INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 16, Issue: 6, Month: December, Year: 2021
Article Number: 4291, https://doi.org/10.15837/ijccc.2021.6.4291

Tversky Similarity based Under Sampling with Gaussian Kernelized Decision Stump Adaboost Algorithm for Imbalanced Medical Data Classification

M. Kamaladevi, V. Venkatraman

M. Kamaladevi
Department of Computer Science,
SASTRA Deemed University-SRC campus Kumbakonam, India
kamaladeviresearch@gmail.com
Corresponding author

V. Venkatraman
Department of Mathematics
SASTRA Deemed University, Thanjavur , India
mathvvr@maths.sastra.edu

Abstract

In recent years, imbalanced data classification are utilized in several domains including, detecting fraudulent activities in banking sector, disease prediction in healthcare sector and so on. To solve the Imbalanced classification problem at data level, strategy such as undersampling or oversampling are widely used. Sampling technique pose a challenge of significant information loss. The proposed method involves two processes namely, undersampling and classification. First, undersampling is performed by means of Tversky Similarity Indexive Regression model. Here, regression along with the Tversky similarity index is used in analyzing the relationship between two instances from the dataset. Next, Gaussian Kernelized Decision stump AdaBoosting is used for classifying the instances into two classes. Here, the root node in the Decision Stump takes a decision on the basis of the Gaussian Kernel function, considering average of neighboring points accordingly the results is obtained at the leaf node. Weights are also adjusted to minimizing the training errors occurring during classification to find the best classifier. Experimental assessment is performed with two different imbalanced dataset (Pima Indian diabetes and Hepatitis dataset). Various performance metrics such as precision, recall, AUC under ROC score and F1-score are compared with the existing undersampling methods. Experimental results showed that prediction accuracy of minority class has improved and therefore minimizing false positive and false negative.

Keywords: Data Imbalance, Undersampling, Tversky, Similarity Indexive Regression, Gaussian Kernelized, Decision Stump AdaBoosting.

1 Introduction

In the field of data mining, one of the significant researches of interests is imbalanced data classification. Imbalanced data constitutes to the fact that the sample frequency in the majority class is higher that the sample frequency in minority class, commonly found in diagnosing malignant tumors concerning medical diagnosis. In general, the class imbalanced problem is solved via classification models, via preprocessing or combining both the preprocessing and classification models. Conventional classification methods proceed in the assumption that the training sets are balanced. Hence, in imbalanced dataset classification, minority class is underrepresented and hence involves a cumbersome process where misclassifying minority class usually tend to some undesirable effect in prediction.

An edited nearest neighbor algorithm [1] split into three parts. First, with the purpose of increasing the samples in minority class, minority over sampling technique was utilized. Then, in second step eliminate the noise from majority samples. Finally, Random Forest (RF) was applied to perform classification where the Matthews Correlation Coefficient (MCC) used as termination condition for the iteration. F-score has shows significance change which exhibit improvement in Classifier performance for Imbalanced dataset.

Random undersampling and ensemble of classifier chains make multi-label learning trained from positive and negative examples of Imbalanced dataset[2]. Here, for each label differing number of binary models and then ensemble chains pertaining to dissimilar sizes were constructed to minimize computational budget . Experimental results show improvement in F- measure value. An undersampling framework for addressing class imbalance using neighborhood based undersampling method, Tomek Link was proposed [3]. Here, class imbalance was addressed by eliminating the significant overlapped data points. Moreover, it also identified and removed majority class instances over the overlapping segment. Removing these irrelevant instances in turn increased the minority class instance visibility and also reduced elevated data elimination, therefore minimizing the information loss to great extent.

2 Related works

The issue of data imbalance is said to arise during classification whenever the observation in one of the major class exceeds the observation in the other minority class. Conventional classification mechanisms are highly prone to imbalance data and hence resulting in bias. This negative aspect influences the classification results. Mutual class potential was utilized in [4] via radial bias that in turn reduced the time complexity involved in imbalanced data classification. However, less concentration was made on the part of false negative rate. To address this issue, unstructured data classification was performed by applying uncertain nearest neighbor decision rule [5], therefore resulting in accuracy.

With the applications of the conventional learning algorithms, though data imbalance status may permit high global accuracy, but it faces a real challenge when taking into consideration the accuracy of minority class. To deal with this aspect, a novel application of the decision tree algorithm to imbalanced data situations was applied in [6]. With this higher prediction was said to be arrived at.

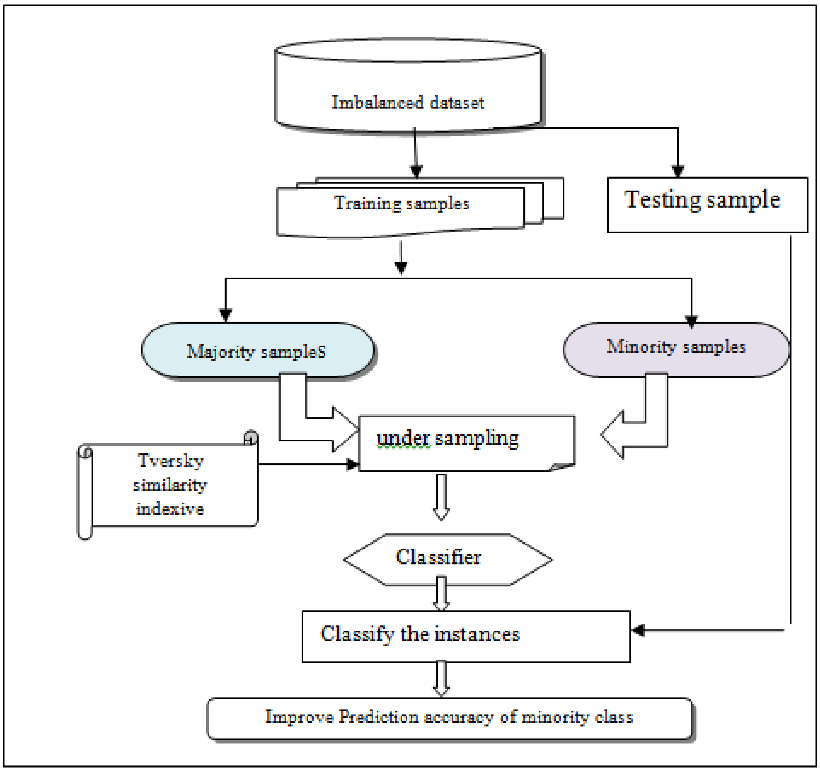
By entirely overlooking the majority class, less importance on class imbalance issue, poses negative influence. To address this issue, a framework for synthetic oversampling that was found to be robust on cases of extreme imbalance data [7]. Despite improvement observed in prediction, however, convergence speed was not said to be improved. In order to speed up the classification process, a branch and bound graph edit distance method was proposed [8], therefore contributing to average execution time.

A novel algorithm that used the furthest neighbor of a candidate example to produce new set of synthetic samples was proposed [9], therefore resulting in the improvement of precision recall space performance. Yet another method used in improving the classification issues by employing local Mahalanobis distance learning (LMDL) method [10]. With this the accuracy and precision factors were said to be improved considerably. However another method called, Particle Stacking Undersampling method was presentedto achieve methodological improvements in terms of time.

In spite of the success of these techniques, all of them have their own advantages and disadvantages and none of them is ideal, and hence there is room for improvement in this research line. This paper proposes an efficient method to enhance learning using Tverskey Similarity Indexive Regression and Gaussian Kernelized Decision Stump-based Adaboosting (TSIR-GKDSA) from imbalanced datasets, whose results indicate that it is better than some amateur, state-of-art methods.

3 Methodology

A synthesis of objectives is adapted in the proposed method with an effective manner to learn from imbalanced data. The objectives considered in this paper are noise and borderline data, where noisy values in the dataset are handled by taking the mean values of that column and in case of borderline data, it is reanalyzed and identify the probability of classes that data that belongs to and then categorizing the data into that specific class. The adapted majority class is then integrated with minority instances to bestow new training set for learning. The proposed method is shown in two steps, undersampling stage and classification stage. The representation of the proposed method Tverskey Similarity Indexive Regression and Gaussian Kernelized Decision Stump-based Adaboosting (TSIR-GKDSA) for imbalanced data classification is shown in Figure 1.



As illustrated in the above figure, in the proposed method, at initial the training sets are fed to the undersampling stage. This stage performs the undersampling of the majority-class instances based on the aspects of noise and borderline data by utilizing Tversky Similarity Indexive Regression Undersampling model. The rectified training set is then given to the classification stage to sort out classification task. The steps are examined in detail as follows.

3.1 Tversky Similarity Indexive Regression Under sampling model

An imbalanced class possesses one or more classes with few examples (i.e., minority class) and one or more classes with many examples (i.e., majority class). Undersampling refers to the process of sampling the majority class and making it equal to the minority class. This process is performed by applying Tversky Similarity Indexive Regression model. Figure 2 shows the flow diagram of Tversky Similarity Indexive Regression Undersampling model. Here, regression a machine learning technique in addition to Tversky Similarity Index is utilized to analyze the relationship between two variables like instances from the dataset. By employing regression the strength of relationship between one dependent variable (i.e. target variable or outcome 'i.e.,Q=(q_1,q_2,. . . ,q_n)') and a series of other independent variables (i.e., predictor variable 'i.e.,P=(p_1,p_2,. . . ,p_n)') are determined. Let us consider the classes $'Cl = cl_1, cl_2, \ldots, cl_n \& Cl \varepsilon maj_{Cl}, min_{Cl}'$, where '$C = c_{1n}, c_{2n}, \ldots, c'_{nn}$ represents the columns in the input dataset (i.e., PIMA Indian Diabetes and Hepatitis). Here 'n=9' and 'n=20' in PIMA Indian Diabetes and Hepatitis respectively. The columns from the respective dataset are split into different numbers of classes into a data matrix and are represented as given below.

$$\begin{bmatrix} c_{11} & c_{12} & \ldots & c_{1n} \\ c_{21} & c_{22} & \ldots & c_{2n} \\ \cdots & \cdots & \ldots & \cdots \\ c_{n1} & c_{n2} & \ldots & c_{nn} \end{bmatrix} \text{———-1}$$

From the above equation (1), 'c'_{11}, represent the column class segregation for the first feature (i.e., Pregnancies in PIMA Indian Diabetes). In a similar manner, 'c_21' represent the column class segregation for the second feature (i.e., Blood Pressure in PIMA Indian Diabetes) and so on. In order to regress sequence between one dependent variable and many independent variable 'Q on P', for each column 'C' in the classes 'Cl', the straightforward model of '$Q = \alpha_0 + \alpha_1 P + \varepsilon'$ is examined.

Here '$\varepsilon = \varepsilon_1, \varepsilon_2, \ldots, \varepsilon'_n$ refers to the error that interprets for the variance between the majority and minority class so that they reduce the error. To reduce the error, both the dependent variable and the independent variable are differentiated with respect to '$\alpha'_0 and`\alpha'_1$ and is mathematically formulated as given below.

$$\frac{\partial \sum_{i=1}^{n} \epsilon_i^2}{\partial \alpha_0} = -2 \sum_{i=1}^{n} [(q_i - (\alpha_0 + \alpha_1 p_i))] = 0 \quad \text{-----2}$$

$$\frac{\partial \sum_{i=1}^{n} \epsilon_i^2}{\partial \alpha_1} = -2 \sum_{i=1}^{n} p_i [(q_i - (\alpha_0 + \alpha_1 p_i))] = 0 \quad \text{---3}$$

By reorganizing the above two equations (2) and (3) are mathematically formulated as given below.

$$\alpha_0 n + \alpha_1 \sum_{i=1}^{n} p_i = \sum_{i=1}^{n} q_i \quad \text{-----4}$$

$$\alpha_0 \sum_{i=1}^{n} p_i + \alpha_1 \sum_{i=1}^{n} p_i^2 = \sum_{i=1}^{n} p_i q_i \quad \text{----5}$$

By figuring out the equations in (4) and (5), '$\alpha'_1 and`\alpha'_0$ are mathematically evaluated as given below.

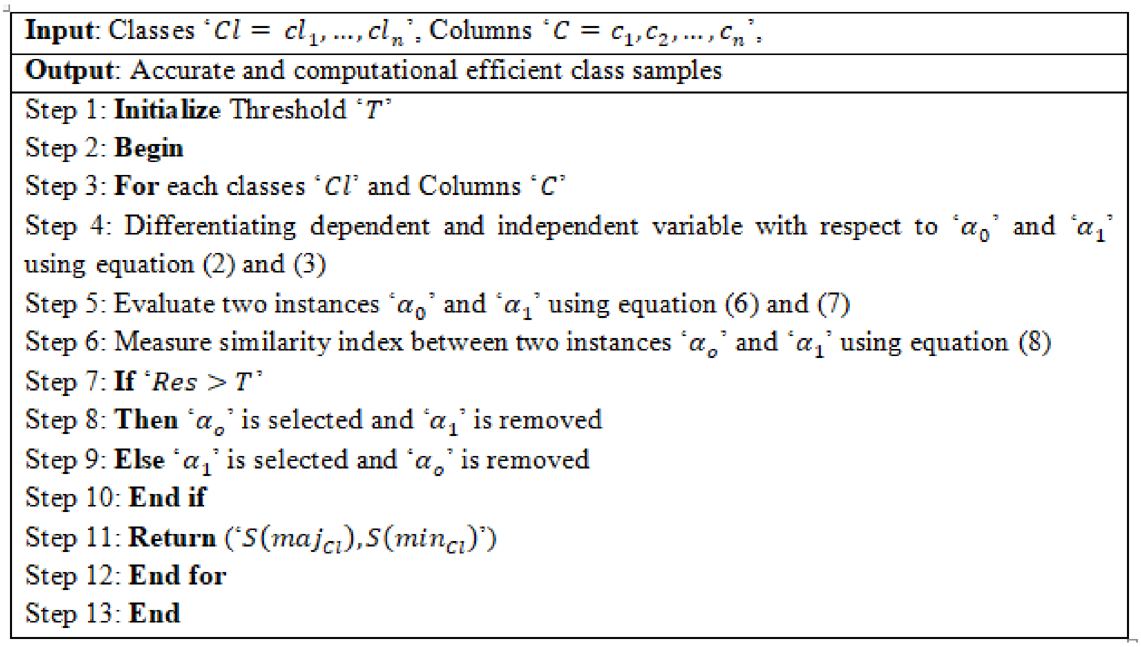
$$\alpha_1 = \frac{\sum_{i=1}^{n} (p_i - P')(q_i - Q')}{\sum_{i=1}^{n} (p_i - P')^2} \quad \text{----6}$$

$$\alpha_o = Q' - \alpha_1 P'; where\ Q' = \frac{1}{n}\sum_{i=1}^{n} q_i\ ;\ P' = \frac{1}{n}\sum_{i=1}^{n} p_i \text{ ----7}$$

Finally, the similarity index between two instances '$\alpha_0' and '\alpha_1'$ is identified via Tversky Similarity Index function. This is mathematically formulated as given below. From the above equation (8), 'FF'

$$Res = SIM(\alpha_0, \alpha_1) = \frac{FF(\alpha_0, \alpha_1)}{UF(\alpha_0) + UF(\alpha_1) + FF(\alpha_0, \alpha_1)} \text{----8}$$

refers to the frequent features and 'UF' refers to the unique features that analyze the relationship between two variables like instances from dataset that in turn minimizes the computation time involved in balancing data with high accuracy. The pseudo code representation of Tversky Similarity Regressive Undersampling is given below. As given in the above Tversky Similarity Regressive Undersampling algorithm, for each classes and columns initialized from two datasets (PIMA Indian Diabetes and Hepatitis), the objective of this algorithm remains in obtaining the undersampling process with minimum time and maximum accuracy. To achieve this objective, first, dependent and independent variable are differentiated by means of regression function. Then with the aid of the Tversky similarity index provides with the output values in the ranges from '0' to '1'. Next, a threshold value is then initialized. If the similarity value is greater than the threshold, one instance/ sample is selected 'α_0' from the majority class and the other one 'α_1' is removed. In this way, all the collected samples are selected and removed from the majority class. Accurate identification and removal of these instances enhance the visibility of the minority class instances and simultaneously reduces the excessive removal of data, which decreases the information loss.



```
Input: Classes '$Cl = cl_1, \ldots, cl_n$', Columns '$C = c_1, c_2, \ldots, c_n$',
Output: Accurate and computational efficient class samples
Step 1: Initialize Threshold '$T$'
Step 2: Begin
Step 3: For each classes '$Cl$' and Columns '$C$'
Step 4: Differentiating dependent and independent variable with respect to '$\alpha_0$' and '$\alpha_1$' using equation (2) and (3)
Step 5: Evaluate two instances '$\alpha_0$' and '$\alpha_1$' using equation (6) and (7)
Step 6: Measure similarity index between two instances '$\alpha_o$' and '$\alpha_1$' using equation (8)
Step 7: If '$Res > T$'
Step 8: Then '$\alpha_o$' is selected and '$\alpha_1$' is removed
Step 9: Else '$\alpha_1$' is selected and '$\alpha_o$' is removed
Step 10: End if
Step 11: Return ('$S(maj_{Cl}), S(min_{Cl})$')
Step 12: End for
Step 13: End
```

Figure 1: Algorithm 1 Tversky Similarity Regressive Undersampling

3.2 Gaussian Kernelized Decision Stump Adaboosting Classification model

With the obtained samples from majority class and initial samples in the minority class, actual classification process is performed. In our work, an ensemble classifier based on Gaussian kernelized decision stump Adaboost classification model is utilized for classifying the instances into two classes. Let '$f^{(l)} = [f_1^{(l)}, f_2^{(l)}, \ldots, f_n^{(l)}]$'represents the 'l-th' training set of feature vectors of Pima Indian diabetes and Hepatitis dataset and 'd^1' representing the actual data (i.e. label) corresponding to 'f^1'. In addition to the weak DS and strong classifiers, AdaBoost comprises of sub-classifier 'S_c_i' to address the limitations of individual classifiers. The main concept is to train different sub-classifier 'SC_i,i=(1,2,. . . ,I)' on the same training samples. The weak classifiers from DS are grouped finally to obtain a stronger classifier. Figure 3 shows the flowchart. As shown in the above figure [2] to start

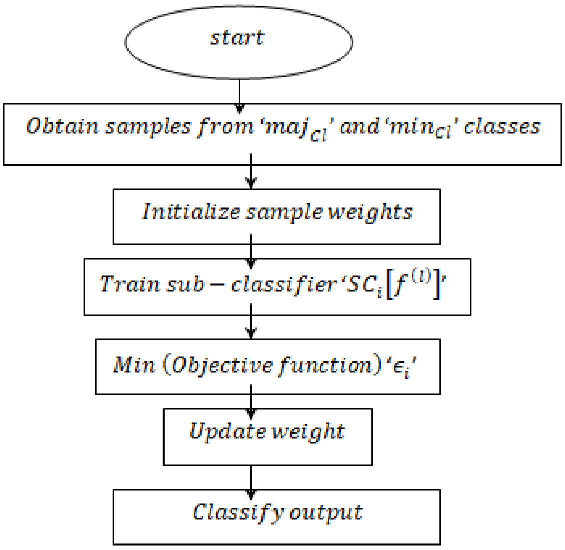

start
Obtain samples from 'maj_cl' and 'min_cl' classes
Initialize sample weights
Train sub – classifier 'SC_i[f^(l)]'
Min (Objective function) 'ϵ_i'
Update weight
Classify output

Figure 2: Flowchart of the Gaussian Kernelized Decision Stump Classification

with the sample weights are initialized as given below. With the weights initialized as given in (9),

$$W(l) = \frac{1}{L}, l = 1, 2, \ldots, L \text{-----}9$$

the sub-classifiers are then trained by minimizing the objective function is mathematically expressed as given below

$$SC_i[f^{(l)}], i = 1, 2, \ldots I \text{---}10$$

$$\epsilon_i = \sum_{l=1}^{L} FD_i(l)C[SC_i(f^{(l)}) \neq d^{(l)}] \text{----}11$$

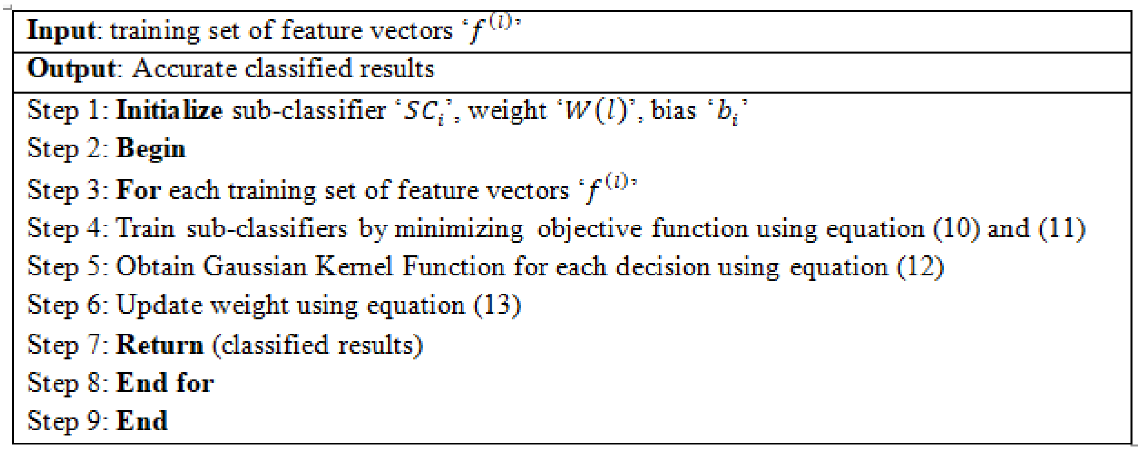
From the above equations (10) and (11), sub-classifier training 'SC_i' are performed for all the training set of feature vector 'f^1'. These are performed using the objective function with feature decision of the 'i-th' instance represented by 'FD_i' for the 'l-th' training instance vector. The criterion function 'C' that takes the value '1' if its argument is true and '0' otherwise using Gaussian Kernel Function by considering average of neighboring points. This is mathematically expressed as given below.

$$d^{(l)} = \begin{cases} 1, if\ SC_i W_i + b_i \geq 1 \\ -1, if\ SC_i W_i + b_i \leq -1 \end{cases} \text{----}12$$

From the above equation (12), 'SC_i W_i+b_i' refers to the output result (i.e. either diabetic or non-diabetic), with 'W_i' and 'b_i' representing the weight and bias of the sub-classifier. Finally, the weight is updated as given below.

$$W_{i+1}(l) = \frac{W_i(l)\exp[-d^{(l)}SC_i(f^{(l)})]}{\sum_{l=1}^{L} W_i(l)\exp[-d^{(l)}SC_i(f^{(l)})]} \text{----}13$$

The pseudo code representation of Gaussian Kernelized Decision Stump Reweight Classification is given below.



```
Input: training set of feature vectors '$f^{(l)}$'
Output: Accurate classified results
Step 1: Initialize sub-classifier '$SC_i$', weight '$W(l)$', bias '$b_i$'
Step 2: Begin
Step 3: For each training set of feature vectors '$f^{(l)}$'
Step 4: Train sub-classifiers by minimizing objective function using equation (10) and (11)
Step 5: Obtain Gaussian Kernel Function for each decision using equation (12)
Step 6: Update weight using equation (13)
Step 7: Return (classified results)
Step 8: End for
Step 9: End
```

Figure 3: lgorithm 2 Gaussian Kernelized Decision Stump Adaboost Classification

As given in the above Gaussian Kernelized Decision Stump Adaboost Classification algorithm, for each training set of feature vectors, the proposed method being an ensemble classifier provides strong classification results by combining the output of weak hypothesis. This is performed in our work by applying a decision stump (DS), a one-level decision tree possessing one root node that is immediately connected to the leaf node. The root node takes a decision based on the Gaussian kernel function, i.e., by considering the average of the neighboring points. Based on the results, the two classes of instances i.e. 1 and 0 are obtained at the leaf node. In addition with the weak hypothesis possessing certain significant training errors that in turn minimizes the classification results, in our work, the problem is rectified by integrating the results of all weak hypotheses to make a strong classification. After combining the weak hypothesis, the error is calculated and accordingly the weights are adjusted to find the best classifier. With this, the prediction accuracy of minority class is said to be improved by reducing false positive and false negative.

4 Experimental setup

Two imbalanced datasets, Pima Indian diabetes (https://www.kaggle.com/uciml/pima-indians-diabetes-database) and Hepatitis (https://www.kaggle.com/harinir/hepatitis) is taken from the Kaggle for as binary classification problems (i.e. 2 classes). The Pima Indian diabetes dataset comprises of 768 instances/ samples and 8 numeric attributes with one class attribute (i.e. outcome 1 or 0). Among the 768 samples, 500 majority class (i.e. class 0) and 268 minority class (i.e. class 1) are obtained. In a similar manner, the Hepatitis dataset comprises of 20 columns. Simulations are conducted with the benchmark dataset. Five performance parameters, such as accuracy, precision, recall, F1-score AUC under ROC score are measured. Fair comparison is made with the proposed Tverskey Similarity Regression and Gaussian Decision Stump-based Adaboosting (TSIR-GKDSA) for imbalanced data classification and existing four undersampling methods, edited nearest neighbor rule [1], random undersampling [2] tomeklinks [3], NearMiss [4] applied to four different classifier .

4.1 Dataset description

The PIMA Indian Diabetes Dataset includes details about female patients with minimum twenty one year age of Pima Indian population obtained from UCI machine learning repository, originally

Table 1: PIMA Indian Diabetes Dataset

Attribute number	**Attribute**
F1	Pregnancy
F2	Plasma glucose concentration
F3	Diastolic blood pressure
F4	Triceps skin fold
F5	Serum insulin
F6	Body mass index
F7	Diabetes pedigree
F8	Age

Table 2: Hepatitis Dataset

Attribute number	**Attribute**
F1	Class
F2	Age
F3	Sex
F4	Steriod
F5	Antivirals
F6	Fatigue
F7	Malaise
F8	Anorexia
F9	Liver big
F10	Liver firm
F11	Spleen palpable
F12	Spiders
F13	Ascites
F14	Bilirubin
F15	Alk phospate
F16	SGOT
F17	Albumin
F18	Protime
F19	Histology

owned by the National Institute of diabetes and digestive and kidney diseases. A total of 768 instances are present in this dataset. They are classified into two different classes, as diabetic and non diabetic involving eight risk factors. The eight risk factors are number of time pregnant, plasma glucose concentration of two hours in an oral glucose tolerance test, diastolic blood pressure, triceps skin fold thickness, two-hour serum insulin, body mass index, diabetes pedigree function and age as provided in table 1 and the features in Hepatitis dataset are provided in table 2.

4.2 Case Scenario for medical imbalanced data classification using PIMA Indian Diabetes Dataset

In this section, a case scenario for medical imbalanced data classification using PIMA Indian Diabetes Dataset is provided. As provided in the above table 1, a distinctly fascinating attribute or feature utilized was the Diabetes Pedigree Function, ‘pedi- F7’, that contributed certain data on diabetes mellitus genetics arising from relatives, also displaying the genetic relationship of those relatives to the patient (i.e., analyzing the relationship between two instances from the dataset). This estimation of genetic impact gave us a conception of the genetic likelihood one might have with the outbreak of diabetes mellitus. On the basis of this consideration in the proceeding section, it is ambiguous how well this diabetes pedigree function predicts or classifies the outbreak of diabetes. Initially, the columns ‘n=8’ in PIMA Indian Diabetes Dataset are split into different numbers of classes into a data matrix as given below.

$$DM = \begin{bmatrix} 6 & 1 & 8 & 1 & 0 \\ 148 & 85 & 183 & 89 & 137 \\ 72 & 66 & 64 & 66 & 40 \\ 35 & 29 & 0 & 23 & 35 \\ 0 & 0 & 0 & 94 & 168 \\ 33.6 & 36.6 & 25.3 & 28.4 & 43.1 \\ 0.627 & 0.351 & 0.672 & 0.167 & 2.288 \\ 50 & 31 & 32 & 21 & 23) \end{bmatrix}$$

To start with each attribute or features in the dataset were inspected and accordingly the scattering features were reviewed and the following inferences were made.

1. Features to be integers were the pregnancy 'F1' and age 'F8'
2. Population is by and large smaller with less than 50 years old
3. From equations (6) and (7) no relationship between two variables (i.e., between age and onset of diabetes is said to exist) and also no relationship between pedi function and onset of diabetes is said to exist.
4. Certain attributes or features possessed a value of zero that are found to be errors in the data, like plasma glucose concentration 'F2', blood pressure 'F3', skin thickness 'F4', insulin 'F5' and BMI 'F6'.
5. Upon implementation of the distribution of several classes, as given above, 500 negatives instances were appeared ('65.1%') and 258 positive instances ('34.9%') were appeared to an overall of 758 instances.
6. From the above results, the minority classes were found to be ('34.9%') and majority classes were found to be ('65.1%').
7. The samples obtained from minority classes and majority classes were then provided as input for classification of results. Initially, the weight and bias was initialized to '-0.2 to 2.0', 1.0 respectively.
8. The sub-classifiers, from both the classes were trained according to the weight and bias value, by considering average of neighboring points. The classified results were arrived at.
9. The classification performance is evaluated in the next section.

4.3 Performance Analysis using precision

First, a significant metric used for analyzing medical data classification is precision. Precision is mathematically expressed as below.

$$P = \frac{TP}{TP+FP} * 100 - - - 14$$

From the above equation (14), precision 'P' is arrived at based on the ratio of the true positive and false positive rate. Here, the true positive 'TP' refers to the number of positive instances classified as positive and false positive 'FP' refers to the number of negative instances classified as positive. Table 3 given below lists the summary results of precision of proposed system TSR-GDSA, and exiting methods such as edited nearest neighbor rule [1], random undersampling [2] and tomeklins [3] and Near Miss[4] applied in four different classifier Logistic regression, Support vector Machine, K-Nearest Neighbour and Random forest Figure 4 shows the graphical representation of precision compared with TSIR-GKDSA and other undersampling method applied in various classifier for the two datasets. Different

Table 3: Summary of precision of PIMA Indian Diabetes and Hepatitis dataset using TSIR-GKDSA, and other Undersampling Methods

Attribute number	**Attribute**
F1	Class
F2	Age
F3	Sex
F4	Steriod
F5	Antivirals
F6	Fatigue
F7	Malaise
F8	Anorexia
F9	Liver big
F10	Liver firm
F11	Spleen palpable
F12	Spiders
F13	Ascites
F14	Bilirubin
F15	Alk phospate
F16	SGOT
F17	Albumin
F18	Protime
F19	Histology

Undersampling procedure mentioned in x axis and the y axis denotes the precision results obtained for all the five method applied in different classifier. The reason for precision improvement using TSIR-GKDSA is due to the application of Tversky Similarity Regressive Undersampling algorithm. By applying this algorithm, initially, dependent and independent variable were delineated by applying the regression function. Then, with the results obtained, Tversky similarity index was utilized that in turn resulted in the output values in the ranges from '0' to '1'. With this, correct instances were classified as positive and reduced the number of negative instances to be classified as positive.

4.4 Performance Analysis using recall

The second metric used in our work for medical data classification analysis is recall. and is mathematically expressed as given below.

$$R = \frac{TP}{TP+FN} * 100 - - - - - - 15$$

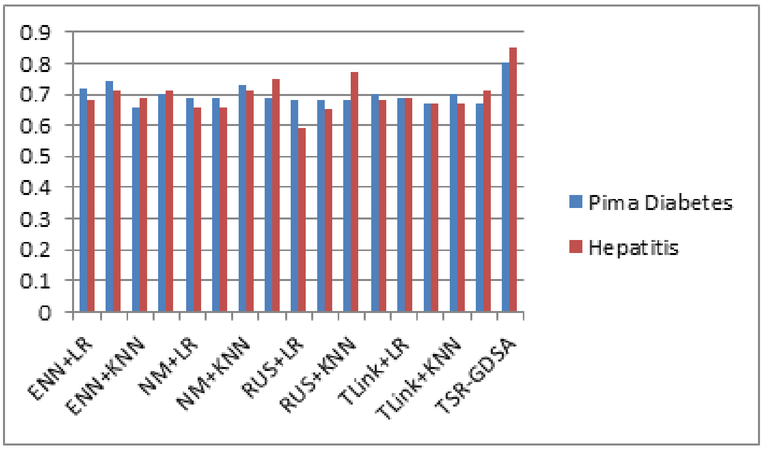


Figure 4: raphical representation of precision

Table 4: Summary of recall using PIMA Indian Diabetes and Hepatitis dataset for TSR-GDSA and existing Undersampling method

Undersampling Technique	**Classifier**	**Pima**	**Hepatitis**
ENN	Logistic regression	0.73	0.76
	SVM	0.75	0.79
	K-Nearest Neighbour	0.67	0.69
	Random Forest	0.72	0.79
Near Miss	Logistic Regression	0.70	0.77
	SVM	0.71	0.77
	K-Nearest Neighbour	0.75	0.79
	Random Forest	0.69	0.88
Random Undersampler	Logistic Regression	0.68	0.65
	SVM	0.68	0.73
	K-Nearest Neighbour	0.68	0.81
	Random Forest	0.7	0.80
Tomeklinks	Logistic Regression	0.68	0.68
	SVM	0.67	0.67
	K-Nearest Neighbour	0.70	0.71
	Random Forest	0.67	0.67
TSIR-GKDSA		0.78	0.83

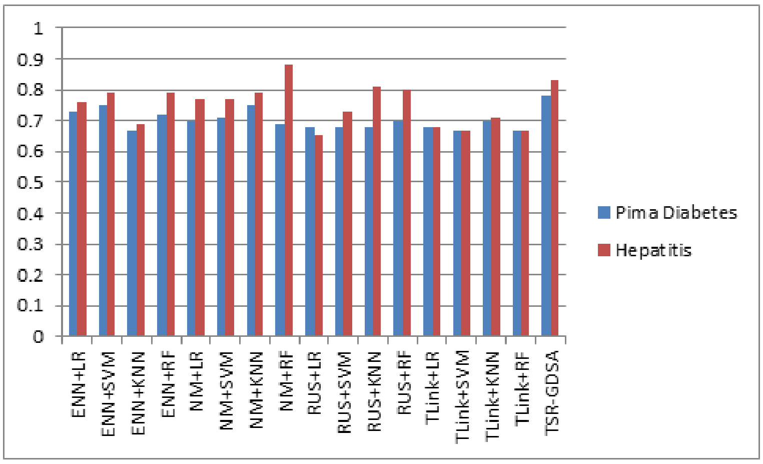


Figure 5: raphical representation of recall

Figure 4 given above depicts the graphical representation of recall using five different methods with the aid of PIMA Indian Diabetes Dataset and Hepatitis Dataset. With recall referring to the ratio of actual positive cases (i.e., correct classification) that were correctly classified as such, analysis were made for different classifier applied in from both the datasets. From the results it is inferred that the recall rate is said to be improved using TSIR-GKDSA when compared to the Four methods. The reason for improvement is the application of Gaussian Kernelized Decision Stump Adaboosting algorithm. By applying this algorithm ensemble classification is said to be performed using our work where the strong classification results are combined with the output of weak hypothesis via decision stump (DS). The advantage of DS being a one-level decision tree possessing one root node that is immediately connected to the leaf node, where classification is made by means of the Gaussian kernel function, therefore reducing the number of positive instances or classes as negative.

4.5 Performance Analysis using F1-score

F1 score refers to the weighted average of precision and recall. This score therefore considers both the false positives and false negatives into consideration. In case of uneven class distribution,

Table 5: Summary of F1-score using four different methods and TSIR-GKDSA.

Undersampling Technique	**Classifier**	**Pima**	**Hepatitis**
ENN	Logistic regression	0.71	0.69
	SVM	0.74	0.74
	K-Nearest Neighbour	0.65	0.69
	Random Forest	0.68	0.74
Near Miss	Logistic Regression	0.69	0.6
	SVM	0.70	0.6
	K-Nearest Neighbour	0.69	0.74
	Random Forest	0.73	0.75
Random Undersampler	Logistic Regression	0.69	0.54
	SVM	0.68	0.65
	K-Nearest Neighbour	0.69	0.79
	Random Forest	0.68	0.64
Tomeklinks	Logistic Regression	0.69	0.69
	SVM	0.67	0.67
	K-Nearest Neighbour	0.67	0.67
	Random Forest	0.70	0.71
TSIR-GKDSA		0.96	0.97

F1 score is considered to be the most useful metric. In this section, the F1 score is evaluated and is mathematically formulated as given below.

$$F1 - score = 2 * \frac{R*P}{R+P} \text{——} 17$$

From the above equation (17), the 'F1-score' is evaluated based on the recall value 'R' and the precision value 'P' respectively. Figure 5 given above shows the F1-score performance of four different methods, PIMA Indian Diabetes and Hepatitis dataset and proposed TSIR-GKDSA. From the results it is inferred that the F1-score using TSIR-GKDSA for both the data sets was improved than when compared to the state-of-the-art methods. The improvement in F1-score is due to the incorporation of Tversky Similarity function that initially obtains the relationship between two variables, therefore resulting in the accurate identification.

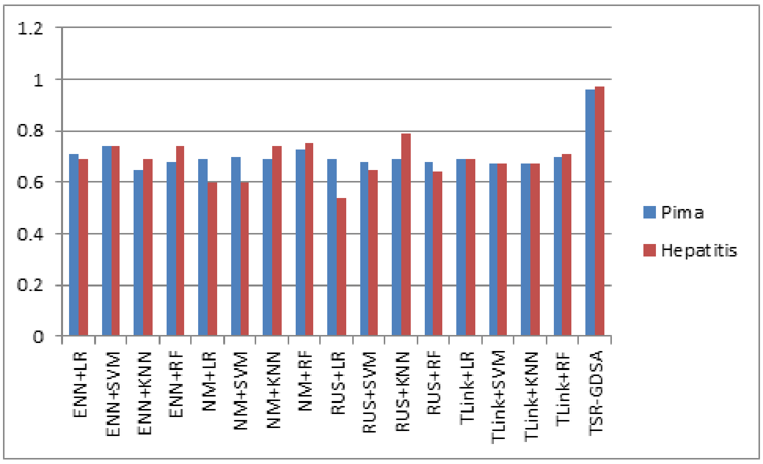


Figure 6: raphical representation of recall

4.6 Performance Analysis using Accuracy

Accuracy is measured by
Accuracy=TP+TN/TP+FP+TN+FN

Table 6: Summary of accuracy of TSIR-GKDSA with other undersampling method

Undersampling Technique	**Classifier**	**Pima**	**Hepatitis**
ENN	Logistic regression	0.72	0.7
	SVM	0.72	0.82
	K-Nearest Neighbour	0.66	0.82
	Random Forest	0.69	0.82
Near Miss	Logistic Regression	0.71	0.64
	SVM	0.71	0.64
	K-Nearest Neighbour	0.72	0.82
	Random Forest	0.74	0.82
Random Undersampler	Logistic Regression	0.71	0.59
	SVM	0.68	0.73
	K-Nearest Neighbour	0.71	0.77
	Random Forest	0.7	0.68
Tomeklinks	Logistic Regression	0.72	0.72
	SVM	0.70	0.70
	K-Nearest Neighbour	0.72	0.7
	Random Forest	0.72	0.73
TSIR-GKDSA		0.75	0.83

Accuracy can be compared with proposed architecture and other state of art undersampling method for the two dataset

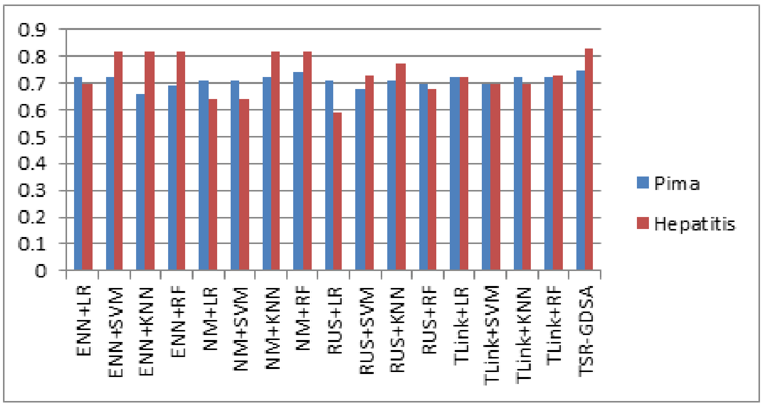


Figure 7: raphical representation of recall

From the fig [6],proposed system show high accuracy when compared with existing methods

4.7 Performance analysis using AUC Under ROC score

Finally, AUR or the area under ROC curve is obtained that evaluates the two dimensional area underneath the ROC curve. The Area Under the Curve (AUC) measures the potentiality of the sampling technique for imbalanced data classification with the purpose of differentiating between classes. The higher the AUC better the performance between the positive and negative classes. In addition, it also provides an aggregate or total measurement of performance across all probable thresholds. Table 7 given below provides the roc_auc_score measure for five different methods, using two different datasets.

Figure 7 given above illustrates the AUC curve performance of four different methods, PIMA Indian Diabetes and Hepatitis dataset for TSR-GDSA, edited nearest neighbor rule, random under sampling and tomeklins and NearMiss The AUC curve is measured with respect to the true positive rate and false positive rate. Here, false positive rate refers to the rate or ratio of probability of falsely rejecting the null hypothesis for a particular test (i.e., falsely rejecting the null hypothesis for imbalanced data

Table 7: Summary of roc_auc_score using PIMA Indian Diabetes and Hepatitis dataset for TSIR-GKDSA with other Undersampling method

Undersampling Technique	**Classifier**	**Pima**	**Hepatitis**
ENN	Logistic regression	0.79	0.87
	SVM	0.81	0.9
	K-Nearest Neighbour	0.82	0.94
	Random Forest	0.76	0.92
Near Miss	Logistic Regression	0.77	0.84
	SVM	0.75	0.94
	K-Nearest Neighbour	0.77	0.92
	Random Forest	0.75	0.95
Random Undersampler	Logistic Regression	0.79	0.84
	SVM	0.81	0.9
	K-Nearest Neighbour	0.81	0.93
	Random Forest	0.75	0.94
Tomeklinks	Logistic Regression	0.79	0.79
	SVM	0.81	0.81
	K-Nearest Neighbour	0.82	0.76
	Random Forest	0.76	0.81
TSIR-GKDSA		0.94	0.95

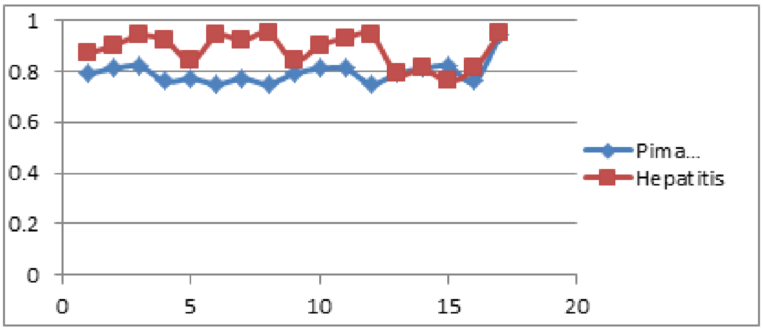


Figure 8: raphical representations of roc_auc score

classification) and on the other hand true positive rate refers to the actual positive will test positive. Higher roc_auc score expresses a good test and from the samples though increase is substantially found in all the foure methods using both the datasets, comparative analysis shows better results using TSIR-GKDSA. This is because of the application of the proposed probability sampling using random sampling is applied for imbalanced data classification where each sample has an equal chance or probability of being selected.

5 Conclusion

In this paper, an undersampling and classification solution to solve the class imbalance issue is proposed. The underlying study reveals the fact that with proper undersampling and classification reduce the false alarms in medical imbalanced data. The proposed method incorporates the elimination of noise and border line data instances to the baseline and explored Tverskey Similarity Regression and Gaussian Decision Stump-based Adaboosting (TSR-GDSA) method. Moreover, Gaussian Kernelized Decision Stump Boosting is employed with the objective of reducing the training errors occurring during classification so that best classifier is obtained. The results obtained show significant improvement over the conventional data classification method. The performance of the models is validated with various performance measures such as, precision, recall, F1 score, Accuracy and AUC under ROC. The results achieved are quite satisfactory.

Ethics Approval and Consent to Participate

No participation of humans takes place in this implementation process

Human and Animal Rights

No violation of Human and Animal Rights is involved.

Funding

No funding is involved in this work.

Conflict of interest

Conflict of Interest is not applicable in this work.

Authorship contributions

There is no authorship contribution.

Acknowledgement

There is no acknowledgement involved in this work.

References

[1] Zhaozhao Xu, Derong Shen, Tiezheng Nie, Yue Kou, (2020). “A hybrid sampling algorithm combining M SMOTE and ENN based on Random Forest for medical imbalanced data”, Journal of Biomedical Informatics, Elsevier, [edited nearest neighbor rule].

[2] Bin Liu, Grigorios Tsoumakas ,(2020). “Dealing with class imbalance in classifier chains via random undersampling”, Pattern Recognition, Elsevier, Volume 102, Pages 1-34 [random undersampling]

[3] Pattaramon Vuttipittayamongko, Eyad Elyan, (2019).“Neighbourhood-based undersampling approach for handling imbalanced and overlapped data”, Information Sciences, Elsevier,[tomeklinks]

[4] MichałKoziarski, ,(2020).“Radial-Based Undersampling for imbalanced data classification”, Pattern Recognition, Elsevier.

[5] Nijaguna Gollara Siddappa, Thippeswamy Kampalappa, “Adaptive Condensed Nearest Neighbor for Imbalance Data Classification “, International Journal of Intelligent Engineering & Systems

[6] Ikram Chaabane, Radhouane Guermazi, Mohamed Hammami, (2019). “Enhancing techniques for learning decision trees from imbalanced data”, Advances in Data Analysis and Classification, Springer.

[7] Colin Bellinger, Shiven Sharma, Nathalie Japkowicz, Osmar R. Zaïane,(2019). “Framework for extreme imbalance classification: SWIM—sampling with themajority class”, Knowledge and Information Systems, Springer,

[8] Zeina Abu-Aisheh, Romain Raveaux, Jean-Yves Ramel (2018).“Efficient k-nearest neighbors search in graph space”, Pattern Recognition Letters, Elsevier

[9] Ahmad S. Tarawneh, Ahmad B. A. Hassanat, Khalid Almohammadi, Dmitry Chetverikov, Colin Bellinge, (2020). “SMOTEFUNA: Synthetic Minority Over-Sampling Technique Based on Furthest Neighbour Algorithm”, IEEE Access.

[10] Nijaguna Gollara Siddappa, Thippeswamy Kampalappa,(2020).“Imbalance Data Classification Using Local Mahalanobis Distance Learning Based on Nearest Neighbor”, SN Computer Science

Cite this paper as:
Kamaladevi M.; Venkatraman V. (2021). Tversky Similarity based Under Sampling with Gaussian Kernelized Decision Stump Adaboost Algorithm for Imbalanced Medical Data Classification, *International Journal of Computers Communications & Control*, 16(6), 4291, 2021.
https://doi.org/10.15837/ijccc.2021.6.4291