Project 2

WU1: Answer questions A, B and C for both OAA and AVA. (Questions about “indicative” are open-ended. Any reasonable answers with analysis will be credited.)

(A) What words are most indicative of *being* Sauvignon-Blanc? Which words are most indicative of not being Sauvignon-Blanc? What about Pinot-Noir (label==2)?

- Most indicative of being Sauvignon-Blanc:
  - OAA: "opens" and "hint" are indicative of it being Sauvignon-Blanc because in the cases where a decision is made based on these words, there are no false negatives, such that Sauvignon-Blanc is classified as "likely not to be" (class 0), even though is a single false positive for "hint".
  - AVA: "citrus" and "lime" are most indicative of the description being for a Sauvignon-Blanc because when this word was looked for and found, it resulted in no false-positives, or in other words, when this word was searched for and found, it meant the description was always for a Sauvignon-Blanc.
- Most indicative of not being Sauvignon-Blanc:
  - OAA: "citrus" and "lime" are the most indicative of a description not belonging to a Sauvignon-Blanc as when they are not present in a description, there is a very high chance that the description does not belong to a Sauvignon-Blanc. 96% (359/375) class 0 descriptions don't include "citrus" and 99% (356/359) class 0 descriptions don't include "lime".
  - AVA: The only cases found in AVA where something was classified as not likely a Sauvignon-Blanc was when none of the following words were present in the description; "citrus", "lime", and "refreshing".
- Most indicative of being Pinot-Noir:
  - OAA: "allspice" is indicative of the description being for a Pinot-Noir because te case where "allspice" is used as a variable and is seen in the description, the 2/23 descriptions with "allspice" in it are classified as Pinot-Noir.
  - AVA: "green" and "red" are indicative of the description being for a Pinot-Noir because in the presence of these words, when tested for, there are no false-positives, meaning everything that is classified as 1 after being seen with these words, is indeed "likely" to be Pinot-Noir.
- Most indicative of not being Pinot-Noir:
  - OAA: "round" is indicative of the description being for something that isn't Pinot-Noir because when tested on the word "round" every description that contained "round" was a class 0 and all 12 without "round" were class 1, aka Pinot-Noir.
  - AVA: "thai" and "crisp" are indicative of the description belonging to a wine that is not Pinot-Noir as they both result in a classification of 0 when present with 0 false false-negatives.

(B) Train depth 3 decision trees on the full WineData task (with 20 labels). What accuracy do you get? How long does this take (in seconds)? One of my least favorite wines is Viognier–what words are indicative of this?

- OAA:

        - Accuracy: 37.2%
        - How long?: .19 seconds
        - What words are indicative of Viognier?: "milk" and "seductive" as any time these are tested for and found, there are no false-positives found, meaning that when "milk" and "seductive" were looked for in the description, the description was always for a Viognier.
    - AVA:
        - Accuracy: 26.1%
        - How long?: .24 seconds
        - What words are indicative of Viognier?: There were not many words I could find that were indicative of a description being used for a Viognier, but if the description had the word "peach", "peaches", or "tannin", it is almost certain to be a drink other than Viognier.

    (C) Compare the accuracy using zero-one predictions versus using confidence. How much difference does it make?

    - OAA: 24.2% accuracy w/ zero-one vs 37.2% accuracy w/ confidence. The choice of prediction type makes a significant difference for OAA with a 13% difference in training accuracy.
    - AVA: 26.4% accuracy w/ zero-one vs 26.1% accuracy w/ confidence. The choice of prediction type makes a little to no difference for AVA with a .3% difference in training accuracy.

WU2: Now, you must implement a tree-based reduction. Most of train is given to you, but predict you must do all on your own. A tree class is provided to help you. Show the test accuracy you get with a balanced tree on the WineData using a DecisionTreeClassifier with max depth 3.

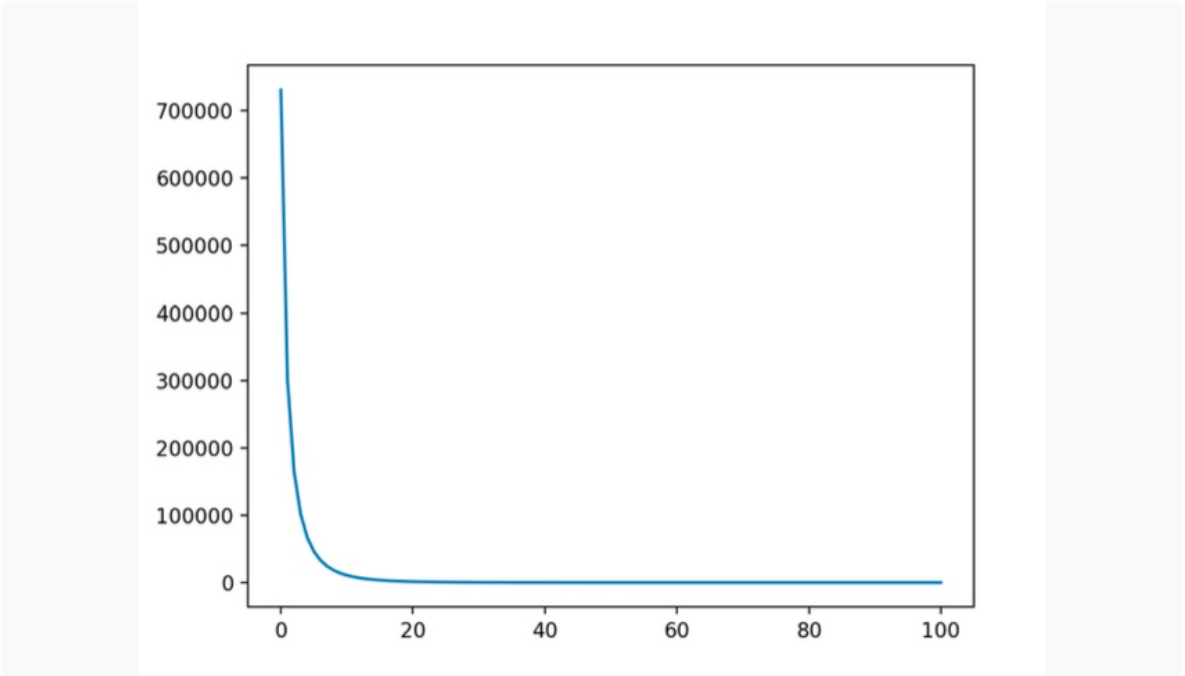
```
>>> import multiclass
>>> import util
>>> from datasets import *
>>>
>>> t = multiclass.makeBalancedTree(range(5))
>>> h = multiclass.MCTree(t, lambda: DecisionTreeClassifier(max_depth=3))
>>> h.train(WineData.X, WineData.Y)
training classifier for [0, 1] versus [2, 3, 4]
training classifier for [0] versus [1]
training classifier for [2] versus [3, 4]
training classifier for [3] versus [4]
>>> P = h.predictAll(WineData.Xte)
>>> mean(P == WineData.Yte)
0.23840445269016697
```

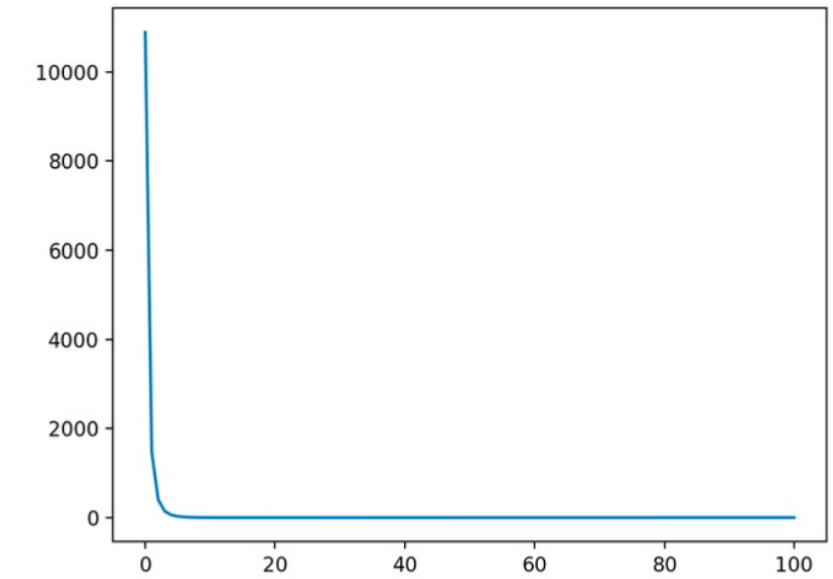
WU3: What is the impact of the step size on convergence? Find values of the step size where the algorithm diverges and converges.

- In general, large step sizes can skip local minima, which negatively impacts the function's performance. For the function x^2 and using 100 iterations, negative step sizes all diverge. Step sizes greater than 0 converge for values less than 6. However, for step sizes greater than 6, it doesn't converge within 100 iterations. Increasing the step size after 0 causes convergence time to increase and eventually fail completely.

WU4: Come up with a *non-convex* univariate optimization problem. Plot the function you're trying to minimize and show two runs of `gd`, one where it gets caught in a local minimum and one where it manages to make it to a global minimum. (Use different starting points to accomplish this.)

- The function $x4 + x(x - 1)(x + 1) - x2$ has a global minimum at (0.64, -0.62) and a local minimum at (0, -1). Starting at x = 29, the function finds the global minimum. When starting at x = 10, the function gets stuck at the local minimum 0.



- 

WU5: For each of the loss functions, train a model on the binary version of the wine data (called WineDataBinary) and evaluate it on the test data. You should use lambda=1 in all cases. Which works best? For that best model, look at the learned weights. Find the *words* corresponding to the weights with the greatest positive value and those with the greatest negative value. Hint: look at WineDataBinary.words to get the id-to-word mapping. List the top 5 positive and top 5 negative and explain.

Loss Function	Training Accuracy	Test Accuracy
Squared	0.243	0.314
Logistic	0.757	0.686
Hinge	0.753	0.686

- Top 5: is, this, tannins, black, wine
- Bottom 5: lime, grapefruit, tropical, crisp, citrus
- The majority of the bottom five words are pertaining to specific flavors, like grapefruit or lime. Additionally, only one of the words doesn't have to do with a specific type of flavor. All of the top five words are general, except for black. Words that carry the most weight are general and can be applied to more wines while words that carry the least weight are more specific to types of wine.