ML: sampling imbalanced dataset leads to selection bias

Asked 4 years, 5 months ago Modified 4 years, 4 months ago Viewed 1k times

5

1

By sampling we make the algorithm think that the prior probabilities of the classes are the same. This seems to affect the predictions as well and therefore the probabilities cannot be interpreted as probabilities anymore and have to be recalibrated.

I am a bit confused on why the equality of the prior class distribution affects the predictions.

Let's assume we have a two-class classification problem with an imbalanced dataset that we oversample to balance the class distribution. We run decision trees on it. The test set is imbalanced but does it really matter? Each sample of the test set just goes through the nodes of the decision trees and it never checks if the sample belongs to the majority or minority class.

So, why does the prior probability of classes affect the prediction of a sample?

machine-learning sampling unbalanced-classes

Share Cite Edit Follow Flag

edited Apr 16, 2018 at 11:14
Satwik Bhattamishra
1,456 9 24

asked Apr 8, 2018 at 10:50
Alina
1,065 3 11 22

1 Answer

Sorted by: Highest score (default)

10

> By sampling we make the algorithm think that the prior probabilities of the classes are the same. This seems to affect the predictions as well and therefore the probabilities cannot be interpreted as probabilities anymore and have to be recalibrated.

You seem mostly correct, the phenomenon is called prior probability shift. It is the case where $P_{train}(x|y) = P_{test}(x|y)$ but $P_{train}(y) \neq P_{test}(y)$. I'm not sure about what you mean by 'probabilities not remaining the same and have to be recalibrated'.

> Lets assume we have a two class classification problem, imbalanced datasets that we oversample to get the same class distribution. We run decision trees on it. The test set in imbalanced but does it really matter?

is imbalanced but does it really matter?

Yes, it matters and that is the cause of the problem.

> Each sample of the test set just goes through the nodes of the decision trees and it never checks if the sample belongs to the majority or minority class.

Correct. The problem is not with how decision tree predicts a given sample point. The issue lies with the way it was trained and with the characterization of the feature space for each class.

Oversampling the minority class is a way to deal with the imbalanced class problem but it is not ideal. When the minority class is over-sampled by increasing amounts, the effect is to identify similar but more specific regions in the feature space as the decision region for the minority class. The decision tree would predict a given point in the way that you mentioned but if its decision regions are not accurate based on the way it was trained then it won't predict well.

> So, why does the prior probability of classes affect the prediction of a sample?

Prior probability shift is a particular type of dataset shift. There's a fair amount of work in the literature on this topic and whether it's a generative or discriminative model, both of them suffer from the problem. The general idea is whether you are trying to train a discriminative model $P(y|x) = \frac{P(x|y)P(y)}{P(x)}$ or a generative model $P(x, y) = P(x|y)P(y)$, the change in $P(y)$ affects $P(y|x)$ and $P(x, y)$. If the $P(x)$ changes in train and test dataset, then the phenomenon is called covariate shift. You can learn more about dataset shift here, it was probably one of the first compilation of the work done on dataset shift.

On a side note, you can refer to this paper on SMOTE. It addresses the oversampling issue with decision tree and provides a better way to rebalance the dataset by creating synthetic points of the minority class. It is widely used and I believe various implementations of this method already exists.

answered Apr 13, 2018 at 9:15

Satwik Bhattamishra

1,456 9 24