Dealing with Imbalanced Classes

Junho Kim

2018-2-13

Imbalanced classes pose a challenge in machine learning. The term imbalance in this context refers to the situation in which the distribution of some particular dataset is extremely skewed. Below are several widely used strategies to cope with imbalanced classes.

1 Collect more data

Obviously, collecting more data would help. However this may be difficult since the data gathering process could often be cumbersome.

2 Change the performance metric

Usually one tries to measure the performance of some classifier by its accuracy. In unbalanced datasets, this might not be the best way to go. We could use some other alternatives mentioned below:

2.1 Precision, recall, and F_β score

Accuracy is a crude measure when it comes to assessing a model's performance. We could deploy other alternative measures such as precision, recall, or F_β score. First of all, precision is defined as follows, where tp stands for true positive and fp stands for false positive:

$$precision = \frac{tp}{tp + fp} \tag{1}$$

Recall is defined in a similar manner, where fn stands for false negative:

$$recall = \frac{tp}{tp + fn} \tag{2}$$

F_β score is a weighted harmonic average of the aforementioned quantities.

$$F_\beta = (1 + \beta^2)\frac{precision * recall}{\beta^2 * precision + recall} \tag{3}$$

These values provide a fine-grained information regarding the performance of the classifier at hand.

2.2 Cohen's Kappa

Cohen's Kappa is a normalized classification accuracy. It is defined as follows:

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{4}$$

p_o and p_e stand for observed agreement probability and hypothetical agreement probability, respectively. For the sake of brevity, let's consider the following example. Suppose that you were analyzing data related to a group of 50 people applying for a grant. Each grant proposal was read by two readers and each reader either said "Yes" or "No" to the proposal. Suppose the disagreement count data were as follows, where A and B are readers: In the above example,

A:Yes B:Yes	A:Yes B:No	A:No B:Yes	A:No B:No
20	5	10	15

we could calculate p_o and p_e as follows:

$$p_o = \frac{20 + 15}{50} = 0.7 \tag{5}$$

$$p_e = 0.5 * 0.6 + 0.5 * 0.4 = 0.5 \tag{6}$$

Now lets take a look at equation 4. If the raters are in complete agreement then $\kappa = 1$. If there is no agreement among the raters other than what would be expected by chance (as given by p_e), $kappa = 0$.

3 Penalize

By deliberately setting penalty parameters, we could wittingly solve the imbalance problem. Let's consider the following example. When we are building support vector machines, we use the following objective function:

$$\frac{\parallel w \parallel^2}{2} + C \sum_{i=1}^{n} \xi_i \tag{7}$$

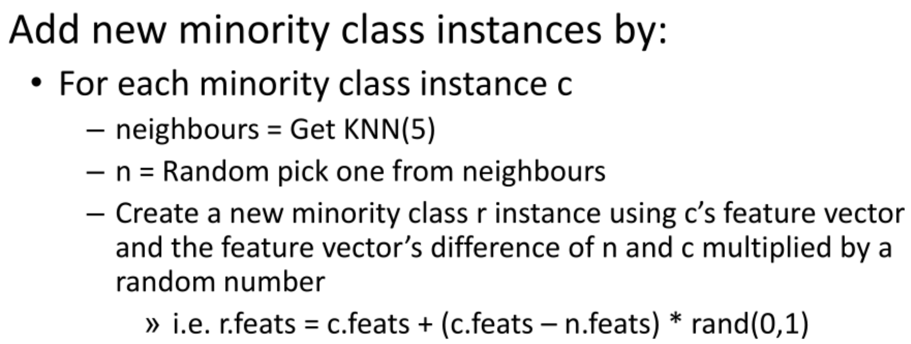
We could modify the above equation by modifying penalty parameters with class imbalance in regard:

$$\frac{\parallel w \parallel^2}{2} + C^+ \sum_{class=positive} \xi^+ + C^- \sum_{class=negative} \xi^- \tag{8}$$

This way we could get a more adaptive solution sensitive of the class imbalance.

4 Smote

Smote is an algorithm that generates synthetic examples from some given dataset. This is done in the following manner:



Add new minority class instances by:

- For each minority class instance c
  - neighbours = Get KNN(5)
  - n = Random pick one from neighbours
  - Create a new minority class r instance using c’s feature vector and the feature vector’s difference of n and c multiplied by a random number
    - » i.e. r.feats = c.feats + (c.feats – n.feats) * rand(0,1)

5 Miscellaneous

We could obviously try under-sampling and over-sampling. Bayesian methods could be deployed as well, since the imbalance imformation could be effectively incorporated in the prior term.