Decision Trees

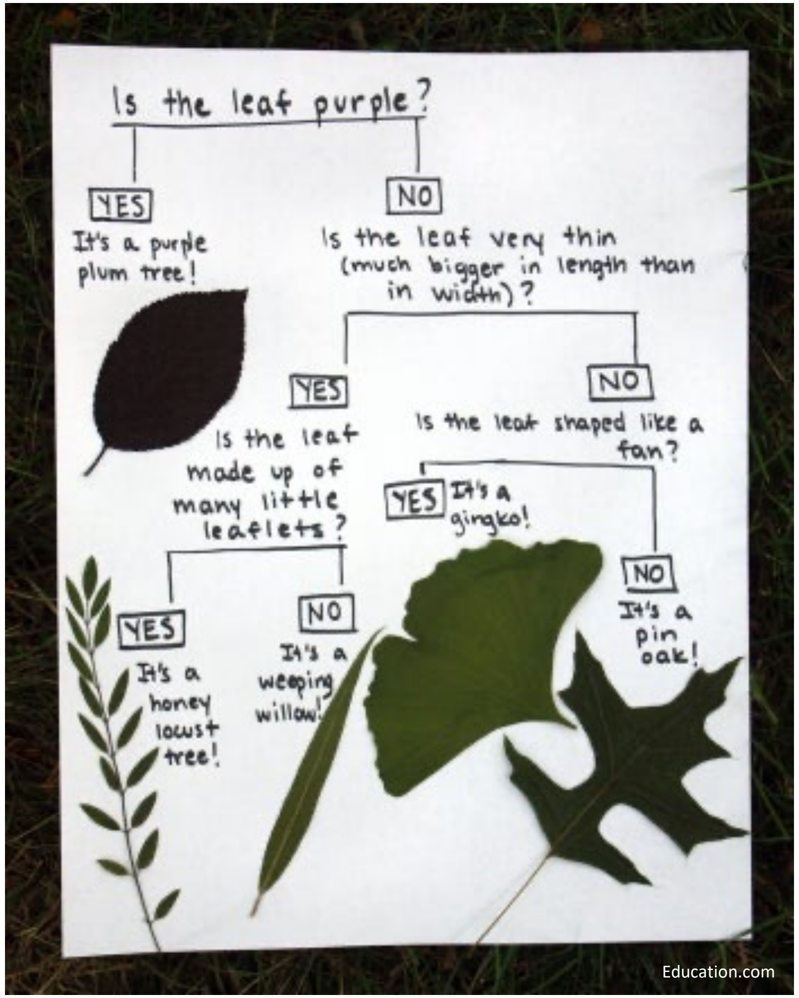
COSC 410: Applied Machine Learning

Spring 2022

Prof. Apthorpe

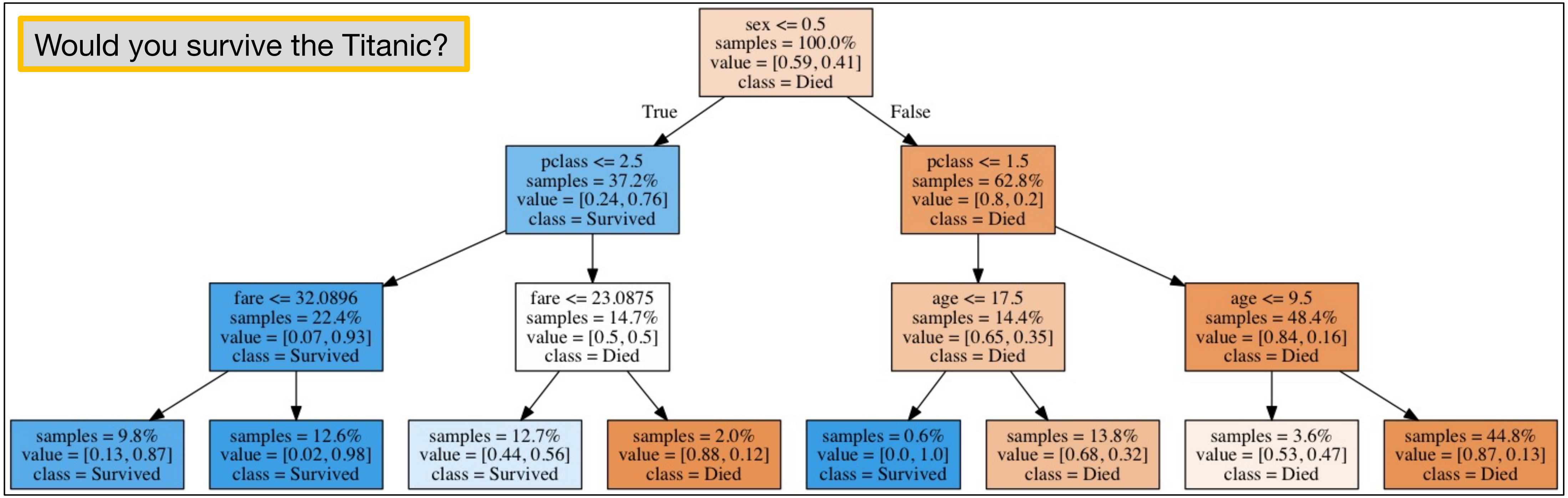
Outline

- Prediction
- Training
- Impurity Metrics
- Feature Importance
- Perks
- Overfitting



Decision Tree Prediction

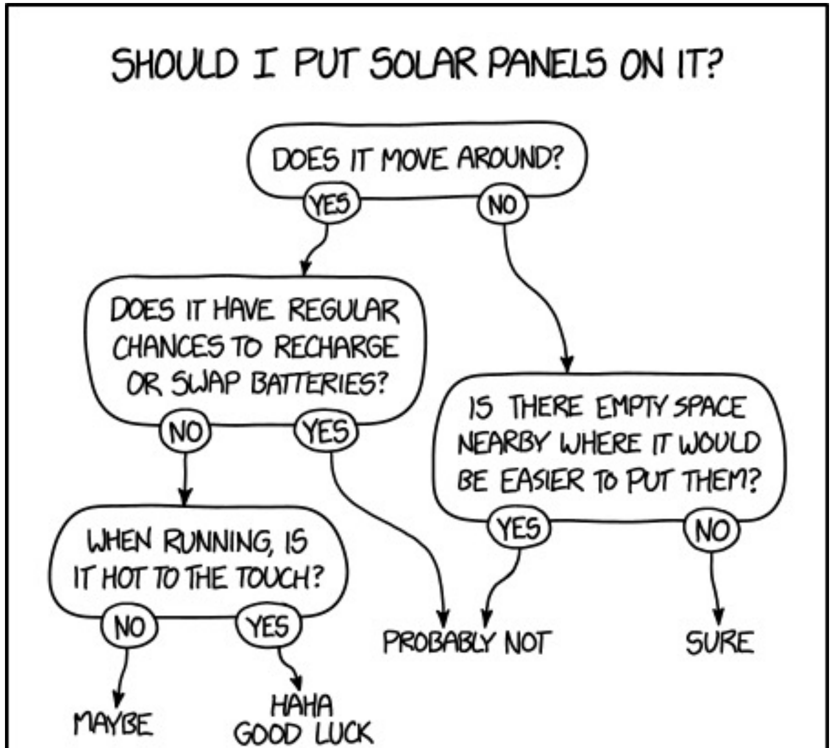
1. Start at root node
2. Continue to child node that satisfies root condition…repeat until you reach a leaf
3. Predict **mode** (classification) or **mean** (regression) of training labels in the leaf



Patrick Triest

Decision Tree Perks

- Little preprocessing required
  - Accepts nominal, numeric, or binary data
  - Standardization/normalization unnecessary
- Trained model is easily interpretable
- Trained model indicates **feature importance**



https://xkcd.com/1924/

Decision Tree Training

- ***Goal:*** Train a **balanced** tree with minimal training error
- Classification and Regression Tree (CART) algorithm

Greedy Algorithm:
Tree many not be optimally balanced
But optimal alg. is NP-complete

- Select a feature *k* and threshold t_k that divide the examples in current node by number and label **as equally as possible** (minimize cost function *J*)

examples in left child

Impurity of left child

$$J(k, t_k) = \frac{m_{left}}{m} G_{left} + \underbrace{\frac{m_{right}}{m} G_{right}}_{\text{Same for right child}}$$

examples in current node

Training examples in **pure nodes** all have the same label

- Repeat for each child node until max depth is reached or all leaf nodes are **pure**

Node Impurity Metrics

- Lowest when all examples have same label
- Highest when examples are spread evenly across labels
- Gini Impurity $$G = 1 - \sum_{k=1}^{n} \left(\frac{||\text{examples in class } k||}{||\text{all examples}||} \right)^2$$
- Entropy $$H = -\sum_{k=1}^{n} \frac{||\text{examples in class } k||}{||\text{all examples}||} \log \left(\frac{||\text{examples in class } k||}{||\text{all examples}||} \right)$$

Skip classes with no examples to avoid undefined log(0)

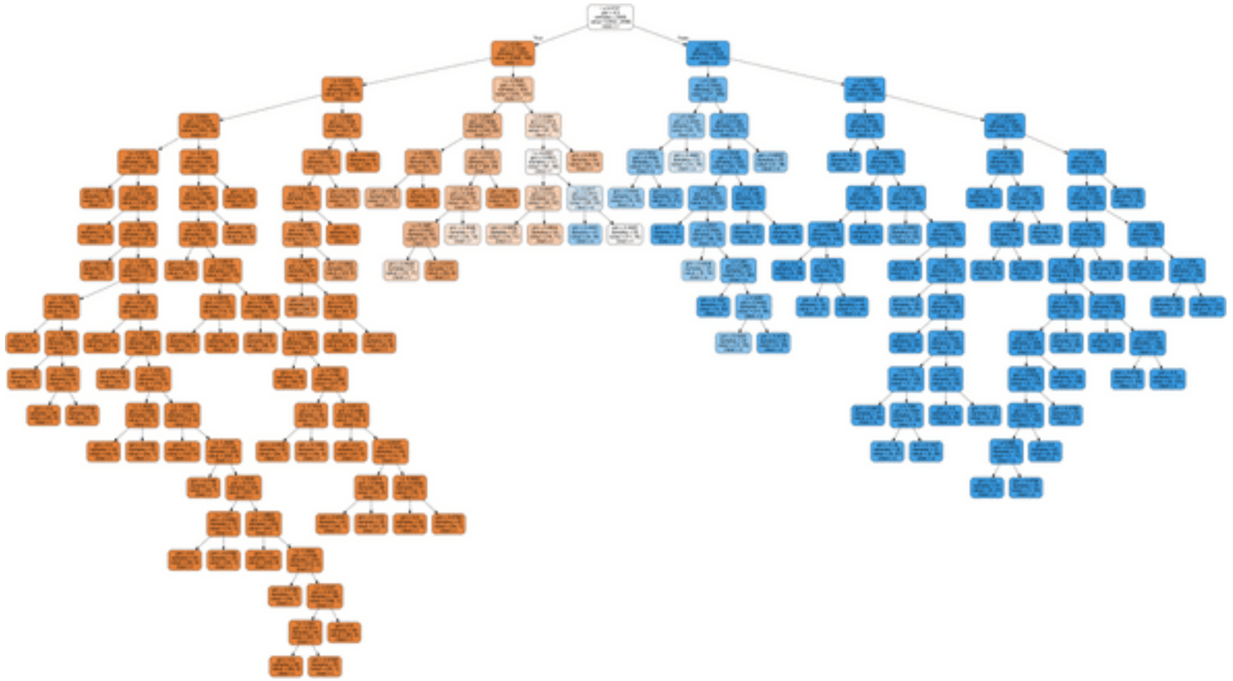
Feature Importance

- Features can be ranked by **importance** to a decision tree
  - Mean **increase in purity** from **splitting on feature** across the tree
  - Varies depending on stochastic tree construction algorithm
    - Best to train several trees and average importance

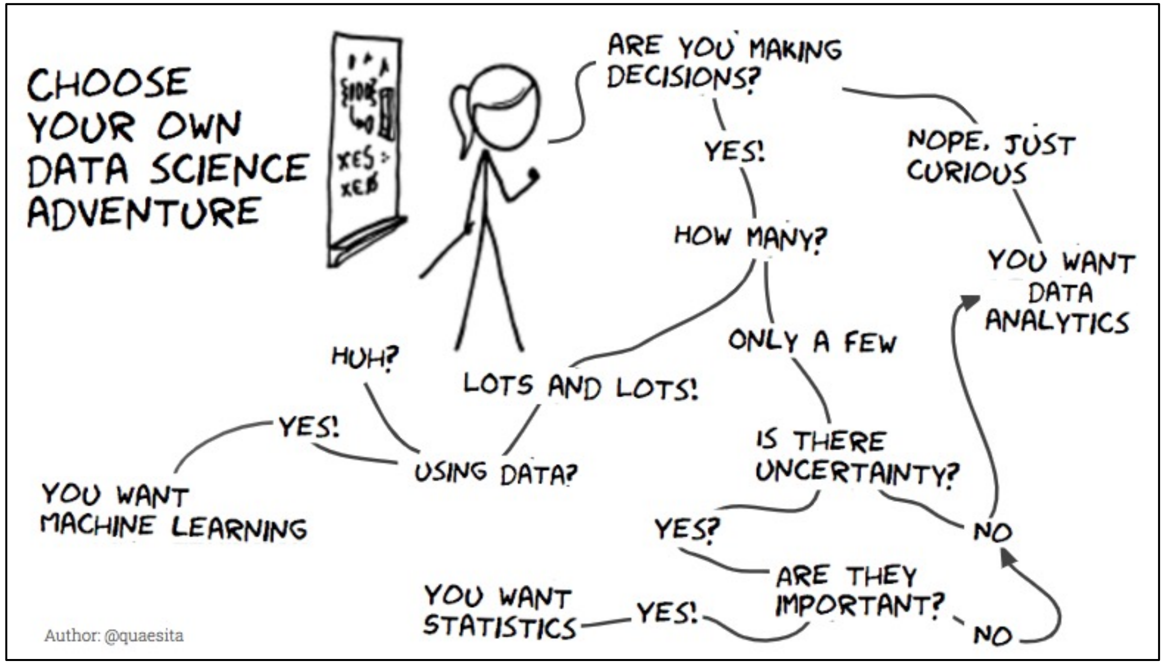
- **More "important" features are more predictive of labels**
  - Provides intuition about underlying phenomenon you are attempt to model

Overfitting

- Decision trees are **non-parametric**
  - Can fit the training data exactly…just keep adding nodes until each leaf is pure
  - Leaf nodes with only a small number of training examples may cause overfitting
- **Max depth** hyperparameter
  - Limit tree to a specific depth
- **Min split** hyperparameter
  - Don't add child nodes if current node has fewer than a threshold # of examples



- **Pruning**
  - Train full tree and iteratively remove nodes that provide less than a threshold decrease in cost



Programming Practice

DecisionTrees.ipynb

Questions?