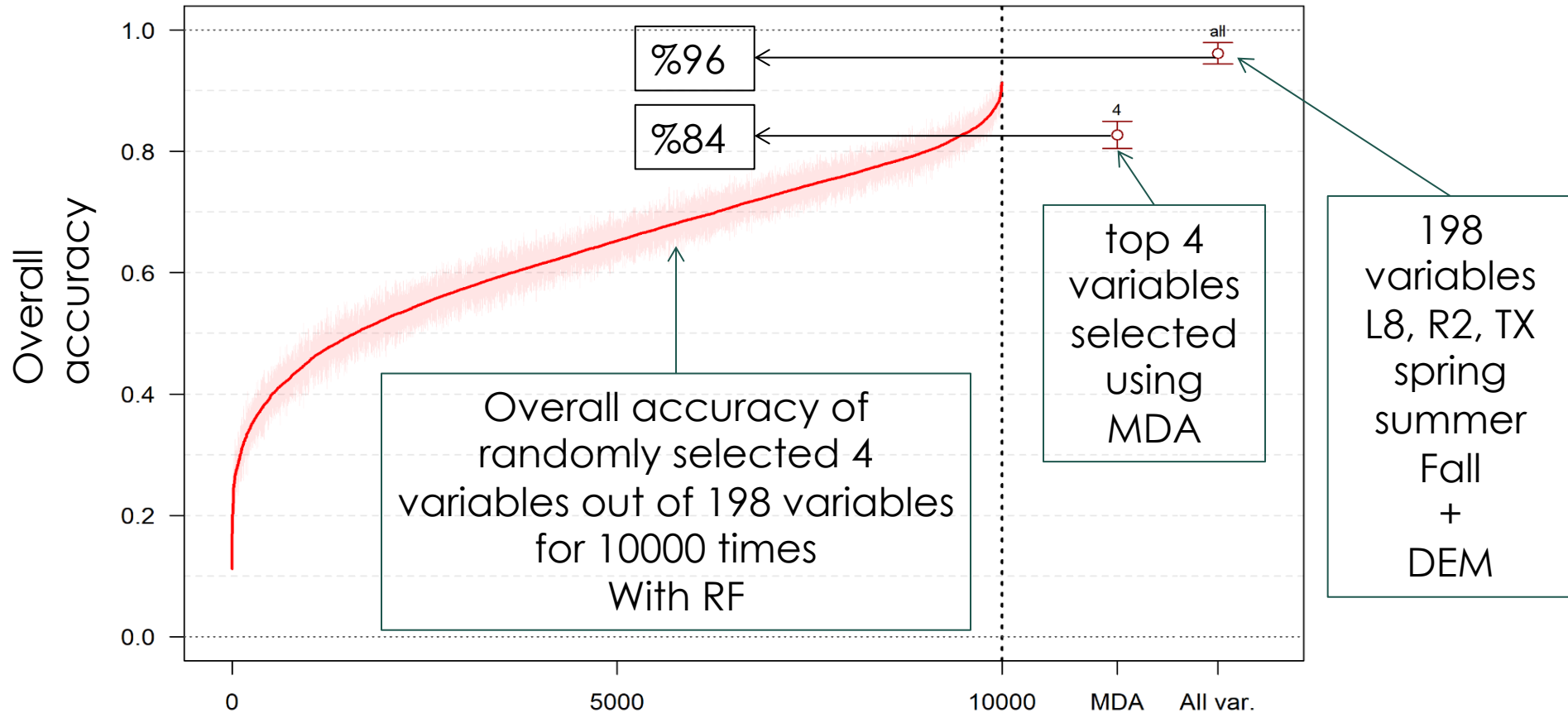


Reducing Correlated Variables: Problems

- Pearson's r can not identify non-linear relationships
- Problem with principal components analysis:
 - Principal components are linear combinations of the data, so they may not contain much useful information in cases where non-linear relationships exist
 - Low-variance information may be useful for separating specific classes
 - Not all observed variance is important for classification purposes
 - Consider the entire scene, so rare classes that contribute few pixels to the total population, may not contribute significantly to any of the top principal components.
- Spearman's ρ can identify non-linear relationships but hard to identify per class relationships

Example: Selecting Optimum Set of Variables

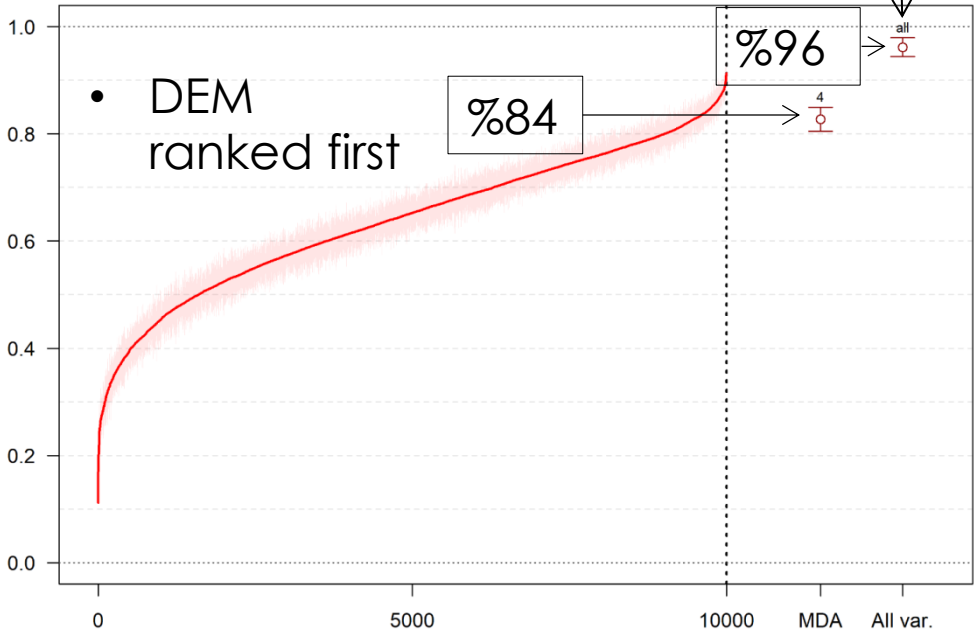
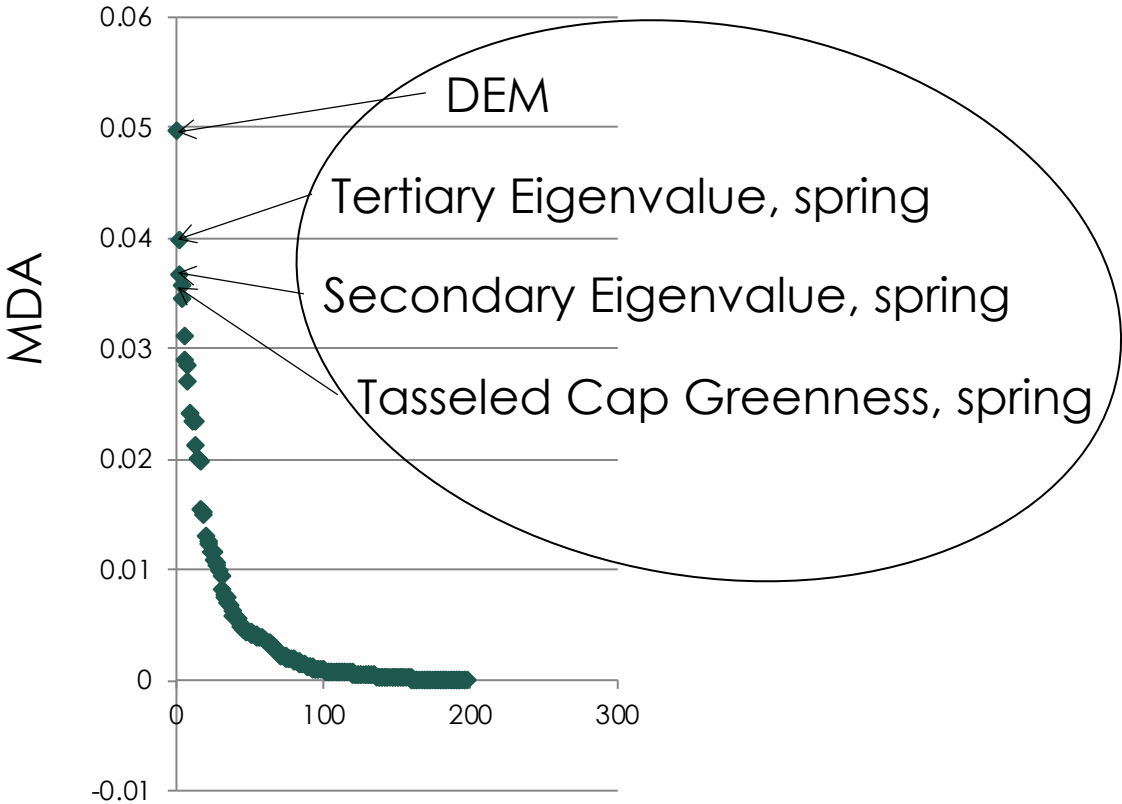
- Objective: least number of variables + maximum classification accuracy



Example: Selecting Optimum Set of Variables

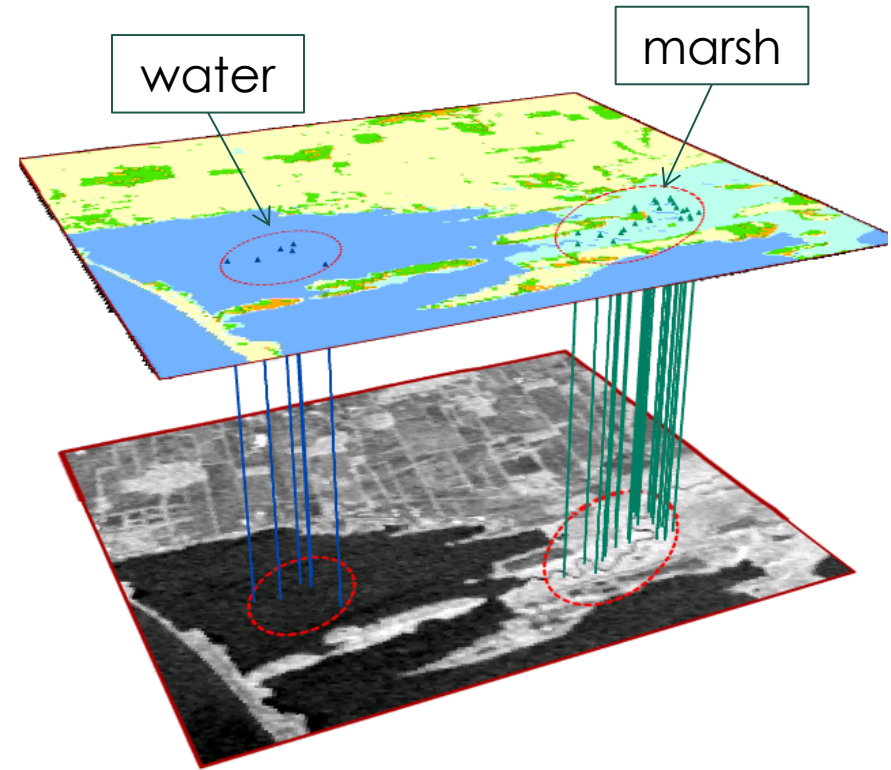
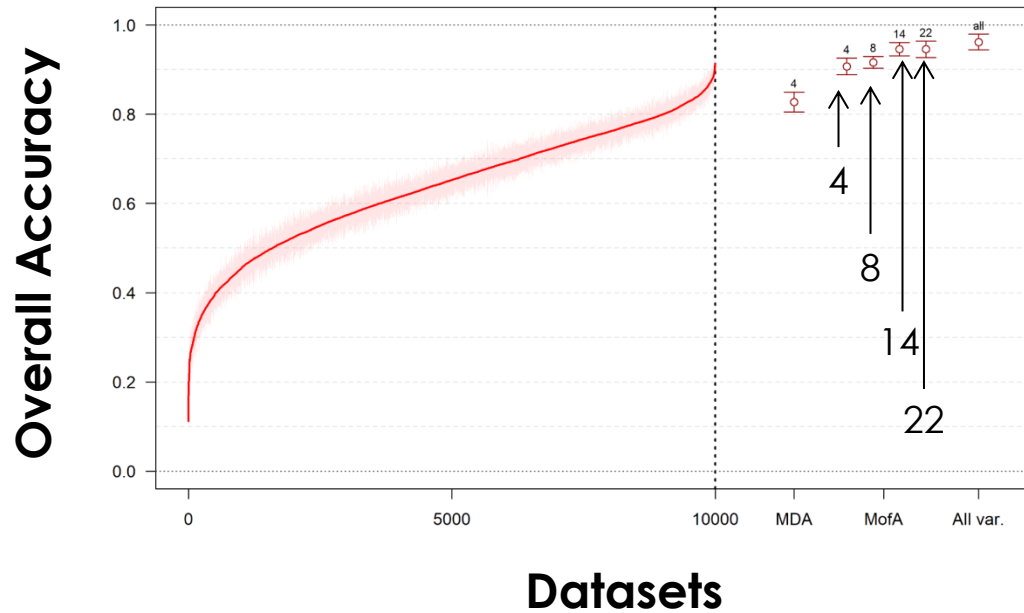
L8, R2, TX
Spring, Summer, Fall
198 variables
RF importance ranking

L8 - Band 1 Coastal Aerosol	FD Double Bounce	Secondary Psi Angle
L8 - Band 2 Blue	FD Volume	Secondary Eigenvalue
L8 - Band 3 Green	FD Rough	Secondary Touzi Alpha S Parameter
L8 - Band 4 Red	Total Power	Secondary Touzi Phase
L8 - Band 5 NIR	Intensity Ratio HH/VV	Secondary Touzi Angle (Helicity)
L8 - Band 6 SWIR	Intensity Ratio HV/HH	Tertiary Psi Angle
L8 - Band 7 SWIR	Entropy	Tertiary Eigenvalue
NDVI (normalized difference vegetation index)	Anisotropy	Tertiary Touzi Alpha S Parameter
SAVI (soil adjusted vegetation index)	Alpha Angle	Tertiary Touzi Phase
Tasseled Cap Brightness	Beta Angle	Tertiary Tau Angle (Helicity)
Tasseled Cap Greenness	Maximum Polarization Response	Real Component of Element 1,1 of Covariance matrix
Tasseled Cap Wetness	Minimum Polarization Response	Real Component of Element 2,2 of Covariance matrix
HH Intensity	Touzi Anisotropy	Real Component of Element 3,3 of Covariance matrix
VV Intensity	Difference between min & max response	Shannon Entropy Intensity
Total Power	Pedestal Height	Shannon Entropy polarimetry
Loss of Polarization During the Scattering Process	Phase Difference HH-VV	OMNRF 2 M DEM
Difference of the HH and VV Intensities	HH, VV: magnitude of the correlation coefficient	OMNRF 2 M ASPECT
Imaginary part of HH and VV	HH, VV: Phase of the correlation coefficient	OMNRF 2 M SLOPE
Double Bounce Scattering from Two Component Decomposition	HH, VV: Real component of the correlation coefficient	
Surface Scattering from Two Component Decomposition	HH, VV: Imaginary component of the correlation coefficient	
	Dominant Psi Angle	
	Dominant Eigenvalue	
	Dominant Touzi Alpha S Parameter	
	Dominant Touzi Phase	
	Dominant Tau Angle (Helicity)	



Example: Selecting Optimum Set of Variables

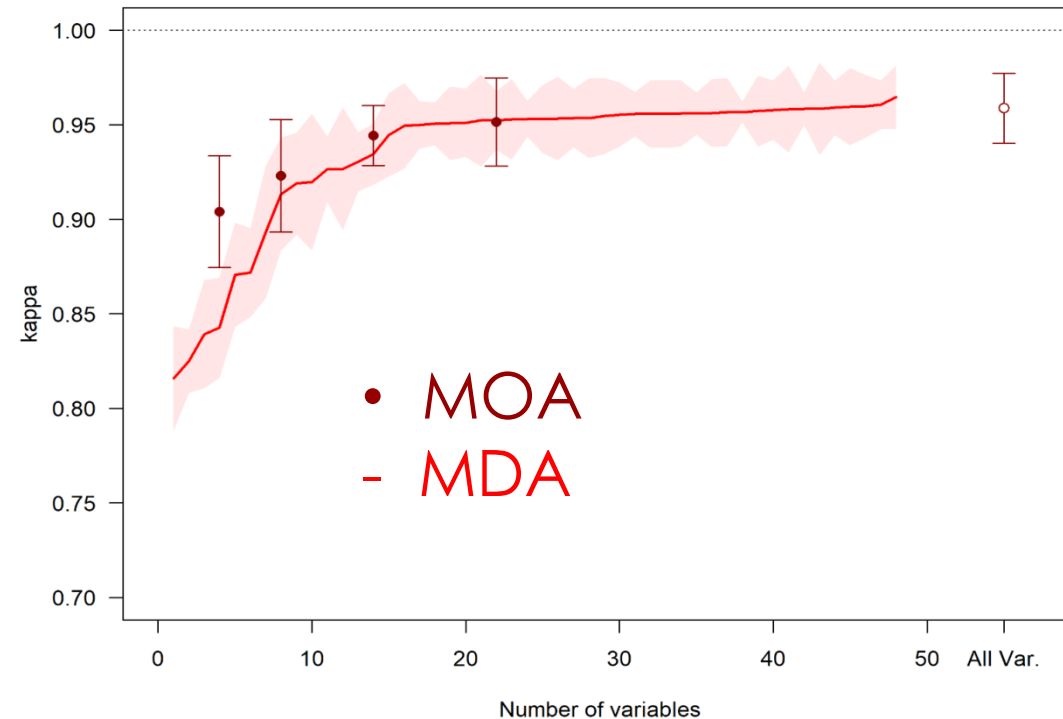
Measure of Association



$$MOA = \frac{M_w - M_m}{\sigma_{tot}} \sqrt{\frac{n_w n_m}{n_{tot}}}$$

MDA vs MOA

MOA	MDA
1 - Tertiary Eigenvalue, 2 - Shannon Entropy, 3 - SAVI, 4 - L8 band 6 SWIR,	1 - DEM 2 - Tertiary Eigenvalue, 3 - Secondary Eigenvalue, 4 - Tasseled Cap Greenness,
spring spring spring fall	spring spring spring



SUMMARY

- Land use classification is used to make maps by grouping pixels into user-defined classes
 - Generally, for all classifiers, you should select your training data carefully
 - Test for spatial autocorrelation
 - Ensure balanced data
- Random Forests is a robust classifier that produces high accuracy results
 - There are only two parameters to set in creating a random forest model
 - *ntree* and *mtry*
 - The appropriate *ntree* should be chosen with care
- Random Forests works well with high dimensional data
 - But do not rely on importance ranking (e.g. MDA) in the presence of correlated variables

REFERENCES

- 1) Breiman, Leo. "Random forests." Machine learning 45.1 (2001): 5-32.
- 2) Genuer, Robin, Jean-Michel Poggi, and Christine Tuleau-Malot. "Variable selection using random forests." Pattern Recognition Letters 31.14 (2010): 2225-2236.
- 3) Liaw, Andy, and Matthew Wiener. "Classification and regression by randomForest." R news 2.3 (2002): 18-22.
- 4) Behnamian, Amir, et al. "A systematic approach for variable selection with random forests: achieving stable variable importance values." IEEE Geoscience and Remote Sensing Letters 14.11 (2017): 1988-1992.
- 5) Banks, Sarah, et al. "Assessing the potential to operationalize shoreline sensitivity mapping: Classifying multiple Wide Fine Quadrature Polarized RADARSAT-2 and Landsat 5 scenes with a single Random Forest model." Remote Sensing 7.10 (2015): 13528-13563.
- 6) Banks, Sarah, et al. "Contributions of Actual and Simulated Satellite SAR Data for Substrate Type Differentiation and Shoreline Mapping in the Canadian Arctic." Remote Sensing 9.12 (2017): 1206.
- 7) Banks, Sarah, et al. "Wetland Classification with Multi-Angle/Temporal SAR Using Random Forests." Remote Sensing 11.6 (2019): 670.
- 8) White, Lori, et al. "Moving to the RADARSAT constellation mission: Comparing synthesized compact polarimetry and dual polarimetry data with fully polarimetric RADARSAT-2 data for image classification of peatlands." Remote Sensing 9.6 (2017): 573.
- 9) Millard, Koreen, and Murray Richardson. "On the importance of training data sample selection in random forest image classification: A case study in peatland ecosystem mapping." Remote sensing 7.7 (2015): 8489-8515.
- 10) Millard, Koreen, and Murray Richardson. "Wetland mapping with LiDAR derivatives, SAR polarimetric decompositions, and LiDAR-SAR fusion using a random forest classifier." Canadian Journal of Remote Sensing 39.4 (2013): 290-307.
- 11) Planet Team (2017). Planet Application Program Interface: In Space for Life on Earth. San Francisco, CA. <https://api.planet.com>
- 12) RADARSAT-2 Data and Products © Maxar Technologies Ltd. (2018) – All Rights Reserved. RADARSAT is an official mark of the Canadian Space Agency.