Multiclass Classifier Design Methods

Two class problems are called ***binary*** classification tasks. If there are 3 or more classes we deal with ***multiclass*** classification problems. There are two major approaches in solving classification (pattern recognition) problems faced with more than two classes in data.

- **One-against-all** i.e., **1-vs-All classification**, in which there is one binary classifier for each class to separate members of that class from members of other classes.
  - If there are C classes one **must design *C* classifiers**!
  - and, for l data (samples) each classifier is designed by using all the l data (samples)
- **One-against-One** i.e., **1-vs-1** i.e., **Pairwise classification** in which there is one binary classifier for each pair of classes to separate members of one class from members of the other.
  - If there are C classes one must **design $C(C-1)/2$** classifiers!
  - and, for l data (samples) assuming the balanced dataset (same number of samples l/C in each class) each classifier is designed by using only $2l/C$ data (samples)

One-Against-All Methods, or One-vs-All, or 1vA

One-against-all (OAA) classifiers were first introduced by Vladimir Vapnik in 1995. The initial formulation of the one-against-all method required unanimity among all classifiers: a data point would be classified under a certain class if and only if that class's classifier accepted it and all other classes' classifiers rejected it. While accurate for tightly clustered classes, this method leaves regions of the feature space undecided where more than one class accepts or, all classes reject.

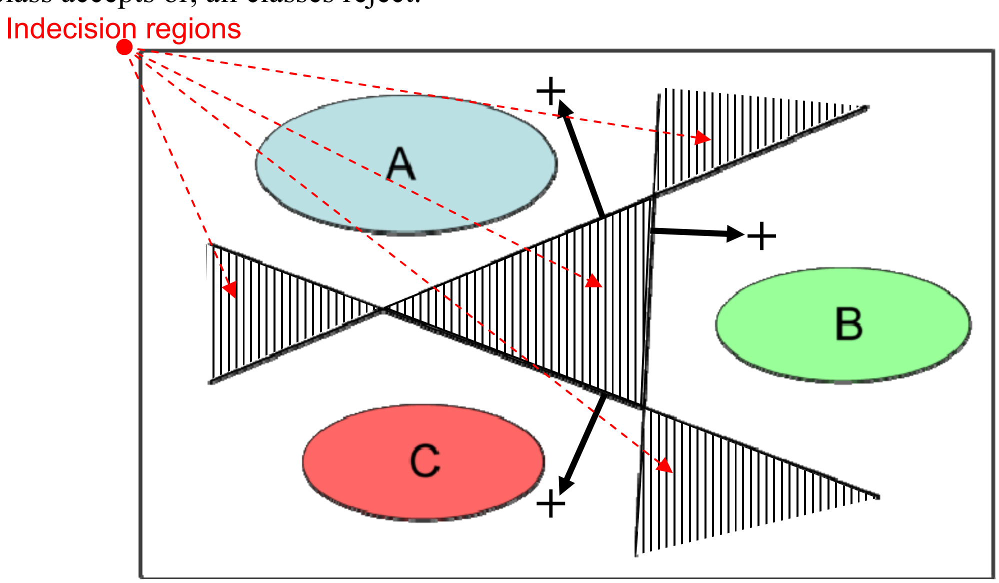


Figure 1: Diagram of binary OAA region boundaries on a basic problem

One method for improving the performance of OAA classifiers was suggested by Vapnik in 1998. It involves **using** the **continuous values of classifier decision functions** rather than simply their signs. **The class of a data point is whichever class has a decision**

function with highest value, regardless of sign (use the MAX operator). This approach and the 1-vs-1 are the most common methods for multiclass classifiers today.

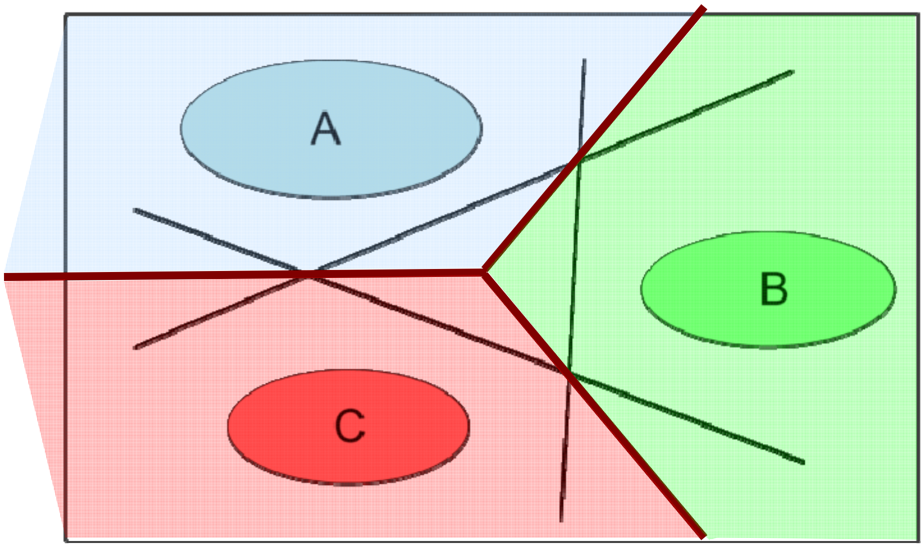
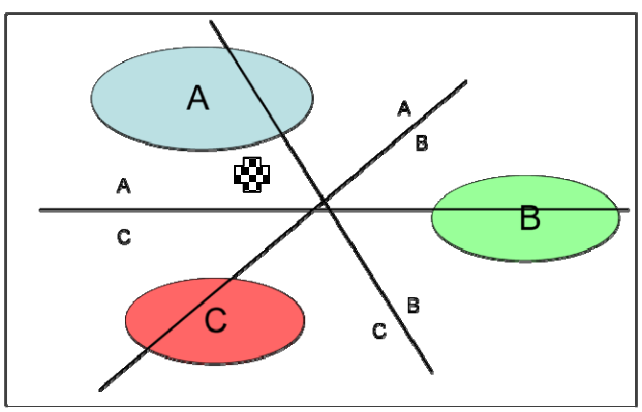


Figure 2: Diagram of continuous OAA (i.e. OvsAll)region boundaries on a basic problem after MAX operation, which is aka the Winner-Takes-All (WTA)

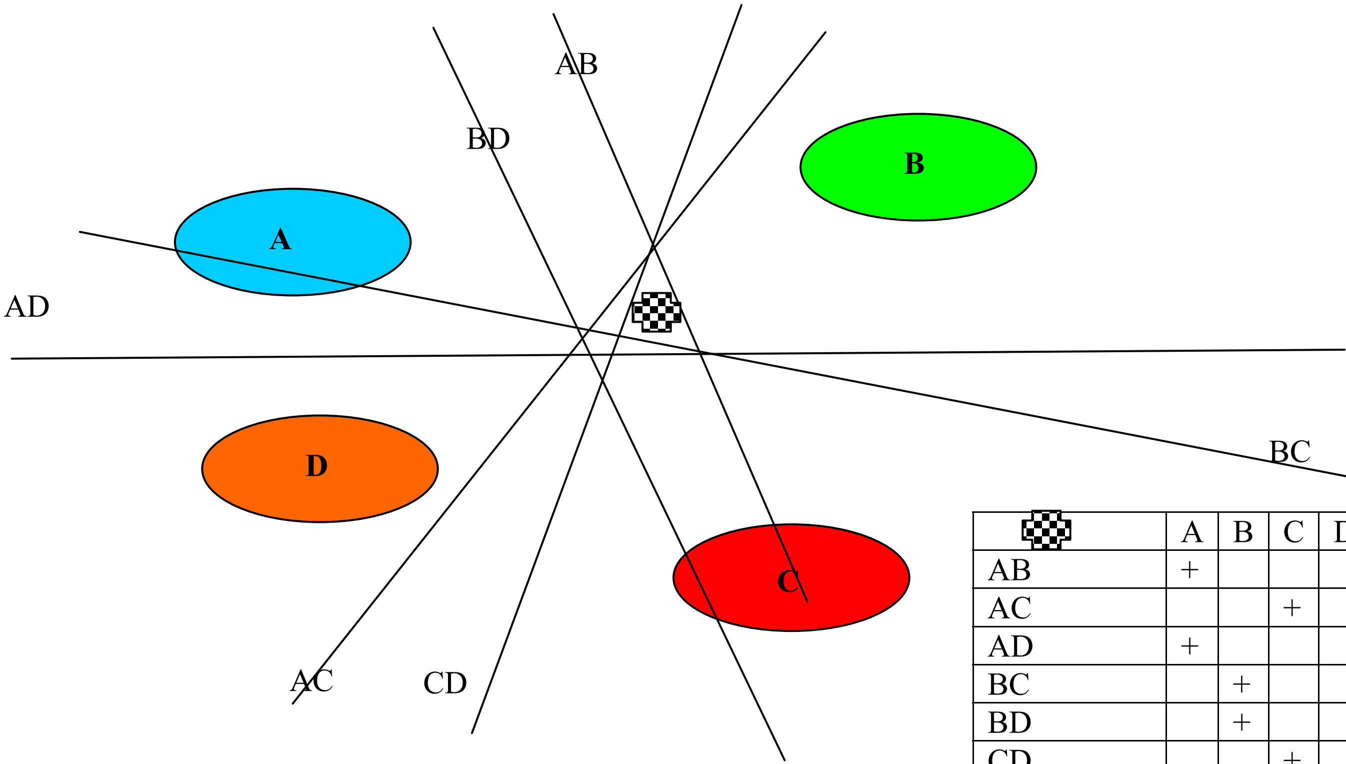
Pairwise Methods or One-vs-One, or 1v1

In pairwise classifiers, there is **one classifier for each pair of classes** trained to separate the data from each. **The simplest form of classification with pairwise classifiers selects the class chosen by the maximal number of pairwise classifiers**; more advanced methods include using decision graphs to **determine the class selected based on the continuous classifier's output values.**



	A	B	C
AB	+		
AC	+		
BC			+
TOTAL	2	0	1
belongs to	**A**		

Figure 2: Diagram of pairwise classifiers boundaries on a basic problem



	A	B	C	D
AB	+			
AC			+	
AD	+			
BC		+		
BD		+		
CD			+	
TOTAL	2	2	2	
belongs to	?	?	?	

So, we have indecisions here !

A determination of the class selected based on the continuous classifier's output values.

		A	B	C	D
	AB	**0.5**			
	AC			**0.8**	
	AD	0.6			
	BC		**0.7**		
	BD		1.3		
	CD			0.5	
First way:	TOTAL	1.1	2.0	1.3	
	belongs to		**B**		
		0.5	**0.7**	**0.8**	
	belongs to			**C**	

Second version to resolve the ties, is looking at the tied classifiers only. Eliminate all classifiers involving D class

2nd version is not quite good? Is it?

There are no significant differences in <u>accuracies</u> achieved between 1-vs-All and 1-vs-1 approach!

Note, however, that in terms of the computational time required and despite the fact that 1-vs-1 method must design $C(C-1)/2$ classifiers, 1-vs-1 training is often faster because each of the model uses much smaller number of training data than the 1-vs-All does. For balanced datasets 1-vs-1 uses only $2l/C$ samples for the training while 1-vs-All uses all l samples for each classifier.

The following math for balanced data might help in understanding the time complexity:

1-vs-All uses l data for C classifiers making $\boldsymbol{Cl}$ data participating in the training, while
1-vs-1 uses $C(C-1)/2 * 2l/C = \mathbf{(C - 1)}\boldsymbol{l}$ data participating in the training.