JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGY
http://jict.uum.edu.my

How to cite this article:
Jamaluddin, A. H., & Mahat, N. I. (2021). Validation assessments on resampling method in imbalanced binary classification for linear discriminant analysis. *Journal of Information and Communication Technology, 20*(1), 83-102.

VALIDATION ASSESSMENTS ON RESAMPLING METHOD IN IMBALANCED BINARY CLASSIFICATION FOR LINEAR DISCRIMINANT ANALYSIS

[1]Ahmad Hakiim Jamaluddin & [2]Nor Idayu Mahat
[1]Department of Mathematics, Universiti Putra Malaysia, Malaysia
[2]Centre for Testing, Measurement and Appraisal,
Universiti Utara Malaysia, Malaysia

Corresponding author: ahmadhakiimjamaluddin@gmail.com
noridayu@uum.edu.my

Received: 12/5/2019 Revised: 29/7/2020 Accepted: 5/8/2020 Published: 4/11/2020

ABSTRACT

The curse of class imbalance affects the performance of many conventional classification algorithms including linear discriminant analysis (LDA). The data pre-processing approach through some resampling methods such as random oversampling (ROS) and random undersampling (RUS) is one of the treatments to alleviate such curse. Previous studies have attempted to address the effect of a resampling method on the performance of LDA. However, some studies contradicted with each other based on different performance measures as well as validation strategies. This manuscript attempted to shed more light on the effect of a resampling method (ROS or RUS) on the performance of LDA based on true positive rate and true negative rate through five validation strategies, i.e. leave-one-out cross-validation, *k*-fold cross-validation, repeated

k-fold cross-validation, naive bootstrap, and .632+ bootstrap. 100 two-group bivariate normally distributed simulated and four real data sets with severe class imbalance ratio were utilised. The analysis on the location and dispersion statistics of the performance measures was further enlightened on: (i) the effect of a resampling method on the performance of LDA, and (ii) the enhancement in the learning fairness of LDA on objects regardless of sample size, hence reducing the effect of the curse of class imbalance.

Keywords: Linear discriminant analysis, pre-processing, resampling method, class imbalance, binary classification.

INTRODUCTION

Classification algorithms including linear discriminant analysis (LDA) often deal with a data set with groups of similar sizes (balanced groups). Since the conventional algorithms learn fairly from each object, they will learn more from a group with more objects, leading to a biased discrimination learning. For a data set with such imbalanced groups, the performance of the classification algorithms favours the majority group and this problem is termed as *the curse of class imbalance* (Das et al., 2018). The fairness of the algorithms is disrupted by which they learn relatively less from the minority group than the majority group. To overcome such curse, three primary approaches have been established, namely: (i) data pre-processing approach, which aims to alter the imbalanced data prior to classification; (ii) algorithm-oriented approach, which comprises some methodologies to improve the existing classification rules algorithmically or structurally to yield flexible rules; and (iii) hybrid approach, which combines both data pre-processing and algorithm-oriented approaches (Kaur et al., 2019). A cluster of methods within the data pre-processing approach that aims to transform a data set with imbalanced group sizes (imbalanced data set) into a data set with balanced group sizes (balanced data set) is termed as resampling methods. Resampling methods comprise oversampling, undersampling, and a combination of both oversampling and undersampling (which is also termed as hybrid resampling). The most basic oversampling method is random oversampling (ROS) and the most basic undersampling method is termed as random undersampling (RUS). At random, ROS selects several objects from the minority group and adds them into the same group, whilst RUS excludes several objects from the majority group so that the groups have similar sizes (Japkowicz, 2000).

In general, an evaluation is required to verify the performance of a developed classification algorithm. Some performance measures such as sensitivity or

true positive rate (TPR), specificity or true negative rate (TNR), accuracy rate, and misclassification rate are computed based on a validation sample. Often, the choice of measure depends on the aim of a study. To evaluate the performance of a classification algorithm, a suitable strategy is needed to ensure an accurate performance estimation. There are several strategies to accurately estimate the performance measures such as: (i) cross-validation including leave- q -out cross-validation (lqocv), k -fold cross-validation (kfcv), and repeated k -fold cross-validation (rkfcv); and (ii) bootstrap including naive bootstrap (B) and .632+ bootstrap (B632).

Cross-validation strategy deals with having some samples of objects used to train a classification algorithm and validate its performance alternatively. Cross-validation is widely utilised due to the universality in data splitting heuristics, and the most basic cross-validation strategy is lqocv (Arlot & Celisse, 2010). Later, to alleviate the high computational cost of lqocv, kfcv was introduced by Geisser (1975). Although the computational has been reduced, the variability of kfcv is still questionable. Thus, rkfcv was proposed and examined by Burman (1989) in regression analysis. Subsequently, Kim (2009) investigated its goodness in classification and revealed that the incurred computational cost is worth borne. The naive bootstrap strategy (B) deals with a selection of *b* samples of *m* observations with replacement. B yields an estimator with less variability for small samples (Efron & Tibshirani, 1993). To improve the accuracy of B, .632+ bootstrap (B632) was introduced by Efron and Tibshirani (1997).

The effects of class imbalance on LDA has been investigated theoretically and empirically using different performance measures including *TPR*, *TNR* , error rate (*ER*), and area under receiver operating characteristics curve (*AUC*). The two earliest related works theoretically and empirically contradicted with each other on the effects of using a resampling method (ROS or RUS) on LDA. Nevertheless, a recent study succeeded to enlighten on the unique contributions of the previous works and justifications towards their conclusions empirically. The findings of the work could have been more convincing if some validation strategies are used for the verification of the performance estimation.

This manuscript attempts to further investigate and verify the novel findings of the recent works through some validation strategies including loocv, kfcv, rkfcv, B, and B632 on the performance of LDA with (or without) a resampling method. This manuscript starts with a thorough discussion on related works primarily based on three main studies. The next section describes the methodology of the study in terms of classification algorithms,

data simulation, and the sets of real data considered. The findings of the study are presented in the subsequent section before discussions and conclusion are finally presented.

RELATED WORKS

Classification tasks are applicable in various domains (Hairuddin et al., 2020; Roy et al., 2018). In a binary classification, Gaussian-based Bayes rule assigns an object with p variables, $\boldsymbol{x}$ to group π_1 if the probability density of $\boldsymbol{x}$ in π_1 is larger than that in π_2:

$$g_1(\boldsymbol{x}) > g_2(\boldsymbol{x}). \quad (1)$$

When π_1 and π_2 have homogeneous covariance matrices such that $\boldsymbol{\Sigma_1} = \boldsymbol{\Sigma_2} = \boldsymbol{\Sigma}$, a future object $\boldsymbol{x_0}$ will be assigned to π_1 if:

$$(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\Sigma^{-1}\boldsymbol{x}_0 - \frac{1}{2}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)'\Sigma^{-1}\boldsymbol{x}_0(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2) \geq ln\left(\frac{p_2}{p_1}\right) \text{ or } \boldsymbol{x}_0 \epsilon\, \pi_2 \text{ otherwise,} \quad (2)$$

where $\boldsymbol{\mu}_1$, $\boldsymbol{\Sigma_1}$ and p_1 refer to the mean vector, covariance matrix, and prior probability of π_1, respectively; while $\boldsymbol{\mu}_2$, $\boldsymbol{\Sigma_2}$ and p_2 refer to the mean vector, covariance matrix, and prior probability of π_2, respectively. Equation 2 holds only if the misclassification cost of both groups is equal. For data with unknown parameters, the parameters could be estimated using sample-based statistics, namely:

sample mean vectors, $\bar{x}_i = \frac{1}{n_i}\sum_{j=1}^{n_i} x_{ij}$, (3)

sample homogeneous covariance, $\boldsymbol{S}_{pooled} = \left[\frac{n_1-1}{(n_1-1)+(n_2+1)}\right]\boldsymbol{S}_1 + \left[\frac{n_2-1}{(n_1-1)+(n_2+1)}\right]\boldsymbol{S}_2$, and (4)

sample group probability, $\hat{p}_i = \frac{n_i}{\sum_{i=1}^{2} n_i}$ (5)

where $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ are the sample covariance matrices of π_1 and π_2, respectively such that: $\boldsymbol{S}_i = \frac{1}{n_i-1}\sum_{j=1}^{n_i}(\boldsymbol{x}_{ij} - \bar{\boldsymbol{x}}_i)(\boldsymbol{x}_{ij} - \bar{\boldsymbol{x}}_i)'$ for $i = 1, 2, \ldots, n$ and $j = 1, 2$. (6)

Resampling method is used prior to the construction of a linear discriminant rule. For a data set with n_1 objects in the minority group π_1 and n_2 objects in the majority group π_2, ROS randomly selects an object $\boldsymbol{x}_i$ from π_1 and adds it into π_1 for $(n_2 - n_1)$ times, so that π_1 has $n_1 + (n_2 - n_1)$ objects. RUS randomly selects an object $\boldsymbol{x}_i$ from π_2 and discards it from π_2 for $n_2 - n_1$ times, so that π_2 consists of $n_2 - (n_2 - n_1)$ objects. Both resampling methods eventually yield a balanced data set.

In general, the decision making of an algorithm for a binary classification is represented by a 2x2 confusion matrix. The matrix has objects termed as positives and negatives, where the positives refer to the objects with the event of interest and the negatives represent the objects without the event of interest.

The confusion matrix has four main cells with four different elements, namely true positives (*TP*), true negatives (*TN*), false positives (*FP*), and false negatives (*FN*). *TP*, *TN*, *FP*, and *FN* consist of correctly classified positives, correctly classified negatives, incorrectly classified negatives, and incorrectly classified positives, respectively. In a typical class imbalance study, the minority group refers to the group with the positives, whilst the majority group contains the negatives. Table 1 shows the confusion matrix.

Table 1

Confusion Matrix of a Two-Group Classification

		Prediction	
		Positive	Negative
Actual	Positive	*TP*	*FN*
	Negative	*FP*	*TN*

Accuracy and error rate (*ER*) are often used as the performance measures in an algorithm performancvaluation for data sets with balanced group sizes. The formulae of the measures are as follows:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \text{ and} \tag{7}$$

$$ER = 1 - \frac{TP+TN}{TP+TN+FP+FN}. \tag{8}$$

Both measures are at a disadvantage in class imbalance studies due to the bias drawn from the favour towards the majority group (Branco et al., 2016). Such disfavour leads to the introduction of balanced accuracy:

$$balanced\ accuracy = \frac{1}{2}\left(\frac{TP+FN}{TP} + \frac{TN+FP}{TN}\right). \tag{9}$$

Meanwhile, balanced accuracy comprehensively estimates the performance of an algorithm. Some other measures, i.e. *TPR* (in Equation 10) and *TNR* (in Equation 11), can specifically estimate the performance of an algorithm in the classification of the positives and the negatives, respectively.

$$TPR = \frac{TP}{TP+FN} \tag{10}$$

$$TNR = \frac{TN}{TN+FP}. \tag{11}$$

LDA has been evaluated in dealing with the curse of class imbalance by primarily three studies (Jamaluddin & Mahat, 2019; Xie & Qiu, 2007; Xue & Titterington, 2008) using different performance measures, i.e.*ER*, *AUC*, *TPR*, and *TNR* . The earliest theoretical and empirical work on LDA in class imbalance used 10 real data sets (Xie & Qiu, 2007). The findings suggested that class imbalance affected the performance of LDA negatively and a resampling method (ROS or RUS) improved its performance based on *AUC* through 4-fold cross-validation strategy. However, Xue and Titterington (2008) revealed that no evidence was found to support the previous findings. Their empirical study employed four simulated data sets of normally distributed data and normal mixture data, and used the same 10 real data sets as in Xie and Qiu (2007). Xue and Titterington (2008) utilised two performance measures, namely *AUC* and *ER* through hold-out validation strategy. Their findings concluded that a resampling method (ROS or RUS) resulted in a relatively small improvement in *AUC*, but significantly reduced *ER*. Recently, a study by Jamaluddin and Mahat (2019) unveiled a conclusive finding that shed light on the contradicting findings from both previous studies. Their empirical study employed 100 bivariate normally distributed simulated and four real data sets. Through 10-fold cross-validation strategy based on *TPR* and *TNR,* they revealed that the performance increment effect in the classification of the positives was more significant than the performance decrement effect in the classification of the negatives. Besides, they found that LDA was significantly biased towards the majority group, hence, they inferred that class imbalance negatively affected the performance of LDA. The use of *TPR* and *TNR* succeeded to relate the findings from the earliest works as both measures allowed the performance estimation of LDA in learning from the minority group objects and the majority group objects individually. Although the study has comprehensively revealed the effect of a resampling method on the performance of LDA, the findings could be explored and verified further using various validation strategies to allow for variability analysis of the estimates through different strategies such as loocv, kfcv, rkfcv, B, and B632. The results could further enlighten on the consistency of estimates between the strategies, which would eventually assist in the effect analysis of a resampling method towards LDA, especially in terms of the fairness in discrimination learning from objects regardless of group sizes.

METHODOLOGY

Classification Algorithm

The classification algorithms based on five different validation strategies are organised as in Algorithms 1 to 5.

Algorithm 1: LDA with resampling method based on leave - one-out cross - validation

Step 1: Let $\overline{f_j}$ be the $\overline{j^{th}}$ test set with one observation for $j = 1, 2, \ldots, n$. Meanwhile, f_{-j} containing the data without the j^{th} observation represents the training set.
1.1 Perform ROS on f_{-j}.
1.2 Construct LDA using the output data of ROS.
1.3 Evaluate the constructed LDA using the test set f_j by computing
1.3.1 true positive rate, $tpr_j = tp_j$, and
1.3.1 true negative rate, $tnr_j = tn_j$,
where tp_j is the number of correctly classified positive object and tn_j is the number of correctly classified negative object in j^{th} test set
1.4 Repeat Steps 1.1 and 1.3 until $j = n$.

Step 2: Compute
2.1 true positive rate $TPR = \sum_j^n \left(\frac{tpr_j}{k}\right)$, and
2.2 true negative rate $TNR = \sum_j^n \left(\frac{tnr_j}{k}\right)$.

Step 3: Repeat Step 1 to Step 2 by replacing ROS with RUS at Step 1.1.

Algorithm 2: LDA with resampling method based on -fold cross-validation

Step 1: Split an imbalanced data set with group sizes of n_1 and n_2 into k folds $(f = 1, 2, \ldots, k)$ so that each fold contains $\frac{n_1}{k}$ and $\frac{n_2}{k}$ observations.

Step 2: Let f_j be the j^{th} test set with n_j number of observations for (j =1,2,...,k). Meanwhile, f_{-j} containing the folders without the j^{th} folder represents the training set.
2.1 Perform ROS on $\overline{f_{-j}}$ such that both groups in the training set have equal size.
2.2 Construct LDA using the output data of ROS.
2.3 Evaluate the constructed LDA using f_j the test set by computing
2.3.1 true positive rate, $tpr_j = \frac{tp_j}{n_j}$, and
2.3.1 true negative rate, $tnr_j = \frac{tn_j}{n_j}$,
where tp_j is the number of correctly classified positive objects and tn_j is the number of correctly classified negative objects in j^{th} test set

(continued)

Step 3: Compute
3.1 true positive rate $TPR = \sum_j^k \left(\frac{tpr_j}{k}\right)$, and
3.2 true negative rate $TNR = \sum_j^k \left(\frac{tnr_j}{k}\right)$.

Step 4: Repeat Step 1 to Step 3 by replacing ROS with RUS at Step 2.1.

Algorithm 3: LDA with resampling method based on repeated k -fold cross-validation

Step 1: Split an imbalanced dataset with group sizes of n_1 and n_2 into folds ($f = 1, 2, \ldots, k$) so that each fold contains $\frac{n_1}{k}$ and $\frac{n_2}{k}$ observations.

Step 2: Let f_j be the test j^{th} set with $\overline{n_j}$ number of observations for j=1,2,...,k. Meanwhile, f_{-j} containing the folders without the j^{th} folder represents the training set.
2.1 Perform ROS on f_{-j} such that both groups in the training set have equal size.
2.2 Construct LDA using the output data of ROS.
2.3 Evaluate the constructed LDA using the test set f_j by computing
2.3.1 true positive rate, $tpr_j = \frac{tp_j}{n_j}$, and
2.3.1 true negative rate, $tnr_j = \frac{tn_j}{n_j}$,
where tp_j is the number of correctly classified positive objects and tn_j is the number of correctly classified negative objects in j^{th} test set
2.4 Repeat Steps 2.1 and 2.3 until j=k .

Step 3: Compute
3.1 true positive rate $TPR = \sum_j^k \left(\frac{tpr_j}{k}\right)$, and
3.2 true negative rate $TNR = \sum_j^k \left(\frac{tnr_j}{k}\right)$.

Step 4: Repeat Step 1 to Step 3 for r repetitions.

Step 5: Repeat Step 1 to Step 4 by replacing ROS with RUS at Step 2.1.

Algorithm 4: LDA with resampling method based on bootstrap

Step 1: Select n observations with replacement from the imbalanced data for b samples (f = 1,2,...,b) at random.

Step 2: Let f_j be the j^{th} sample with n number of observations for j = 1,2,...,b.
2.1 Perform ROS on f_j .
2.2 Construct LDA using the output data of ROS.
2.3 Evaluate the constructed LDA using f_j by computing
2.3.1 true positive rate, $tpr_j = \frac{tp_j}{n_j}$, and
2.3.1 true negative rate, $tnr_j = \frac{tn_j}{n_j}$,
where tp_j is the number of correctly classified positive objects and tn_j is the number of correctly classified negative objects in j^{th} sample.
2.4 Repeat Steps 2.1 and 2.3 until $j = b$.

(continued)

Step 3:	Compute 3.1 true positive rate $TPR = \sum_j^b \left(\frac{tpr_j}{b}\right)$, and 3.2 true negative rate $TNR = \sum_j^b \left(\frac{tnr_j}{b}\right)$.
Step 4:	Repeat Step 1 to Step 3 by replacing ROS with RUS at Step 2.1.

Algorithm 5: LDA with resampling method based on .632+bootstrap

Step 1:	Select *n* observations with replacement from the imbalanced data for *b* samples (*f* = 1,2,...,*b*) at random.
Step 2:	Split f_j into two sub-samples $(i = 1, 2)$ for *j* = 1,2,...,*b*.
Step 3:	Let f_{ji} be the j^{th} sample for i^{th} sub-sample.
Step 4:	Let f_{j1} be.368 of *n* observations and the remaining observations as f_{j2}(.632*n*). 4.1 Perform ROS on f_{j1} 4.2 Construct LDA using the output data of ROS. 4.3 Evaluate the constructed LDA using f_{j2} by computing 4.3.1 true positive rate, $tpr_j = \frac{tp_j}{n_j}$, and 4.3.1 true negative rate, $tnr_j = \frac{tn_j}{n_j}$, where tp_j is the number of correctly classified positive objects and tn_j is the number of correctly classified negative objects in f_{j2}. 4.4 Repeat Steps 4.1 and 4.3 until *j* = *b*.
Step 5:	Compute 5.1 true positive rate $TPR = \sum_j^b \left(\frac{tpr_j}{b}\right)$, and 5.2 true negative rate $TNR = \sum_j^b \left(\frac{tnr_j}{b}\right)$.
Step 6:	Repeat Step 1 to Step 5 by replacing ROS with RUS at Step 4.1.

Parameters used in the algorithms included: (i) k = 10 (Kim, 2009; Kohavi, 1995), which was used in Algorithms 2 and 3; (ii) b = 50 (Efron & Tibshirani, 1993), which was used in Algorithms 4 and 5; and (iii) r =5, which was used in Algorithm 3 to ensure a fair comparison since $b = 50$ was used in Algorithms 4 and 5 following Kim (2009). *caret* R package (Kuhn, 2008) was used to execute all five algorithms. Specific settings of R codes included several functions. First, *trainControl(method, sampling, number, repeats)*, where (i) *method = c("LOOCV", "cv", "repeatedcv", "boot", "boot632")*, with *"LOOCV"*, *"cv"*, *"repeatedcv"*, *"boot"* and *"boot632"* referring to loocv, kfcv, rkfcv, B, and B632 respectively; (ii) *sampling = c("up", "down")*, with *"up"* and *"down"* referring to ROS and RUS, respectively; (iii) *number* only applies to kfcv and B strategies. It represents (a) the number of folds for kfcv or (b) the number of bootstrap samples for B; and (iv) *repeats* only applies to rkfcv and it represents the number of repetitions performed. Second, *train(x, y, method, trControl)*, where (i) *x* refers to the data for model development and

training of *n* observations (in rows) times independent variables (in columns); (ii) *y* refers to the data for model development and training of *n* observations (in rows) times dependent variable (in column); (iii) *method* refers to the classification method used and, this study used *method = "lda"* as *"lda"* represents LDA; and (iv) *trControl* refers to the output of *trainControl()*.

Simulated and Real Data Sets

100 two-group bivariate normally distributed simulated data sets were produced and utilised for the performance assessment of the algorithms. Individual data sets had unique characteristics based on different parameter settings, i.e. (i) group size ratio (minority to majority), $\pi_1 : \pi_2 = 100:900$; mean, $\boldsymbol{\mu}_d$ = (-$\boldsymbol{\mu}$, + $\boldsymbol{\mu}$) where $\boldsymbol{\mu}$ = 0.5, 1.0, 1.5, 2.0, 2.5; (iii) variance, $\boldsymbol{\sigma}_d^2$ = ($\boldsymbol{\sigma}^2$, $\boldsymbol{\sigma}^2$) where $\boldsymbol{\sigma}^2$ = 0.5, 1.0, 1.5, 2.0, 2.5; and (iv) bivariate correlation, $cor(x_1, x_2)_d$ = 0.1, 0.2, 0.3, 0.4. The parameters were used in such settings to ensure group distance, bivariate correlation, and group overlapping vary across data sets in order to mimic many real data structures (Jamaluddin & Mahat, 2019). The simulated data sets were generated using a newly created function based on *rmvnorm(n, mean, v)* function from *mvtnorm* R package (Genz et al., 2020). *rmvnorm* requires at least three input parameters, i.e. *n* represents the number of objects, *mean* refers to the mean and standard deviation variable vector, while *v* represents the variance-covariance variable matrix. Four real data sets acquired from Knowledge Extraction Evolutionary Learning (KEEL) website (as in Table 2) had less than 10% minority to majority group ratio in parallel with such ratio of the simulated data sets (Alcalá-Fdez et al., 2011).

Table 2

Details of Real Data Sets

Name	Number of variables	Number of observations	Minority: Majority (%)
abalone9-18	9	731	0.05746: 0.94254
shuttle-c0-vs-c4	10	1829	0.06725: 0.93275
glass-0-1-6_vs_2	10	192	0.08854: 0.91146
page-blocks-1-3_vs_4	11	472	0.05932: 0.94068

FINDINGS

To analyse the performance of the algorithms in classifying the objects from individual groups, the mean (TPR_μ), standard deviation (TPR_σ), and range

(TPR_R) of TPR as well as the mean (TNR_μ), standard deviation (TNR_σ),and range (TNR_R) of TNR were computed as presented in Tables 3 and 4, respectively.

Table 3

TPR_μ, TPR_σ, and TPR_R of Simulated Data Sets

Algorithm	Validation strategy	TPR_μ	TPR_σ	TPR_R
LDA	loocv	0.67590	0.36164	[0.01000, 1.00000]
	kfcv	0.67700	0.36104	[0.01000, 1.00000]
	rkfcv	0.67666	0.36104	[0.01000, 1.00000]
	B	0.67542	0.36093	[0.00891, 1.00000]
	B632	0.67563	0.36083	[0.00944, 1.00000]
ROS-LDA	loocv	0.91530	0.10219	[0.69000, 1.00000]
	kfcv	0.91370	0.10368	[0.68000, 1.00000]
	rkfcv	0.91460	0.10279	[0.68400, 1.00000]
	B	0.91383	0.10401	[0.67595, 1.00000]
	B632	0.91370	0.10419	[0.67703, 1.00000]
RUS-LDA	loocv	0.91420	0.10300	[0.68000, 1.00000]
	kfcv	0.91300	0.10528	[0.61000, 1.00000]
	rkfcv	0.91448	0.10334	[0.67500, 1.00000]
	B	0.91360	0.10408	[0.67044, 1.00000]
	B632	0.91363	0.10402	[0.67254, 1.00000]

Tables 3 and 4 depict several findings. First, the TPR_μ for conventional LDA, ROS-LDA, and RUS-LDA based on all validation strategies were [0.67542, 0.67700], [0.91370, 0.91530], and [0.91300, 0.91448], respectively. Meanwhile, the TNR_μ for conventional LDA, ROS-LDA, and RUS-LDA based on all validation strategies were [0.99439, 0.99457], [0.89546, 0.89606], and [0.89429, 0.89454], respectively. The ranges in TPR_μ and TNR_μ exhibited that on average, LDA with a resampling method could classify the objects from the minority and majority groups better than the conventional LDA respectively. Second, the TPR_σ for conventional LDA, ROS-LDA, and RUS-LDA based on all validation strategies were [0.36083, 0.36164], [0.10219, 0.10419], and [0.10300, 0.10528], respectively. Meanwhile, the TNR_σ for conventional LDA, ROS-LDA, and RUS-LDA based on all validation strategies were [0.00536, 0.00545], [0.11537, 0.11581], and [0.11668, 0.11722], respectively. In a nutshell, the TPR_σ and TNR_σ of LDA with a resampling method were stable approximately between 0.10 to 0.12. Unfortunately, the TPR_σ and TNR_σ

of the conventional LDA were unstable as they were approximately between 0.01 to 0.36. These findings further discovered that LDA with a resampling method learned from objects regardless of the group sizes fairer than the conventional LDA.

Table 4

TNR_μ, TNR_σ, and TNR_R of Simulated Data Sets

Algorithm	Validation strategy	TNR_μ	TNR_σ	TNR_R
LDA	loocv	0.99452	0.00544	[0.98556, 1.00000]
	kfcv	0.99451	0.00545	[0.98444, 1.00000]
	rkfcv	0.99457	0.00539	[0.98556, 1.00000]
	B	0.99440	0.00536	[0.98499, 1.00000]
	B632	0.99439	0.00538	[0.98475, 1.00000]
ROS-LDA	loocv	0.89606	0.11554	[0.63778, 1.00000]
	kfcv	0.89599	0.11537	[0.63667, 1.00000]
	rkfcv	0.89596	0.11567	[0.63533, 1.00000]
	B	0.89546	0.11581	[0.63795, 1.00000]
	B632	0.89554	0.11568	[0.63795, 1.00000]
RUS-LDA	loocv	0.89454	0.11718	[0.62889, 1.00000]
	kfcv	0.89434	0.11687	[0.64667, 1.00000]
	rkfcv	0.89429	0.11722	[0.63356, 1.00000]
	B	0.89447	0.11668	[0.63684, 1.00000]
	B632	0.89443	0.11670	[0.63659, 1.00000]

Despite the general insights, the significant difference in the ranges in TPR_μ, TNR_μ, TPR_σ, and TNR_σ between the conventional LDA and the LDA with a resampling method could suggest that the advantage of using a resampling method still outweighed its disadvantage. This was in view of the fact that the range differences in TPR_μ and TNR_μ reflected that a resampling method increased approximately 26% (91% - 67%) of TPR but reduced only roughly 10% (100% - 90%) of TNR on average. Moreover, the range differences in TPR_σ and TNR_σ depicted that the consistency of TPR increased by approximately 26% (36% - 10%), whereas TNR reduced by only 11.5% (12% - 0.5%) on average when a resampling method was employed. This could be further explained as the difference in the lowest TPR of the conventional LDA and the LDA with a resampling method was approximately 60% (61% - 1%). On the other hand, the difference in the lowest TNR of the conventional LDA and the LDA with

a resampling method was roughly only 35% (98% - 63%). To further examine the significance in range differences of TPR and TNR, the average difference between the conventional LDA and the LDA with a resampling method (LDA-ROS or LDA-RUS) was computed and tabulated in Table 5. The formulae of average difference in TPR and TNR are as follows:

$$\bar{\Delta}\left(TPR_{id}, TPR_{jd}\right) = \frac{\sum_{d=1}^{100} TPR_i - \sum_{d=1}^{100} TPR_j}{100}, \quad (12)$$

$$\bar{\Delta}\left(TNR_{id}, TNR_{jd}\right) = \frac{\sum_{d=1}^{100} TNR_i - \sum_{d=1}^{100} TNR_j}{100} \quad (13)$$

where i = LDA, j = ROS-LDA or RUS-LDA for d data sets.

Based on Table 5, LDA with a resampling method (ROS or RUS) outweighed the conventional LDA by approximately 24% in classifying the objects from the minority group, while the conventional LDA outperformed the LDA with a resampling method by approximately 10% in the classification of majority group objects on average. Relatively, this finding suggested that the improvement in *TPR* of LDA was twice the deterioration in *TNR* of LDA when employing a resampling method (ROS or RUS). In the other perspective, resampling method increased the fairness of LDA when it allowed a better learning process on the objects from both groups for a just discrimination and classification. In order to learn more about the fair learning process of LDA on each object regardless of group sizes, the number of false negatives (*FN*) and errors (*E*) were examined. Table 6 summarises the *FN* and *E* for both the conventional LDA and the LDA with a resampling method.

Three scenarios of classification case were analysed based on Table 6. The first scenario referred to cases where errors were due to only the misclassification of the positives $((FN = E) \neq 0)$. Table 6 shows that there was no case of errors due to the misclassified positives alone for the LDA with a resampling method. However, the range of such case for the conventional LDA was from 0% to 4%. Hence, a resampling method was proved to avoid such cases by 0% to 4%. In addition, the finding demonstrated that the conventional LDA was biased towards the majority group objects by neglecting the minority group objects for 0% to 4% of the data sets. The second scenario referred to cases with perfect classification $(FN = E = 0)$. The number of perfect classification cases for the conventional LDA was [13, 17] while that of the LDA with a resampling method was [7, 10]. This implied that the conventional LDA had 6% to 7% higher number of cases with perfectly classified objects than that of the LDA with a resampling method. The third scenario referred to cases where errors were due to the misclassification of both the positives and the negatives $(FN \neq E)$. The number of data sets with errors comprising misclassified objects

from both groups for the conventional LDA and the LDA with a resampling method were [79, 87] and [89, 93], respectively. These figures signified the better fairness in discrimination of objects acquired by employing a resampling method as the LDA with a resampling method had more cases for which errors consisted of both misclassified positives and negatives. To understand more, the third scenario was further investigated by evaluating the percentage of cases for this scenario by which at least 50%, 75%, and 90% of *E* were contributed by *FN* for both the conventional LDA and the LDA with a resampling method. None of the data sets had at least 50% of *E* due to *FN* for the LDA with a resampling method (ROS-LDA or RUS-LDA). Thus, the investigation was only performed on the conventional LDA. Table 7 displays the findings of further analysis on the third scenario for the conventional LDA.

Table 5

Average Difference in and Based on Simulated Data Sets

	Validation strategy	Average difference value
$\bar{\Delta}(TPR_{LDA}, TPR_{ROS-LDA})$	loocv	-0.23940
	kfcv	-0.23670
	rkfcv	-0.23794
	B	-0.23841
	B632	-0.23807
$\bar{\Delta}(TPR_{LDA}, TPR_{RUS-LDA})$	loocv	-0.23830
	kfcv	-0.23830
	rkfcv	-0.23782
	B	-0.23818
	B632	-0.23800
$\bar{\Delta}(TNR_{LDA}, TNR_{ROS-LDA})$	loocv	0.09847
	kfcv	0.09852
	rkfcv	0.09860
	B	0.09893
	B632	0.09884
$\bar{\Delta}(TNR_{LDA}, TNR_{RUS-LDA})$	loocv	0.09998
	kfcv	0.10017
	rkfcv	0.10028
	B	0.09993
	B632	0.09995

Table 6

FN And E of LDA With and Without Resampling Methods

Algorithm	Validation strategy	(FN=E) ≠ 0 (%)	FN= E = 0 (%)	F ≠ E (%)
LDA	loocv	4	17	79
	kfcv	2	17	81
	rkfcv	2	17	81
	B	0	13	87
	B632	0	13	87
ROS-LDA	loocv	0	10	90
	kfcv	0	11	89
	rkfcv	0	10	90
	B	0	7	93
	B632	0	8	92
RUS-LDA	loocv	0	10	90
	kfcv	0	10	90
	rkfcv	0	9	91
	B	0	8	92
	B632	0	7	93

Table 7

Percentage of Data Sets with at Least 50%, 75%, and 90% Of due to for the Conventional LDA

Validation strategy	Data set with at least 50% of due to (%)	Data set with at least 75% of due to (%)	Data set with at least 90% of due to (%)
B	78	33	17
B632	78	33	17
kfcv	79	40	15
loocv	77	38	12
rkfcv	78	38	15

Based on Table 7, the percentages of cases for which at least 50%, 75%, and 90% of the errors were due to the misclassified positives were [77%, 79%], [33%, 40%], and [12%, 17%], respectively. The results implied that a majority

of the positives were misclassified in most data sets. Thus, it was evidenced that the conventional LDA could not fairly learn from the objects regardless of the size of the groups and a resampling method (ROS or RUS) alleviated such drawback.

Table 8

$\Delta(TPR_i, TPR_j)$ *and* $\Delta(TNR_i, TNR_j)$ *Based on Real Data Where I = LDA and J = ROS-LDA Or RUS-LDA*

Validation strategy	Algorithm	$\Delta(TPR_i, TPR_j)$			
		Data 1	Data 2	Data 3	Data 4
loocv	ROS-LDA	0.00813	-0.23810	-0.21429	-0.82353
	RUS-LDA	0.01626	-0.23810	-0.25000	-0.76471
kfcv	ROS-LDA	0.00813	-0.23810	-0.21429	-0.88235
	RUS-LDA	-0.00813	-0.21429	-0.28571	-0.70588
rkfcv	ROS-LDA	0.00650	-0.26190	-0.20714	-0.77059
	RUS-LDA	0.00407	-0.25952	-0.25357	-0.79412
B	ROS-LDA	0.00058	-0.26412	-0.17495	-0.72663
	RUS-LDA	0.00186	-0.25596	-0.22248	-0.73798
B632	ROS-LDA	0.00197	-0.26567	-0.17565	-0.72096
	RUS-LDA	0.00110	-0.26396	-0.22989	-0.73644
		$\Delta(TNR_i, TNR_j)$			
loocv	ROS-LDA	0.00000	0.06096	0.08108	0.21143
	RUS-LDA	0.00000	0.08418	0.11712	0.31429
kfcv	ROS-LDA	0.00000	0.06241	0.07432	0.18857
	RUS-LDA	0.00000	0.06386	0.09910	0.31429
rkfcv	ROS-LDA	-0.00006	0.06415	0.08086	0.20857
	RUS-LDA	-0.00012	0.07417	0.12027	0.29200
B	ROS-LDA	0.00004	0.06752	0.06976	0.20286
	RUS-LDA	0.00006	0.07865	0.10993	0.27962
B632	ROS-LDA	0.00006	0.06667	0.06855	0.19952
	RUS-LDA	0.00006	0.07837	0.11263	0.27664

Data 1, Data 2, Data 3, and Data 4 refer to *shuttle-c0-vs-c4*, *abalone9-18*, *page-blocks-1-3_vs_4*, and *glass-0-1-6_vs_2*, respectively.

In summary, although the conventional LDA was found to better classify the majority group objects without a resampling method (ROS or RUS), a resampling method significantly improved the performance of LDA to classify the minority group objects. Nevertheless, the performance increment effect of the minority group object's classification was more significant than the performance decrement effect of the majority group object's classification.

On top of that, a resampling method improved the fairness of LDA in learning from objects regardless of group sizes. Consequently, it discounted the negative effect of the curse of class imbalance. Subsequently, four real data sets were used to compare the classification performance of the conventional LDA and the LDA with a resampling method. The results are tabulated in Table 8.

Based on Table 8, the advantage of using a resampling method could be witnessed clearly from three of the data sets, i.e. *abalone9-18*, *page-blocks-1-3_vs_4*, and *glass-0-1-6_vs_2*. For *shuttle-c0-vs-c4*, the performance difference in classifying the minority and majority group objects was negligible. For *abalone9-18*, the absolute differences in *TPR* and *TNR* between the conventional LDA and the LDA with a resampling method were approximately [0.21, 0.26] and [0.06, 0.08], respectively. The relative difference between the performance measures of the algorithms implied that the positive impact of employing a resampling method in the classification of the positives was 3.25 to 3.5 times larger than its negative impact in classifying the negatives. For *page-blocks-1-3_vs_4*, the absolute differences in *TPR* and *TNR* between the conventional LDA and the LDA with a resampling method were approximately [0.17, 0.29] and [0.07, 0.12], respectively. The relative difference between the performance measures of the algorithms indicated that the positive impact of employing a resampling method in the classification of the positives was roughly 2.42 to 2.43 times larger than its negative impact in classifying the negatives. For *glass-0-1-6_vs_2*, the absolute differences in *TPR* and *TNR* between the conventional LDA and the LDA with a resampling method were approximately [0.71, 0.88] and [0.19, 0.31], respectively. The relative difference between the performance measures of the algorithms signified that the positive impact of employing a resampling method in the classification of the positives was roughly 3.84 to 3.74 times larger than its negative impact in classifying the negatives. The results from primarily *abalone9-18*, *page-blocks-1-3_vs_4*, and *glass-0-1-6_vs_2* data sets further exhibited the improvement in the fairness of LDA towards discrimination learning from objects regardless of the group sizes through the employment of a resampling method (ROS or RUS).

DISCUSSION AND CONCLUSION

The findings from the means of *TPR* and *TNR* are in line with the findings from Jamaluddin and Mahat (2019) by which they enlightened the fact that the increment of the LDA's performance in classifying the minority group objects was more significant than its performance decrement in the majority group object classification relatively. On top of that, this study succeeded to

clearly quantify in detail on the significance of the increment as compared to the decrement using several statistics. This implies that if a study is interested in a minority group event, it is better to use a resampling method prior to LDA. Besides, the standard deviations of *TPR* and *TNR* have unveiled the reduction in the performance estimation bias of LDA upon employing a resampling method. The quantified significant reduction in the standard deviation of both *TPR* and *TNR* upon the employment of ROS or RUS could be witnessed based on the assessments using both simulated and real data sets. The significance of bias has been further explained by the three scenarios of classification cases and the derived explanation from the third scenario. This quantified reduction in the performance bias estimation depicts that the fairness of LDA to learn from objects regardless of the group sizes is improved by employing a resampling method.

In line with the findings from Jamaluddin and Mahat (2019), the findings of this study do not contradict with that from Xie and Qiu (2007) as well as Xue and Titterington (2008). The improvement in *AUC* as found by Xie and Qiu (2007) is directly parallel with the findings of this study as it was revealed that *TPR* has increased significantly. ROS or RUS succeeded to allow the significant improvement in the classification of the minority group objects (or the positives) and it directly led to the enhancement in *AUC* . Moreover, the findings of this study are indirectly parallel with that from Xue and Titterington (2008). It was clearly revealed that the decrement in *TNR* is significant. Although the ratio in the increment of *TPR* to the decrement of *TNR* is significantly large, a small decrement in *TNR* would yield a large decrement in *ER* as the majority group objects largely outnumbered the minority group objects in a data set with class imbalance. Nevertheless, this study succeeded to prove that a resampling method has improved the overall fairness of LDA in the classification of objects regardless of group sizes.

This manuscript recommends future studies to explore on the effect of a resampling method on LDA with varying class imbalance ratios. This is vital to quantify the needs of a treatment regarding the curse of class imbalance to avoid unnecessary computational costs as well as preserving the original characteristics of the imbalanced data. Apart from that, the separability of the groups of objects could affect the significance of class imbalance. If the groups are far from each other, then the objects could be separated easily even though the groups are largely imbalanced in sizes. Thus, the effect of class imbalance with varying group separability should be studied. This study also suggested for the investigation on the approaches to simultaneously cater several issues at once such as the curses of class imbalance and high dimensionality (either large number of variables or high-dimensionality-small-sample-size issue). In the event that the treatment of both curses require a sequence of distinct treatments, future work will have to identify which cure to treat first.

ACKNOWLEDGEMENT

This research is partially supported by the Ministry of Higher Education Malaysia under the Fundamental Research Grant Scheme [FRGS/1/2015/SG04/UUM/02/3]. An utmost note of gratitude is addressed to the reviewers for their constructive recommendations.

REFERENCES

Alcalá-Fdez, J., Fernández, A., Luengo, J., Derrac, J., García, S., Sánchez, L., & Herrera, F. (2011). KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *J. of Mult.-Valued Logic & Soft Computing*, *17*, 255–287.

Arlot, S., & Celisse, A. (2010). A survey of cross-validation procedures for model selection. *Statistics Surveys*, *4*, 40–79. https://doi.org/10.1214/09-SS054

Branco, P., Torgo, L., & Ribeiro, P. R. (2016). A survey of predictive modeling on imbalanced domains. *ACM Computing Surveys*, *49*(2), 31:1–31:50. https://doi.org/10.1145/2907070

Burman, P. (1989). A comparative study of ordinary cross-validation, v-fold cross-validation and the repeated learning-testing methods. *Biometrika*, *76*(3), 503–514. https://doi.org/10.2307/2336116

Das, S., Datta, S., & Chaudhuri, B. B. (2018). Handling data irregularities in classification: Foundations, trends, and future challenges. *Pattern Recognition*, *81*, 674–693.

Efron, B., & Tibshirani, R. (1993). *An introduction to the bootstrap*. New York: Chapman and Hall.

Efron, B., & Tibshirani, R. (1997). Improvements on cross-validation: The .632+ bootstrap method. *Journal of the American Statistical Association*, *92*(438), 548–560.

Geisser, S. (1975). The predictive sample reuse method with applications. *Journal of the American Statistical Association*, *70*(350), 320–328. https://doi.org/10.1080/01621459.1975.10479865

Genz, A., Bretz, F., Miwa, T., Mi, X., Leisch, F., Scheipl, F., & Hothorn, T. (2020). *mvtnorm: Multivariate normal and t distributions. R package version 1.1-1*.

Hairuddin, N. L., Mi Yusuf, L., & Othman, M. S. (2020). Gender classification on skeletal remains: Efficiency of metaheuristic algorithm method and optimized back propagation neural network. *Journal of Information and Communication Technology*, *19*(2), 251–277.

Jamaluddin, A.H & Mahat, N. I. (2019). The effects of resampling methods on linear discriminant analysis for data set with two imbalanced groups: An empirical evidence. *Advances and Applications in Statistics*, *59*(1), 17–42. https://doi.org/10.17654/AS059010017

Japkowicz, N. (2000). Learning from imbalanced data sets: A comparison of various strategies. In *AAAI Workshop on Learning from Imbalanced Data Sets* (Vol. 68, pp. 10–15).

Kaur, H., Pannu, H. S., & Malhi, A. K. (2019). A systematic review on imbalanced data challenges in machine learning: Applications and solutions. *ACM Comput. Surv.*, *52*(4). https://doi.org/10.1145/3343440

Kim, J.-H. (2009). Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap. *Computational Statistics & Data Analysis*, *53*(11), 3735–3745.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*, 1137–1143. http://dl.acm.org/citation.cfm?id=1643031.1643047

Kuhn, M. (2008). Building predictive models in R using the caret package. *Journal of Statistical Software*, *28*. https://doi.org/10.18637/jss.v028.i05

Roy, S., Ahmed, M., & Akhand, M. A. H. (2018). Noisy image classification using hybrid deep learning methods. *Journal of Information and Communication Technology*, *17*, 233–269.

Xie, J., & Qiu, Z. (2007). The effect of imbalanced data sets on LDA: A theoretical and empirical analysis. *Pattern Recognition*, *40*(2), 557–562. https://doi.org/10.1016/j.patcog.2006.01.009

Xue, J.-H., & Titterington, D. M. (2008). Do unbalanced data have a negative effect on LDA? *Pattern Recognition*, *41*(5), 1575–1588. https://doi.org/10.1016/j.patcog.2007.11.008