Multiclass Classification

Instructor: Jessica Wu -- Harvey Mudd College

The instructor gratefully acknowledges Eric Eaton (UPenn), David Kauchak (Pomona), Tommi Jaakola (MIT) and the many others who made their course materials freely available online.

Multiclass Classification

Learning Goals

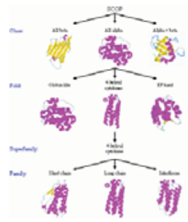
- Describe common strategies for multiclass classification and the trade-offs of each
  - one-versus-all, one-versus-one
  - black-box approach using output codes
- Describe how to evaluate multiclass problems
  - micro-averaging, macro-averaging
- Describe the difference between multiclass and multilabel problems (extra)

Setup

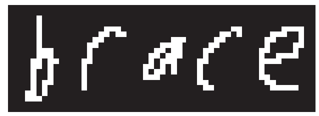
Most classification problems are multiclass (more than 2 labels).

Can you name some real-world examples?

document classification

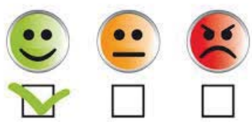


protein classification



handwriting recognition

face recognition



sentiment analysis

autonomous vehicles

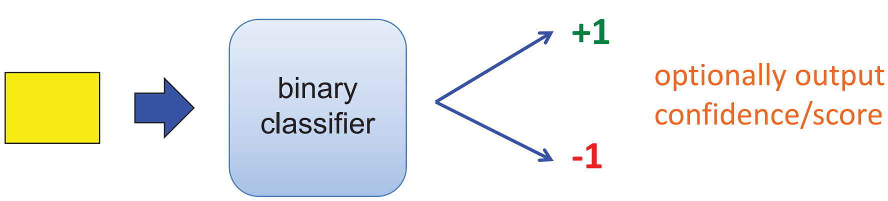
emotion recognition

Which of our current classifiers work out-of-the-box?

Based on slide by David Kauchak

Black-Box Approach

Abstraction: We have a generic binary classifier.



Can we use it to solve a multiclass problem?

Consider simple three-class classification task.

Based on slide by David Kauchak

Approach 1: One-versus-All (OVA)

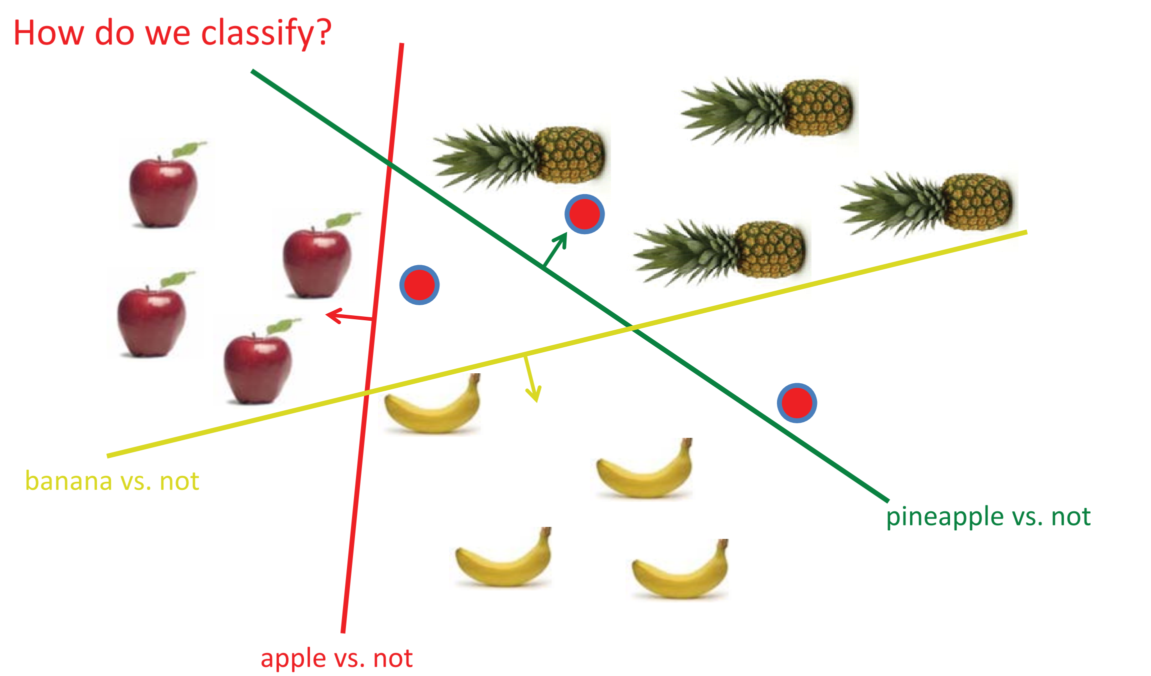
For each class, pose a binary classification task.

- All examples of this class 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**

Based on slide by David Kauchak

OVA with Linear Classifiers



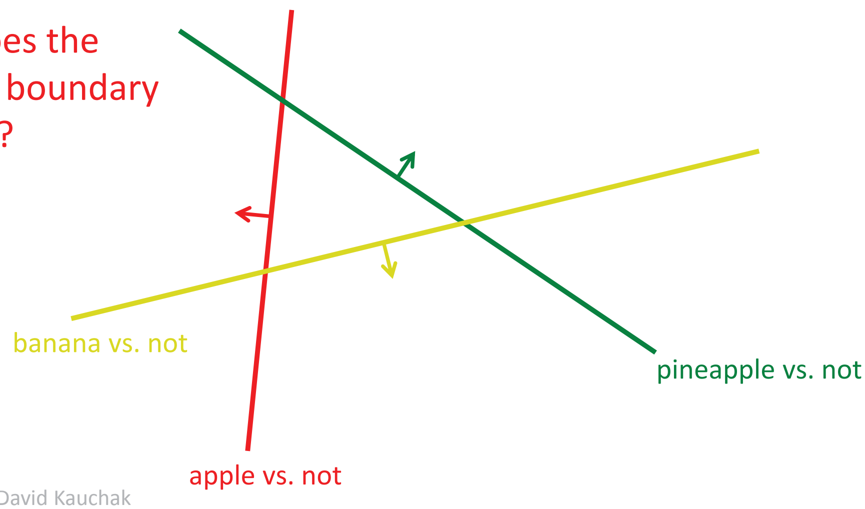
Based on slide by David Kauchak

OVA: Classify

- If classifier does not provide confidence (this is rare) and there is ambiguity, pick one of the ones in conflict.
- Else
  - Pick the most confident positive.
  - If none vote positive, pick least confident negative.

e.g. as measured by distance to hyperplane

What does the decision boundary look like?



Based on slide by David Kauchak

Generalizing to Output Codes

Let $y = \{1,2,3\}$.

To turn a m-class classification task into m binary classification tasks:

Each column defines how classes are translated to binary classes in each component task.

$$R = \begin{array}{c} \begin{array}{ccc} \text{task 1} & \text{task 2} & \text{task 3} \end{array} \\ \begin{bmatrix} +1 & -1 & -1 \\ -1 & +1 & -1 \\ -1 & -1 & +1 \end{bmatrix} \end{array} \begin{array}{l} \\ y = 1 \\ y = 2 \\ y = 3 \end{array}$$

Each row corresponds to one of original classes.

Ex: To solve task 3, any example with class $y = 2$ has target binary class $R(2,3) = -1$.

Based on notes by Tommi Jaakola

Training

Separately train classifiers $h_1(\boldsymbol{x})$, $h_2(\boldsymbol{x})$, $h_3(\boldsymbol{x})$ to solve each associated binary task.

If $\left\{\left(\boldsymbol{x}^{(i)}, y^{(i)}\right)\right\}_{i=1}^{n}$ is original 3-class training set, then $h_1(\boldsymbol{x})$ is trained with $\left\{\left(\boldsymbol{x}^{(i)}, y_1^{(i)}\right)\right\}_{i=1}^{n}$, where

$$y_1^{(i)} = R\left(y^{(i)}, 1\right)$$

so that $y_1^{(i)}$ corresponds to first column of R.

Based on notes by Tommi Jaakola

Prediction

Combine outputs of trained binary classifiers into full multiclass classifier.

Clearly, prediction of original labels has to be based on output code R. If $h_i(\boldsymbol{x}) \in \{+1, -1\}$, we can simply predict class label that best agrees binary predictions:

$$\hat{y} = \arg\max_{y \in \{1,2,3\}} \sum_{j=1}^{k} \boxed{R(y, j) h_j(\boldsymbol{x})}$$

each product determines whether binary task j agrees or disagrees with possible label y

where $k = 3$ is the number of binary tasks.

Notes:

- If $R(y,j)$ and $h_j(\boldsymbol{x})$ match (in sign), then $h_j(\boldsymbol{x})$ agrees with predicting (multiclass) y.
- If each component classifier $h_j(\boldsymbol{x})$ predicts $R(y,j)$ consistent with true label y, then above sum will be maximized for that y.

Based on notes by Tommi Jaakola

Example

$$R = \begin{bmatrix} +1 & -1 & -1 \\ -1 & +1 & -1 \\ -1 & -1 & +1 \end{bmatrix}$$

(OVA code)

$$h_1(\boldsymbol{x}) = +1,\ h_2(\boldsymbol{x}) = -1,\ h_3(\boldsymbol{x}) = -1$$

$$\hat{y} = \arg\max_{y \in \{1,2,3\}} \sum_{j=1}^{k} R(y,j) h_j(\boldsymbol{x})$$

What is predicted label $\hat{y}$?

Extensions

Problems

- Classifier outputs may be contradictory.
  Ex: If $h_1(\boldsymbol{x}) = +1$, $h_2(\boldsymbol{x}) = -1$, $h_3(\boldsymbol{x}) = +1$, then predict $y = 1$ or $y = 3$.
- Using $h_i(\boldsymbol{x}) \in \{+1, -1\}$ omits how strongly each classifier insists on its binary label.

Solutions

- Use discriminant function values

$$h_i(\boldsymbol{x}) = \boldsymbol{\theta}^T \boldsymbol{x}$$

- Use loss function, then predict class most consistent with classifiers (minimum loss):

$$\hat{y} = \arg\min_{y \in \{1,2,3\}} \sum_{j=1}^{k} \text{Loss}\left(R(y,j) h_j(\boldsymbol{x})\right)$$

Note:
$L(y, \hat{y}) = L(z)$
where $z = y\hat{y}$

  Loss function measures how poorly discriminant function matches particular binary class [explore in homework].

Problems with OVA?

What kind of training set will be difficult for OVA?

One-versus-One (All-versus-All)

$$
R = \begin{array}{cc}
\begin{array}{ccc} \text{task 1} & \text{task 2} & \text{task 3} \end{array} & \\
\begin{bmatrix} +1 & +1 & 0 \\ -1 & 0 & +1 \\ 0 & -1 & -1 \end{bmatrix} & \begin{array}{c} y = 1 \\ y = 2 \\ y = 3 \end{array}
\end{array}
$$

0 in output code indicates examples not part of binary classification task.
Ex. class 3 excluded from task 1.

Based on notes by Tommi Jaakola

Comparing Output Codes

Some output codes require more binary tasks

OVA: OVO:

Some binary tasks require larger training set

OVA: OVO:

Some binary tasks are harder to solve than others

OVA: OVO:

If using binary classifiers, OVA is the most common.

`scikit-learn`: OVA for all linear models except SVC

OVO for SVC

Based on notes by Tommi Jaakola

Minimal Output Code?

$$R = \begin{bmatrix} +1 & +1 \\ -1 & +1 \\ +1 & -1 \end{bmatrix}$$

Minimal number of binary tasks but …

- one task could be hard, leading to poorly performing classifier
- with only two tasks, a single poorly performing classifier degrades performance considerably

With more binary classifiers

- multiclass classifier has chance to “error correct”

Based on notes by Tommi Jaakola

Multiclass Evaluation

	class	prediction
	apple	orange
	orange	orange
	apple	apple
	banana	pineapple
	banana	banana
	pineapple	pineapple

How should we evaluate?

recall = (TP) / (TP + FN)

Problems?

Based on slide by David Kauchak

Micro- and Macro-Averaging

	class	prediction
	apple	orange
	orange	orange
	apple	apple
	banana	pineapple
	banana	banana
	pineapple	pineapple

recall

apple = 1/2 orange = 1/1

banana = 1/2 pineapple = 1/1

Micro-averaging

- average over examples
- "normal" way of calculating

recall = 4/6

Macro-averaging

- calculate metric for each class, then average over classes
- put more emphasis on rarer classes

recall = 3/4

Based on slide by David Kauchak

Confusion Matrices

prediction

actual

	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

entry (i, j) represents number of examples of class i that were predicted to be class j

Based on slide by David Kauchak

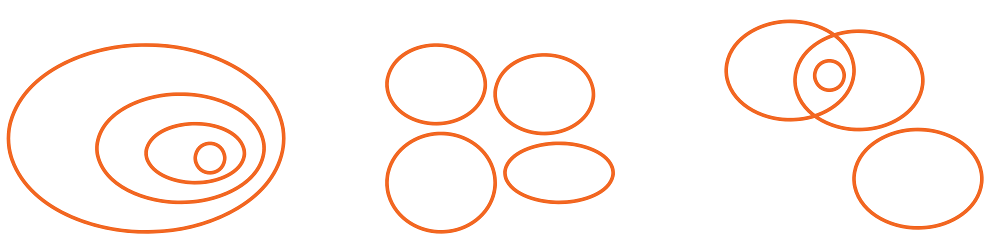
Multilabel Classification

- 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?

Differences?



nested / hierarchical

exclusive/ multiclass

general / structured

Based on slide by David Kauchak

Multiclass vs Multilabel

Multiclass

each example has **one label and exactly one label**

Multilabel (also called annotation)

each example has **zero or more labels**

Multilabel applications?

- image annotation
- document topics
- medical diagnosis

Based on slide by David Kauchak