Biased support vector machine and weighted-SMOTE in handling class imbalance problem

Hartono \textsuperscript{a,1,7}, Opim Salim Sitompul \textsuperscript{b,2}, Tulus \textsuperscript{b,3}, Erna Budhiarti Nababan \textsuperscript{b,4}

\textsuperscript{a} STMIK IBBI, Jl. Sei Deli No. 18, Medan 20114, Indonesia
\textsuperscript{b} Universitas Sumatera Utara, Jl. Dr. T. Mansyur No. 9, Medan 20155, Indonesia
\textsuperscript{1} hartonoibbi@gmail.com; \textsuperscript{2} opim@usu.ac.id; \textsuperscript{3} tulus_jp@yahoo.com; \textsuperscript{4} ernabrn@usu.ac.id
* corresponding author

1. Introduction

Class imbalance problem occurs when instances in a class are much higher than in other classes. This machine learning major problem can affect the predicted accuracy. Support Vector Machine (SVM) is robust and precise method in handling class imbalance problem but weak in the bias data distribution, Biased Support Vector Machine (BSVM) became popular choice to solve the problem. BSVM provide better control sensitivity yet lack accuracy compared to general SVM. This study proposes the integration of BSVM and SMOTEBoost to handle class imbalance problem. Non Support Vector (NSV) sets from negative samples and Support Vector (SV) sets from positive samples will undergo a Weighted-SMOTE process. The results indicate that implementation of Biased Support Vector Machine and Weighted-SMOTE achieve better accuracy and sensitivity.

ARTICLE INFO

Article history
Received December 27, 2017
Revised February 9, 2018
Accepted March 9, 2018

Keywords
Class Imbalance
Biased Support Vector Machine
Weighted-SMOTE
Accuracy
Sensitivity
The SMOTE method will generally initialize a sample number of minorities and after that it will generate equal numbers of synthetic data, but this method should consider all the remaining minority data samples [15]. A Weighted-SMOTE proposed by Prusty et al. [15] will oversampling each minority data sample and carried out based on the weight assigned to it. These weights are determined by using the Euclidean distance of a particular minority data sample with consideration to all remaining minority data samples.

This paper introduced the implementation of Biased Support Vector Machine and Weighted-SMOTE in Handling Class Imbalance Problem. Biased Support Vector Machine will give better sensitivity control compare to Support Vector Machine. Biased Support Vector Machine works by providing different cost functions for positive samples (minority class) and negative samples (majority class) will group both positive samples and negative samples into Support Vector (SV) Sets and Non Support Vector (NSV) Sets.

Biased Support Vector Machine has been combined with SMOTEBoost in handling class imbalance problem to get the better accuracy. SMOTEBoost should consider all the minority data samples and for this reason, Weighted-SMOTE has been introduced. NSV Sets from negative samples and SV Sets from Positive Samples will then undergo a Weighted-SMOTE process. The traditional performance evaluation in imbalanced evaluation are accuracy and sensitivity [5].

On condition high imbalance ratio of 99%, which minority (positive) classes have only 1% instances as well as error rates may be reduced by learning algorithm through classifying all instances into majority class, until the result give a 1% error rate. Nonetheless, in data mining work, minority (positive) class usually a focus [7] and in case of class imbalance, minority classes frequently grow into the more important class [16]. To be able to recognize unusual behavior differs from the general access patterns, the focus would be placed on the positive class and Biased Support Vector Machine (BSVM) became popular choice to solve the problem. The purpose of this research is to get a better sensitivity and accuracy using the implementation of BSVM and Weighted-SMOTE in handling class imbalance problem.

2. Method

2.1. Biased Support Vector Machine and Weighted-SMOTE Algorithm

Traditional Support Vector Machine (SVM) cannot handling class imbalance problem with significant bias in data distribution. It is because the separation hyperplane learned by the SVM very close to the minority class which explains why SVM has a degrading performance on highly imbalanced datasets [17]. A Geometric Mean SVM (GSVM) proposed by Gonzalez-Abril et al. [18] will get the ideal hyperplane by calculating the moving of the original bias in the SVM to improve the geometric mean between the true positive rate and the true negative rate. However, in some problem, focus of interest was given to the positive class since it may contain unusual behavior differs from the general access pattern and to deal with datasets where one class, the positive class, is considered more relevant than another class in binary classification problems, the Biased Support Vector Machine (BSVM) method [13] can be considered.

BSVM designed for cases when it is non-critical to increase the true positive ratio in exchange for an increase in the false positive rate [19]. BSVM achieve better performance in sensitivity, but have a significant reduction in accuracy [19]. To overcome this issue, integration with Weighted-Smote Algorithm can be made. Biased Support Vector Machine combined with SMOTEBoost to handle class imbalance problem [14] but this method should consider all remaining minority data samples [15] and this is the reason why Weighted-SMOTE has an advantage to use in this research.

In Biased Support Vector Machine, minority samples are given with a larger cost function and the mathematical model of Biased Support Vector Machine is defined by (1) [13].
\[
\min \frac{1}{2} |w|^2 + C^+ \sum_{i \in I^+} \xi_i^+ + C^- \sum_{i \in I^-} \xi_i^-
\]
\[s.t. y_i (w^T \phi(x_i) + b) \geq 1 - \xi_i \quad \xi_i \geq 0, \forall i \]

Where \(w\) and \(b\) represent the normal vector and the intercept of hyperplane, respectively \(\xi \geq 0\) are slack variable.

**Step 1**, Take some data from minority class and majority class. Some data from the minority class are expressed as positive samples and some data from the majority class is expressed as negative samples.

**Step 2**, Determine the hyperplane combination of positive samples and negative samples. Where the combined value of positive samples is 1 and the combined value of negative samples is \(-1\).

**Step 3**, Minimise the margin value of positive samples and negative samples to obtain Hyperplane using Biased Support Vector Machine which is done using equation (2) and (3) [14].

\[
\frac{1}{2} |w|^2 = \frac{1}{2} (w_1^2 + w_2^2 + w_3^2 + w_4^2)
\]
\[s.t.
\]
\[
y_i (w \cdot x_i + b) \geq 1, \quad i = 1, 2, 3, \ldots, N
\]
\[
y_i (w_1 \cdot x_1 + w_2 \cdot x_2 + w_3 \cdot x_3 + w_4 \cdot x_4 + b) \geq 1
\]

**Step 4**, Determine Positive Samples and Negative Samples using (4) [14].

\[
f(x) = \text{sign} \left( \sum_{i=1}^{N} y_i (w \cdot x_i) + b \right)
\]

If the sign function gives a result greater than 0 then it is included in the minority class and if the sign function gives a result smaller than 0 then it is included in the majority class. Conduct training for minority class and majority class based on hyperplane equation obtained. The training process can be done using (4)

**Step 5**, Obtain positive result for minority class and negative result for majority class. If the training result for an instance in the minority class gives a negative result then move the instance into the majority class and if the training result for an instance in the majority class gives a positive result then move that instance into the minority class.

**Step 6**, Take some data back from the minority class and make it as positive samples and retrieve some data back from majority class and make it as negative samples. Determine the hyperplane of each positive samples and negative samples using (2) and (3).

**Step 7**, Perform training process on minority class using (5). For the process of determining SV Sets and NSV Sets by using hyperplane from positive samples. If the calculation result gives \( \leq 1 \) then it is categorized into SV Sets and otherwise categorized as NSV Sets.

\[
H_1 = SV \text{ Sets for Minority Class} \ (w \cdot x + b = 1)
\]

**Step 8**, Do the training process on majority class using (6). For the process of determining SV Sets and NSV Sets by using hyperplane of negative samples. If the calculation result gives \( \leq -1 \) then it is categorized into SV Sets and otherwise categorized as NSV Sets.

\[
H_2 = SV \text{ Sets for Majority Class} \ (w \cdot x + b = -1)
\]
Step 9, SV Eliminate noise using Sets on the Minority Class and then process the Weighted-SMOTE to be combined with NSV Sets on the minority class to become a new minority class. The Weighted-SMOTE process is as follows [15]:

1. Calculate Euclidean Distance for each of T minority data samples using (7) [20].

\[
d_{ij} = \sqrt{\sum_{k=1}^{P} (x_{ik} - x_{jk})^2}
\]  

Here, \(i = [1,2,\ldots,T]\) and \(j = [1,2,\ldots,T]\) and \(j \neq 1\). Euclidean Distance for all the minority data are calculated and stored in a column matrix using (8) [10].

\[
ED = [ED_1, ED_2, \ldots, ED_T]
\]

2. Normalized ED matrix using maximum of the ED (\(ED_{max}\)) and minimum of ED (\(ED_{min}\)) then called it as normalized ED (NED) that can be seen in (9).

\[
NED = \frac{ED - ED_{min}}{ED_{max} - ED_{min}}
\]

3. Convert NED matrix using (10) to get the Remodeled Normalized Euclidean Distance Matrix (RNED).

\[
[RNED]_{Tx1} = \text{sum}(NED) - [NED]_{Tx1}
\]

4. Calculated Weight Matrix for each minority of T samples considering the sum of RNED Matrix using (11).

\[
[Weight Matrix]_{Tx1} = \frac{[RNED]_{Tx1}}{\text{sum}(RNED)}
\]

5. Using this Weight matrix to find the SMOTE generation matrix using (12).

\[
[SMOTE Generation Matrix]_{Tx1} = \frac{NT}{100} [Weight Matrix]_{Tx1}
\]

Step 10, NSV Sets on Majority Class go through Weighted-SMOTE process and combined with SV Sets on Majority Class to become a new majority class. The Weighted-SMOTE process is similar with Step 9.

2.2. Research Design

The experiments conducted using Iris Dataset and Balanced Scale Weight & Distance Database using software deployed in R Language. Research Method can be seen in Fig. 1.

This research calculates the Accuracy and Sensitivity from the result of implementation of Biased Support Vector Machine and Weighted-SMOTE using (13) [15] and (14) [13].

\[
Accuracy = \frac{TN+TP}{TN+FP+FN+TP}
\]

\[
Sensitivity = \frac{TP}{TP+FN}
\]

Confusion Matrix [13] is used for determining the value of TN, TP, FN, and FP which can be seen in Table 1.

<table>
<thead>
<tr>
<th>Classified as Positive</th>
<th>Classified as Negative</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Positive Samples</strong></td>
<td>True Positive (TP)</td>
</tr>
<tr>
<td><strong>Negative Samples</strong></td>
<td>False Positive (FP)</td>
</tr>
</tbody>
</table>

Hartono et al. (Biased support vector machine and weighted-SMOTE ...
From the Fig. 1, shows that Biased Support Vector Machine will classify both majority class (negative samples) and minority class (positive samples) into Support Vector Sets (SV Sets) and Non Support Vector Sets (NSV Sets). Noise from SV Sets Negative Samples removed and the NSV Sets Negative Samples processed using Weighted-SMOTE. Then SV Sets Negative Samples combined with NSV Sets Negative Samples, generating a new Negative Samples Sets. After this process, noise from the SV Sets in Positive Samples removed and this SV Sets processed using Weighted-SMOTE. Then SV Sets Positive Samples combined with NSV Sets Positive Samples, generating a new Positive Samples Sets. New Negative Sample Sets and New Positive Sample Sets combined to make Result Dataset.

Fig. 1. Research Method

3. Results and Discussion

Testing conducted using Iris Dataset and Balanced Scale Weight & Distance Database using software deployed in R Language. Result of the testing using Iris Dataset can be seen in Table 2.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Before Processing</th>
<th>After Processing</th>
<th>Accuracy</th>
<th>Sensitivity</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Number of Majority</td>
<td>Number of Minority</td>
<td>Number of Majority</td>
<td>Number of Minority</td>
</tr>
<tr>
<td>1</td>
<td>60</td>
<td>40</td>
<td>52</td>
<td>48</td>
</tr>
<tr>
<td>2</td>
<td>60</td>
<td>34</td>
<td>49</td>
<td>45</td>
</tr>
<tr>
<td>3</td>
<td>62</td>
<td>39</td>
<td>52</td>
<td>49</td>
</tr>
<tr>
<td>4</td>
<td>64</td>
<td>36</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>5</td>
<td>55</td>
<td>44</td>
<td>50</td>
<td>49</td>
</tr>
<tr>
<td>6</td>
<td>71</td>
<td>29</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>7</td>
<td>66</td>
<td>30</td>
<td>50</td>
<td>46</td>
</tr>
<tr>
<td>8</td>
<td>62</td>
<td>37</td>
<td>50</td>
<td>49</td>
</tr>
<tr>
<td>9</td>
<td>67</td>
<td>35</td>
<td>54</td>
<td>48</td>
</tr>
<tr>
<td>10</td>
<td>65</td>
<td>35</td>
<td>52</td>
<td>48</td>
</tr>
<tr>
<td>Average</td>
<td>63.2</td>
<td>35.9</td>
<td>50.9</td>
<td>48.2</td>
</tr>
</tbody>
</table>

Based on Table 2 it can be seen that from the 10 experiments it was found that Biased Support Vector Machine and Weighted-SMOTE can decrease the difference in the number of members between majority and minority class. This can be done because Biased Support Vector Machine and Weighted-SMOTE can improve classification accuracy. Better accuracy impacts the misclassification on the lower
majority and minority classes. In general, class imbalance issues will arise if there is misclassification of the placement of instances in the minority class to be placed in the majority class.

Result of testing using Balanced Scale Weight & Distance Database can be seen in Table 3. Similar results were obtained in experiments using the Balance Scale Weight & Distance Database which can be seen in Table 3. The results show that the Biased Support Vector Machine and Weighted-SMOTE can also correct the difference in the number of majority and minority that can be seen on the results before and after processing. This confirms that an increase in classification accuracy can provide an advantage in overcoming class imbalance problems.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Before Processing</th>
<th>After Processing</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Number of Majority</td>
<td>Number of Minority</td>
<td>Number of Majority</td>
<td>Number of Minority</td>
<td>Accuracy</td>
</tr>
<tr>
<td>1</td>
<td>325</td>
<td>251</td>
<td>290</td>
<td>288</td>
<td>0.83</td>
</tr>
<tr>
<td>2</td>
<td>353</td>
<td>226</td>
<td>291</td>
<td>288</td>
<td>0.84</td>
</tr>
<tr>
<td>3</td>
<td>345</td>
<td>242</td>
<td>301</td>
<td>286</td>
<td>0.91</td>
</tr>
<tr>
<td>4</td>
<td>363</td>
<td>213</td>
<td>295</td>
<td>281</td>
<td>0.87</td>
</tr>
<tr>
<td>5</td>
<td>347</td>
<td>232</td>
<td>291</td>
<td>288</td>
<td>0.83</td>
</tr>
<tr>
<td>6</td>
<td>319</td>
<td>257</td>
<td>292</td>
<td>284</td>
<td>0.88</td>
</tr>
<tr>
<td>7</td>
<td>321</td>
<td>256</td>
<td>294</td>
<td>283</td>
<td>0.9</td>
</tr>
<tr>
<td>8</td>
<td>316</td>
<td>260</td>
<td>293</td>
<td>283</td>
<td>0.86</td>
</tr>
<tr>
<td>9</td>
<td>343</td>
<td>236</td>
<td>290</td>
<td>289</td>
<td>0.79</td>
</tr>
<tr>
<td>10</td>
<td>361</td>
<td>219</td>
<td>294</td>
<td>286</td>
<td>0.85</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>339.3</strong></td>
<td><strong>239.2</strong></td>
<td><strong>293.1</strong></td>
<td><strong>285.4</strong></td>
<td><strong>0.86</strong></td>
</tr>
</tbody>
</table>

From Table 2 and Table 3, there is an improvement in the differentiation on number of majority and number of minority before and after processing with Biased Support Vector Machine and Weighted-SMOTE. The accuracy and sensitivity is satisfying. The high value of accuracy and sensitivity means that the implementation of Biased Support Vector Machine and Weighted-SMOTE can handle the class imbalance problem. The results also show that if focus will be given to positive samples or minority classes then Biased Support Vector Machine and Weighted-SMOTE can be considered.

4. Conclusion

The study implemented Biased Support Vector Machine and Weighted-SMOTE in handling class imbalance problem. Experiments of two datasets showed the satisfactory result that indicated in value of the accuracy and sensitivity with the average accuracy was 0.87 and average sensitivity was 0.86. From the study, it can be conclude that implementation of BSV and Weighted-SMOTE can handle class imbalance problem with the positive class in a higher priority. The proposed methods give the satisfactory result in accuracy and sensitivity.

Acknowledgment

This work was supported by the Grant of Ministry of Research, Technology, and Higher Education (RistekDikti) of the Republic of Indonesia.

References


