MULTICLASS CONTINUED AND RANKING

David Kauchak
CS 451 – Fall 2013

Admin

Assignment 4

Course feedback

Midterm

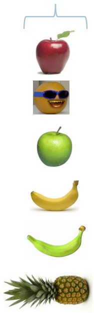
Java tip for the day

private vs. public vs. protected

Debugging tips

Multiclass classification

examples



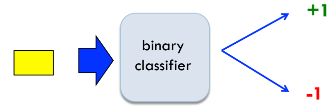
label
apple
orange
apple
banana
banana
pineapple

Same setup where we have a set of features for each example

Rather than just two labels, now have 3 or more

Black box approach to multiclass

Abstraction: we have a generic binary classifier, how can we use it to solve our new problem



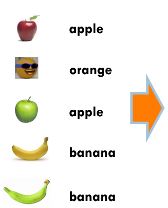
optionally: also output a confidence/score

Can we solve our multiclass problem with this?

Approach 1: One vs. all (OVA)

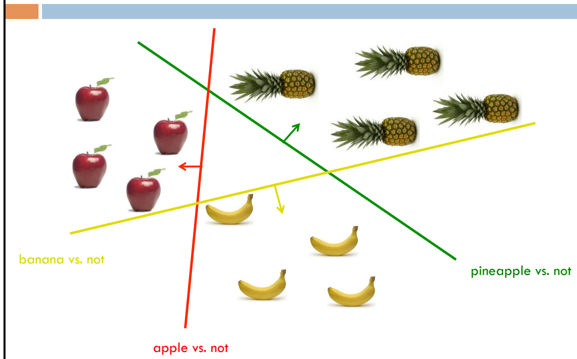
Training: for each label *L*, pose as a binary problem

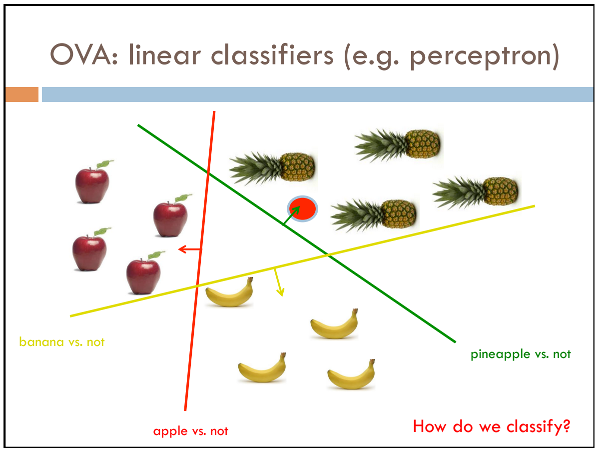
- all examples with label *L* are positive
- all other examples are negative

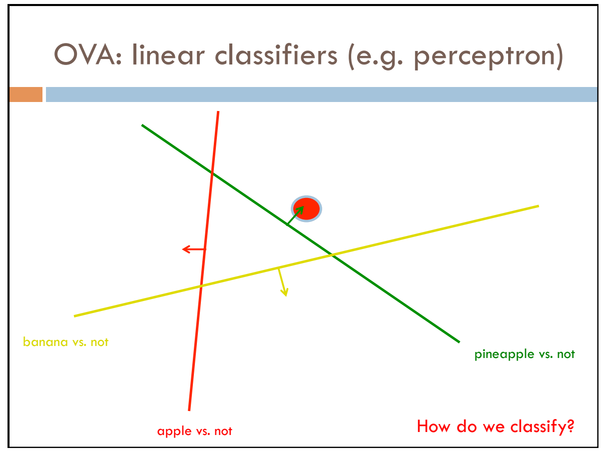


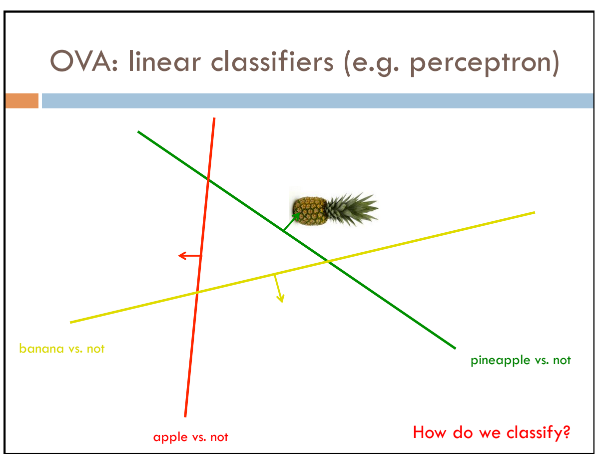
	apple vs. not	orange vs. not	banana vs. not
apple	+1	-1	-1
orange	-1	+1	-1
apple	+1	-1	-1
banana	-1	-1	+1
banana	-1	-1	+1

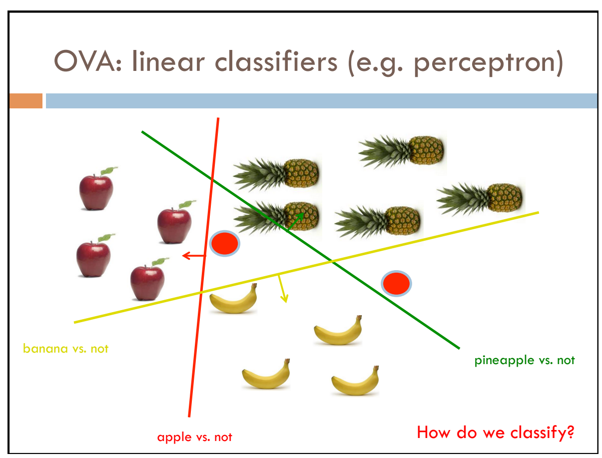
OVA: linear classifiers (e.g. perceptron)



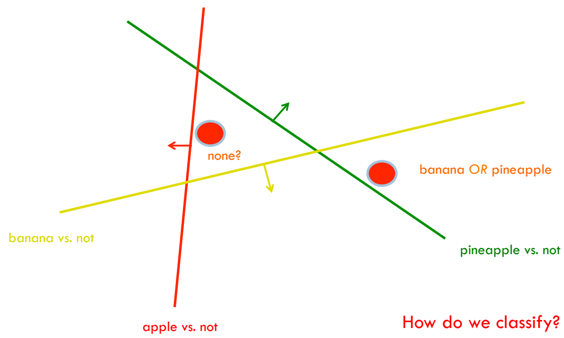

OVA: linear classifiers (e.g. perceptron)
banana vs. not
pineapple vs. not
apple vs. not
How do we classify?


OVA: linear classifiers (e.g. perceptron)
banana vs. not
pineapple vs. not
apple vs. not
How do we classify?


OVA: linear classifiers (e.g. perceptron)
banana vs. not
pineapple vs. not
apple vs. not
How do we classify?

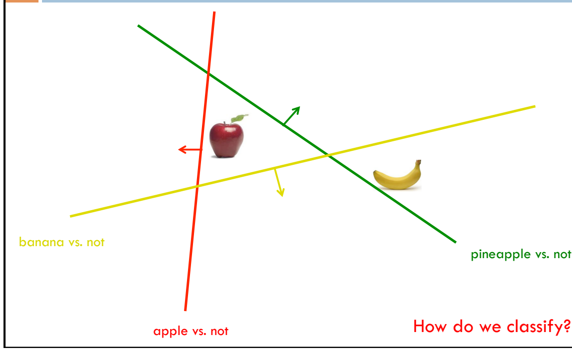

OVA: linear classifiers (e.g. perceptron)
banana vs. not
pineapple vs. not
apple vs. not
How do we classify?

OVA: linear classifiers (e.g. perceptron)



How do we classify?

OVA: linear classifiers (e.g. perceptron)



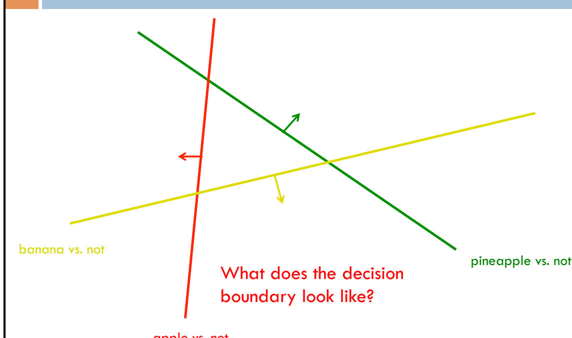
How do we classify?

OVA: classify

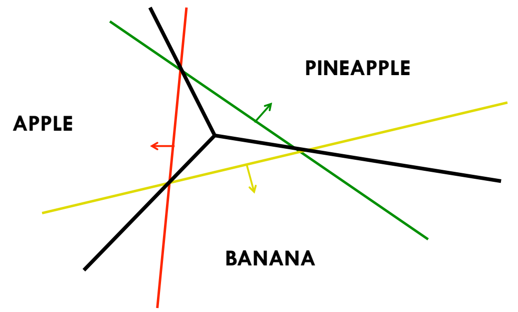
Classify:

- If classifier doesn't provide confidence (this is rare) and there is ambiguity, pick one of the ones in conflict
- Otherwise:
  - pick the most confident positive
  - if none vote positive, pick *least* confident negative

OVA: linear classifiers (e.g. perceptron)



OVA: linear classifiers (e.g. perceptron)



OVA: classify, perceptron

Classify:

- If classifier doesn't provide confidence (this is rare) and there is ambiguity, pick majority in conflict
- Otherwise:
  - pick the most confident positive
  - if none vote positive, pick *least* confident negative

How do we calculate this for the perceptron?

OVA: classify, perceptron

Classify:

- If classifier doesn't provide confidence (this is rare) and there is ambiguity, pick majority in conflict
- Otherwise:
  - pick the most confident positive
  - if none vote positive, pick *least* confident negative

$$prediction = b + \sum_{i=1}^{n} w_i f_i$$

Distance from the hyperplane

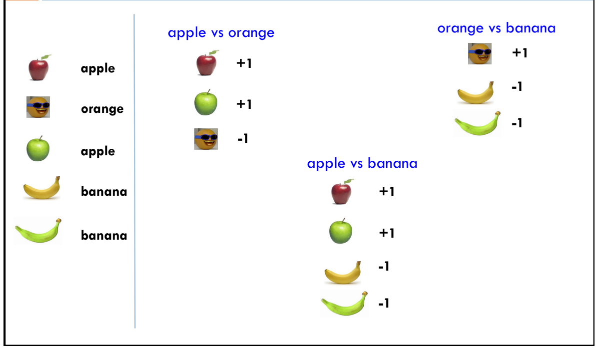
Approach 2: All vs. all (AVA)

Training:

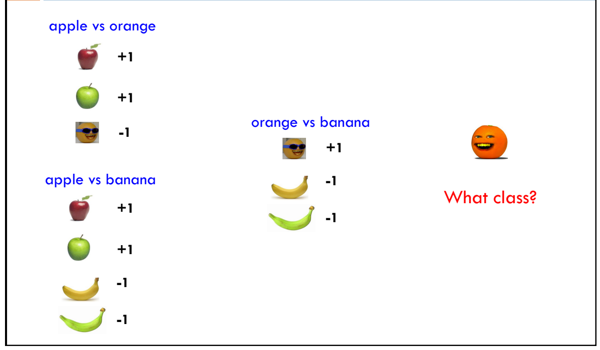
For each pair of labels, train a classifier to distinguish between them

for i = 1 to number of labels:
  for k = i+1 to number of labels:
    train a classifier to distinguish between $label_j$ and $label_k$:
      - create a dataset with all examples *with* $label_j$ labeled positive and all examples with $label_k$ labeled negative
      - train classifier on this subset of the data

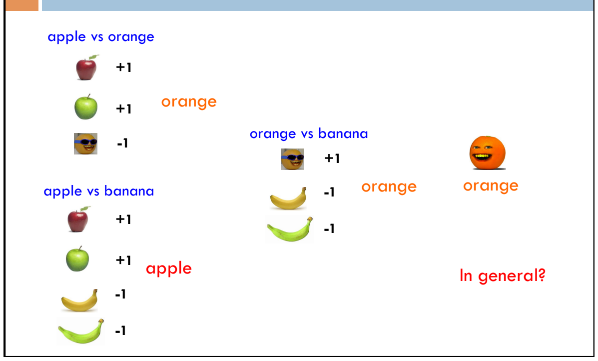
AVA training visualized



AVA classify



AVA classify



AVA classify

To classify example e, classify with each classifier f_{jk}

We have a few options to choose the final class:

- Take a majority vote
- Take a weighted vote based on confidence
  - $y = f_{jk}(e)$
  - $score_j$ += y
  - $score_k$ -= y

How does this work?

*Here we're assuming that y encompasses both the prediction (+1,-1) and the confidence, i.e. y = prediction * confidence.*

AVA classify

Take a weighted vote based on confidence

- $y = f_{jk}(e)$
- $score_j$ += y
- $score_k$ -= y

If y is positive, classifier thought it was of type j:
- raise the score for j
- lower the score for k

if y is negative, classifier thought it was of type k:
- lower the score for j
- raise the score for k

OVA vs. AVA

Train/classify runtime?

Error? Assume each binary classifier makes an error with probability ε

OVA vs. AVA

Train time:

AVA learns more classifiers, however, they're trained on much smaller data this tends to make it faster if the labels are equally balanced

Test time:

AVA has more classifiers

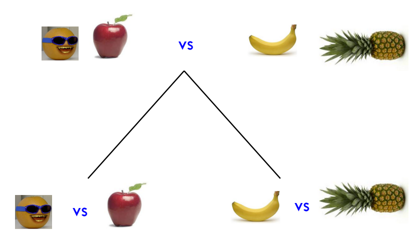
Error (see the book for more justification):
- AVA trains on more balanced data sets
- AVA tests with more classifiers and therefore has more chances for errors

- Theoretically:

-- OVA: ε (number of labels -1)

-- AVA: 2 ε (number of labels -1)

Approach 3: Divide and conquer



Pros/cons vs. AVA?

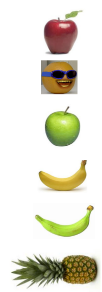
Multiclass summary

If using a binary classifier, the most common thing to do is OVA

Otherwise, use a classifier that allows for multiple labels:

- DT and k-NN work reasonably well
- We'll see a few more in the coming weeks that will often work better

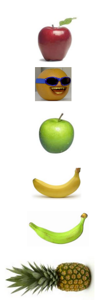
Multiclass evaluation



label	prediction
apple	orange
orange	orange
apple	apple
banana	pineapple
banana	banana
pineapple	pineapple

How should we evaluate?

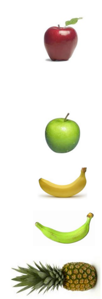
Multiclass evaluation



label	prediction
apple	orange
orange	orange
apple	apple
banana	pineapple
banana	banana
pineapple	pineapple

Accuracy: 4/6

Multiclass evaluation imbalanced data



label	prediction
apple	orange
…	
apple	apple
banana	pineapple
banana	banana
pineapple	pineapple

Any problems?

Data imbalance!

Macroaveraging vs. microaveraging

microaveraging: average over examples (this is the "normal" way of calculating)

macroaveraging: calculate evaluation score (e.g. accuracy) for each label, then average over labels

What effect does this have?
Why include it?

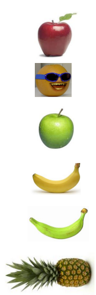
Macroaveraging vs. microaveraging

microaveraging: average over examples (this is the "normal" way of calculating)

macroaveraging: calculate evaluation score (e.g. accuracy) for each label, then average over labels

- Puts more weight/emphasis on rarer labels
- Allows another dimension of analysis

Macroaveraging vs. microaveraging

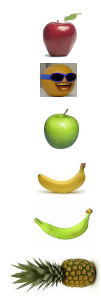


label	prediction
apple	orange
orange	orange
apple	apple
banana	pineapple
banana	banana
pineapple	pineapple

microaveraging: average over examples

macroaveraging: calculate evaluation score (e.g. accuracy) for each label, then average over labels

Macroaveraging vs. microaveraging



label	prediction
apple	orange
orange	orange
apple	apple
banana	pineapple
banana	banana
pineapple	pineapple

microaveraging: 4/6

macroaveraging:

apple = 1/2
orange = 1/1
banana = 1/2
pineapple = 1/1
total = (1/2 + 1 + 1/2 + 1)/4
= 3/4

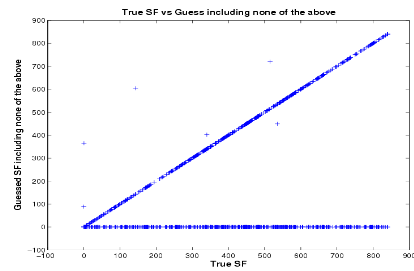
Confusion matrix

entry *(i, j)* represents the number of examples with label *i* that were predicted to have label *j*

another way to understand both the data and the classifier

	Classic	Country	Disco	Hiphop	Jazz	Rock
Classic	86	2	0	4	18	1
Country	1	57	5	1	12	13
Disco	0	6	55	4	0	5
Hiphop	0	15	28	90	4	18
Jazz	7	1	0	0	37	12
Rock	6	19	11	0	27	48

Confusion matrix



BLAST classification of proteins in 850 superfamilies

Multilabel vs. multiclass classification

- Is it edible?
- Is it sweet?
- Is it a fruit?
- Is it a banana?

Is it a banana?
Is it an apple?
Is it an orange?
Is it a pineapple?

Is it a banana?
Is it yellow?
Is it sweet?
Is it round?

Any difference in these labels/categories?

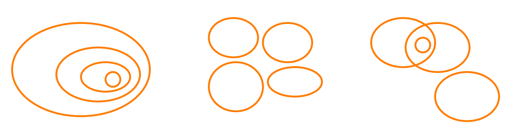
Multilabel vs. multiclass classification

- Is it edible?
- Is it sweet?
- Is it a fruit?
- Is it a banana?

Is it a banana?
Is it an apple?
Is it an orange?
Is it a pineapple?

Is it a banana?
Is it yellow?
Is it sweet?
Is it round?

Different structures



Nested/ Hierarchical — Exclusive/ Multiclass — General/Structured

Multiclass vs. multilabel

Multiclass: each example has one label and exactly one label

Multilabel: each example has ***zero or more*** labels. Also called annotation

Multilabel applications?

Multilabel

Image annotation

Document topics

Labeling people in a picture

Medical diagnosis

Multiclass vs. multilabel

Multiclass: each example has one label and exactly one label

Multilabel: each example has ***zero or more*** labels. Also called annotation

Which of our approaches work for multilabel?