# Detailed Experimental Results of RUEO

Zahra Nouri, Vahid Kiani and Hamid Fadishei

Computer Engineering Department, University of Bojnord, Bojnord, 9453155111, North Khorasan, Iran.

*Corresponding author(s). E-mail(s): v.kiani@ub.ac.ir; Contributing authors: zahranoori.1995@gmail.com; fadishei@ub.ac.ir;

**Abstract**

To address the classification of consecutive data instances within imbalanced data streams, this research introduces a new ensemble classification algorithm called Rarity Updated Ensemble with Oversampling (RUEO). This document presents detailed experimental results of the original paper considering four classification performance criteria including average-accuracy, G-Mean, Kappa, and Prequential AUC. The source code and experimental results of this research work will be publicly available at https://github.com/vkiani/RUEO.

**Keywords:** data stream mining, ensemble learning, adaptive learning, imbalanced data, data stream classification

## 1 Introduction

We evaluate the accuracy of each algorithm in the data stream classification task in terms of four criteria: Average-Accuracy, G-Mean, Prequential AUC, and Kappa. By using these criteria we can gain insights into how well our algorithm performs on both minority and majority classes in imbalanced data streams.

Accuracy is a commonly used metric to evaluate the performance of a classifier and estimate the true accuracy of the model in the real-world data population. However, when the dataset is imbalanced, accuracy can be misleading. It fails to capture the model's ability to correctly identify the minority class. Average-accuracy takes into account classification accuracy in every class and provides a more balanced evaluation of the model's performance. Average-accuracy is defined as the average recall rate obtained in each class as follows:

$$Average - accuracy = \frac{1}{C}\sum_{i=1}^{C}\frac{TP_i}{n_i} \tag{1}$$

where $C$ indicates the number of classes in the dataset, $TP_i$ is the number of true positive decisions in class $i$, and $n_i$ is the number of data points in class $i$. If the dataset is quite balanced, accuracy and average-accuracy will be the same. However, for an imbalanced dataset, average-accuracy is a better criterion, giving more weight to the minority class than its proportional influence on the entire dataset.

The G-Mean, also known as the geometric mean, is a statistical metric used to evaluate the performance of a classification model, particularly in imbalanced datasets. The G-Mean measure utilizes the concept of geometric mean, which is a statistical measure of central tendency. In binary classification tasks, it combines the sensitivity (true positive rate) and specificity (true negative rate) of the model into a single value. In multi-class classification tasks, G-Mean considers the recall rate in every class. Unlike accuracy, which can be biased towards the majority class, G-Mean measures the balance between classification performance for both minority and majority classes. It is defined as:

$$G\text{–}mean = \sqrt[C]{\prod_{i=1}^{C}\frac{TP_i}{n_i}} \tag{2}$$

The Kappa measure is a statistical metric employed to assess the concordance between observed and expected classifications, accounting for the potential occurrence of chance agreement. The Kappa measure is derived from Cohen's Kappa coefficient, which is a statistical measure of inter-rater agreement. Essentially, Cohen's Kappa statistic extends the concept of measuring the agreement between predicted and true labels, treating them as two random categorical variables, and compares it with the agreement that could happen by chance alone. This is accomplished by constructing a confusion matrix and determining the distributions of the marginal rows and columns. Considering the confusion matrix $M$, the Kappa statistic for multi-class classification problems is defined as [1]:

$$Kappa = \frac{c \times s - \sum_{k=1}^{C} p_k \times t_k}{s \times s - \sum_{k=1}^{C} p_k \times t_k} \tag{3}$$

where $c = \sum_{i=1}^{C} M_{ii}$ is the total number of data instances correctly predicted, $s = \sum_{i=1}^{C}\sum_{j=1}^{C} M_{ij}$ is the total number of data instances, $p_k = \sum_{i=1}^{C} M_{ki}$ is number of times that class $k$ was predicted (column total), and $t_k = \sum_{i=1}^{C} M_{ik}$ is number of times that class $k$ truly occurs (row total).

The AUC (Area Under the Curve) is a statistical metric used to evaluate the performance of a predictive model by assessing its ability to discriminate between positive and negative instances across various classification thresholds [2]. In a binary classification scenario, the Receiver Operating Characteristic (ROC) curve is created by plotting the true positive rate (proportion of correctly classified positives) against the false positive rate (proportion of incorrectly classified negatives). By decreasing the decision threshold of the classification model, which determines the point at which a data instance with a higher score is classified as positive, the true and false positive rates will increase. This results in a piecewise linear curve known as the ROC curve. The AUC metric summarizes the relationship depicted in the ROC curve by calculating the area under it. Remarkably, AUC is equivalent to the Wilcoxon-Mann-Whitney (WMW) U statistic test of ranks [3]. This statistical interpretation has paved the way for the development of algorithms that can compute AUC without constructing the ROC curve itself. These algorithms achieve this by quantifying the number of misorderings between positive and negative data instances in the ranking produced by classifier scores [4].

The aforementioned classification accuracy criteria are designed for static data sets. In the case of streaming data, a sliding window of fixed length is usually used, which calculates the classification performance criterion for a window of data points each time. In the experiments of this paper, after the value of the criterion was calculated for the consecutive non-overlapping windows of the data, the average of these values was calculated and reported as the performance measure on the data stream. In the case of AUC criterion, we used prequential-AUC [5]. In order to compute value of prequential-AUC in our experiments, we employed WindowImbalancedClassificationPerformanceEvaluator, WindowAUCImbalancedPerformanceEvaluator, and WindowAUCMultiClassImbalancedPerformanceEvaluator classes of the MOA library in Java[1]. Command line scripts for reproducing our results are available at the RUEO project repository[2].

# 2 Evaluation on Synthetic Datasets

In this section, we will compare the performance of the proposed method with the baseline algorithms in the classification of data streams with different degrees of imbalance. Also, with the help of imbalanced synthetic data streams, we will investigate the effect of increasing imbalance ratio on the performance of algorithms. To generate imbalanced data streams and adjust the amount of imbalance, we used the imbalanced data generator classes of MOA[3,4].

[1] Performance evaluators of MOA: https://github.com/canoalberto/ROSE.
[2] RUEO project webpage: https://github.com/vkiani/RUEO.
[3] Imbalanced stream generators: https://github.com/dabrze/imbalanced-stream-generator.
[4] ROSE project webpage: https://github.com/canoalberto/ROSE.

Average classification performance for synthetic data streams is given in Table 1, Table 2, Table 3, and Table 4 in terms of average-accuracy, G-Mean, prequential-AUC, and Kappa criterion, respectively. In each row of these tables, the ratio of the minority class is presented in the column "imbalance ratio". In every cell, the decimal number denotes the accuracy while the integer number between parenthesis indicates the rank of the algorithm among 15 compared algorithms. The table cells with a green background and italic style highlight the algorithms with lower performance than RUEO.

**Table 1** Performance of ensemble classification algorithms on synthetic data streams in terms of Average-Accuracy criterion

| Average Accuracy | Our Method | Chunk-based Algorithms | | | Online Algorithms | | | | | | Imbalance-specific Algorithms | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Imbalance Ratio | RUEO | AWE | AUE | KUE | WMAJ | DWM | DACC | OSBoost | OCBoost | OAUE | RStream | ORUSBoost | OAdaBoost | CSMOTE | OAdaC2 |
| 10 % | 0.81 (1) | 0.60 (15) | 0.78 (4) | 0.77 (9) | 0.76 (10) | 0.72 (13) | 0.61 (14) | 0.78 (5) | 0.78 (7) | 0.78 (6) | 0.79 (3) | 0.73 (11) | 0.72 (12) | 0.77 (8) | 0.80 (2) |
| 20 % | 0.89 (1) | 0.68 (14) | 0.86 (3) | 0.86 (6) | 0.84 (8) | 0.77 (13) | 0.67 (15) | 0.87 (2) | 0.85 (7) | 0.86 (4) | 0.86 (5) | 0.81 (10) | 0.79 (11) | 0.83 (9) | 0.79 (12) |
| 30 % | 0.91 (1) | 0.80 (13) | 0.90 (3) | 0.89 (7) | 0.88 (8) | 0.84 (11) | 0.73 (15) | 0.90 (2) | 0.89 (5) | 0.90 (4) | 0.89 (6) | 0.85 (10) | 0.84 (12) | 0.85 (9) | 0.75 (14) |
| 40 % | 0.92 (2) | 0.84 (13) | 0.92 (3) | 0.91 (5) | 0.90 (8) | 0.87 (9) | 0.77 (14) | 0.92 (1) | 0.91 (6) | 0.92 (4) | 0.91 (7) | 0.87 (10) | 0.86 (12) | 0.86 (11) | 0.71 (15) |
| 50 % | 0.92 (4) | 0.86 (13) | 0.92 (3) | 0.92 (5) | 0.91 (8) | 0.89 (9) | 0.79 (14) | 0.93 (1) | 0.91 (7) | 0.92 (2) | 0.91 (6) | 0.88 (10) | 0.87 (11) | 0.87 (12) | 0.67 (15) |

**Table 2** Performance of ensemble classification algorithms on synthetic data streams in terms of G-Mean criterion

| G-Mean | Our Method | Chunk-based Algorithms | | | Online Algorithms | | | | | | Imbalance-specific Algorithms | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Imbalance Ratio | RUEO | AWE | AUE | KUE | WMAJ | DWM | DACC | OSBoost | OCBoost | OAUE | RStream | ORUSBoost | OAdaBoost | CSMOTE | OAdaC2 |
| 10 % | 0.79 (1) | 0.25 (15) | 0.71 (7) | 0.69 (10) | 0.68 (11) | 0.58 (13) | 0.39 (14) | 0.72 (6) | 0.72 (5) | 0.70 (8) | 0.73 (4) | 0.70 (9) | 0.63 (12) | 0.76 (3) | 0.79 (2) |
| 20 % | 0.88 (1) | 0.54 (15) | 0.84 (5) | 0.83 (7) | 0.81 (9) | 0.69 (13) | 0.57 (14) | 0.85 (2) | 0.84 (3) | 0.84 (6) | 0.84 (4) | 0.80 (10) | 0.76 (11) | 0.83 (8) | 0.75 (12) |
| 30 % | 0.91 (1) | 0.78 (13) | 0.89 (3) | 0.88 (7) | 0.87 (8) | 0.83 (11) | 0.70 (14) | 0.90 (2) | 0.89 (4) | 0.89 (5) | 0.89 (6) | 0.84 (10) | 0.83 (12) | 0.85 (9) | 0.69 (15) |
| 40 % | 0.92 (2) | 0.83 (13) | 0.91 (3) | 0.91 (5) | 0.90 (8) | 0.87 (9) | 0.76 (14) | 0.92 (1) | 0.91 (6) | 0.91 (4) | 0.91 (7) | 0.86 (10) | 0.86 (12) | 0.86 (11) | 0.62 (15) |
| 50 % | 0.92 (4) | 0.86 (13) | 0.92 (3) | 0.92 (5) | 0.91 (8) | 0.89 (9) | 0.79 (14) | 0.93 (1) | 0.91 (7) | 0.92 (2) | 0.91 (6) | 0.88 (10) | 0.87 (11) | 0.87 (12) | 0.54 (15) |

**Table 3** Performance of ensemble classification algorithms on synthetic data streams in terms of prequential-AUC criterion

| Prequential AUC | Our Method | Chunk-based Algorithms | | | Online Algorithms | | | | | | Imbalance-specific Algorithms | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Imbalance Ratio | RUEO | AWE | AUE | KUE | WMAJ | DWM | DACC | OSBoost | OCBoost | OAUE | RStream | ORUSBoost | OAdaBoost | CSMOTE | OAdaC2 |
| 10 % | 0.86 (5) | 0.81 (11) | 0.87 (1) | 0.86 (4) | 0.84 (8) | 0.82 (10) | 0.64 (15) | 0.87 (3) | 0.78 (14) | 0.87 (2) | 0.84 (9) | 0.80 (13) | 0.85 (7) | 0.80 (12) | 0.85 (6) |
| 20 % | 0.92 (4) | 0.86 (12) | 0.93 (1) | 0.92 (5) | 0.90 (7) | 0.87 (11) | 0.73 (15) | 0.93 (2) | 0.85 (14) | 0.93 (3) | 0.90 (9) | 0.89 (10) | 0.90 (8) | 0.86 (13) | 0.91 (6) |
| 30 % | 0.95 (3) | 0.89 (13) | 0.95 (2) | 0.95 (5) | 0.93 (6) | 0.91 (11) | 0.79 (15) | 0.95 (1) | 0.89 (12) | 0.95 (4) | 0.92 (8) | 0.92 (9) | 0.92 (10) | 0.88 (14) | 0.93 (7) |
| 40 % | 0.96 (2) | 0.89 (14) | 0.96 (4) | 0.96 (5) | 0.94 (6) | 0.92 (11) | 0.82 (15) | 0.96 (1) | 0.91 (12) | 0.96 (3) | 0.93 (7) | 0.93 (9) | 0.92 (10) | 0.89 (13) | 0.93 (8) |
| 50 % | 0.96 (3) | 0.89 (14) | 0.96 (4) | 0.96 (5) | 0.94 (7) | 0.93 (8) | 0.83 (15) | 0.96 (1) | 0.91 (12) | 0.96 (2) | 0.94 (6) | 0.93 (10) | 0.92 (11) | 0.90 (13) | 0.93 (9) |

**Table 4** Performance of ensemble classification algorithms on synthetic data streams in terms of Kappa criterion

| Kappa | Our Method | Chunk-based Algorithms | | | Online Algorithms | | | | | | Imbalance-specific Algorithms | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Imbalance Ratio | RUEO | AWE | AUE | KUE | WMAJ | DWM | DACC | OSBoost | OCBoost | OAUE | RStream | ORUSBoost | OAdaBoost | CSMOTE | OAdaC2 |
| 10 % | 0.56 (8) | 0.21 (15) | 0.64 (1) | 0.61 (5) | 0.58 (7) | 0.49 (10) | 0.24 (14) | 0.64 (2) | 0.59 (6) | 0.63 (4) | 0.64 (3) | 0.43 (12) | 0.52 (9) | 0.40 (13) | 0.47 (11) |
| 20 % | 0.74 (6) | 0.43 (14) | 0.77 (2) | 0.76 (4) | 0.72 (8) | 0.59 (12) | 0.37 (15) | 0.78 (1) | 0.73 (7) | 0.77 (3) | 0.75 (5) | 0.66 (9) | 0.65 (10) | 0.62 (11) | 0.46 (13) |
| 30 % | 0.81 (4) | 0.63 (13) | 0.82 (2) | 0.80 (5) | 0.78 (8) | 0.72 (10) | 0.49 (14) | 0.83 (1) | 0.80 (6) | 0.82 (3) | 0.80 (7) | 0.73 (9) | 0.71 (11) | 0.69 (12) | 0.41 (15) |
| 40 % | 0.84 (4) | 0.68 (13) | 0.84 (2) | 0.83 (5) | 0.80 (8) | 0.75 (9) | 0.55 (14) | 0.85 (1) | 0.82 (6) | 0.84 (3) | 0.82 (7) | 0.74 (10) | 0.73 (11) | 0.72 (12) | 0.38 (15) |
| 50 % | 0.85 (4) | 0.72 (13) | 0.85 (3) | 0.84 (5) | 0.82 (8) | 0.77 (9) | 0.58 (14) | 0.85 (1) | 0.82 (7) | 0.85 (2) | 0.82 (6) | 0.75 (10) | 0.74 (11) | 0.74 (12) | 0.33 (15) |

# 3 Evaluation on Real-world Datasets

To ensure the better performance of the proposed method than the baseline methods in real-world conditions, the performance of the proposed method and the baseline methods were also evaluated on 14 real-world datasets. The results of the classification accuracy evaluation for the proposed algorithm and the baselines are summarized in Table 5, Table 6, Table 7, and Table 8 in terms of average-accuracy, G-Mean, prequential-AUC, and Kappa criteria. In every cell, the decimal number denotes the accuracy while the integer number between parenthesis indicates the rank of the algorithm among 15 evaluated algorithms. Similar to the previous section, in these tables, the cells with a green background and italic font style highlight cases where the proposed algorithm has improved classification performance compared to the baseline algorithm. Also, average results for all real-world datasets are reported in the last row of each table.

The performance of the proposed RUEO method on every data chunk is evaluated in Figure 1 and compared with other chunk-based algorithms for real-world data streams. The evaluation is based on average-accuracy. Figure 2 presents a similar comparison between the RUEO method and online algorithms for consecutive data chunks. Moreover, Figure 3 compares the RUEO method with imbalance-specific ensemble classification algorithms.

Table 9 presents the outcomes of the Wilcoxon statistical test where the lower the p-value the bigger the differences between the algorithms. Considering a significance level of 0.05 indicates a 5% risk of concluding that a difference exists when there is no actual difference between the algorithms.

**Table 5** Performance of ensemble classification algorithms on real-world data streams in terms of average-accuracy criteria

| Average Accuracy | Our Method | Chunk-based Algorithms | | | Online Algorithms | | | | | | Imbalance-specific Algorithms | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Dataset | RUEO | AWE | AUE | KUE | WMAJ | DWM | DACC | OSBoost | OCBoost | OAUE | Rstream | ORUSBoost | OAdaBoost | CSMOTE | OAdaC2 |
| adult | 0.80 (1) | 0.71 (12) | 0.74 (6) | 0.74 (5) | 0.73 (9) | 0.71 (13) | 0.64 (14) | 0.75 (4) | 0.73 (10) | 0.74 (7) | 0.75 (3) | 0.73 (8) | 0.72 (11) | 0.79 (2) | 0.56 (15) |
| airlines | 0.62 (1) | 0.56 (13) | 0.61 (5) | 0.60 (8) | 0.61 (6) | 0.57 (12) | 0.55 (14) | 0.61 (4) | 0.62 (2) | 0.62 (3) | 0.61 (7) | 0.59 (9) | 0.59 (11) | 0.59 (10) | 0.53 (15) |
| coil2000 | 0.59 (2) | 0.50 (12) | 0.50 (11) | 0.50 (10) | 0.51 (6) | 0.50 (13) | 0.50 (7) | 0.50 (9) | 0.50 (8) | 0.50 (14) | 0.50 (15) | 0.57 (3) | 0.51 (5) | 0.56 (4) | 0.61 (1) |
| connect-4 | 0.48 (4) | 0.41 (15) | 0.45 (14) | 0.45 (12) | 0.47 (7) | 0.46 (10) | 0.45 (11) | 0.50 (3) | 0.47 (8) | 0.47 (9) | 0.48 (5) | 0.58 (1) | 0.56 (2) | 0.45 (13) | 0.47 (6) |
| fars | 0.57 (1) | 0.50 (7) | 0.54 (2) | 0.53 (4) | 0.49 (8) | 0.49 (9) | 0.42 (11) | 0.52 (6) | 0.27 (14) | 0.54 (3) | 0.53 (5) | 0.12 (15) | 0.40 (12) | 0.29 (13) | 0.46 (10) |
| GMSC | 0.72 (2) | 0.51 (13) | 0.55 (8) | 0.53 (10) | 0.51 (14) | 0.53 (9) | 0.52 (12) | 0.55 (7) | 0.60 (5) | 0.53 (11) | 0.55 (6) | 0.61 (4) | 0.62 (3) | 0.72 (1) | 0.50 (15) |
| IntelLabSensors | 0.99 (7) | 0.05 (14) | 0.99 (6) | 0.99 (5) | 0.89 (10) | 1.00 (2) | 1.00 (1) | 0.96 (8) | 0.09 (12) | 0.05 (13) | 0.90 (9) | 1.00 (4) | 0.38 (11) | 1.00 (3) | 0.04 (15) |
| kr-vs-k | 0.72 (4) | 0.21 (12) | 0.71 (5) | 0.68 (6) | 0.48 (10) | 0.97 (2) | 1.00 (1) | 0.67 (7) | 0.10 (14) | 0.38 (11) | 0.53 (9) | 0.11 (13) | 0.60 (8) | 0.94 (3) | 0.09 (15) |
| letter | 0.66 (2) | 0.64 (4) | 0.62 (7) | 0.61 (9) | 0.63 (5) | 0.61 (10) | 0.16 (12) | 0.64 (3) | 0.07 (14) | 0.62 (8) | 0.63 (6) | 0.08 (13) | 0.72 (1) | 0.58 (11) | 0.04 (15) |
| magic | 0.99 (9) | 0.64 (13) | 0.99 (9) | 0.96 (11) | 1.00 (5) | 1.00 (7) | 1.00 (2) | 1.00 (4) | 1.00 (1) | 0.66 (12) | 1.00 (7) | 1.00 (3) | 0.63 (14) | 1.00 (6) | 0.63 (15) |
| poker | 0.63 (5) | 0.40 (15) | 0.55 (8) | 0.73 (3) | 0.54 (10) | 0.52 (11) | 0.54 (9) | 0.74 (2) | 0.50 (13) | 0.56 (7) | 0.77 (1) | 0.64 (4) | 0.58 (6) | 0.51 (12) | 0.41 (14) |
| powersupply | 0.16 (2) | 0.17 (1) | 0.16 (3) | 0.16 (6) | 0.16 (4) | 0.16 (9) | 0.02 (15) | 0.16 (5) | 0.02 (14) | 0.16 (10) | 0.16 (8) | 0.06 (13) | 0.16 (7) | 0.12 (12) | 0.12 (11) |
| shuttle | 0.86 (1) | 0.72 (10) | 0.82 (3) | 0.80 (7) | 0.81 (6) | 0.78 (8) | 0.51 (13) | 0.81 (5) | 0.28 (14) | 0.83 (2) | 0.82 (4) | 0.67 (12) | 0.72 (11) | 0.76 (9) | 0.24 (15) |
| thyroid | 0.90 (3) | 0.74 (10) | 0.84 (8) | 0.87 (6) | 0.73 (11) | 0.89 (5) | 0.62 (13) | 0.91 (2) | 0.57 (14) | 0.85 (7) | 0.89 (4) | 0.63 (12) | 0.76 (9) | 0.91 (1) | 0.33 (15) |
| Average | 0.69 (3.14) | 0.48 (10.79) | 0.65 (6.79) | 0.66 (7.29) | 0.61 (7.93) | 0.66 (8.57) | 0.57 (9.64) | 0.67 (4.93) | 0.42 (10.21) | 0.53 (8.36) | 0.65 (6.36) | 0.53 (8.14) | 0.57 (7.93) | 0.66 (7.14) | 0.36 (12.64) |

**Table 6** Performance of ensemble classification algorithms on real-world data streams in terms of G-Mean criteria

| G-Mean | Our Method | Chunk-based Algorithms | | | Online Algorithms | | | | | | Imbalance-specific Algorithms | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Dataset | RUEO | AWE | AUE | KUE | WMAJ | DWM | DACC | OSBoost | OCBoost | OAUE | Rstream | ORUSBoost | OAdaBoost | CSMOTE | OAdaC2 |
| adult | 0.79 (1) | 0.66 (12) | 0.70 (10) | 0.71 (6) | 0.70 (8) | 0.66 (13) | 0.58 (14) | 0.72 (4) | 0.70 (7) | 0.70 (11) | 0.73 (3) | 0.72 (5) | 0.70 (9) | 0.78 (2) | 0.34 (15) |
| airlines | 0.61 (1) | 0.43 (14) | 0.57 (4) | 0.57 (8) | 0.55 (11) | 0.46 (13) | 0.49 (12) | 0.57 (7) | 0.58 (3) | 0.56 (9) | 0.56 (10) | 0.58 (2) | 0.57 (6) | 0.57 (5) | 0.32 (15) |
| coil2000 | 0.51 (3) | 0.00 (12) | 0.04 (11) | 0.05 (10) | 0.07 (8) | 0.00 (12) | 0.08 (7) | 0.05 (9) | 0.11 (6) | 0.00 (12) | 0.00 (12) | 0.53 (2) | 0.17 (5) | 0.46 (4) | 0.61 (1) |
| connect-4 | 0.37 (3) | 0.19 (13) | 0.21 (11) | 0.19 (12) | 0.26 (5) | 0.22 (8) | 0.22 (7) | 0.27 (4) | 0.00 (15) | 0.21 (9) | 0.25 (6) | 0.49 (1) | 0.46 (2) | 0.21 (10) | 0.15 (14) |
| fars | 0.35 (1) | 0.17 (7) | 0.19 (6) | 0.06 (10) | 0.06 (11) | 0.30 (2) | 0.28 (4) | 0.05 (13) | 0.00 (15) | 0.15 (8) | 0.06 (12) | 0.00 (14) | 0.24 (5) | 0.08 (9) | 0.29 (3) |
| GMSC | 0.71 (1) | 0.04 (14) | 0.30 (8) | 0.21 (9) | 0.05 (13) | 0.19 (10) | 0.13 (12) | 0.30 (6) | 0.47 (5) | 0.19 (11) | 0.30 (7) | 0.52 (3) | 0.50 (4) | 0.70 (2) | 0.01 (15) |
| IntelLabSensors | 0.99 (9) | 0.05 (14) | 0.99 (10) | 0.99 (8) | 0.99 (7) | 1.00 (2) | 1.00 (1) | 1.00 (5) | 0.09 (12) | 0.06 (13) | 1.00 (6) | 1.00 (4) | 0.38 (11) | 1.00 (3) | 0.04 (15) |
| kr-vs-k | 0.74 (10) | 0.23 (12) | 0.76 (8) | 0.75 (9) | 0.93 (6) | 1.00 (2) | 1.00 (1) | 0.97 (4) | 0.09 (14) | 0.31 (11) | 0.95 (5) | 0.12 (13) | 0.91 (7) | 0.99 (3) | 0.09 (14) |
| letter | 0.62 (2) | 0.60 (4) | 0.58 (7) | 0.56 (10) | 0.60 (5) | 0.57 (9) | 0.01 (12) | 0.60 (3) | 0.00 (13) | 0.58 (8) | 0.60 (6) | 0.00 (13) | 0.71 (1) | 0.52 (11) | 0.00 (13) |
| magic | 0.97 (9) | 0.63 (13) | 0.97 (9) | 0.95 (11) | 1.00 (5) | 1.00 (7) | 1.00 (2) | 1.00 (4) | 1.00 (1) | 0.66 (12) | 1.00 (7) | 1.00 (3) | 0.62 (14) | 1.00 (6) | 0.62 (15) |
| poker | 0.57 (3) | 0.16 (13) | 0.34 (12) | 0.53 (4) | 0.40 (7) | 0.40 (9) | 0.44 (6) | 0.59 (2) | 0.05 (15) | 0.38 (11) | 0.67 (1) | 0.52 (5) | 0.38 (10) | 0.40 (8) | 0.14 (14) |
| powersupply | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (1) | 0.00 (2) | 0.00 (2) |
| shuttle | 0.54 (1) | 0.22 (11) | 0.48 (5) | 0.42 (7) | 0.46 (6) | 0.35 (9) | 0.16 (13) | 0.50 (3) | 0.00 (14) | 0.50 (2) | 0.50 (4) | 0.23 (10) | 0.19 (12) | 0.40 (8) | 0.00 (14) |
| thyroid | 0.88 (5) | 0.65 (11) | 0.80 (8) | 0.85 (6) | 0.66 (10) | 0.88 (4) | 0.55 (12) | 0.90 (2) | 0.00 (14) | 0.81 (7) | 0.89 (3) | 0.43 (13) | 0.73 (9) | 0.90 (1) | 0.00 (14) |
| Average | 0.62 (3.64) | 0.29 (10.86) | 0.49 (7.93) | 0.49 (8.00) | 0.48 (7.43) | 0.50 (7.29) | 0.42 (7.50) | 0.54 (4.86) | 0.22 (9.71) | 0.37 (9.00) | 0.53 (6.00) | 0.44 (6.43) | 0.47 (6.86) | 0.57 (5.29) | 0.19 (11.71) |

**Table 7** Performance of ensemble classification algorithms on real-world data streams in terms of prequential-AUC criteria

| Prequential AUC | Our Method | Chunk-based Algorithms | | | Online Algorithms | | | | | | Imbalance-specific Algorithms | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Dataset | RUEO | AWE | AUE | KUE | WMAJ | DWM | DACC | OSBoost | OCBoost | OAUE | Rstream | ORUSBoost | OAdaBoost | CSMOTE | OAdaC2 |
| adult | 0.88 (5) | 0.87 (6) | 0.88 (3) | 0.89 (1) | 0.87 (7) | 0.80 (13) | 0.74 (14) | 0.89 (2) | 0.73 (15) | 0.88 (4) | 0.86 (8) | 0.85 (10) | 0.86 (9) | 0.83 (12) | 0.85 (11) |
| airlines | 0.66 (4) | 0.58 (14) | 0.67 (2) | 0.65 (6) | 0.67 (3) | 0.61 (12) | 0.55 (15) | 0.66 (5) | 0.62 (11) | 0.68 (1) | 0.65 (7) | 0.64 (8) | 0.63 (9) | 0.60 (13) | 0.63 (10) |
| coil2000 | 0.65 (3) | 0.50 (15) | 0.65 (1) | 0.60 (8) | 0.51 (13) | 0.51 (12) | 0.52 (11) | 0.61 (6) | 0.50 (14) | 0.64 (4) | 0.58 (9) | 0.61 (7) | 0.62 (5) | 0.57 (10) | 0.65 (2) |
| connect-4 | 0.69 (9) | 0.63 (14) | 0.72 (7) | 0.69 (10) | 0.73 (6) | 0.66 (12) | 0.65 (13) | 0.77 (3) | 0.50 (15) | 0.75 (5) | 0.71 (8) | 0.81 (2) | 0.81 (1) | 0.67 (11) | 0.77 (4) |
| fars | 0.43 (9) | 0.46 (4) | 0.48 (1) | 0.47 (2) | 0.45 (5) | 0.42 (10) | 0.36 (12) | 0.45 (7) | 0.28 (14) | 0.47 (3) | 0.43 (8) | 0.27 (15) | 0.38 (11) | 0.33 (13) | 0.45 (6) |
| GMSC | 0.79 (7) | 0.68 (12) | 0.85 (1) | 0.84 (3) | 0.81 (5) | 0.67 (13) | 0.50 (15) | 0.85 (2) | 0.60 (14) | 0.84 (4) | 0.80 (6) | 0.70 (10) | 0.77 (9) | 0.78 (8) | 0.69 (11) |
| IntelLabSensors | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (1) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) | 0.00 (2) |
| kr-vs-k | 0.00 (6) | 0.00 (5) | 0.00 (6) | 0.00 (6) | 0.00 (15) | 0.00 (1) | 0.00 (3) | 0.00 (13) | 0.00 (6) | 0.00 (6) | 0.00 (14) | 0.00 (4) | 0.00 (12) | 0.00 (2) | 0.00 (6) |
| letter | 0.80 (11) | 0.95 (2) | 0.95 (6) | 0.94 (8) | 0.95 (5) | 0.86 (10) | 0.55 (12) | 0.95 (1) | 0.50 (15) | 0.94 (7) | 0.95 (4) | 0.51 (13) | 0.95 (3) | 0.92 (9) | 0.50 (14) |
| magic | 0.99 (11) | 0.99 (11) | 0.99 (11) | 0.99 (11) | 1.00 (1) | 1.00 (8) | 1.00 (2) | 1.00 (2) | 1.00 (7) | 1.00 (4) | 1.00 (4) | 0.99 (9) | 0.99 (9) | 1.00 (4) | 0.99 (11) |
| poker | 0.06 (13) | 0.06 (15) | 0.07 (3) | 0.07 (4) | 0.10 (1) | 0.10 (2) | 0.06 (12) | 0.07 (5) | 0.06 (14) | 0.06 (8) | 0.06 (11) | 0.06 (9) | 0.06 (6) | 0.06 (10) | 0.06 (7) |
| powersupply | 0.77 (2) | 0.77 (1) | 0.76 (3) | 0.76 (4) | 0.76 (5) | 0.61 (12) | 0.39 (15) | 0.76 (6) | 0.50 (13) | 0.76 (7) | 0.75 (9) | 0.49 (14) | 0.75 (10) | 0.64 (11) | 0.76 (8) |
| shuttle | 0.19 (11) | 0.24 (7) | 0.29 (3) | 0.28 (4) | 0.32 (2) | 0.32 (1) | 0.18 (12) | 0.23 (8) | 0.17 (14) | 0.27 (5) | 0.19 (10) | 0.17 (14) | 0.25 (6) | 0.22 (9) | 0.18 (13) |
| thyroid | 0.94 (7) | 0.92 (11) | 0.95 (6) | 0.95 (4) | 0.94 (8) | 0.94 (9) | 0.73 (13) | 0.96 (1) | 0.50 (15) | 0.95 (5) | 0.96 (3) | 0.89 (12) | 0.93 (10) | 0.96 (2) | 0.50 (14) |
| Average | 0.56 (7.14) | 0.55 (8.50) | 0.59 (3.93) | 0.58 (5.21) | 0.58 (5.57) | 0.54 (8.29) | 0.45 (10.79) | 0.59 (4.50) | 0.43 (12.07) | 0.59 (4.64) | 0.57 (7.36) | 0.50 (9.21) | 0.57 (7.29) | 0.54 (8.29) | 0.50 (8.50) |

**Table 8** Performance of ensemble classification algorithms on real-world data streams in terms of Kappa criteria

| Kappa | Our Method | Chunk-based Algorithms | | | Online Algorithms | | | | | | Imbalance-specific Algorithms | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Dataset | RUEO | AWE | AUE | KUE | WMAJ | DWM | DACC | OSBoost | OCBoost | OAUE | Rstream | ORUSBoost | OAdaBoost | CSMOTE | OAdaC2 |
| adult | 0.50 (7) | 0.46 (13) | 0.53 (4) | 0.53 (3) | 0.51 (6) | 0.47 (11) | 0.33 (14) | 0.54 (1) | 0.48 (9) | 0.52 (5) | 0.54 (2) | 0.47 (10) | 0.47 (12) | 0.49 (8) | 0.16 (15) |
| airlines | 0.22 (5) | 0.12 (13) | 0.23 (3) | 0.21 (8) | 0.22 (6) | 0.15 (12) | 0.10 (14) | 0.23 (4) | 0.23 (2) | 0.24 (1) | 0.22 (7) | 0.19 (9) | 0.18 (11) | 0.18 (10) | 0.05 (15) |
| coil2000 | 0.10 (1) | 0.00 (12) | 0.01 (11) | 0.01 (10) | 0.02 (5) | 0.00 (13) | 0.02 (7) | 0.02 (8) | 0.01 (9) | 0.00 (14) | 0.00 (15) | 0.06 (3) | 0.04 (4) | 0.02 (6) | 0.07 (2) |
| connect-4 | 0.24 (12) | 0.15 (15) | 0.24 (13) | 0.25 (10) | 0.30 (7) | 0.28 (8) | 0.28 (9) | 0.36 (3) | 0.33 (4) | 0.30 (6) | 0.32 (5) | 0.48 (1) | 0.47 (2) | 0.25 (11) | 0.21 (14) |
| fars | 0.63 (5) | 0.58 (7) | 0.62 (6) | 0.69 (1) | 0.58 (8) | 0.51 (9) | 0.41 (11) | 0.67 (2) | 0.15 (14) | 0.65 (4) | 0.66 (3) | 0.02 (15) | 0.36 (12) | 0.19 (13) | 0.44 (10) |
| GMSC | 0.22 (4) | 0.02 (13) | 0.16 (8) | 0.09 (9) | 0.02 (14) | 0.09 (10) | 0.05 (12) | 0.16 (6) | 0.24 (3) | 0.09 (11) | 0.16 (7) | 0.20 (5) | 0.26 (2) | 0.29 (1) | 0.00 (15) |
| IntelLabSensors | 0.97 (6) | 0.05 (13) | 0.97 (4) | 0.97 (4) | 0.41 (9) | 0.98 (2) | 1.00 (1) | 0.44 (8) | 0.09 (12) | 0.05 (14) | 0.30 (11) | 0.98 (3) | 0.38 (10) | 0.97 (7) | 0.04 (15) |
| kr-vs-k | 0.47 (6) | 0.15 (11) | 0.49 (4) | 0.49 (5) | 0.16 (10) | 0.65 (2) | 0.94 (1) | 0.19 (8) | 0.10 (14) | 0.20 (7) | 0.16 (9) | 0.10 (13) | 0.12 (12) | 0.63 (3) | 0.09 (15) |
| letter | 0.64 (2) | 0.62 (4) | 0.61 (8) | 0.60 (9) | 0.62 (5) | 0.59 (10) | 0.13 (12) | 0.62 (3) | 0.03 (14) | 0.61 (7) | 0.62 (6) | 0.05 (13) | 0.72 (1) | 0.56 (11) | 0.00 (15) |
| magic | 0.97 (5) | 0.62 (14) | 0.97 (5) | 0.95 (9) | 0.99 (4) | 0.92 (10) | 1.00 (2) | 0.99 (3) | 1.00 (1) | 0.66 (12) | 0.92 (10) | 0.97 (7) | 0.62 (13) | 0.97 (8) | 0.62 (14) |
| poker | 0.54 (7) | 0.17 (15) | 0.49 (10) | 0.78 (2) | 0.45 (11) | 0.40 (12) | 0.51 (9) | 0.80 (1) | 0.68 (5) | 0.52 (8) | 0.78 (3) | 0.68 (4) | 0.62 (6) | 0.34 (13) | 0.20 (14) |
| powersupply | 0.13 (2) | 0.13 (1) | 0.13 (3) | 0.12 (6) | 0.13 (4) | 0.12 (9) | -0.03 (15) | 0.12 (5) | -0.02 (14) | 0.12 (10) | 0.12 (7) | 0.01 (13) | 0.12 (8) | 0.08 (12) | 0.09 (11) |
| shuttle | 0.97 (1) | 0.93 (9) | 0.95 (6) | 0.96 (2) | 0.93 (10) | 0.95 (7) | 0.65 (13) | 0.95 (8) | 0.40 (14) | 0.96 (3) | 0.96 (5) | 0.66 (12) | 0.96 (4) | 0.82 (11) | 0.00 (15) |
| thyroid | 0.70 (8) | 0.63 (10) | 0.78 (6) | 0.81 (3) | 0.61 (11) | 0.80 (4) | 0.51 (12) | 0.85 (1) | 0.02 (14) | 0.78 (5) | 0.82 (2) | 0.42 (13) | 0.69 (9) | 0.77 (7) | 0.00 (15) |
| Average | 0.52 (5.07) | 0.33 (10.71) | 0.51 (6.50) | 0.53 (5.79) | 0.42 (7.86) | 0.49 (8.50) | 0.42 (9.43) | 0.50 (4.36) | 0.27 (9.21) | 0.41 (7.64) | 0.47 (6.57) | 0.38 (8.64) | 0.43 (7.57) | 0.47 (8.64) | 0.14 (13.21) |


**Fig. 1** Comparison of the windowed classification performance of the proposed RUEO method with chunk-based ensemble classification algorithms on real-world data streams in terms of average-accuracy.

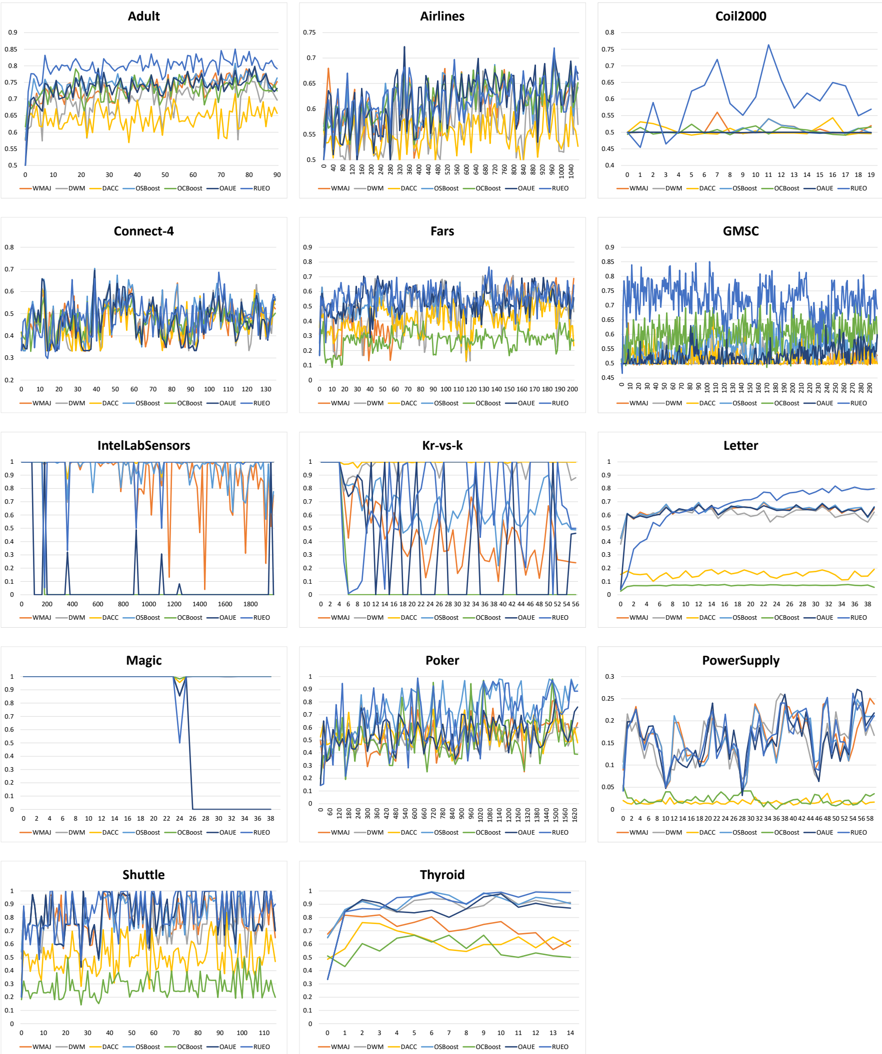


**Fig. 2** Comparison of the windowed classification performance of the proposed RUEO method with online ensemble classification algorithms on real-world data streams in terms of average-accuracy.

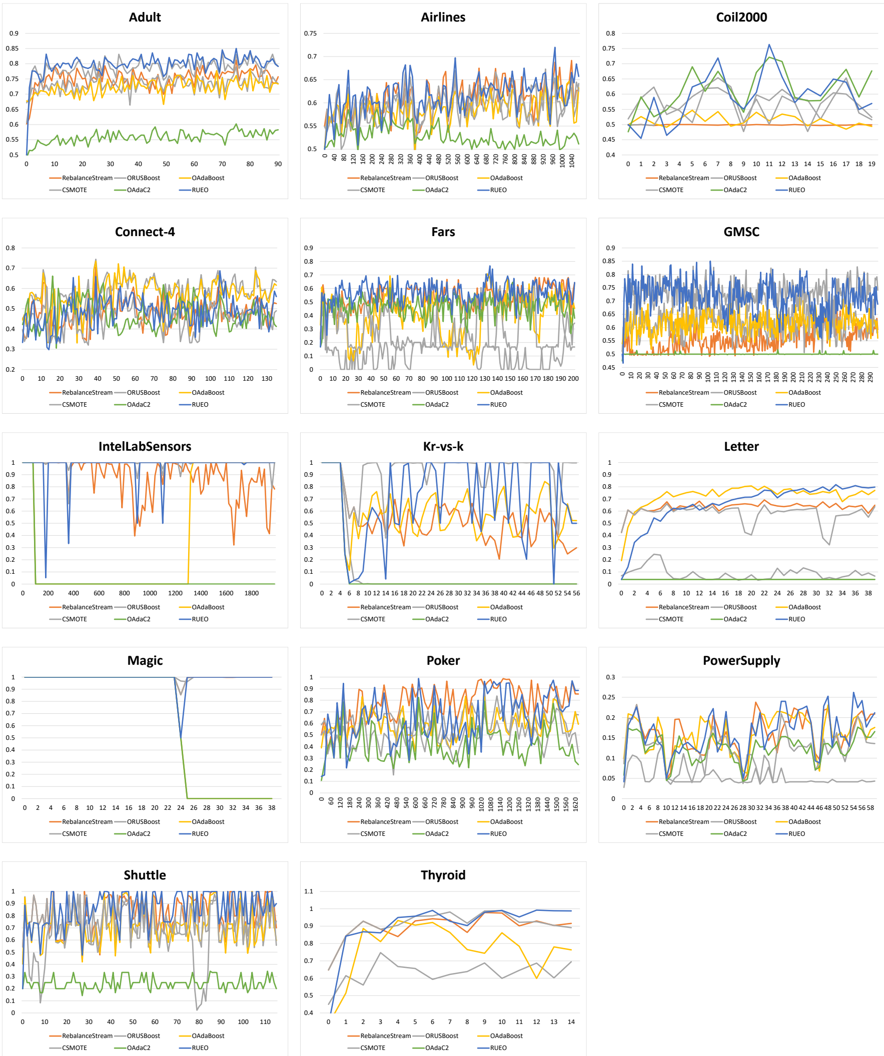


**Fig. 3** Comparison of the windowed classification performance of the proposed RUEO method with imbalance-specific ensemble classification algorithms on real-world data streams in terms of average-accuracy.

**Table 9** Wilcoxon test for real-world data streams.

| RUEO vs. | Average-Accuracy p-value | G-Mean p-value | Prequential AUC p-value | Kappa p-value |
|---|---|---|---|---|
| AWE | 0.000 | 0.001 | 0.433 | 0.000 |
| AUE | 0.002 | 0.004 | 0.004 | 0.463 |
| KUE | 0.017 | 0.003 | 0.213 | 0.855 |
| WMAJ | 0.001 | 0.019 | 0.152 | 0.017 |
| DWM | 0.035 | 0.046 | 0.326 | 0.173 |
| DACC | 0.013 | 0.016 | 0.007 | 0.049 |
| OSBoost | 0.091 | 0.116 | 0.046 | 0.952 |
| OCBoost | 0.000 | 0.002 | 0.004 | 0.035 |
| OAUE | 0.000 | 0.001 | 0.012 | 0.217 |
| RStream | 0.020 | 0.133 | 0.600 | 0.583 |
| ORUSBoost | 0.011 | 0.023 | 0.039 | 0.035 |
| OAdaBoost | 0.007 | 0.020 | 0.753 | 0.217 |
| CSMOTE | 0.078 | 0.133 | 0.279 | 0.173 |
| OAdaC2 | 0.000 | 0.002 | 0.131 | 0.000 |

## References

[1] Grandini, M., Bagli, E., Visani, G.: Metrics for Multi-Class Classification: an Overview (2020). https://doi.org/10.48550/arXiv.2008.05756. arXiv:2008.05756 [cs, stat]. Accessed 2023-09-22

[2] Chawla, N.V.: Data Mining for Imbalanced Datasets: An Overview. In: Maimon, O., Rokach, L. (eds.) Data Mining and Knowledge Discovery Handbook, pp. 875–886. Springer, Boston, MA (2010). https://doi.org/10.1007/978-0-387-09823-4_45. https://doi.org/10.1007/978-0-387-09823-4 Accessed 2023-09-22

[3] Hanley, J.A., McNeil, B.J.: The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology **143**(1), 29–36 (1982). https://doi.org/10.1148/radiology.143.1.7063747

[4] Wu, S., Flach, P., Ferri, C.: An Improved Model Selection Heuristic for AUC. In: Kok, J.N., Koronacki, J., Mantaras, R.L.d., Matwin, S., Mladenič, D., Skowron, A. (eds.) Machine Learning: ECML 2007. Lecture Notes in Computer Science, pp. 478–489. Springer, Berlin, Heidelberg (2007). https://doi.org/10.1007/978-3-540-74958-5_44

[5] Brzezinski, D., Stefanowski, J.: Prequential AUC: properties of the area under the ROC curve for data streams with concept drift. Knowledge and Information Systems **52**(2), 531–562 (2017). https://doi.org/10.1007/s10115-017-1022-8. Accessed 2023-09-22