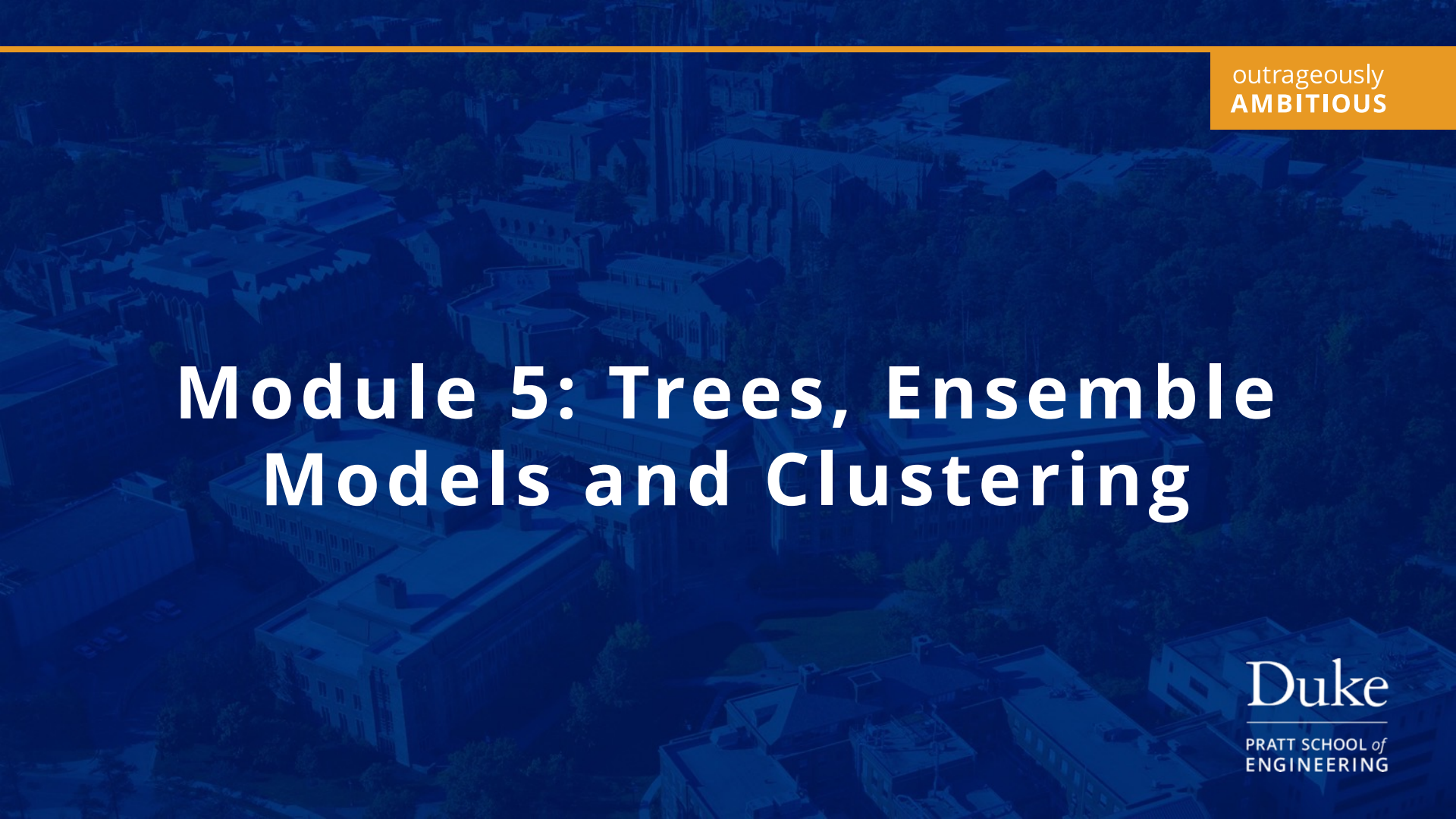
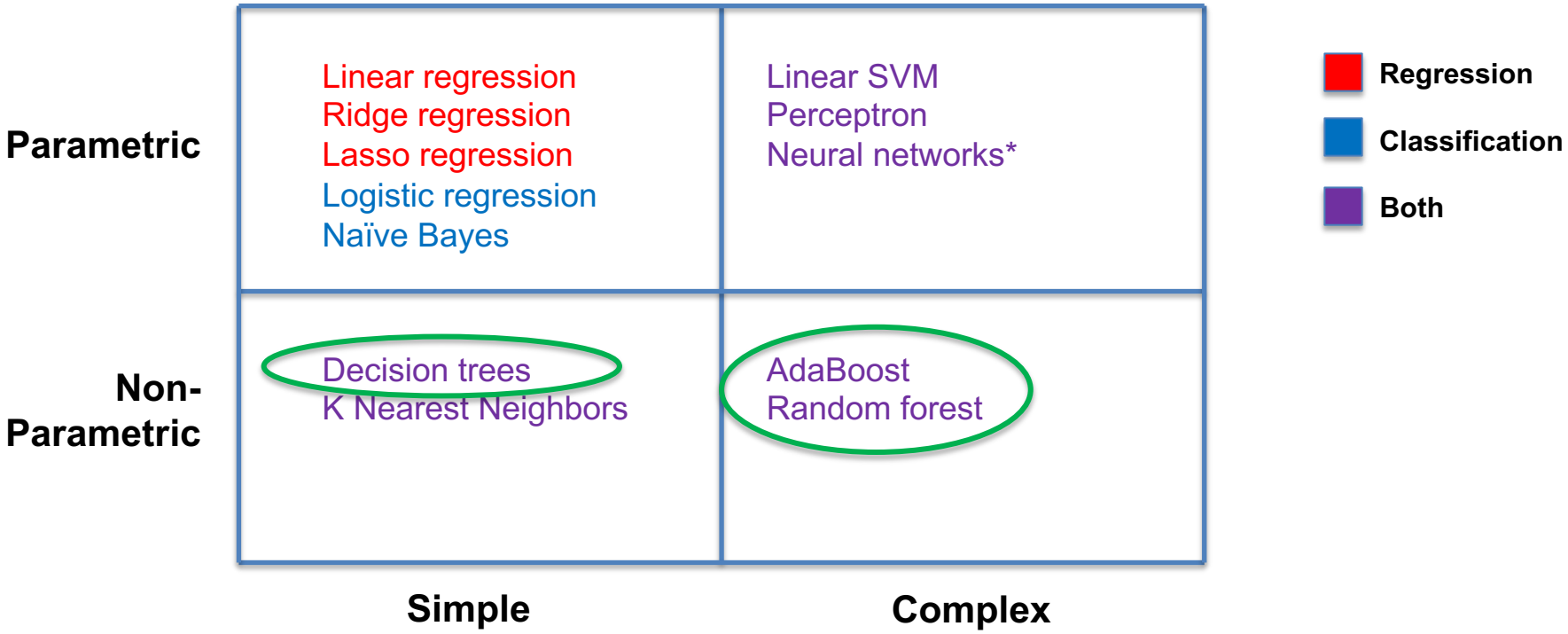

outrageously
AMBITIOUS
Module 5: Trees, Ensemble Models and Clustering
Duke
PRATT SCHOOL of ENGINEERING

Supervised Learning Algorithms



Module 5 Objectives:

At the conclusion of this module, you should be able to:

1) Describe how tree-based models differ from linear models
2) Identify the advantages of ensemble models and how they are assembled
3) Explain what clustering is and how K-Means clustering works


outrageously
AMBITIOUS
Tree Models
Duke
PRATT SCHOOL of
ENGINEERING

Decision Trees

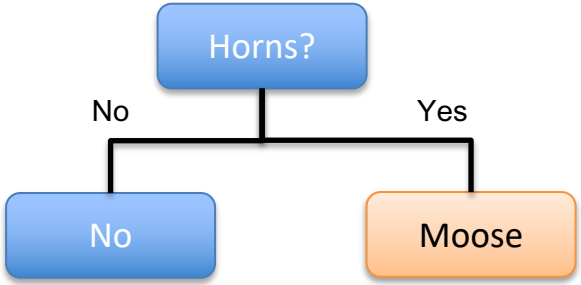
Ask a series of questions to narrow in on the label

	Horns	Color	# Legs	Label
Animal 1	No	Brown	4	Dog
Animal 2	No	Green	4	Lizard
Animal 3	No	Black	2	Bird
Animal 4	Yes	Brown	4	Moose

Decision Trees

Ask a series of questions to narrow in on the label

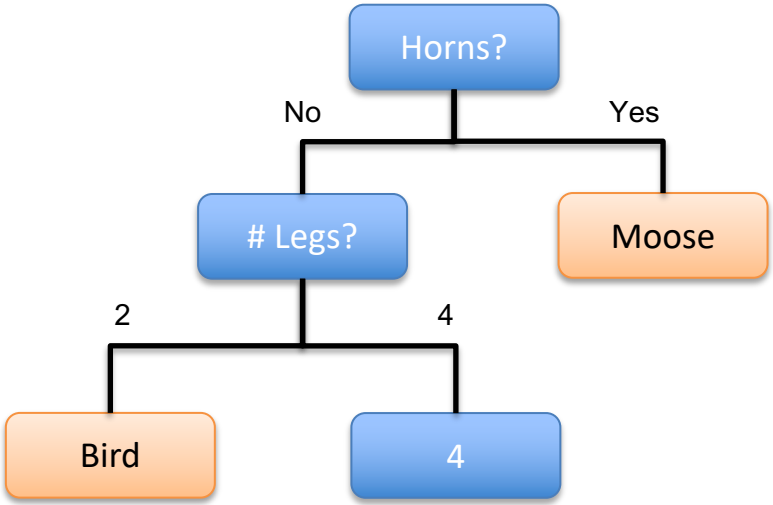
	Horns	Color	# Legs	Label
Animal 1	No	Brown	4	Dog
Animal 2	No	Green	4	Lizard
Animal 3	No	Black	2	Bird
Animal 4	Yes	Brown	4	Moose



Decision Trees

Ask a series of questions to narrow in on the label

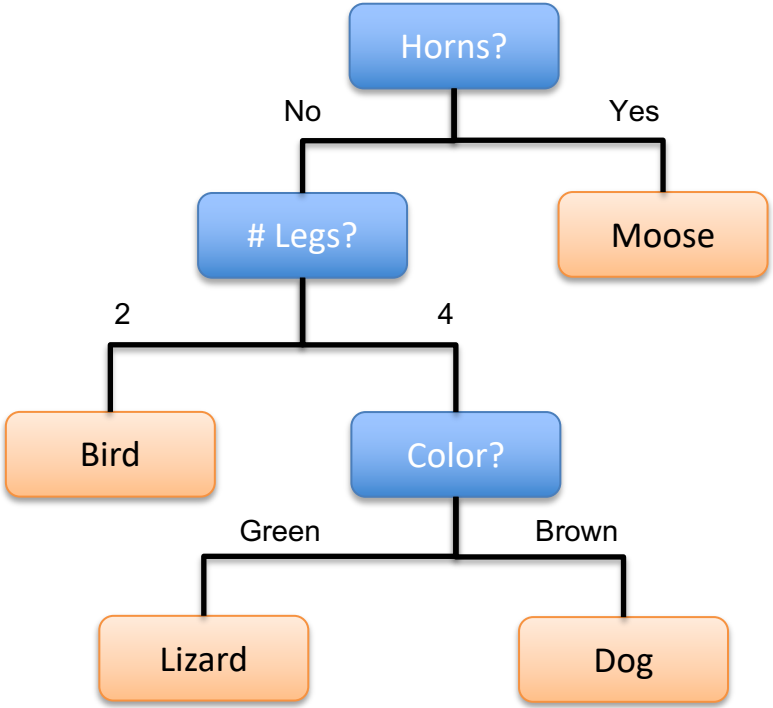
	Horns	Color	# Legs	Label
Animal 1	No	Brown	4	Dog
Animal 2	No	Green	4	Lizard
Animal 3	No	Black	2	Bird
Animal 4	Yes	Brown	4	Moose



Decision Trees

Ask a series of questions to narrow in on the label

	Horns	Color	# Legs	Label
Animal 1	No	Brown	4	Dog
Animal 2	No	Green	4	Lizard
Animal 3	No	Black	2	Bird
Animal 4	Yes	Brown	4	Moose



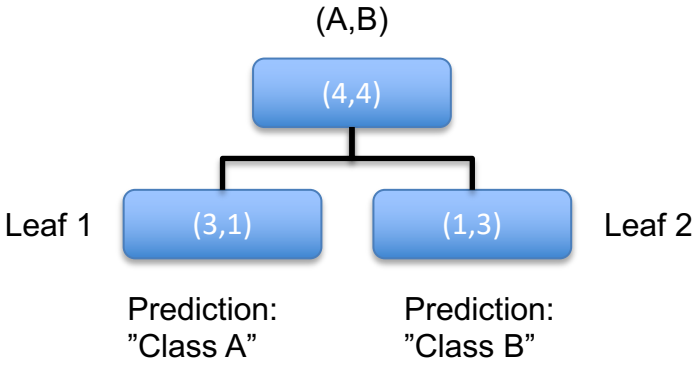
How to choose the splits

- Goal is to build the most efficient tree – the one that splits the data using the minimum number of splits
- We define an objective function to optimize via the decision tree
- Our objective function is to maximize the **Information Gain (IG)** at each split:

$$\mathrm{IG} = Decrease\ in\ impurity$$
$$= Impurity(Parent) - Impurity(Children)$$

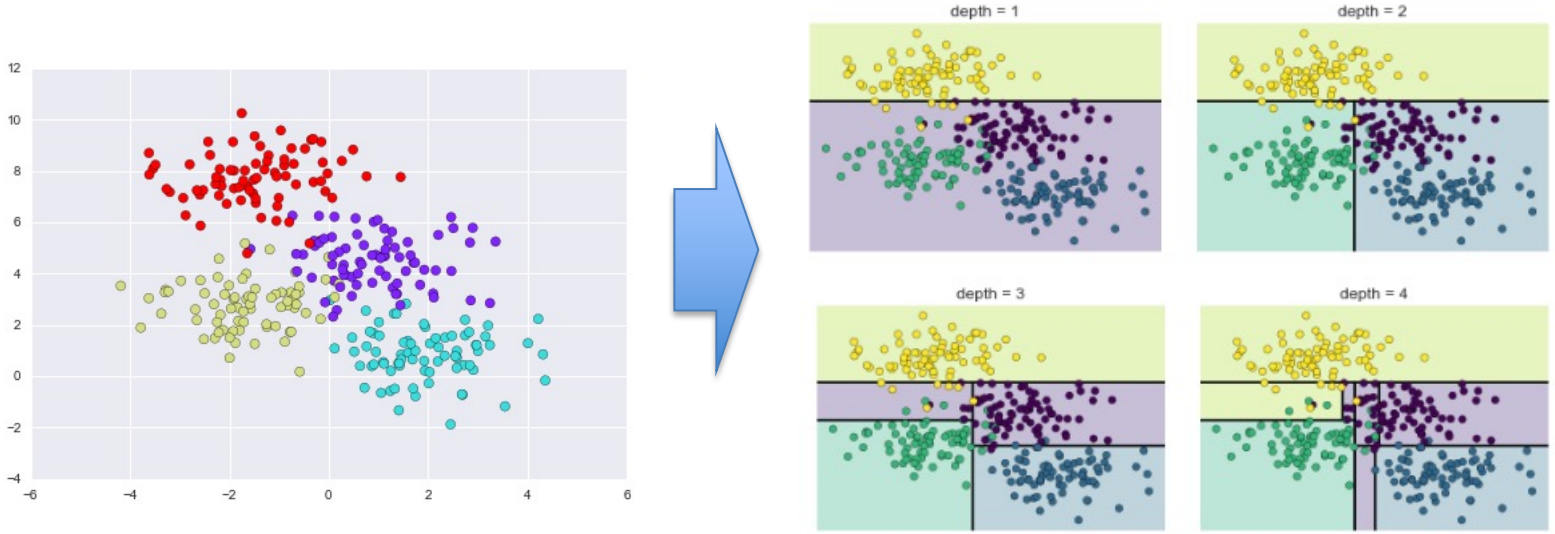
Getting predictions from the tree

- We continue splitting the tree until we cannot split any further or we stop
- The bottom nodes are called “leaves”
- We take the majority average class and label all examples at the leaf with that class



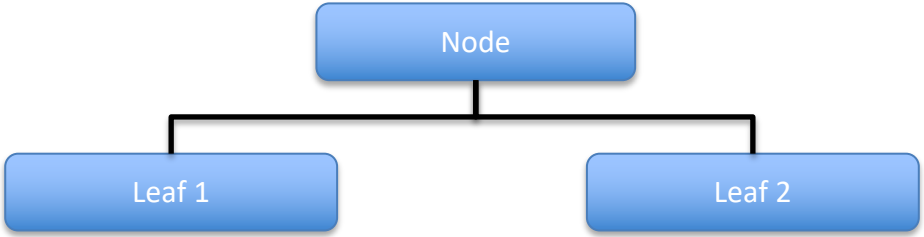
Tree depth

- Depth of our tree (max # of splits of a branch) is a hyperparameter
- Very shallow trees will underfit the data
- Deep trees will overfit the data (every example ends up as own leaf)



Regression trees

Rather than taking the majority vote of samples in a leaf, we calculate the mean target value of the samples



4 samples
Target values: [5,9,6,8]

Prediction for leaf:
(5+9+6+8)/4 = 7

3 samples
Target values: [4,2,1]

Prediction for leaf:
(4+2+3)/3 = 3

Benefits & challenges of trees

Benefits

- Highly interpretable
- Train quickly
- Handle non-linear relationships well
- Do not require scaling or one-hot encoding variables

Benefits & challenges of trees

Challenges

- Highly sensitive to hyperparameters (tree depth)
- Very prone to overfitting

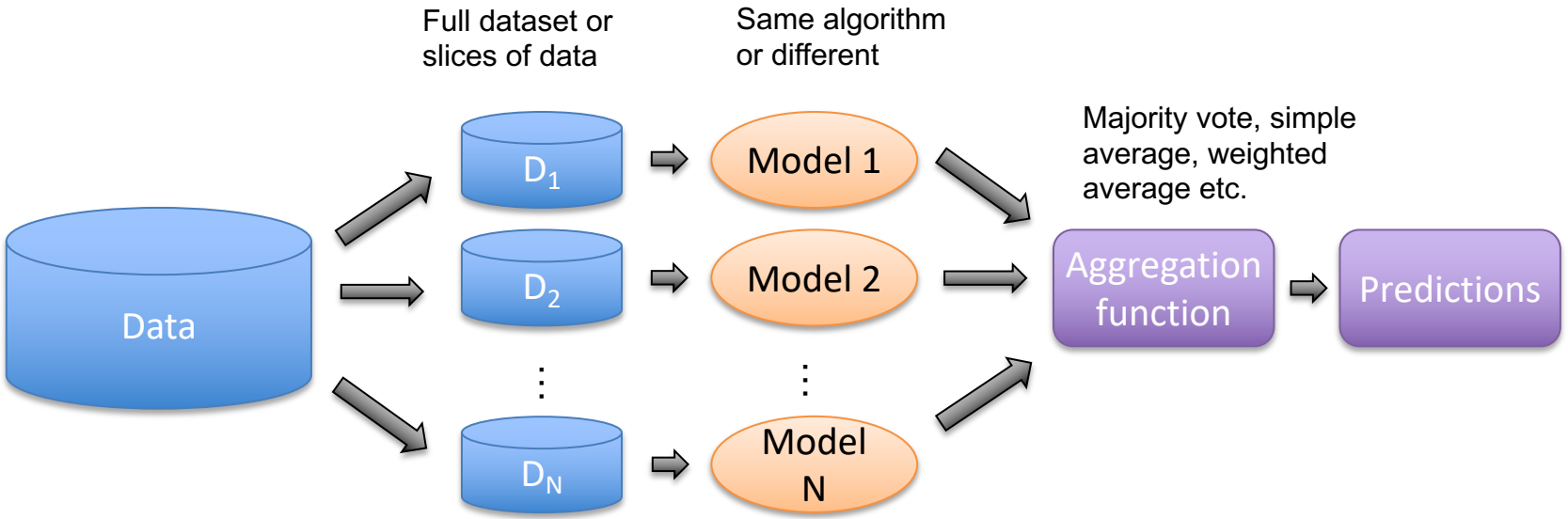

outrageously
AMBITIOUS
Ensemble Models
Duke
PRATT SCHOOL of
ENGINEERING

Ensemble models

- Goal is to combine multiple models together into a meta-model that has better generalization performance
- Averaging multiple models makes the aggregate model less likely to overfit and better at generalizing to new data
  - If model outputs are independent (or close), the variance of the average prediction is lower than the variance of the individual model predictions

Ensemble models

Goal of ensembling is to combine multiple models together into a meta-model that has better generalization performance



Challenges of ensemble models

- Time and compute resources to train
- Computational cost of running multiple models
- Decrease in interpretability

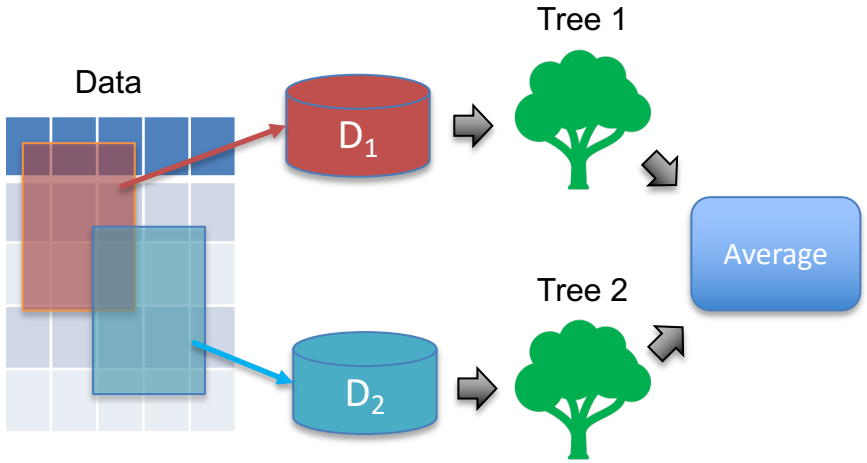

outrageously
AMBITIOUS
Bagging and the Random Forest
Duke
PRATT SCHOOL of ENGINEERING

Bootstrap aggregating (bagging)

- In bagging, we use **bootstrapped** samples to train each model
  - Bootstrapping = sampling with replacement
  - We select the size of each bagging subset
- Because each model is trained on different data, their output predictions can be considered close to independent
- Thus, when we use the average of their predictions we reduce the variance

From tree to random forest

- Decision trees tend to overfit
- We can grow several trees and take the majority vote
- We use bagging to ensure each tree is trained on a different subset
- To predict on new data, we take the majority vote / simple average



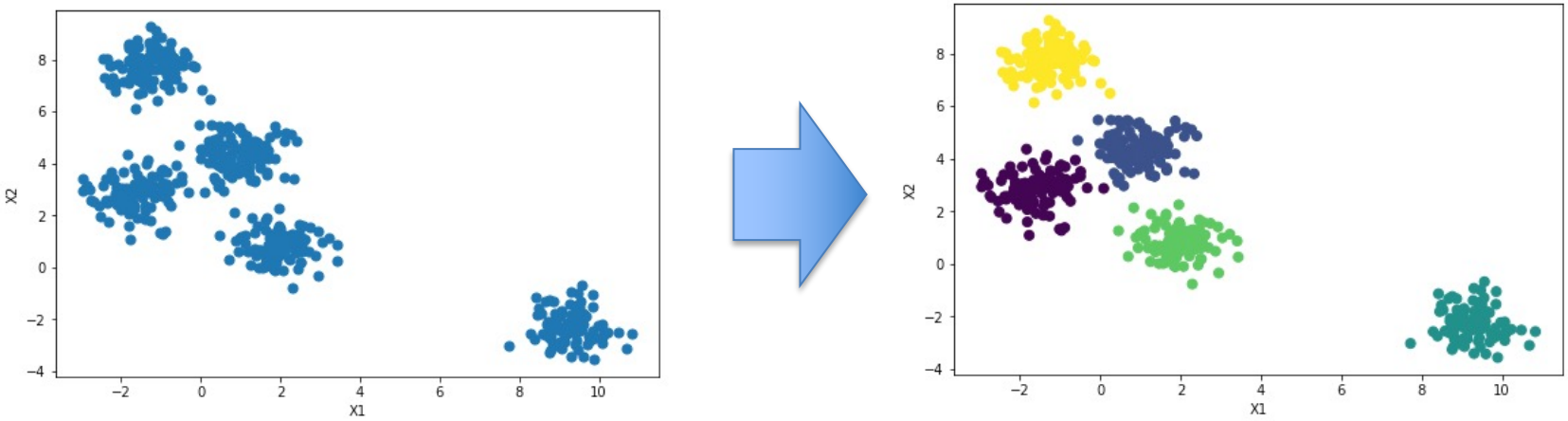
Random Forest design

1. **Number of trees** in the forest
2. **Sampling strategy** for bagging
   - Bagging sample size as % of total rows in training set
   - Max % of features represented in each bagging sample
3. **Depth of trees**
   - Maximum depth
   - Minimum samples per leaf

Clustering

What is clustering?

- A technique used to organize data points into logical groups without using explicit group labels
- Sorts similar data points into the same clusters, and different points into different clusters




HOW MON
ROUND
the veracity
bankers who were
to the LOUs said
their
branches to
antecedents of
Pacific Diamond
FZE

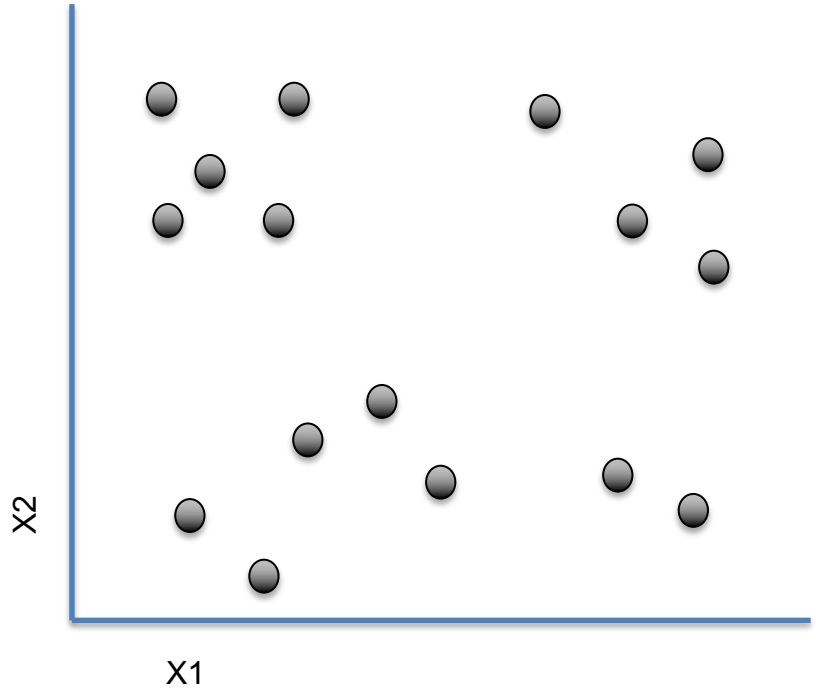
Determining similarity

How do we determine whether things are similar or different?

1. What is our basis for similarity?
2. How do we calculate similarity?


outrageously
AMBITIOUS
K-Means Clustering
Duke
PRATT SCHOOL of
ENGINEERING

K-Means Clustering



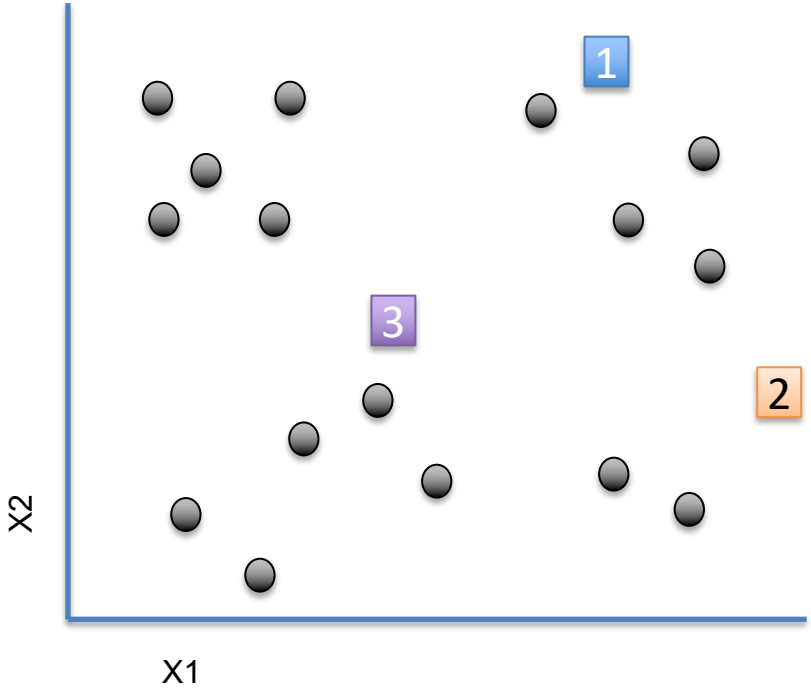
K-Means clustering groups points into clusters based on distance from the nearest cluster center

Objective:
Minimize the sum of the distances from each point to its assigned cluster center

$$min \sum_{i=1}^{k} \sum_{x \in S_i} ||x - \mu_i||$$

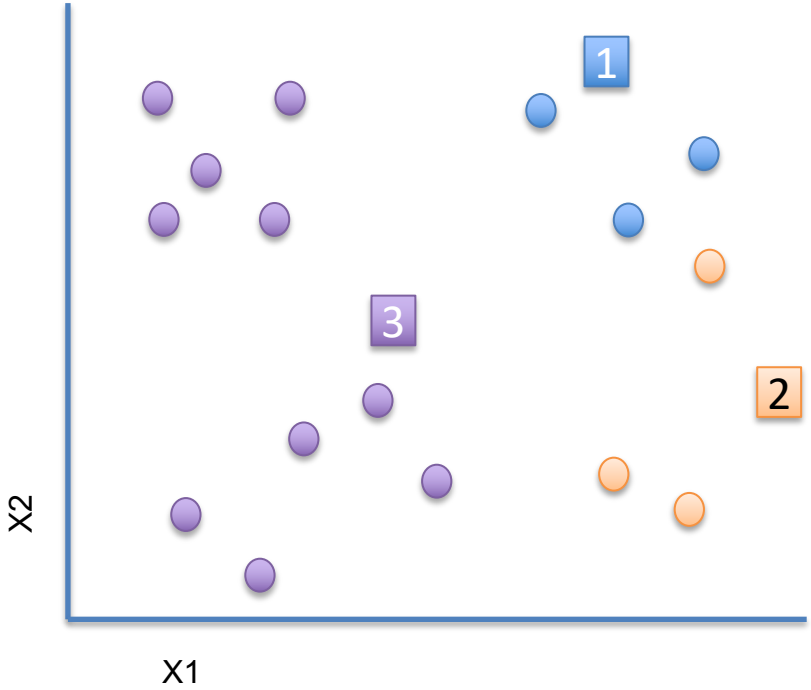
μ_i is the center in cluster S_i

K-Means Clustering



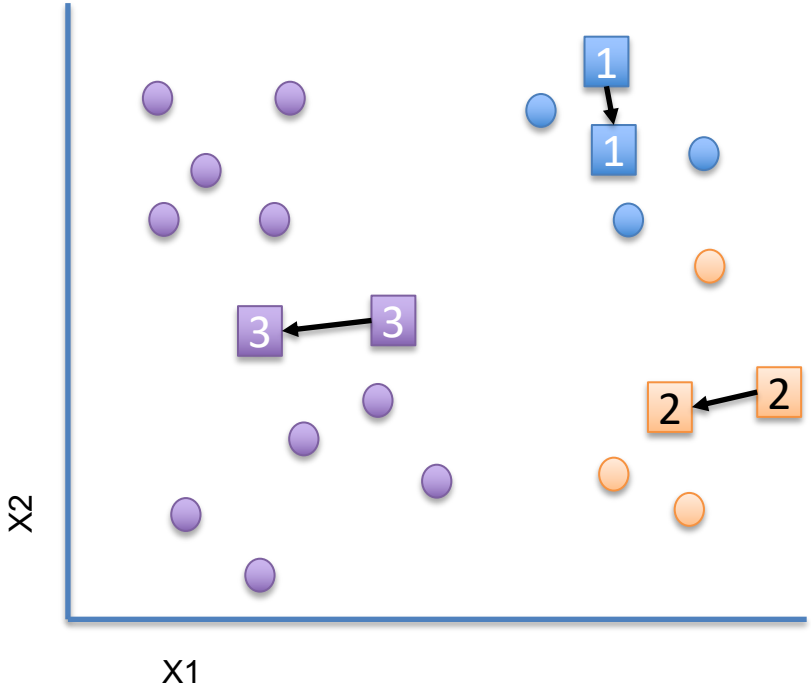
Step 1: Select the number of clusters (k) and randomly select locations for each cluster center

K-Means Clustering



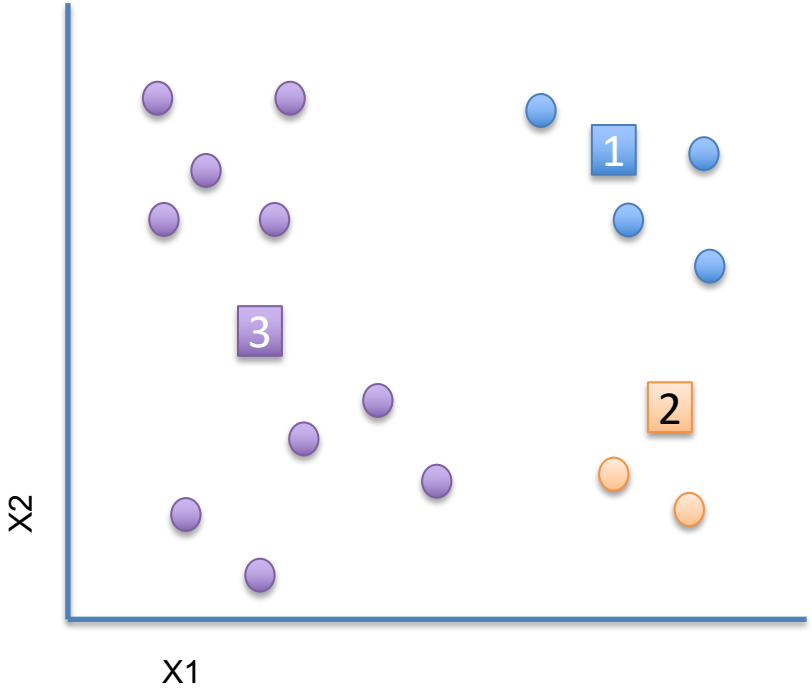
Step 2: Assign each datapoint to the cluster associated with the nearest cluster center

K-Means Clustering



Step 3: Move the cluster centers to the mean location of the assigned points in the cluster

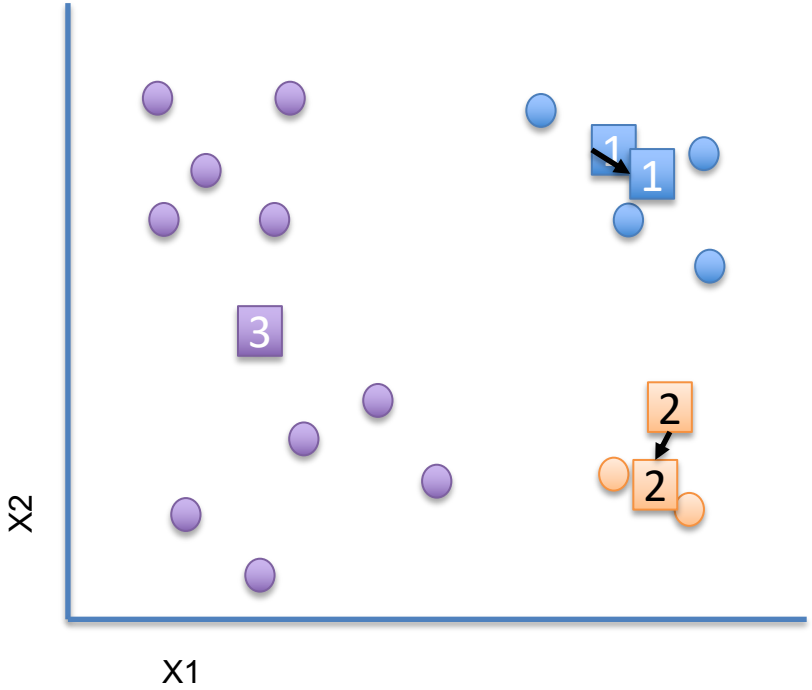
K-Means Clustering



Repeat steps 2-3 until no cluster centers move:

- Assign points to nearest cluster center
- Update location of each cluster center to mean location

K-Means Clustering



Repeat steps 2-3 until no cluster centers move:

- Assign points to nearest cluster center
- Update location of each cluster center to mean location

K-Means Strengths & Weaknesses

Strengths

- Easy to implement
- Quick to converge
- Often a very good starting point for clustering tasks

Weaknesses

- Requires user to specify the number of clusters in advance
- Forms linear boundaries – does not work well for geographically complex data


outrageously
AMBITIOUS
Wrap-up
Duke
PRATT SCHOOL of
ENGINEERING

Wrap Up

- As we evaluate supervised learning algorithms on a given problem, we should consider:
  - Performance
  - Interpretability
  - Computational cost

- For unsupervised learning, an important consideration is how we define similarity