Class Imbalance learning method for imbalanced dataset.

SUVAMJYOTI PANDA
AITS
techsystems.ai@gmail.com
Faridabad, India
likun54@gmail.com

ABSTRACT

It has been well studied that normal machine learning algorithm doesn't perform well on imbalanced dataset.Thus we require some methods to overcome this difficulty.It can be done in two ways externally and internally.External method deals with processing the dataset before feeding for learning whearas internal methods involves change in the learning algorithm itself.

It's the Assignment-3 given to me as a Machine Learning Intern.I have performed CIL methods on imbalanced dataset for improving the performance of machine learning algorithm on the same.

Keywords—dataset,oversampling,undersampling, imbalanced

1. INTRODUCTION

Imbalanced dataset are created when we have a situation in which the data for a particular is monotonously large as compared to other class.Due to which the machine learning technique gets biased due to lack of ample amount of information for all class. There are various class imbalance learning techniques which can be used for imbalanced dataset.These methods can be broadly classified into two category i.internall ii.external.In this paper we will focus on external methods.

A. *Internal*

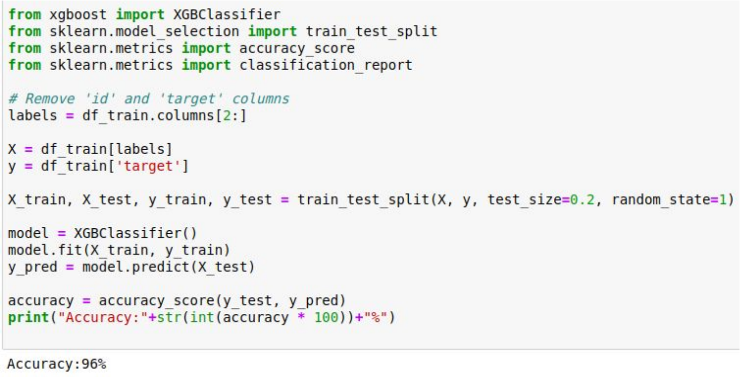
Internal methods involves modifying the learning algorithm itself to provide more accurate classification and make it less sensitive towards minority class.Modification have been proposed for various classification algorithms such as neural network[1], decision trees [2], fuzzy systems [3], [4] etc., for imbalanced dataset learning.For svm Veropoulos et al. [5] has proposed a method called different error costs (DEC)

B. *External*

External methods are independent of learning algorithm and involves preprocessing the training data, to make it balanced training the classifier on the same.Different methods such as focused undersampling, random oversampling etc falls into this category.Random oversampling involves duplicating the minority class examples until the proper ratio between two class is achieved.Focused undersampling involves removing random data from majority class until required class ratio is achieved.

2. PROBLEMS

When we train using imbalanced data,the machine learning algorithm gets skewed towards the minority class, amateur data scientist check the model by accuracy alone, and this biasedness is not visible in the same. It is only after deployment we came to know that our model is classifying each and every example as a majority class and is not able to classify minority class at all.It can be a huge problem if used this model to predict defaulters for bank loan as even the one with the worst financial condition will be classified as eligible for loan.



```
from xgboost import XGBClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from sklearn.metrics import classification_report

# Remove 'id' and 'target' columns
labels = df_train.columns[2:]

X = df_train[labels]
y = df_train['target']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=1)

model = XGBClassifier()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:"+str(int(accuracy * 100))+"%")
```

```
Accuracy:96%
```

As accuracy score are quite deceiving therefore data scientist also use recall and precision to classify
a model into a good or bad one.

Precision:

can be defined as total no of predicted true positive divided by the sum of total no of actual positive and
total no of false positives.High precision indicate that the instances which are classified as positive are

indeed positive.

Precision = TP(predicted)/TP(actual)+FP

Recall:

can be defined as total no of predicted true positive divided by the sum of total no of actual positive and total no of false negative.High recall indicate that the class is correctly classified.

Recall = TP(predicted)/TP(actual)+FN

High recall, low precision:

This means that most of the positive examples are correctly recognized (low FN) but there are a lot of false positives.

Low recall, high precision:

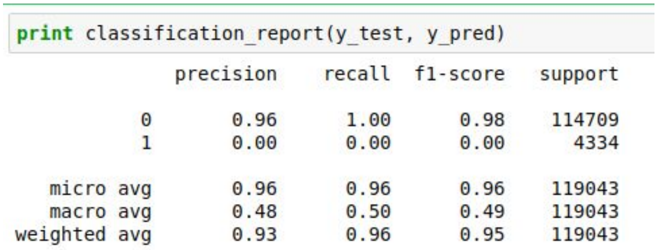
This shows that we miss a lot of positive examples (high FN) but those we predict as positive are indeed positive (low FP)

F-measure:

as their are two values to measure the goodness,hence it will be quite helpful if we could combine the both into one,thus F-measure is used.It takes harmonic mean of both instead of arithmetic mean that is punishing the extreme value more,therefore f measure generally will be nearer to lower value of precision and recall.

F-measure = (2*recall*precision)/(recall+precision)

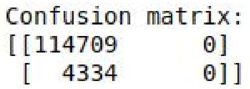
if we apply these to our imbalanced dataset we find



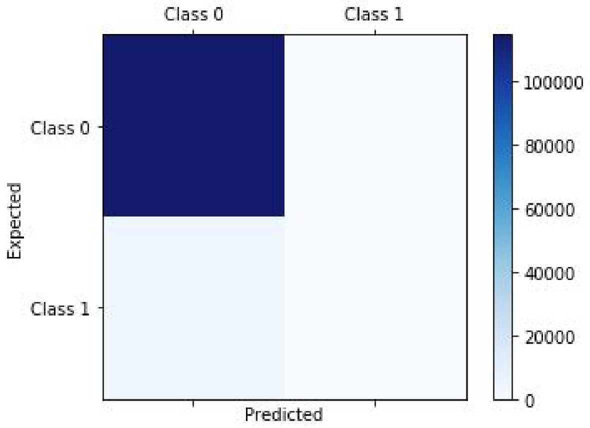
```
print classification_report(y_test, y_pred)
              precision    recall  f1-score   support

           0       0.96      1.00      0.98    114709
           1       0.00      0.00      0.00      4334

   micro avg       0.96      0.96      0.96    119043
   macro avg       0.48      0.50      0.49    119043
weighted avg       0.93      0.96      0.95    119043
```

which clearly shows that our models fails badly in classifying the minority class.

Another method to test the model is confusion matrix



```
Confusion matrix:
[[114709      0]
 [  4334      0]]
```



we can clearly infer from this that our model has classified all minority class as majority class.So to solve this we can use both external and internal CIL method in this we are concentrating on external methods.

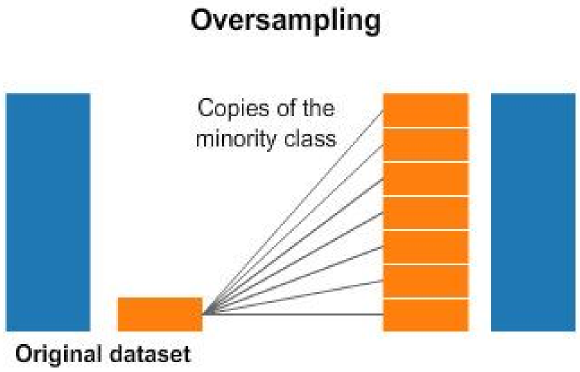
3. EXTERNAL METHODS

● Oversampling

one way to fight class imbalance is to increase the no of examples for minority clas**s.**

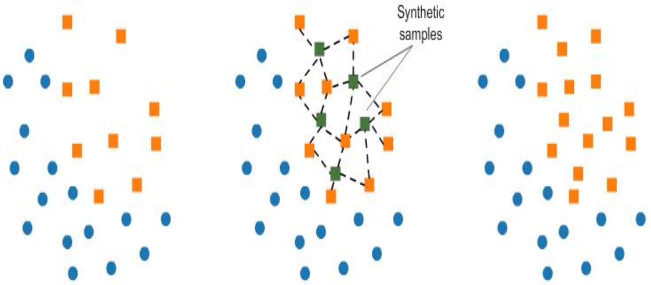
1. Random oversampling

Most naive method to do it is to randomly duplicating the already existing examples from minority class until the desirable ratio between both the class is achieved.



2. SMOTE

SMOTE(synthetic minority oversampling technique) is used for creating synthetic value of minority class by using already exicting values.It works by picking a random minority value and calculate its k-nearest neighbour and place a synthetic value between the choosen value and neighbour.

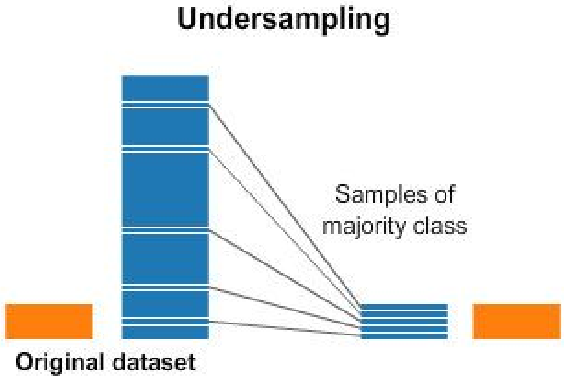


- **Undersampling**

another way to overcome the same will be to remove examples from majority class until the desired ratio is achieved.

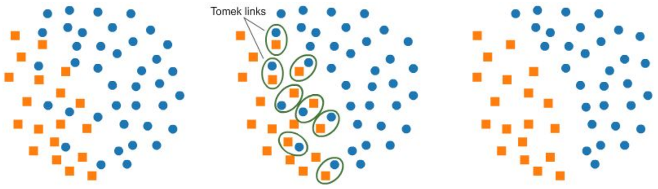
1. Random Undersampling

It involves removing random values from the class with high no of instances until both class have similar no of instances.



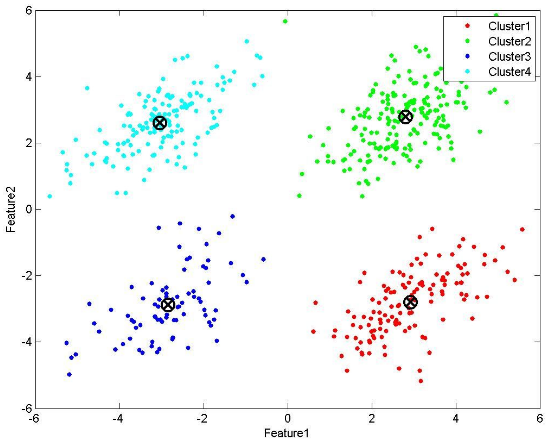
2. Tomek links

Tomek link are defined as pair of value close to each other but belonging to different class.By removing the instance of majority class ,creates a gap between two facilitating easier classification.



3. Clustered Centroid

Cluster centroid is a technique of undersampling under which we first we cluster our data into to preserve information.Method that under samples the majority class by replacing a cluster of majority samples by the cluster centroid of a KMeans algorithm. This algorithm keeps N majority samples by fitting the KMeans algorithm with N cluster to the majority class and using the coordinates of the N cluster centroids as the new majority samples.



4. CONCLUSION

These methods do a pretty good job at balancing the dataset and now we can easily use any machine learning algorithm as per requirement without the fear of biasedness.

5. REFRENCES

[1] Rafael Alencar "Resampling strategies for imbalanced datasets" https://www.kaggle.com

[2] Z.-H. Zhou and X.-Y. Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem," IEEE Trans. Knowl. Data Eng., vol. 18, no. 1, pp. 63–77, Jan. 2006

[3] D. Cieslak and N. Chawla, "Learning decision trees for unbalanced data," in Machine Learning and Knowledge Discovery in Databases. Berlin, Germany: Springer-Verlag, 2008, pp. 241–256

[4] A. Fernandez, M. Jesus, and F. Herrera, "Hierarchical fuzzy rule based classification systems with genetic rule selection for imbalanced datasets," Int. J. Approx. Reason., vol. 50, no. 3, pp. 561–577, 2009.

[5] A. Fernandez, M. Jesus, and F. Herrera, "Improving the performance ´ of fuzzy rule based classification systems for highly imbalanced datasets using an evolutionary adaptive inference system," in Bio-Inspired Systems: Computational and Ambient Intelligence. Berlin, Germany: Springer-Verlag, 2009, pp. 294–301.

[6]] K. Veropoulos, C. Campbell, and N. Cristianini, "Controlling the sensitivity of support vector machines," in Proc. Int. Joint Conf. Artif. Intell., Stockholm, Sweden, 1999, pp. 55–60