Research Article
Journal of Intelligent Systems: Theory and Applications 5(2) 2022: 119-126
DOI: 10.38016/jista.1033354

# A New Instance Selection Method for Enlarging Margins Between Classes

Fatih Aydın[1*]

[1] Balıkesir University, Department of Computer Engineering, Balıkesir, Turkey
fatih.aydin@balikesir.edu.tr

**Abstract**
As discarding superfluous instances in data sets shortens the learning process, it also increases learning performance because of eliminating noisy data. Instance selection methods are commonly utilized to undertake the abovementioned tasks. In this paper, we propose a new supervised instance selection algorithm called Border Instances Reduction using Classes Handily (BIRCH). BIRCH considers k-nearest neighbors of each instance and selects instances that have neighbors from the only same class, namely, but not having neighbors from the different classes. It has been compared with one traditional and four state-of-the-art instance selection algorithms by using fifteen data sets from various domains. The empirical results show BIRCH well delivers the trade-off between accuracy rate and reduction rate by tuning the number of neighbors. Furthermore, the proposed method guarantees to yield a high classification accuracy. The source code of the proposed algorithm can be found in https://github.com/fatihaydin1/BIRCH.

**Keywords:** Machine learning, nearest neighbors, instance reduction, instance selection, big data.

## Sınıflar Arası Kenar Payını Genişletmek İçin Yeni Bir Örnek Seçim Algoritması

**Öz**
Veri kümelerindeki gereksiz örneklerin atılması öğrenme sürecini kısalttığı gibi gürültülü verileri ortadan kaldırdığı için öğrenme performansını da arttırmaktadır. Örnek seçim yöntemleri, yukarıda belirtilen görevleri yerine getirmek için yaygın olarak kullanılmaktadır. Bu makalede, "Border Instances Reduction using Classes Handily (BIRCH)" adlı yeni bir denetimli örnek seçim algoritması öneriyoruz. BIRCH, her örneğin k-en yakın komşularını dikkate alarak, sadece aynı sınıftan komşuları olan, yani farklı sınıflardan komşuları olmayan örnekleri seçer. BIRCH, çeşitli alanlardan on beş veri kümesi kullanılarak biri geleneksel ve dördü son teknoloji örnek seçim algoritması ile karşılaştırılmıştır. Ampirik sonuçlar, BIRCH'in komşu sayısının ayarlanmasıyla doğruluk oranı ve azaltma oranı arasındaki dengeyi iyi sağladığını göstermektedir. Ayrıca önerilen yöntem, yüksek bir sınıflandırma doğruluğunu sağlamayı garanti eder. Önerilen algoritmanın kaynak kodu https://github.com/fatihaydin1/BIRCH web adresinde bulunabilir.

**Anahtar Kelimeler:** Makine öğrenmesi, en yakın komşular, örnek azaltma, örnek seçimi, büyük veri.

## 1. Introduction

Machine Learning (ML) is a discipline, which intends to redound learning capability for automata to discover patterns in real-world data. But Some ML algorithms such as Support Vector Machine (SVM) and k-Nearest Neighbors (kNN) suffer from big data in terms of running time. Instance selection is a process of getting rid of unnecessary data (Olvera-López *et al.*, 2010). In other words, the common goal of the instance selection methods is to discard redundant data from the data set. After the instance selection stage, the desired end is the classification performances over the original data set and the selected subset are close to each other. Instance selection would be beneficial at reducing the training and test time for lazy learners and function learners such as SVM and Neural Networks (NN). Besides, instance reduction methods are used to address the challenges in the different areas such as class-imbalanced data sets, time series, distributed learning, monotonic data sets, noise sensitiveness, and lazy learners. In the literature review, it is seen that the nearest neighbor, evolutionary methods, meta-approaches, computational strategies, probabilistic approaches, cluster-based approach, geometrical

[*] Corresponding Author
E-mail: fatih.aydin@balikesir.edu.tr

Recieved : 6 Dec 2021
Revision : 29 Dec 2021
Accepted : 23 Mar 2022

approaches, and ranking approach have been utilized to develop instance selection algorithms. There exist several joint characteristics in instance selection methods: type of selection, the direction of search, and evaluation of search (Olvera-López *et al.*, 2010; García-Pedrajas, 2011; Garcia *et al.*, 2012). Furthermore, the criteria such as storage requirement, noise resistance, classification accuracy, and running time have been used to compare instance selection algorithms (Garcia *et al.*, 2012).

In the literature, Condensed Nearest Neighbor (CNN) is the first approach that has been designed to discard irrelevant or noisy data (Hart, 1968). CNN is an iterative method and begins with a blank subset. In the next stage, CNN indiscriminately selects a point from the training data and joins it to the subset if the instance is misclassified while using the subset as training data. The halt rule is that there remain no more instances. CNN does not promise to attain the optimal subset. Besides, it forms different subsets at each run because of selecting instances arbitrarily (Alpaydin, 1997). Modified Condensed Nearest Neighbor (MCNN) has been proposed to enhance CNN. MCNN produces the subset by regarding the centroid of the misclassified instances in each class. MCNN achieves better performance if the data is normally distributed (Susheela Devi and Murty, 2002). Edited Nearest Neighbor (ENN) is one of the first algorithms that focus on eliminating noisy instances (Wilson, 1972).

Wilson and Martinez proposed six reduction algorithms abbreviated DROP1-DROP5 (i.e., Decremental Reduction Optimization Procedure Family), and DEL (Wilson and Martinez, 2000). DROP3-DROP5 methods are hybrid methods that fuse condensing and editing techniques.

In respect of meta approaches, Alpaydin introduced a voting approach combining predictions from a sequence of models after training multiple subsets by using two voting schemes such as simple voting and weighted voting (Alpaydin, 1997).

As for the use of local-sensitive hashing family (LSH), LSH-IS-S and LSH-IS-F methods proposed based on LSH are with quadratic and log-linear complexities and rely on unveiling similarities between instances (Arnaiz-González *et al.*, 2016). Data Reduction with Locality-Sensitive Hashing (DR.LSH) is a new instance selection method using LSH. The proposed method tries to rapidly detect similar and redundant data and discard them from the original data set (Aslani and Seipel, 2020). The Border Point extraction based on Locality-Sensitive Hashing (BPLSH) that has been suggested as a novel instance selection method holds instances that are close to the decision borders and eliminates interior instances (Aslani and Seipel, 2021).

Rico-Juan et al proposed two instance selection algorithms based on the ranking approach. The goal of the first extension is to obtain greater robustness against noise according to the nearest neighbors in the selection process. The second method employs a new parameter-free approach to select instances (Rico-Juan, Valero-Mas and Calvo-Zaragoza, 2019). Ruiz and Gómez-Nieto proposed a novel instance selection algorithm to build Quantitative Structure-Activity Relationship (QSAR) classification models by using the Rivality Index NeighborHood (RINH) algorithm. The method can get significant reduction rate in the size of the training data as maintaining the classification performance (Ruiz and Gómez-Nieto, 2020). Fast Data Reduction with Granulation-based Instances Importance Labeling (FDR-GIIL) has been proposed as a fast instance selection method using granular computing to select the instances that contribute to the classification performance (Sun *et al.*, 2019).

For the solution suggestions to data sets with different properties, Wang et al introduced two data cleaning algorithms to address class-imbalanced data sets. The former examines whether realizing instance selection to eliminate several noisy data from the majority class can improve the performance of one-class classifiers. The latter handles instance selection and missing value problems jointly for incomplete data sets (Wang, Tsai and Lin, 2021).

Constraint Nearest Neighbor-based Instance Reduction (CNNIR) has been proposed as a novel instance selection algorithm based on the concept of natural neighbor, removes noises, and searches core instances. It defines a constraint nearest-neighbor chain that only consists of three instances to choose boundary instances that can build a smooth decision boundary, next the subset is obtained by merging boundary and inner instances (Yang *et al.*, 2019).

Shell Extraction (SE) is a new instance selection method, which considers an unbalanced distribution of instances and a strategy with self-adaption from the geometrical perspective (Liu *et al.*, 2017). Akinyelu and Adewumi introduced two novel instance selection methods for SVM Speed Optimization: FFA-based Instance Selection (FFA_IS) and Edge Instance Selection Algorithm (EISA). FFA_IS is inspired by the flashing behavior of fireflies. EISA relies on the idea of edge detection in image processing (Akinyelu and Adewumi, 2017). Akinyelu and Ezugwu suggested two instance selection methods for SVM speed optimization called the Flower Pollination Instance Selection Algorithm (FPISA) and the Social Spider Instance Selection Algorithm (SSISA), which are respectively a nature-inspired metaheuristic algorithm and normal individual-based swarm intelligence algorithm (Akinyelu and Ezugwu, 2019).

In this paper, we propose a condensing approach that performs to eliminate the boundary instances instead of preserving them. Thus, large margins between classes are formed. The reason for applying to the first stage is to reduce the error that a model makes due to variance. This approach especially supports the learners that suffer from high variance. The time complexity of our proposed method is log-linear in the best case and quadratic in the worst case. Besides, the proposed algorithm has obtained remarkable results on the data sets used in the experiments. The main contributions of the proposed method are as follows:

- The proposed algorithm can faster process big data compared to the similar approaches.
- The algorithm is easy to implement.
- The proposed method guarantees a high accuracy rate.
- The algorithm has only two parameters to adjust.

The rest of the paper is organized as the following. In Section 2, we introduce the proposed method. In Section 3, we explain the experimental setup. In Section 4, we present the experimental results. Finally, we put forth the conclusions of the paper in Section 5.

## 2. The related work

In this section, we provide the description of the proposed method and calculate the time and space complexities of the proposed algorithm.

### 2.1. The description of the proposed method

The proposed algorithm performs to remove boundary instances and thus, enlarges the margin between the classes. In this end, the proposed method selects instances that have neighbors from the only same class, namely, but not having neighbors from the different classes by considering the k-nearest neighbors of each instance. The contributions of removing boundary instances are: (i) keeping up with streaming data that changes over time, (ii) increasing resistance against noise, and (iii) reinforcing learners that suffer from variance. As a result of filtering up boundary instances by using 1-Nearest Neighbors (1NN), the removed error rate corresponds to at most twice the Bayes error rate as proved by Cover and Hart (Cover and Hart, 1967) in (1):

$$R^* \leq R \leq 2R^*(1-R^*) \leq 2R^* \tag{1}$$

where $R^*$ denotes Bayes error rate (i.e., irreducible error) and $R$ denotes 1NN error rate. The Bayes classifier is optimal since its risk is the minimum expected error rate $R^*$. For a data set with two classes ($c_1\ and\ c_2$) and any point $x$, let $P(c_1|x)$ and $P(c_2|x)$ be error rates for each class. Accordingly, the n-sample 1NN risk is shown in (2).

$$R = E[P(c_1|x)P(c_2|x) + P(c_2|x)P(c_1|x)]$$

$$= E[2P(c_1|x)P(c_2|x)] \tag{2}$$

Since $P(c_1|x) + P(c_2|x) = 1$, we have

$$R = E\big[2P(c_1|x)\big(1 - P(c_1|x)\big)\big]$$

Since $R^* = E[P(c_1|x)]$, we have

$$R = 2R^*(1-R^*) - 2 \times Var\big(P(c_1|x)\big)$$

Considering the case in which the variance of $P(c_1|x)$ is zero, we arrive at (3).

$$R \leq 2R^*(1-R^*) \tag{3}$$

Consequently, removing the boundary instances on the training set decreases the generalization error since it removes the noisy instances or the instances that can cause errors due to high variance.

The proposed method runs according to the number of the nearest neighbors to eliminate instances. We propose a new instance selection algorithm called Border Instances Reduction using Classes Handily (BIRCH) and describe it in Algorithm 1.

```
Algorithm 1: BIRCH
Input:
T = {(x_1, y_1), …, (x_m, y_m)} ∈ $\mathbb{R}^{m\times d}$: Data set
δ: The distance metric (by default, 'cityblock')
k: The number of neighbors (by default, 1)

Output:
S = {(x_1, y_1), …, (x_t, y_t)} ∈ $\mathbb{R}^{t\times d}$: Selected points
S ← T
$N^{m\times k}$ ← Find the k-nearest neighbors of each instance
$C^{m\times k}$ ← Find the class of N
A = {x: x ∈ X, x is neighbor instance from the different class in C}
B = {x: x ∈ X, x is neighbor instance from the same class in C}
S = B\A
```

### 2.2. The time and space complexities

Accordingly, we carry out the calculation of the time and space complexities of the algorithm. In the first stage, BIRCH searches for the k-nearest neighbors and removes instances, depending on the case that they are from the same or different class. The determination of the k-nearest neighbors is calculated with time complexity $O(kmlog_2m)$ and space complexity $O(md)$. Finding the classes of the neighbors is calculated with time complexity $O(mk)$ and space complexity $O(mk)$. The upper bound time and space complexities that are needed to search unique instances are $O(2mlog_2m)$ and $O(mk)$, respectively. The time and space complexities of difference between two sets are $O(mk)$. As a result, the total time complexity of the first stage is $O\big((k+2)mlog_2m + 2mk\big)$ and the total space complexity of the first stage is $O(md)$. We neglect the expressions owing less effect on high order terms.

Accordingly, the time complexity of BIRCH is found as $O(kmlog_2m + mk)$. Consequently, the time complexity of BIRCH is log-linear in the best case and log-quadratic in the worst case (i.e., $k \approx m$).

## 3. Experimental Setup

In this section, we explain the experimental setup, including experimental data sets, instance selection algorithms used in the experiments, evaluation metrics, and implementations.

### 3.1. Data sets

BIRCH has been compared with the state-of-the-art instance selection algorithms to measure its efficiency by using fifteen data sets from the UCI database[1], OpenML[2], and MATLAB[3]. The data sets have been picked from the various domains. Besides, the selected data sets contain the different number of instances, features and classes. The descriptive information belonging to those data sets is shown in Table 1. The imbalance ratio denotes the ratio of the number of classes with the most instances to the number of those with the least instances.

### 3.2. Instance selection methods

**Table 1.** The characteristics of the data sets used experiments

| # | Data set | Instances | Features | Classes | Imbalance ratio |
|---|---|---|---|---|---|
| 1 | Arrhythmia | 452 | 279 | 13 | 122.50 |
| 2 | Avila | 20867 | 10 | 12 | 857.20 |
| 3 | BostonHousing2 | 506 | 18 | 92 | 30.00 |
| 4 | EEG_EyeState | 14980 | 14 | 2 | 1.23 |
| 5 | Electricity | 45312 | 8 | 2 | 1.36 |
| 6 | HTRU2 | 17898 | 8 | 2 | 9.92 |
| 7 | HumanActivity | 24075 | 60 | 5 | 2.34 |
| 8 | LetterRecognition | 20000 | 16 | 26 | 1.11 |
| 9 | Madelon | 2000 | 500 | 2 | 1.00 |
| 10 | MAGIC Gamma Telescope | 19020 | 10 | 2 | 1.84 |
| 11 | Mozilla4 | 15545 | 5 | 2 | 2.04 |
| 12 | Nomao | 34465 | 118 | 2 | 2.50 |
| 13 | Ovariancancer | 216 | 4000 | 2 | 1.27 |
| 14 | Seeds | 210 | 7 | 3 | 1.00 |
| 15 | Shuttle | 58000 | 9 | 7 | 4558.60 |

The proposed method has been compared with one conventional and four state-of-the-art instance selection algorithms in Table 2. The parameter values and other characteristics of the algorithms used in the experiments are also shown Table 2. In addition, we have conducted all the experiments by the default values of the algorithms. All the methods used in the experiments benefit from class information and they adopt the filter approach.

### 3.3. Implementations

The baseline method means that the 1NN algorithm applies to the original data set. Additionally, we apply 10-fold cross-validation to all the experiments and repeat each experiment five times to select the training data with different combinations. The experiments have been conducted in the MATLAB R2021a on an i5-8265U CPU at 1.6 GHz with 8 GB of RAM on Windows 11 Pro (64-bit). Further, we use the default number of neighbors and default distance metric as 1 and 'city block', respectively for BIRCH.

### 3.4. Evaluation metrics

We have used three criteria such as classification accuracy, reduction rate, and running time have been used to compare instance selection methods.

**Table 2.** The instance selection methods used in the experiments

| Algorithm | Supervision | Type | Technique | Parameter(s) |
|---|---|---|---|---|
| BPLSH[4] | ✓ | Filter | Condensation | M=30, L=10, W=1 |
| DR.LSH[5] | ✓ | Filter | Hybrid | M=25, L=10, W=1, ST=9 |
| LSH-IS-S[6] | ✓ | Filter | Hybrid | L=0, Y=10, O=4, W=1, S=1 |
| LSH-IS-F[6] | ✓ | Filter | Hybrid | L=0, Y=10, O=4, W=1, S=1 |
| Wilson's ENN[7] | ✓ | Filter | Edit | k=3 |

## 4. Results and Discussion

In this section, we report the results regarding the comparative results of the instance selection methods.

The illustration in which the proposed algorithm reduces the boundary instances on the seeds data set is shown in Figure 1. The seeds data set is a data set related

[1] http://archive.ics.uci.edu/ml

[2] https://www.openml.org/

[3] https://www.mathworks.com/help/stats/sample-data-sets.html

[4] https://github.com/mohaslani/BPLSH

[5] https://github.com/mohaslani/DR.LSH

[6] https://github.com/alvarag/LSH-IS

[7] https://github.com/LucyKuncheva/Instance_selection

to Life sciences. According to the results, the reduction rates of the instances are respectively 27.14%, 39.05%, and 43.33% for k = 1, k = 2, and k = 3.

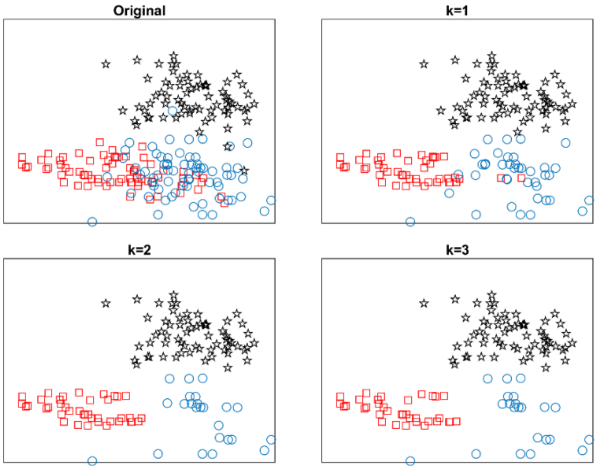

**Figure 1.** The illustration of reduction of the boundary instances on the seeds data set (3rd and 7th features), according to nearest neighbors (i.e., the value k)

Figure 2 shows the illustration in which the proposed algorithm reduces the boundary instances on the HTRU2 data set. The HTRU2 data set is a data set related to Physical sciences. According to the results, the reduction rates of the instances are respectively 4.34%, 7.96%, and 10.99% for k = 1, k = 3, and k = 5.

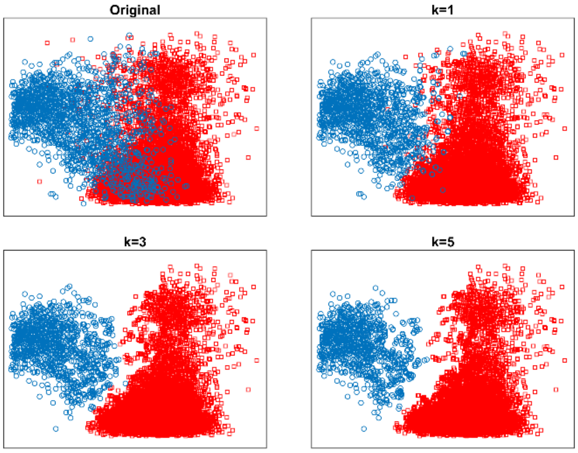

**Figure 2.** The illustration of reduction of the boundary instances on the HTRU2 data set (1st and 6th features), according to nearest neighbors (i.e., the value k)

Figure 3 shows the illustration in which the proposed algorithm reduces the boundary instances on the Human Activity data set. The Human Activity data set is a data set related to Health sciences. According to the results, the reduction rates of the instances are respectively 6.95%, 16.10%, and 21.09% for k = 1, k = 3, and k = 5.

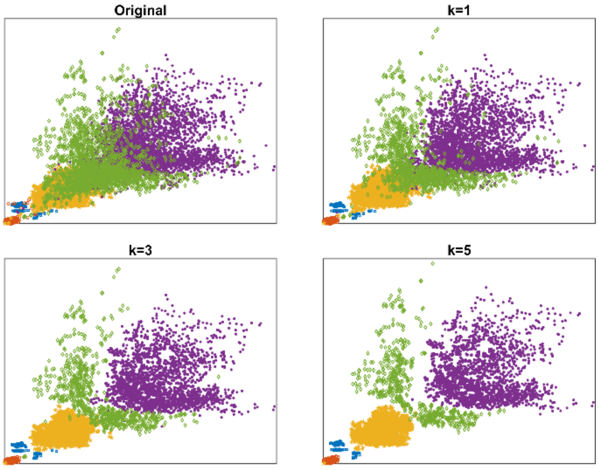

**Figure 3.** The illustration of reduction of the boundary instances on the human activity data set (4th and 6th features), according to nearest neighbors (i.e., the value k)

The illustration in which the proposed algorithm reduces the boundary instances on the madelon data set is shown in Figure 4. The madelon data set is an artificial data set. According to the results, the reduction rates of the instances are respectively 50.95%, 77.85%, and 90.25% for k = 1, k = 2, and k = 3.

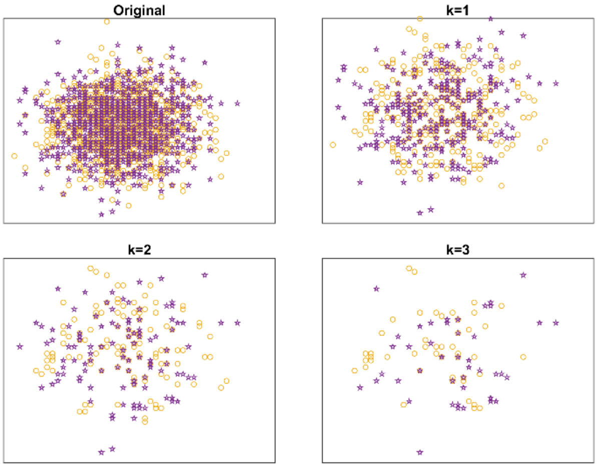

**Figure 4.** The illustration of reduction of the boundary instances on the madelon data set (1st and 4th features), according to nearest neighbors (i.e., the value k)

The comparative results of the algorithms in terms of accuracy rate are shown in Figure 5. BIRCH yields the highest rate with 87.01% classification accuracy after the baseline with 87.45%. Besides, BIRCH delivers the highest classification accuracy on two data sets (#1 and #5) in comparison to the other methods. The lowest and highest classification accuracy of the proposed method are 58.08% and 99.82%, respectively. BIRCH ranks third in terms of average reduction rate. This situation demonstrates that there is a trade-off between accuracy rate and reduction rate. According to Kruskal-Wallis test results, accuracy rates do not have mean ranks significantly different from each other.

Figure 6 shows the comparative results of the algorithms in terms of average reduction rate. According to the results, DR.LSH has the highest average reduction rate. According to Kruskal-Wallis test results, reduction rates do not have mean ranks significantly different from each other. BIRCH ranks third. Wilson’s ENN ranks last, as well. Accordingly, it is apparent that BIRCH is comparable to the hybrid, edition, or condensation methods. Although BIRCH and Wilson’s ENN search nearest neighbors of an instance, BIRCH is faster than the well-known similar approaches. As is known to all, Wilson’s ENN is faster than CNN and so on.

The comparative results of the algorithms in terms of average running time are shown in Figure 7. According to Kruskal-Wallis test results, running time of Wilson’s ENN has mean ranks significantly different from other methods. It is obvious that the slowest method is Wilson’s ENN. LSH-IS-S and LSH-IS-F are the fastest methods. BIRCH ranks fourth. Considering these three criteria, we would like to remark that BIRCH is more balanced compared to the other methods.

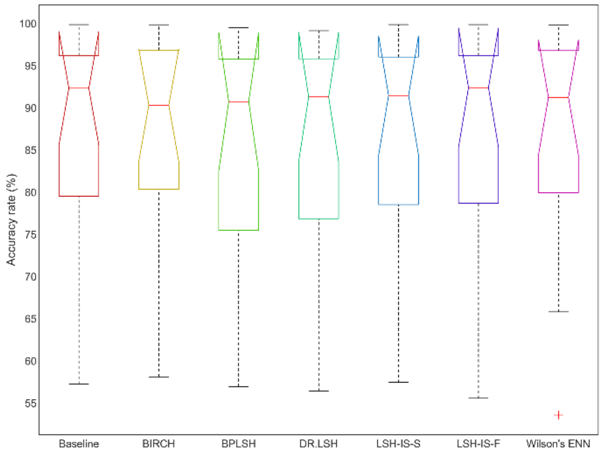

**Figure 5.** The comparative results of the algorithms in terms of accuracy rate (%)

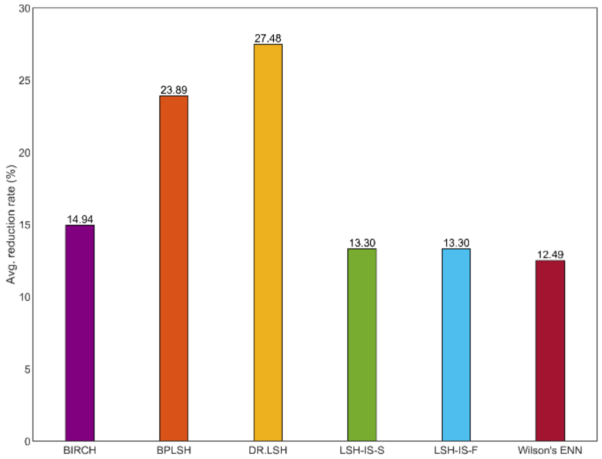

**Figure 6.** The comparative results of the algorithms in terms of reduction rate (%)

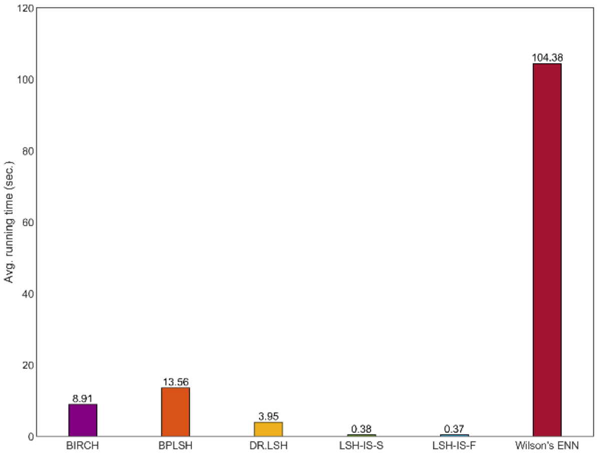

**Figure 7.** The comparative results of the algorithms in terms of running time

The trade-off between accuracy rate and reduction rate according to the number of neighbors is shown in Figure 8. Apart from the Boston Housing data set, while the accuracy rates on the other data sets decrease to an insignificant extent their reduction rates increase to a remarkable extent. We draw attention to the Boston housing data sets has 92 classes. Accordingly, as the k value increases on data sets where have the many classes the accuracy rate decreases quickly. BIRCH can attain a good trade-off between accuracy rate and reduction rate by adjusting the number of neighbors. Finally, we take the number of the nearest neighbors as 1 by default to obtain the maximum accuracy rate. In general, as the number of neighbors increases the accuracy rate decreases. Hence, it is more suitable to empirically determine the appropriate value of k. Thus, the accuracy rate also maintains as possible while the reduction rate increases. Further, the geometric average of k-value for maximum classification accuracy on the data sets is calculated as approximately 1.49. Hence, we set the k-value as 1.

Figure 9 shows the variation in the running time of BIRCH in terms of the number of neighbors. Considering the results, the running time of BIRCH does not rise excessively as the number of neighbors increases. This situation shows that BIRCH maintains a stable runtime performance. Thereby, the suitable accuracy rate-reduction rate balance can be obtained by increasing the k-value without performance loss. Consequently, the proposed method provides a satisfactory trade-off between classification accuracy and the reduction rate. The time complexity of the proposed method is log-linear, and it can achieve both high classification accuracy and reduction rates over many data sets by tuning the number of neighbors. Finally, the proposed method promises to remove more boundary instances by providing to reach high accuracy rates over data sets.

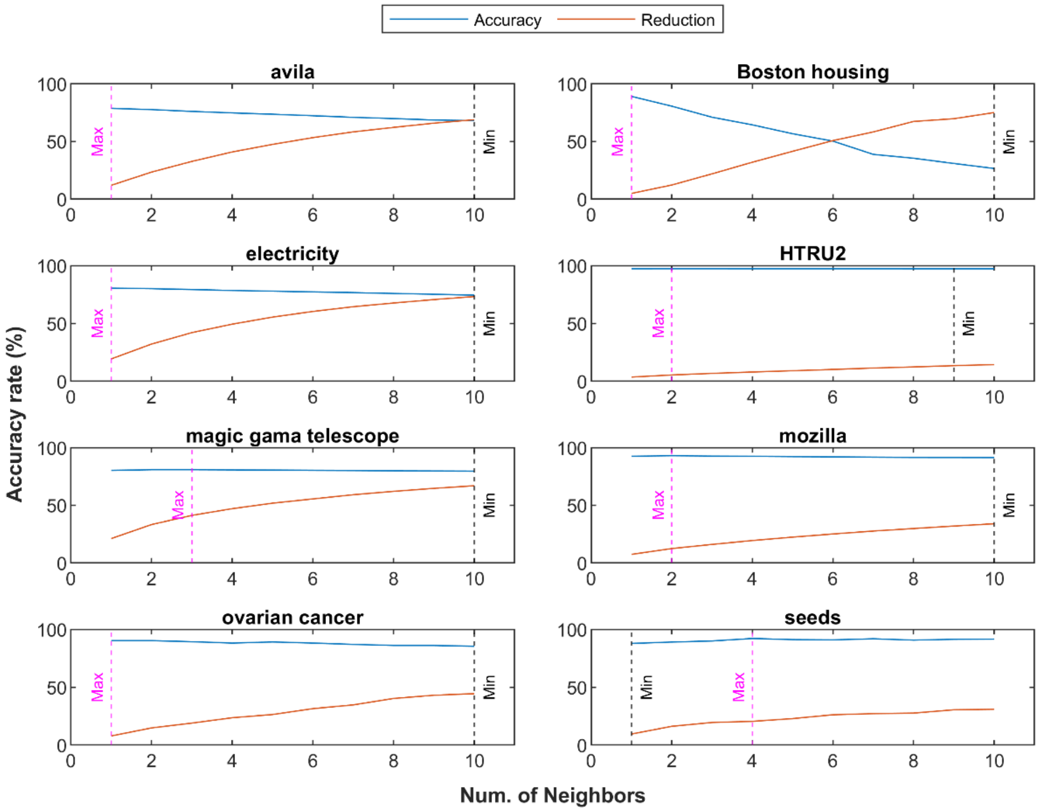

**Figure 8.** The trade-off between accuracy rate and reduction rate according to the number of neighbors

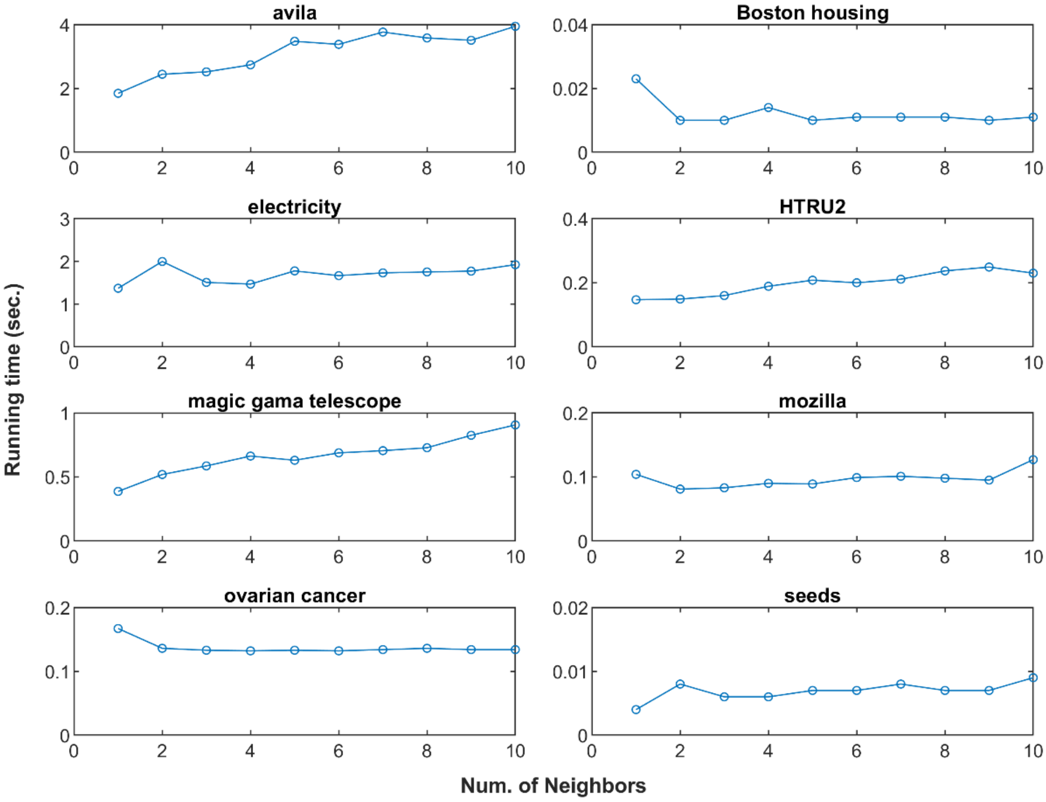

**Figure 9.** The variation in the running time of BIRCH in terms of the number of neighbors

## 5. Conclusions

In this paper, we propose a new instance selection algorithm called Border Instances Reduction using Classes Handily (BIRCH). We have tested the performance of BIRCH by using fifteen data sets from different domains and compared it with one traditional and four state-of-the-art instance selection methods in recent literature. Accordingly, BIRCH delivers a better trade-off between the accuracy rate and the reduction rate in comparison to the other methods. The time complexity of BIRCH is log-linear in the best case and log-quadratic in the worst case. BIRCH can acquire both high accuracy rates and reduction rates over lots of data sets by adjusting the number of neighbors. Principally, the reduction rate decreases as the accuracy rate increases. The proper tradeoff between accuracy rate, reduction rate, and speed-up is what is supposed to be focused on. BIRCH guarantees to discard more boundary instances by allowing to attain high classification accuracy over data sets. The future work of this study is to develop an unsupervised extension of BIRCH.

## References

Akinyelu, A. A. and Adewumi, A. O. (2017) 'Improved Instance Selection Methods for Support Vector Machine Speed Optimization', *Security and Communication Networks*, 2017, pp. 1–11. doi: 10.1155/2017/6790975.

Akinyelu, A. A. and Ezugwu, A. E. (2019) 'Nature Inspired Instance Selection Techniques for Support Vector Machine Speed Optimization', *IEEE Access*, 7, pp. 154581–154599. doi: 10.1109/ACCESS.2019.2949238.

Alpaydin, E. (1997) 'Voting over Multiple Condensed Nearest Neighbors', *Artificial Intelligence Review*, 11(1/5), pp. 115–132. doi: 10.1023/A:1006563312922.

Arnaiz-González, Á. *et al.* (2016) 'Instance selection of linear complexity for big data', *Knowledge-Based Systems*, 107, pp. 83–95. doi: 10.1016/j.knosys.2016.05.056.

Aslani, M. and Seipel, S. (2020) 'A fast instance selection method for support vector machines in building extraction', *Applied Soft Computing*, 97, p. 106716. doi: 10.1016/j.asoc.2020.106716.

Aslani, M. and Seipel, S. (2021) 'Efficient and decision boundary aware instance selection for support vector machines', *Information Sciences*, 577, pp. 579–598. doi: 10.1016/j.ins.2021.07.015.

Cover, T. and Hart, P. (1967) 'Nearest neighbor pattern classification', *IEEE Transactions on Information Theory*, 13(1), pp. 21–27. doi: 10.1109/TIT.1967.1053964.

García-Pedrajas, N. (2011) 'Evolutionary computation for training set selection', *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(6), pp. 512–523. doi: 10.1002/widm.44.

Garcia, S. *et al.* (2012) 'Prototype Selection for Nearest Neighbor Classification: Taxonomy and Empirical Study', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(3), pp. 417–435. doi: 10.1109/TPAMI.2011.142.

Hart, P. (1968) 'The condensed nearest neighbor rule (Corresp.)', *IEEE Transactions on Information Theory*, 14(3), pp. 515–516. doi: 10.1109/TIT.1968.1054155.

Liu, C. *et al.* (2017) 'An efficient instance selection algorithm to reconstruct training set for support vector machine', *Knowledge-Based Systems*, 116, pp. 58–73. doi: 10.1016/j.knosys.2016.10.031.

Olvera-López, J. A. *et al.* (2010) 'A review of instance selection methods', *Artificial Intelligence Review*, 34(2), pp. 133–143. doi: 10.1007/s10462-010-9165-y.

Rico-Juan, J. R., Valero-Mas, J. J. and Calvo-Zaragoza, J. (2019) 'Extensions to rank-based prototype selection in k-Nearest Neighbour classification', *Applied Soft Computing*, 85, p. 105803. doi: 10.1016/j.asoc.2019.105803.

Ruiz, I. L. and Gómez-Nieto, M. Á. (2020) 'Prototype Selection Method Based on the Rivality and Reliability Indexes for the Improvement of the Classification Models and External Predictions', *Journal of Chemical Information and Modeling*, 60(6), pp. 3009–3021. doi: 10.1021/acs.jcim.0c00176.

Sun, X. *et al.* (2019) 'Fast Data Reduction With Granulation-Based Instances Importance Labeling', *IEEE Access*, 7, pp. 33587–33597. doi: 10.1109/ACCESS.2018.2889122.

Susheela Devi, V. and Murty, M. N. (2002) 'An incremental prototype set building technique', *Pattern Recognition*, 35(2), pp. 505–513. doi: 10.1016/S0031-3203(00)00184-9.

Wang, Z., Tsai, C.-F. and Lin, W.-C. (2021) 'Data cleaning issues in class imbalanced datasets: instance selection and missing values imputation for one-class classifiers', *Data Technologies and Applications*, ahead-of-p(ahead-of-print). doi: 10.1108/DTA-01-2021-0027.

Wilson, D. L. (1972) 'Asymptotic Properties of Nearest Neighbor Rules Using Edited Data', *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-2(3), pp. 408–421. doi: 10.1109/TSMC.1972.4309137.

Wilson, D. R. and Martinez, T. R. (2000) 'Reduction techniques for instance-based learning algorithms', *Machine Learning*, 38, pp. 257–286.

Yang, L. *et al.* (2019) 'Constraint nearest neighbor for instance reduction', *Soft Computing*, 23(24), pp. 13235–13245. doi: 10.1007/s00500-019-03865-z.