# Discussion Session 5 - hw3 written part

Sikun Lin

sikun@ucsb.edu

# Announcements

- Hw2 solution & written part grades are posted
- Make sure to download the latest version’s hw (after prof.’s announcement date!)

# Today

- Decision tree with twoing criterion [example]
- hw pick up & QA

# Definition of the twoing criterion

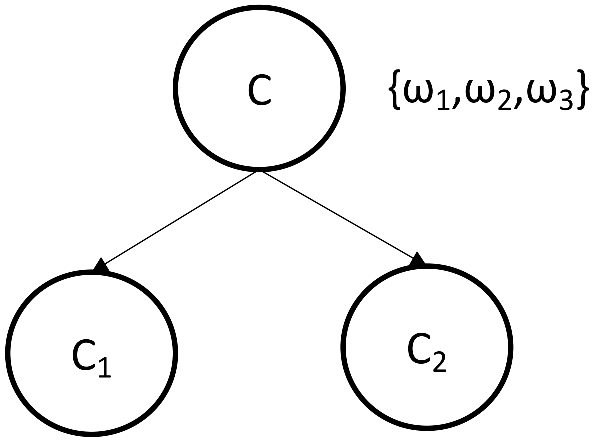
TWOING CRITERION

In multiclass binary tree creation, the *twoing criterion* may be useful.* The overall goal is to find the split that best splits *groups* of the $c$ categories, i.e., a candidate "supercategory" $\mathcal{C}_1$ consisting of all patterns in some subset of the categories, and candidate "supercategory" $\mathcal{C}_2$ as all remaining patterns. Let the class of categories be $\mathcal{C} = \{\omega_1, \omega_2, \ldots, \omega_c\}$. At each node, the decision splits the categories into $\mathcal{C}_1 = \{\omega_{i_1}, \omega_{i_2}, \ldots, \omega_{i_k}\}$ and $\mathcal{C}_2 = \mathcal{C} - \mathcal{C}_1$. For every candidate split $s$, we compute a change in impurity $\Delta i(s, \mathcal{C}_1)$ as though it corresponded to a standard two-class problem. That is, we find the split $s^*(\mathcal{C}_1)$ that maximizes the change in impurity. Finally, we find the supercategory $\mathcal{C}_1^*$ which maximizes $\Delta i(s^*(\mathcal{C}_1), \mathcal{C}_1)$. The benefit of this impurity is that it is *strategic* — it may learn the largest scale structure of the overall problem

* From Duda et al. 's book

# A 3-class toy example

- *"Let the class of categories be $C = \{\omega_1, \omega_2, \ldots, \omega_c\}$. At each node, the decision splits the categories into $C_1 = \{\omega_{i1}, \omega_{i2}, \ldots, \omega_{ik}\}$ and $C_2 = C - C_1$. "*




Possible super category divisions (possible $C_1$ $C_2$)?

# A 3-class toy example *(cont.)*

- Now only consider division $C_1 = \omega_1$ & $C_2 = \omega_2,\omega_3$
- Assume each feature only has a fixed number (i.e. a scalar) of outcomes
  - E.g. These c=3 classes has d=5 features, each feature only has 3 outcomes (-1, 0, 1)
  - How many possible splits? (Hint: you can split between every outcome of every feature)

# A 3-class toy example *(cont.) – step 1*

- *"For every candidate splits, we compute a change in impurity $\Delta i(s, C_1)$ as though it corresponded to a standard two-class problem. That is, we find the split $s^*(C_1)$ that maximizes the change in impurity."*

- From above page, we got 5*2 possible splits. Now we consider a single candidate split **s**:
  - Calculate entropy impurity for both parent node C and children nodes $C_1$ & $C_2$
    - [Board]
    - What if the split is perfect? (split all node belong to $\omega_1$ to one side, and $\omega_2$, $\omega_3$ to another)
  - get the entropy drop $\Delta i(\mathbf{s}, C_1)$ for THIS PARTICULAR **s**
    - If **s** is a perfect split, its entropy drop will be the largest among all 10 s! (children's entropy = 0)
  - Choose the s with the largest entropy drop
    - Prefer perfect splits!

# Reference for entropy impurity & impurity change

ENTROPY IMPURITY

The most popular measure is the *entropy impurity* (or occasionally *information impurity*):

$$i(N) = -\sum_j P(\omega_j) \log_2 P(\omega_j), \tag{1}$$

where $P(\omega_j)$ is the fraction of patterns at node $N$ that are in category $\omega_j$.* By the well-known properties of entropy, if all the patterns are of the same category, the impurity is 0; otherwise it is positive, with the greatest value occuring when the different classes are equally likely.

We now come to the key question — given a partial tree down to node $N$, what value $s$ should we choose for the property test $T$? An obvious heuristic is to choose the test that decreases the impurity as much as possible. The drop in impurity is defined by

$$\Delta i(N) = i(N) - P_L i(N_L) - (1 - P_L) i(N_R), \tag{5}$$

where $N_L$ and $N_R$ are the left and right descendent nodes, $i(N_L)$ and $i(N_R)$ their impurities, and $P_L$ is the fraction of patterns at node $N$ that will go to $N_L$ when property test $T$ is used. Then the "best" test value $s$ is the choice for $T$ that maximizes $\Delta i(T)$.

# A 3-class toy example *(cont.) – step 2*

- *“Finally, we find the supercategory $C_1^*$ which maximizes $\Delta i(s^*(C_1),C_1)$. ”*

- Now we finished calculation for division $C_1 = \omega_1$ & $C_2 = \omega_2,\omega_3$ ..
  - With its best split (largest impurity change) $s^*(C_1 = \omega_1)$
- Re-do above for division $C_1 = \omega_2$ & $C_2 = \omega_1,\omega_3$ ; $C_1 = \omega_3$ & $C_2 = \omega_1,\omega_3$
  - Gathered 3 best splits $s^*(C_1 = \omega_1)$, $s^*(C_1 = \omega_2)$, $s^*(C_1 = \omega_3)$ [one for each division]
  - Compare the impurity change of these 3 splits. Pick the largest one
  - Suppose $s^*(C_1 = \omega_2)$ is the largest among these 3
    - **Pick $C_1 = \omega_2$ & $C_2 = \omega_1,\omega_3$ as the supercategory division**
    - **Pick $s^*(C_1 = \omega_2)$ as the node split**

# Summary & clarification

- A nested problem
  - Outer loop → supercategory division
  - Inner loop → Finding best split for each division

- Regarding time complexity
  - Need big O notation, instead of saying “it’s complex..”