GENERAL ASSEMBLY

DECISION TREES AND RANDOM FORESTS

Jonathan Balaban

DAT2

LEARNING OBJECTIVES

- Build decision tree models for classification and regression
- Understand the differences between linear and non-linear models
- Build random forest models for classification and regression
- Extract important predictors in a random forest model

PRE-WORK REVIEW

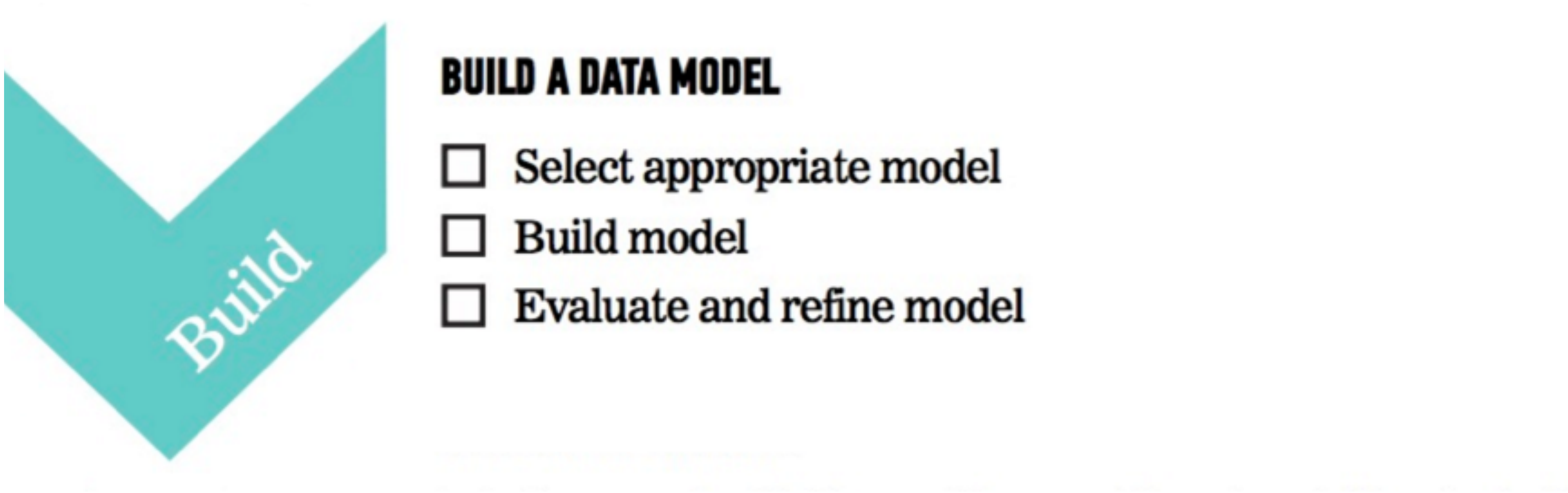
- Use Seaborn to create plots
- Know how to bootstrap
- Explain cross-validation, logistic regression, and overfitting
- Build and evaluate *some* classification model in sckit-learn using cross-validation and AUC
- Precision and Recall

OPENING

DECISION TREES AND RANDOM FORESTS

OVERVIEW OF THE DATA SCIENCE WORKFLOW

- In this lesson, we will focus on mining a dataset and building a decision model, then refining for best predictive ability.



BUILD A DATA MODEL

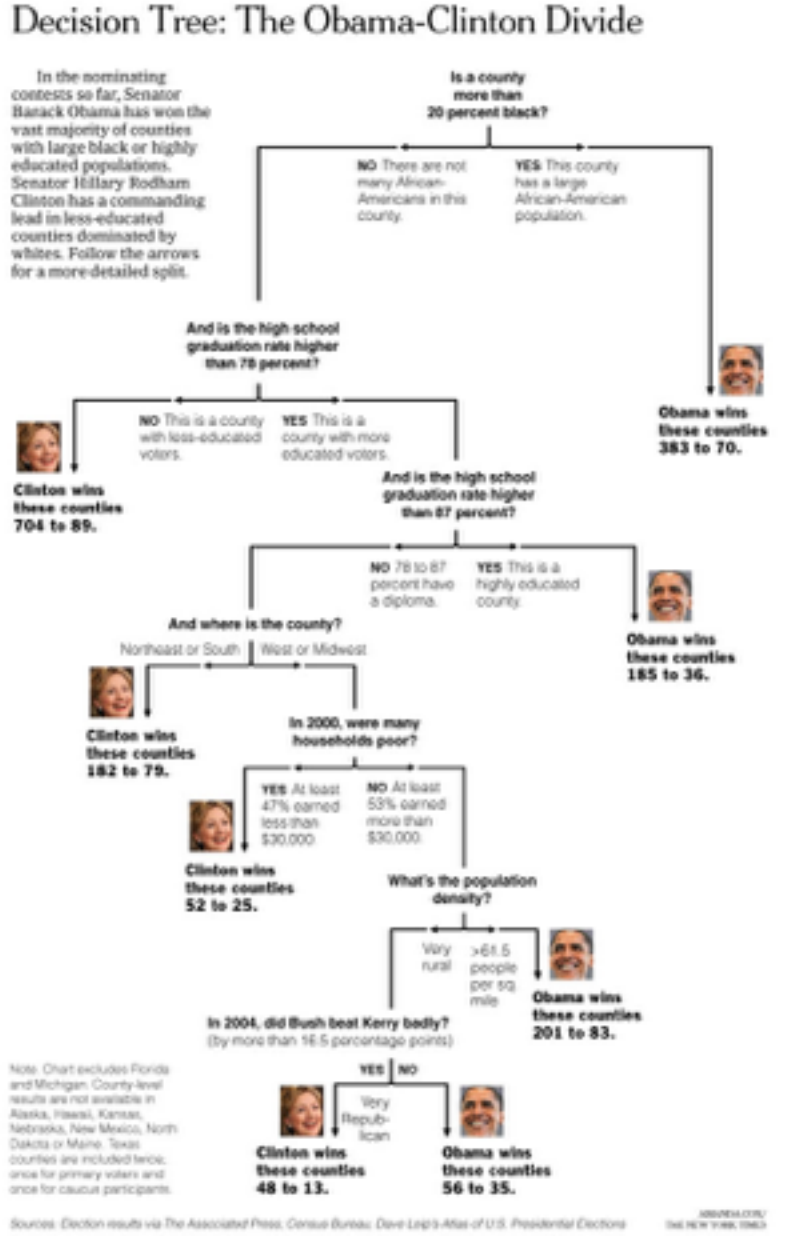
- □ Select appropriate model
- □ Build model
- □ Evaluate and refine model

INTUITION BEHIND DECISION TREES

- Decision trees are like the game “20 questions”. They make decision by answering a series of questions, most often binary questions (yes or no).

- We want the smallest set of questions to get to the right answer.

- Each questions should reduce the search space as much as possible.



Decision Tree: The Obama-Clinton Divide

In the nominating contests so far, Senator Barack Obama has won the vast majority of counties with large black or highly educated populations. Senator Hillary Rodham Clinton has a commanding lead in less-educated counties dominated by whites. Follow the arrows for a more detailed split.

Is a county more than 20 percent black?
NO There are not many African-Americans in this county.
YES This county has a large African-American population.
Obama wins these counties 383 to 70.
And is the high school graduation rate higher than 78 percent?
NO This is a county with less-educated voters.
YES This is a county with more educated voters.
Clinton wins these counties 704 to 89.
And is the high school graduation rate higher than 87 percent?
NO 78 to 87 percent have a diploma.
YES This is a highly educated county.
Obama wins these counties 185 to 36.
And where is the county?
Northeast or South
West or Midwest
Clinton wins these counties 182 to 79.
In 2000, were many households poor?
YES At least 47% earned less than $30,000
NO At least 53% earned more than $30,000
Clinton wins these counties 52 to 25.
What's the population density?
Very rural
>61.5 people per sq mile
Obama wins these counties 201 to 83.
In 2004, did Bush beat Kerry badly?
(by more than 16.5 percentage points)
YES
NO
Very Repub-lican
Clinton wins these counties 48 to 13.
Obama wins these counties 56 to 35.

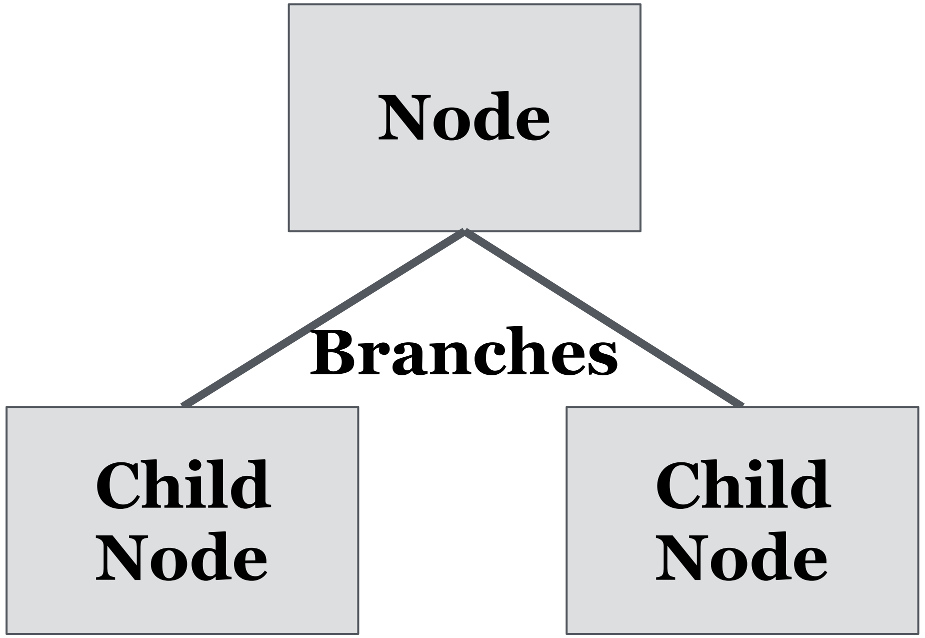
Note: Chart excludes Florida and Michigan. County-level results are not available in Alaska, Hawaii, Kansas, Nebraska, New Mexico, North Dakota or Maine. Texas counties are included twice, once for primary voters and once for caucus participants.

Sources: Election results via The Associated Press; Census Bureau; Dave Leip's Atlas of U.S. Presidential Elections

THE NEW YORK TIMES

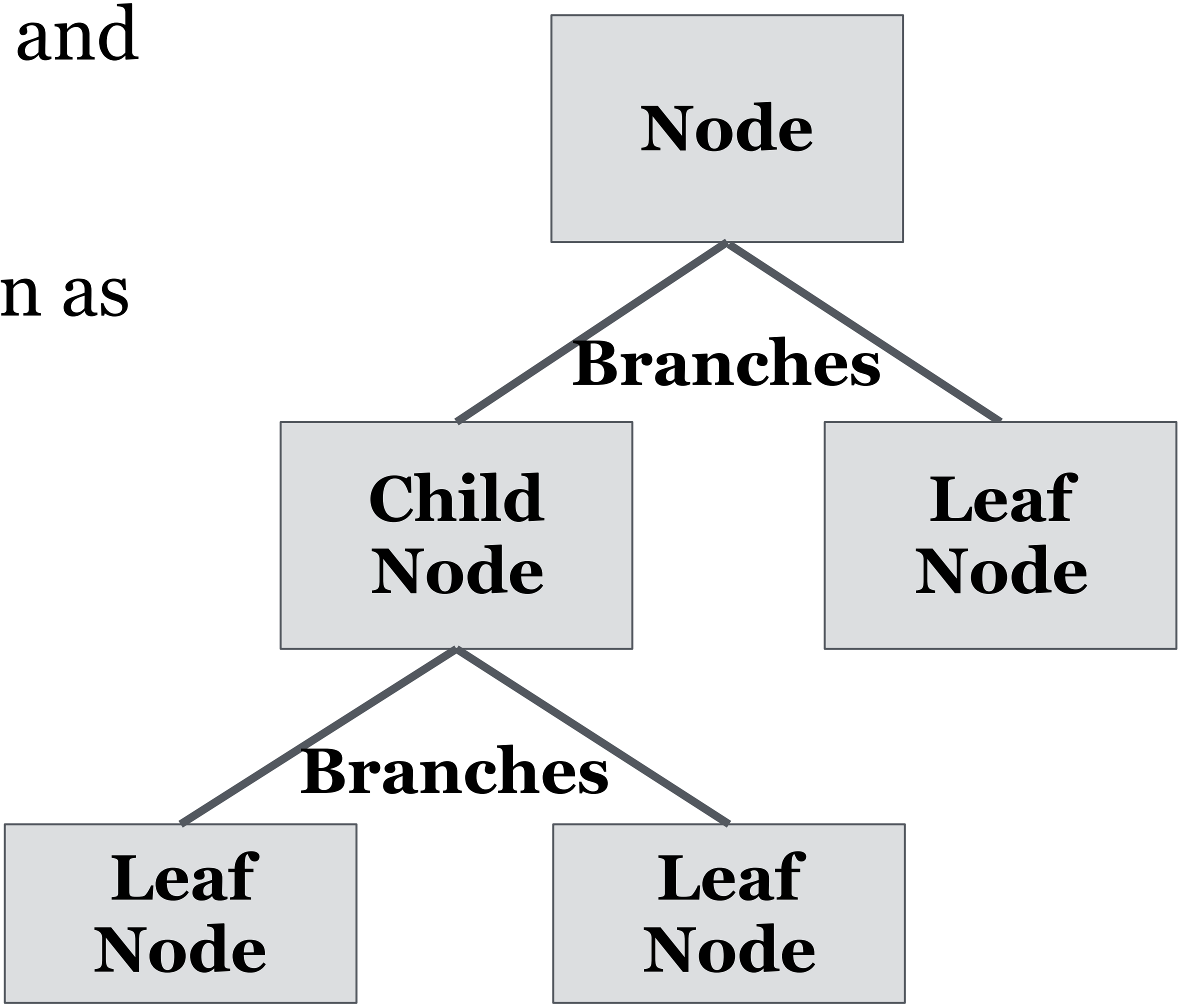
TREES

- Trees are a data structure made up of *nodes* and *branches*.
- Each node *typically* has two or more branches that connect it to its children.



TREES

- Each child is another node in the tree and contains its own *subtree*.
- Nodes without any children are known as *leaf* nodes.



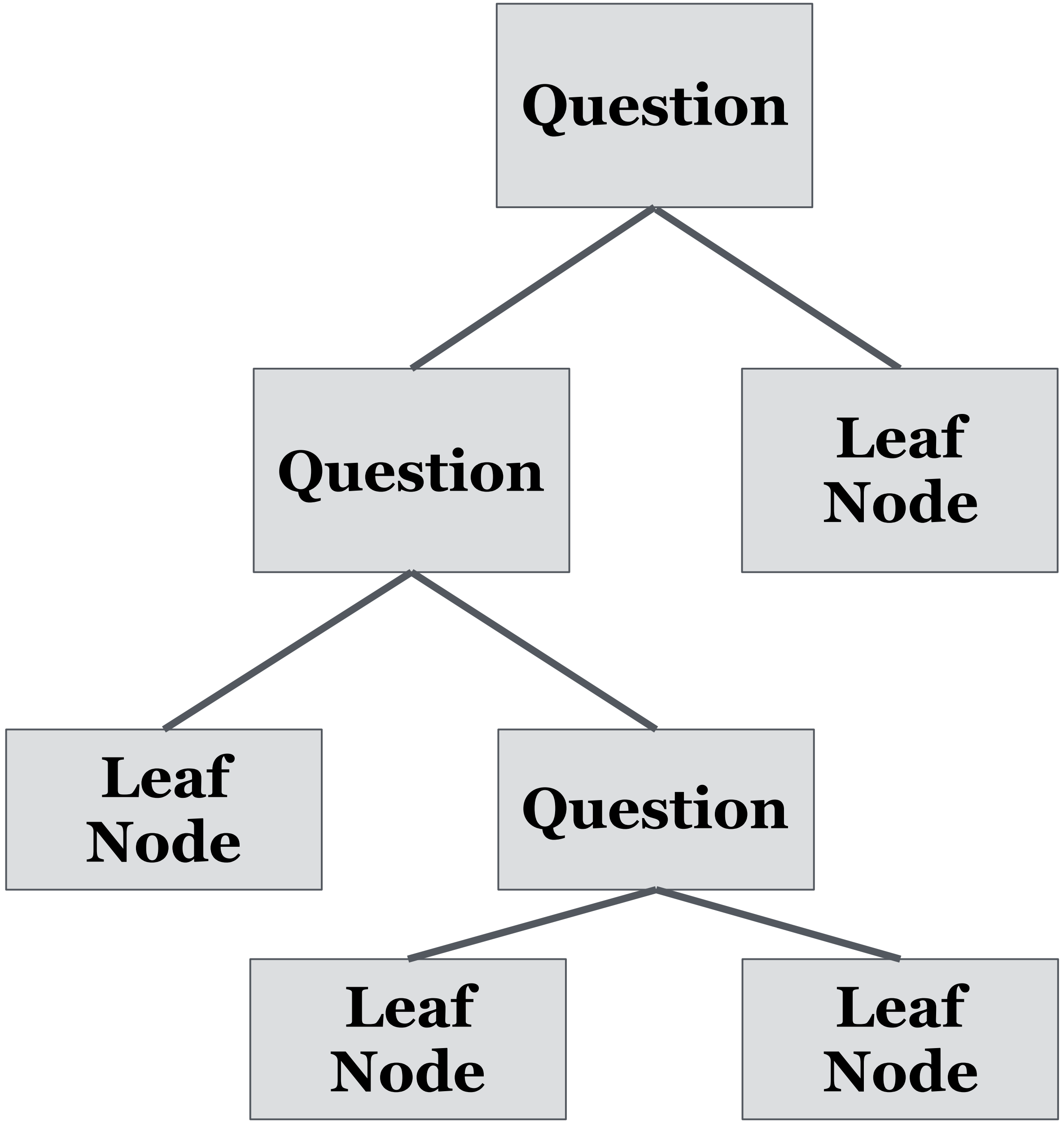
DECISION TREES

- A *decision tree* contains a question at every node.

- Depending upon the answer to the question, we proceed down the left or right branch of the tree and ask another question.

- Once we don’t have any more questions (at the *leaf* nodes), we make a prediction.

- Note: The next question is always dependent on the last.



DECISION TREES

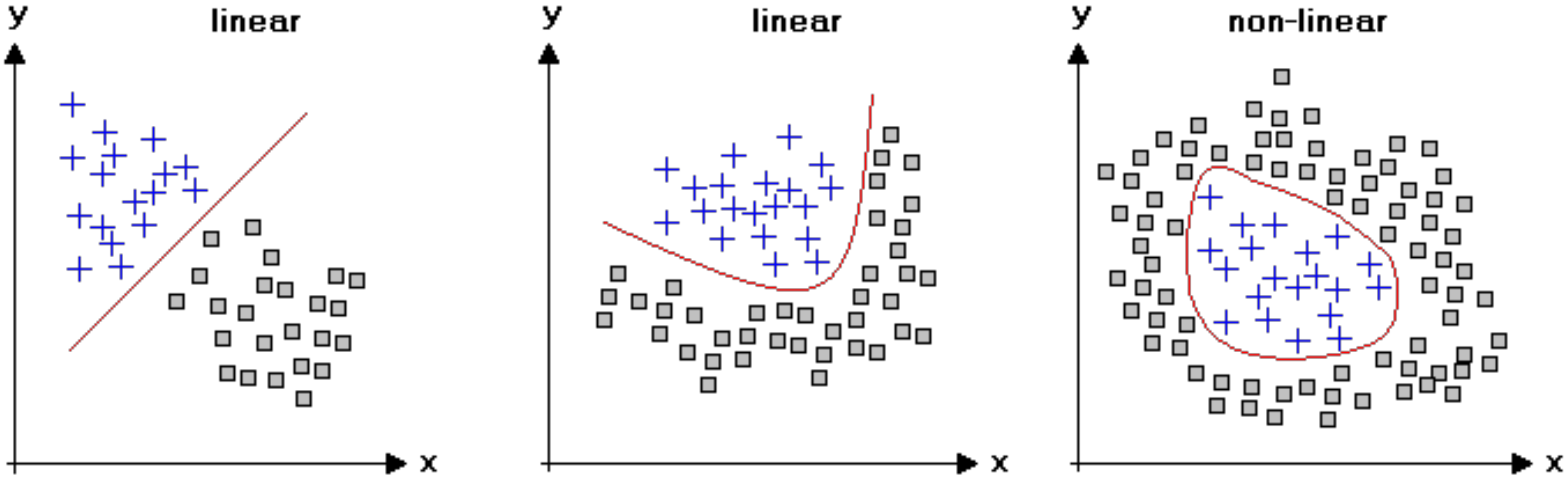
- Let’s suppose we want to predict if an article is a news article.

- What questions should we ask to make a prediction?

- How many questions should we ask?

COMPARISON TO PREVIOUS MODELS

- Linear vs. non-linear classification models



COMPARISON TO PREVIOUS MODELS

- Decision trees are *non-linear*, an advantage over logistic regression.

- A *linear* model is one in which a change in an input variable has a constant change on the output variable.

- Trees automatically contain interaction of features, since each question is dependent on the last.

TRAINING A DECISION TREE MODEL

- Training a decision model is deciding the best set of questions to ask.
- A good question will be one that best segregates the positive group from the negative group and then narrows in on the correct answer.
- For example, in our news article decision tree, the best question is one that creates two groups, one that is mostly news stories and one that is mostly non-news stories.

TRAINING A DECISION TREE MODEL

- We can quantify the *purity* of the separation of groups using Classification Error, Entropy, or Gini Coefficient.

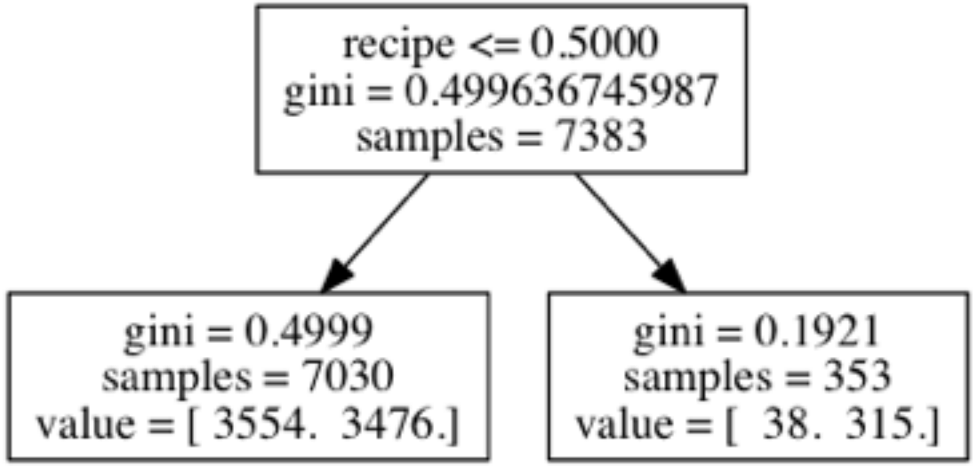
- We want to choose the question that gives us the best *change* in our purity measure. At each step, we can ask, “Given our current set of data points, which question will make the largest change in purity?”

- This is done *recursively* for each new set of two groups until we reach a stopping point.

TRAINING A DECISION TREE MODEL

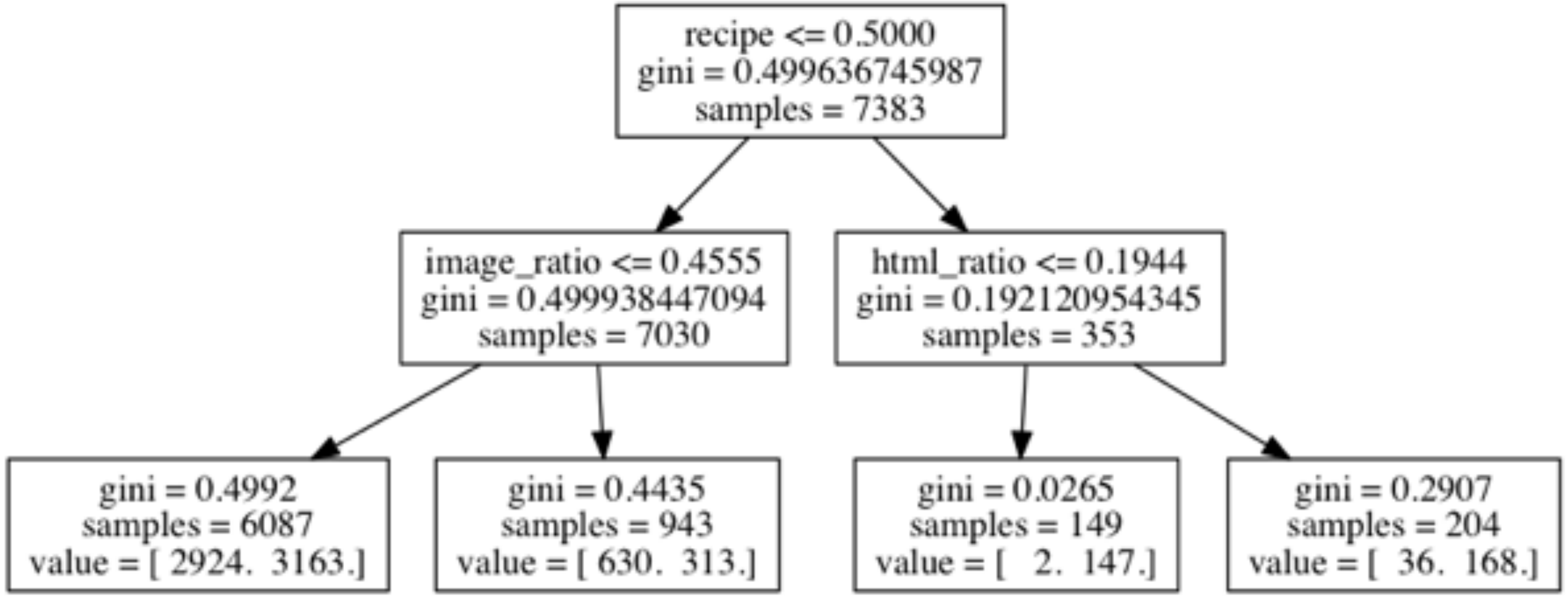
- Let's build a sample tree for our evergreen prediction problem. Assume our features are whether the article contains a recipe, the image ratio, the html ratio.

- First, let's choose the feature that gives us the highest purity, the recipe feature.



TRAINING A DECISION TREE MODEL

- We can take each side of the tree and repeat the process.



- We can continue this process until we have asked as many questions as we want or until our leaf nodes are completely pure.

MAKING PREDICTIONS FROM A DECISION TREE

- Predictions are made by answering each of the questions.
- Once we reach a leaf node, our prediction is made by taking the majority label of the training samples that fulfill the questions.
- In our sample tree, if we want to classify a new article, ask:
    - Does the article contain the word recipe?
    - If it doesn’t, does the article have a lot of images?
    - If it does, then 630 / 943 article are evergreen.
        - So we can assign a 0.67 probability for evergreen sites.

GUIDED PRACTICE

DECISION TREES IN SCIKIT-LEARN

ACTIVITY: EXPLORE THE DATASET

EXERCISE

DIRECTIONS (25 minutes)

We will be using a dataset from StumpleUpon, a service that recommends webpages to users based upon their interests. They like to recommend “evergreen” sites, ones that are always relevant. This usually means websites that avoid topical content and focus on recipes, how-to guides, art projects, etc. We want to determine important characteristics for “evergreen” websites. Follow these prompts to get started:

1. Break into groups.
2. Prior to looking at the data, brainstorm 3-5 characteristics that would be useful for predicting evergreen websites.
3. After looking at the dataset, can you model or quantify any of the characteristics you wanted? See the Notebook for data dictionary and starter code.
4. Does being a news site affect “evergreen-ness”? Compute or plot the percent of evergreen news sites.

ACTIVITY: EXPLORE THE DATASET

EXERCISE

DIRECTIONS (25 minutes)

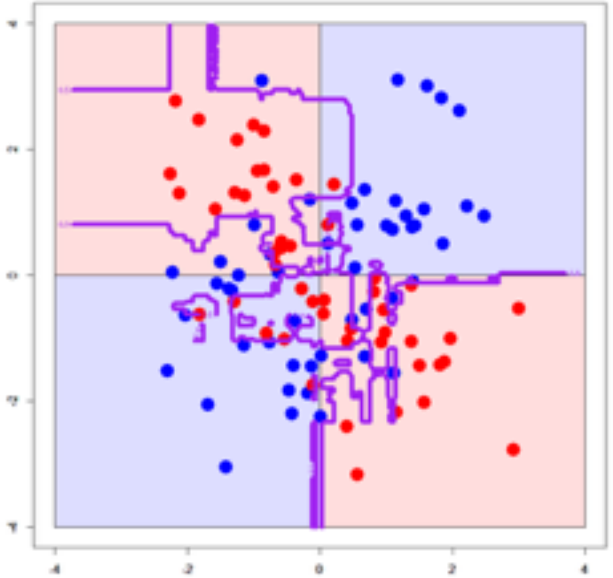
5. Does category affect evergreeness? Plot rate of evergreen for Alchemy categories.
6. How many articles are there per category?
7. Create a feature for the title containing “recipe”. Is the percentage of evergreen websites higher or lower on pages that have “recipe” in the title?

Check: Were you able to plot the requested features? Can you explain how you would approach this type of dataset?

OVERFITTING IN DECISION TREES

- Decision trees tend to be weak models because they can easily memorize or overfit to a dataset.

- A model is *overfit* when it memorizes or bends to a few specific data points rather than picking up general trends in the data.



OVERFITTING IN DECISION TREES

- An unconstrained model can learn an extreme tree (e.g. one feature for each word in a news article).
- We can limit our decision trees using a few methods:
  - Limiting the number of questions (nodes) a tree can have.
  - Limiting the number of samples in the leaf nodes.

ACTIVITY: KNOWLEDGE CHECK

EXERCISE

ANSWER THE FOLLOWING QUESTIONS

1. Why are decision trees generally thought of as weak models?
2. How can we limit our decision trees?

INTRODUCTION

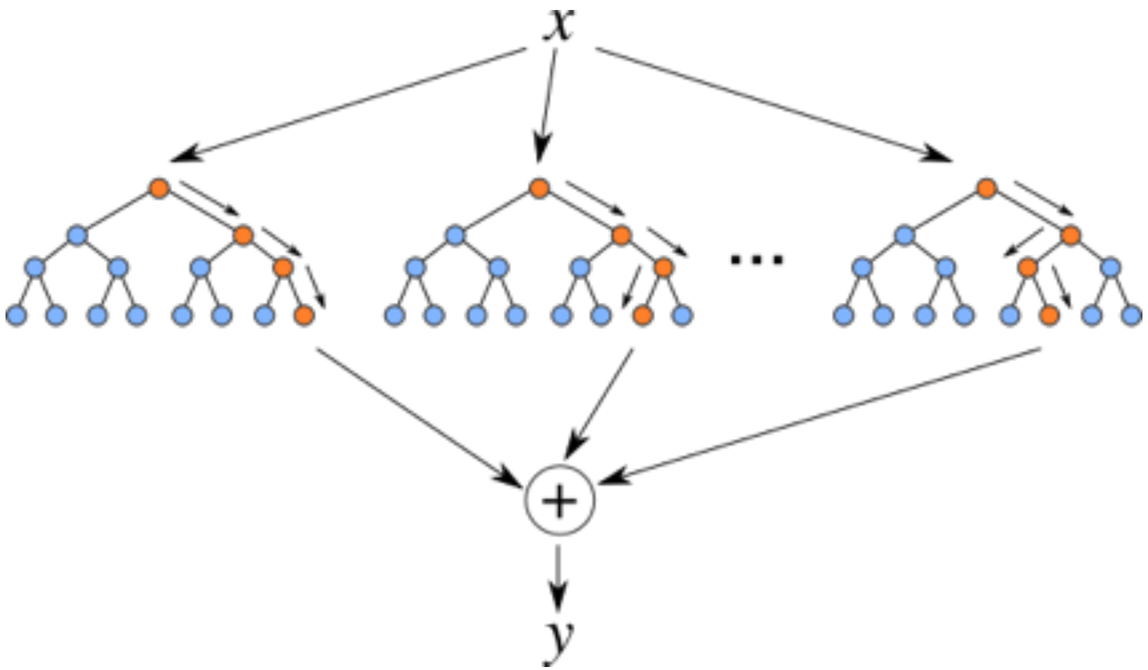
RUNNING THROUGH THE RANDOM FORESTS

RUNNING THROUGH THE RANDOM FORESTS

- Random forest models are one of the most widespread classifiers used.

- They are relatively simple to use and help avoid overfitting.

- Random Forests are an *ensemble* or collection of individual decision trees.



TRAINING A RANDOM FOREST

- Training a random forest model involves training many decision tree models.

- Since decision trees overfit easily, we use many decision trees together and randomize the way they are created.

TRAINING A RANDOM FOREST

- Random Forest Algorithm

  a. Take a bootstrap sample of the dataset.

  a. Train a decision tree on the bootstrap sample. For each split/feature selection, only evaluate a *limited* number of features to find the best one.

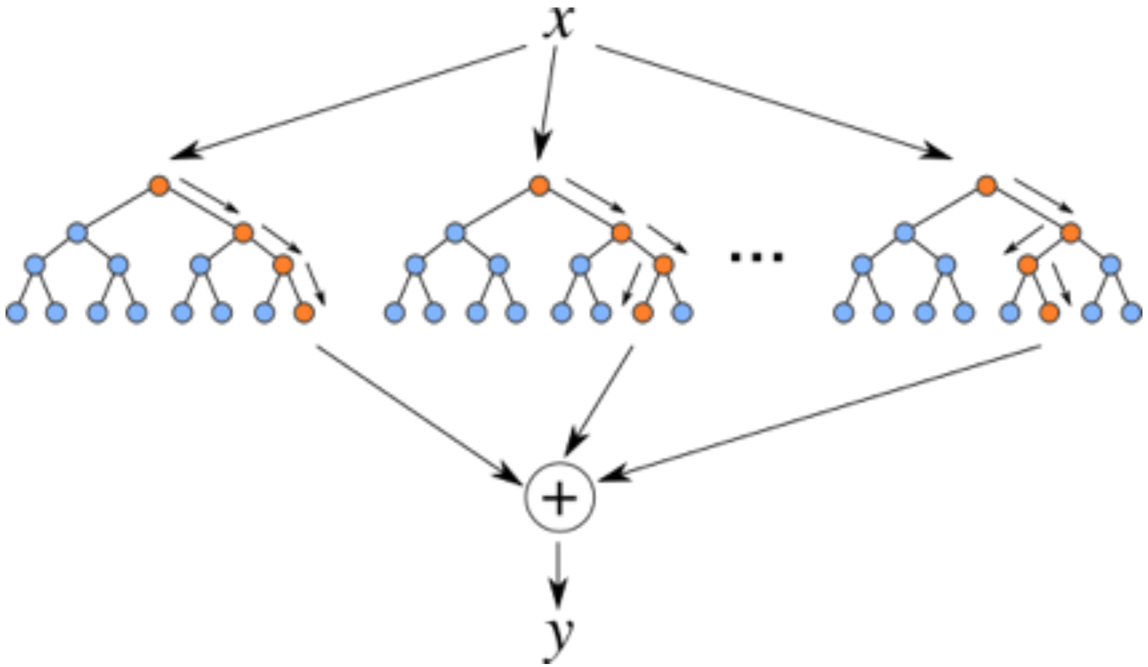
  a. Repeat this for *N* trees.

PREDICTIONS USING A RANDOM FOREST

- Predictions for a random forest model come from each decision tree.

- Make an individual prediction with each decision tree.

- Combine the individual predictions and take the majority vote.



PROS AND CONS OF RANDOM FORESTS

- Advantages
  - Easy to tune
  - Built-in protection against overfitting
  - Non-linear
  - Built-in interaction effects

- Disadvantages
  - Slow
  - Black-box
  - No “coefficients”
  - Harder to explain

ACTIVITY: REGRESSION WITH DECISION TREES & FORESTS

EXERCISE

DIRECTIONS (20 minutes)

1. Build a random forest model to predict the evergreeness of a website. Remember to use the parameter `n_estimators` to control the number of trees used in the model.
2. Take note of the features that give the best splits to determine the most important features.
3. Decision trees and random forests can be used for both classification and regression. In regression, predictions are made by taking the average value of the samples in the leaf node. You can take the average of the individual trees' predictions. Build a regression based random forest model.

EVALUATE RANDOM FORESTS USING CROSS-VALIDATION

EXERCISE

DIRECTIONS (25 minutes)

1. Building upon the previous Guided Practice, add any input variables to the model that you think may be relevant.

1. For each feature:
    a. Evaluate the model for improved predictive performance using cross-validation.
    b. Evaluate the importance of the feature.

1. **Bonus**: Just like the ‘recipe’ feature, add in similar text features and evaluate their performance.

REVIEW Q&A

- What are decision trees?
- What does training involve?
- What are some common problems with decision trees?
- What are random forests?
- What are some common problems with random forests?

COURSE

BEFORE NEXT CLASS

Final: Part 2

LESSON

Q & A

LESSON

EXIT TICKET

DON’T FORGET TO FILL OUT YOUR EXIT TICKET