Machine learning

Multi-class Classifiers

Hamid Beigy

Sharif University of Technology

December 10, 2021

Table of contents

Introduction

- In classification, the goal is to find a mapping from inputs X to outputs $t \in \{1, 2, \ldots, C\}$ given a labeled set of input-output pairs.
- We can extend the binary classifiers to C class classification problems or use multiple binary classifiers.
- For C-class, we have four extensions for using binary classifiers.

  Single $C-$class discriminant
  One-against-all
  One-against-one
  Hierarchical classification
  Error correcting coding

$C-$class discriminant function

$C-$class discriminant function

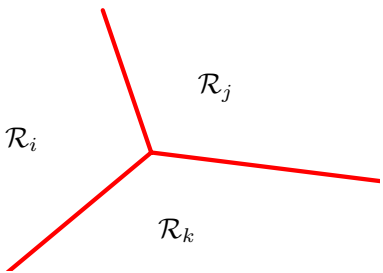
- We can consider a single $C-$class discriminant comprising C linear functions of the form

$$g_k(x) = w_k^T x + w_{k0}$$

- Then assigning a point x to class C_k if $g_k(x) > g_j(x)$ for all $j \neq k$.
- The decision boundary between class C_k and class C_j is given by $g_k(x) = g_j(x)$ and corresponds to hyperplane

$$(w_k - w_j)^T x + (w_{k0} - w_{j0}) = 0$$

- This has the same form as decision boundary for the two-class case.

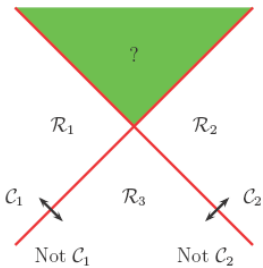


One-against-all classification

One-against-all classification

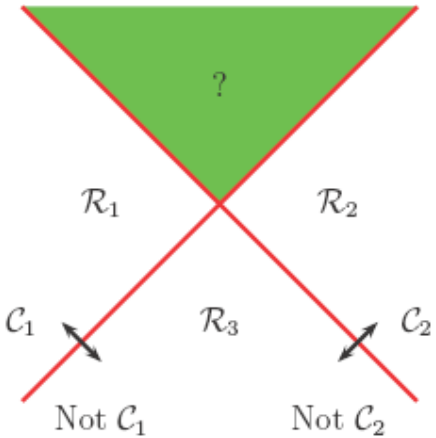
- This extension is to consider a set of C two-class problems.
- For each class, we seek to design an optimal discriminant function, $g_i(x)$ (for $i = 1, 2, \ldots, C$) so that $g_i(x) > g_j(x)$, $\forall j \neq i$, if $x \in C_i$.
- Adopting the SVM methodology, we can design the discriminant functions so that $g_i(x) = 0$ to be the optimal hyperplane separating class C_i from all the others. Thus, each classifier is designed to give $g_i(x) > 0$ for $x \in C_i$ and $g_i(x) < 0$ otherwise.
- Classification is then achieved according to the following rule:

$$\text{Assign } x \text{ to class } C_i \text{ if } \quad i = \underset{k}{argmax} g_k(x)$$



Properties of one-against-all classification

- The number of classifiers equals to C.
- Each binary classifier deals with a rather asymmetric problem in the sense that training is carried out with many more negative than positive examples. This becomes more serious when the number of classes is relatively large.
- This technique, however,may lead to indeterminate regions, where more than one $g_i(x)$ is positive



Properties of one-against-all classification

- The implementation of OVA is easy.
- It is not robust to errors of classifiers. If a classifier make a mistake, it is possible that the entire prediction is errorneous.

Theorem (OVA error bound)

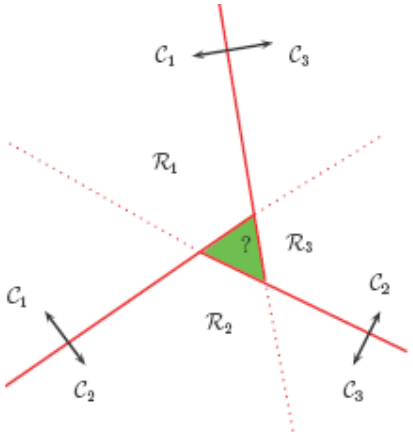
Suppose the average binary error of C binary classifiers is ϵ. Then the error rate of the OVA multi–class classifier is at most $(C-1)\epsilon$.

- Please prove the above theorem.

One-against-one classification

One-against-one classification

- In this case, $C(C-1)/2$ binary classifiers are trained and each classifier separates a pair of classes.
- The decision is made on the basis of a majority vote.



- The obvious disadvantage of the technique is that a relatively large number of binary classifiers has to be trained.

One-against-one classification

- This technique, however,may lead to indeterminate regions, where more than one $g_{ij}(x)$ is positive

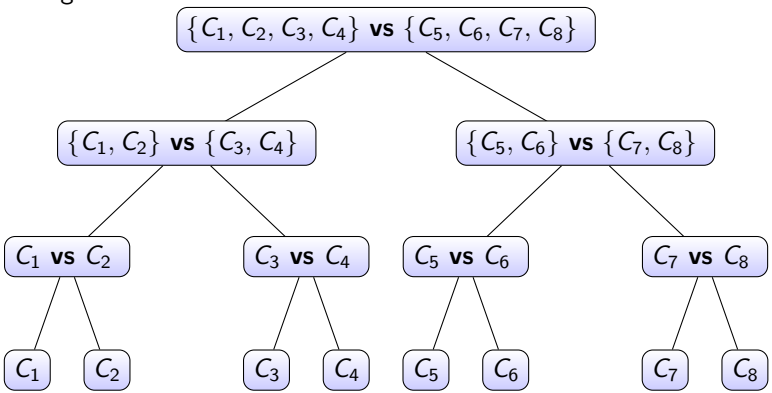
Theorem (AVA error bound)

Suppose the average binary error of the $C(C-1)/2$ *binary classifiers is at most* ϵ*. Then the error rate of the AVA multi–class classifier is at most* $2(C-1)\epsilon$*.*

- Please prove the above theorem.
- The bound for AVA is $2(C-1)\epsilon$ and the bound for OVA is $(C-1)\epsilon$. Does this mean that OVA is neccessarily better than AVA? Why or why not? Please do it as a homework.

Hierarchical classification

- In hierarchical classification, the output space is hierarchically divided i.e. the classes are arranged into a tree.



- One thing to keep in mind with hierarchical classifiers is that you have control over how the tree is defined.
- In OVA and AVA you have no control in the way that classification problems are created.
- In hierarchical classifiers, the only thing that matters is that, at the root, half of the classes are considered positive and half are considered negative.
- You want to split the classes in such a way that this classification decision is as easy as possible.

Theorem (Hierarchical classification error bound)

Suppose the average binary classifiers error is ϵ. Then the error rate of the hierarchical classifier is at most $\lceil \log_2 C \rceil \epsilon$.

- Can you do better than $\lceil \log_2 C \rceil \epsilon$? Yes. Using error-correcting codes.

Error correcting coding classification

Error correcting coding classification

- In this approach, the classification task is treated in the context of error correcting coding.
- For a $C-$class problem a number of, say, L binary classifiers are used,where L is appropriately chosen by the designer.
- Each class is now represented by a binary code word of length L.
- During training of i^{th} classifier, the desired labels are chosen from $\{-1, +1\}$.
- For each class, the desired labels may be different for the various classifiers.
- This is equivalent to constructing a matrix $C \times L$ of desired labels. For example, if $C = 4$ and $L = 6$, such a matrix can be

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

Error correcting coding classification (cont.)

- For example, if $C = 4$ and $L = 6$, such a matrix can be

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

- During training, the first classifier (corresponding to the first column of the previous matrix) is designed in order to respond $(-1, +1, +1, -1)$ for examples of classes C_1, C_2, C_3, C_4, respectively.
- The second classifier will be trained to respond $(-1, -1, +1, -1)$, and so on.
- The procedure is equivalent to grouping the classes into L different pairs, and, for each pair, we train a binary classifier accordingly.
- Each row must be distinct and corresponds to a class.

- When an unknown pattern is presented, the output of each one of the binary classifiers is recorded, resulting in a code word.
- Then,the Hamming distance of this code word is measured against the C code words, and the pattern is classified to the class corresponding to the smallest distance.
- This feature is the power of this technique. If the code words are designed so that the minimum Hamming distance between any pair of them is, say, d, then a correct decision will still be reached even if the decisions of at most $\lfloor \frac{d-1}{2} \rfloor$ out of the L, classifiers are wrong.

Theorem (Error-correcting error bound)

Suppose the average binary classifiers error is ϵ. *Then the error rate of the classifier created using error correcting codes is at most* 2ϵ .

- You can prove a lower bound that states that the best you could possible do is $\frac{\epsilon}{2}$.

Learning with Imbalanced Data

Readings

1. An imbalanced data set is one in which there are too many positive examples and too few negative examples (or vice versa).
2. Examples of imbalanced data set are
   - Fraud detection
   - Intrusion detection
   - Spam detection
3. If we have a good binary classification algorithm, can we use it for imbalanced dataset?
4. To use such a classifier, we use the following transformations of dataset.
   - Sub-sampling
   - Oversampling

Reading

Readings

1. Section 4.1.2 of Pattern Recognition and Machine Learning Book (Bishop 2006).

 Bishop, Christopher M. (2006). *Pattern Recognition and Machine Learning*. Springer-Verlag.

Questions?