Pitfalls/FAQ

- **Tips on learning how to build a decision tree**
  - Go through the textbook pages listed on the class website. Follow this psuedocode
  - Using a small dataset, draw a decision tree from scratch and compare it to a real decision tree.
  - Write a separate entropy function, IG function, and best attribute to split on function. Make your code modular and try to keep your fit function as simple as possible.
- **How to go about making leaves (recursion base case)**
  - To encode this, you can keep a target value inside self.value. If this is None, then the node is not a leaf. Otherwise, it is a leaf.
  - The Tree class is a singular node. When a new Tree is instantiated, you can pass in the final target value for that leaf into the constructor.
- **What information should nodes include?**
  - Examples with that target value
  - Eg. Say there are 5 examples. Attribute A is what your IG calculation tells you to split on. When this split is done, say that 3 examples go down the left branch and 2 go down the right branch. Within the children (right and left) nodes, store the 3 examples and 2 examples respectively.
  - Self.branches is list of more node instances (children)
- **Clarifying base cases**
  - Leaves are
    - Nodes with no more examples left to run an IG calculation on
    - Nodes where are all the examples have the same target value
- **Predict should return something**
  - Simply go down the tree and return a single target value when it reaches a leaf
- **Recursion and passing arguments in**
  - Avoid using global variables
  - Great guide on understanding Python parameter passing
    - https://robertheaton.com/2014/02/09/pythons-pass-by-object-reference-as-explained-by-philip-k-dick/
- **Can a single attribute/feature appear multiple times throughout the decision tree; that is, can the same attribute/feature be found in both the right side and the left side of the starting node?**
  - Yes
- **Use a helper function when implementing fit and predict**
- **Code runs and never stops.**
  - Infinite recursion → problem with base cases
- **I'm getting a divide by 0 warning or "invalid value encountered in long_scalars"**
  - Warning usually occurs in *metrics.py* and caused by dividing by 0

    - To fix: E.g. If TruePositive + FalsePositive == 0, then just return 0 as precision.
- **I implemented load_data and I'm printing the data but it looks completely wrong**
    - Test case for load_data randomly generates data
- **No errors but wrong result?**
    - Fit tree to data by hand with ur own data then run code on data and compare the result from the code from your own results
- **Prior probability explanation**
    - Question 44 on CampusWire