**Journal of Mathematical Modeling**
Vol. 4, No. 3, 2021, pp. 677-690. Research Article

JMM

# Streamwise feature selection on big data using noise resistant rough functional dependency

**Sadegh Eskandari**[*]

*Department of Computer Science, University of Guilan, Rasht, Iran*
*Email(s): eskandari@guilan.ac.ir*

**Abstract.** Online Streaming Features (OSF) is a data streaming scenario, in which the number of instances is fixed while feature space grows with time. This paper presents a rough sets-based online feature selection algorithm for OSF. The proposed method, which is called OSFS-NRFS, consists of two major steps: (1) online noise resistantly relevance analysis that discards irrelevant features and (2) online noise resistanlty redundancy analysis, which eliminates redundant features. To show the efficiency and accuracy of the proposed algorithm, it is compared with two state-of-the-art rough sets-based OSFS algorithms on eight high-dimensional data sets. The experiments demonstrate that the proposed algorithm is faster and achieves better classification results than the existing methods.

*Keywords*: Feature selection, online feature selection, streaming feature selection, rough sets.
*AMS Subject Classification 2010*: 68T10, 68T20.

## 1 Introduction

Feature selection has become an integral part of data mining tasks in the big data era where we encounter huge datasets with thousands of irrelevant and redundant information [9,14]. For example, the educational data mining dataset from KDD CUP 2010 [36] has about 29 million features. Most of the traditional feature selection methods process all the features together and hence, are not applicable in such big data due to the computational concerns.

In addition to feature space size, the growth of feature space is another issue in big data that the traditional feature selection methods are not able to handle. For example, since the outbreak of the Covid-19, hundreds of new features have been reported daily by various laboratories and research centers around the world on the virus and it's detection [26,34,37]. A rudimentary approach in these dynamically growing scenarios is to wait a long time for all features to become

[*] Corresponding author.

Received: 14 January 2021/ Revised: 1 April 2021/ Accepted: 22 April 2021
DOI: 10.22124/jmm.2021.18707.1603

available and then carry out the feature selection methods. However, due to the importance of optimal decisions at every time step, a more rational approach is to design an online streaming features selection (OSFS) method which selects a best feature subset from so-far-seen information and updates the subset on the fly when new features stream in.

Designing a general purpose OSFS algorithm is a very challenging problem. The main reason for this challenge is the lack of domain knowledge, such as the size, distribution, order, type and etc, about the streaming feature space. Although domain knowledge is an integral part of most of the traditional feature selection algorithms, an OSFS algorithm should not require such knowledge. This constraint makes the rough sets [28] an ideal tool to work, because rough sets-based operations do not require any domain knowledge other than the given data.

Several rough sets based OSFS algorithms are proposed in the literature [9,17,19]. Although these algorithms are able to handle streaming features, the accuracy of selected subsets is still an open problem and we need more accurate algorithms that allow reliable classification and learning tasks at each time instance. Motivated by this problem, a new OSFS algorithm, called OSFS-NRFS, is proposed in this paper. This algorithm uses a newly defined functional dependency concept called noise resistance rough functional dependency. OSFS-NRFS consists two major steps: 1) Online noise resistant relevance analysis that discards irrelevant features and, 2) Online noise resistant redundancy analysis that discards redundant features.

In summary, the unique contributions that distinguish the proposed work from existing approaches are threefold: 1) Our work advances the OSFS problem one step further for handling big and noisy data; 2) a novel feature relevance/redundancy criterion based on newly defined noise resistance rough functional dependency is proposed which needs no human input or domain knowledge; and 3) a new OSFS algorithm is proposed, with extensive comparisons and experimental studies to prove its accuracy and speed.

The remainder of this paper is structured as follows: Sections 2 and 3 summarize the related work and theoretical background, respectively. Section 4 discusses the proposed noise resistant functional dependency concepts and presents the OSFS-NRFD algorithm. Section 5 reports experimental results, and Section 6 concludes the paper.

## 2 Related work

Hundreds of FS algorithms have been proposed in the literature [3,14,43]. In terms of selection approach, FS algorithms can be classified into wrapper, filter and embedded methods. Wrappers [5,12,13,20,25], train a learning algorithm for each candidate feature subset and select the subset with maximum accuracy. Although these algorithms are highly accurate, they are computationally very expensive. Filters [2,7,8,16,23,29,33,39], evaluate each candidate subset using criteria such as consistency, relevance, information and etc. As the evaluation is independent of the learning process, filters are very fast and unbiased to any learner. Eembedded approaches [24,30,44], consider FS as a regularization term in loss function. These algorithms try to make a trade-off between the accuracy and complexity of the training model, by removing or selecting candidate features.

Several filter and embedded OSFS algorithms are proposed in the literature:

- **Embedded methods.** Online grafting [31], Information-Investing [40] and alpha-investing

[45] are three embedding OSFS algorithms. These algorithms try to make a trade-off between the accuracy and complexity of the training model, by removing or selecting candidate features. Online grafting algorithm [31] considers the feature selection task as part of a risk minimization problem. This algorithm selects a new incoming feature if it improves the model accuracy more than a predefined threshold. Information-Investing algorithm [40], selects a new incoming feature if it reduces the model entropy more than the cost of the feature coding. Alpha-Investing [45] is a version of Information-Investing, which uses the statistical p-value as a criterion for selecting or discarding features. Although these embedding algorithms are designed to handle streaming features, they are unable to deal with true OSF scenarios for three reasons: (1) they all need to access global feature space for hyper-parameter tuning. For example, suitable $\lambda$ setting in online grafting requires information about the entire feature space; (2) Using these algorithms, the selected subset size grows incrementally with time. This is because of the fact that these algorithms does not eliminate the previously selected features, even if they became redundant due to new streamed features. 3) Considering OSFS as a training part of a special model, results feature subsets which are biased to the model.

- **Filter methods.** Several filter OSFS algorithms are proposed in the literature [9, 17, 19, 21, 22, 32, 42, 46]. In these methods, the feature subset evaluation is independent of the classifier or induction algorithm. For each candidate subset of features, evaluation measures such as information, consistency, relevance and etc are applied and the best feature subset is selected. The fast-OSFS algorithm [42] gradually generates a Markov-blanket of feature space using causality-based measures. For any new incoming feature, this algorithm executes two processes: an online relevance analysis and then an online redundancy analysis. SFS-RS [17] is a rough sets-based OSFS algorithm that uses classical feature significance measure to eliminate irrelevant features in a top-down manner. Using rough sets, this algorithm does not require any human input or domain knowledge other than the given data. OS-NRRSAR-SA [9] is an extension to the SFS-RS algorithm that adopts a noise resistance dependency measure for the significance analysis. OSFS-MRMS [19] is another extension of SFS-RS that filters out the redundant features before the significance analysis step. This algorithm uses a redundancy measure based on functional dependency concept in a bottom-up fashion. OFS-Density [46] is a similar algorithm to SFS-RS (and OS-NRRSAR-SA) that use neighborhood rough sets-based measure for feature significance analysis. OSFSMI [32] uses the well-known mutual information to eliminate irrelevant and/or redundant features in OSF.

## 3 Rough sets

There are several tools for expressing uncertainty in data, including probability theory, fuzzy set theory, and rough set theory. Rough sets theory has been introduced by Pawlak [28] to express vagueness by means of boundary region of a set. The main advantage of this theory is that it requires no domain knowledge other than the given dataset [27]. This section describes the fundamentals of the theory.

### 3.1 Information system and indiscernibility

An information system is a tuple $IS = (U, F)$, where $U$ is a non-empty finite set of objects called the universe and $F$ is a non-empty finite set of features such that $f : U \to V_f$, for every $f \in F$. For any set $B \subseteq F$, the B-indiscernibility relation is defined as:

$$IND_{IS}(B) = \{(x, y) \in U \times U | \forall f \in B, f(x) = f(y)\}. \tag{1}$$

Equivalence classes of the relation $IND_{IS}(B)$ are denoted $[x]_B$ and referred to as B-elementary sets. The partitioning of U into B-elementary subsets, denoted $U/B$, is the common computational routine for rough set-based operations. The worst case time complexity of this routine is $O\left(|U|^2|B|\right)$ [19].

### 3.2 Lower and upper approximations

Lower and upper approximations are two fundamental concepts of rough sets that define information contained in a set. Let $B \subseteq F$ and $X \subseteq U$, the $B-$lower approximation of $X$ specifies all the elements in $U$ that certainly belong to $X$. This approximation is defined as:

$$\underline{B}X = \{x |\, [x]_B \subseteq X\}. \tag{2}$$

The $B-$upper approximation of $X$ specifies all the elements in $U$ that may or may not belong to $X$. This approximation is defined:

$$\overline{B}X = \{x |\, [x]_B \cap X \neq \emptyset\}. \tag{3}$$

By the definition of $\underline{B}X$ and $\overline{B}X$, the objects in $U$ can be partitioned into three parts, called the positive, boundary and negative regions:

$$POS_B(X) = \underline{B}X, \tag{4}$$

$$BND_B(X) = \overline{B}X - \underline{B}X, \tag{5}$$

$$NEG_B(X) = U - \overline{B}X. \tag{6}$$

### 3.3 Noise resistant dependency measure

Quantifying dependencies between feature subsets is the main issue in feature selection tasks. Let $D$ and $C$ be subsets of $F$. The noise resistant dependency measure proposed in [18] tries to quantify a combination of two dependency types:

1. The classical dependency represented by positive region which is defined as:

$$\gamma(C, D) = \frac{|\bigcup_{X \in U/D} \underline{C}X|}{|U|}. \tag{7}$$

2. The dependency that probably lost due to noise. This dependency uses an impurity rate value to calculate the noisy portion of a set. Let $A$ and $B$ be two sets. The impurity rate of $A$ with respect to $B$ can be defined as follows:

$$c(A, B) = \frac{|A - B|}{|A|}. \tag{8}$$

This value calculates the portion of the elements that must be eliminated from $A$ to make it totally included in $B$. Using the impurity rate, the $B$-related information that could be retrieved after removing impurities from $A$ can be formulated as:

$$\xi(A,B) = \begin{cases} 1 - c(A,B), & \text{if } c(A,B) \leq 0.5, \\ 0, & \text{if } c(A,B) > 0.5. \end{cases} \tag{9}$$

This formulation can be applied to elementary sets to extract information that may be unseen in calculating lower approximations. In this regard, the noise measure function, $\phi$, is defined as:

$$\phi_B(X) = \frac{\sum_{Y\in U/B} \xi(Y,X) \quad [\xi(Y,X) \neq 1]}{|U/B|}. \tag{10}$$

This function quantifies the possibility of transferring some objects from the boundary to the positive region of a set, if the noise elements could be removed. Using this function, the noisy dependency of $D$ on $C$ can be defined as follow:

$$\nu(C,D) = \sum_{Y\in U/D} \phi_C(Y). \tag{11}$$

As $\gamma(C,D)$ only operates on the objects in positive region and $\nu(C,D)$ only on the objects in boundary region, the two operators are combined to create a noise resistant evaluation measure $\rho$ [18]:

$$\rho(C,D) = \frac{\nu(C,D) + \gamma(C,D)}{2}. \tag{12}$$

# 4 The proposed OSFS method

In this section, we first define the notations of noise resistant functional dependency and then based on this concept, we define the notations of redundant and irrelevant features. Finally, we propose an OSFS algorithm that uses the redundant and irrelevant features concepts for feature selection with streaming features.

## 4.1 Noise resistant functional dependency

In supervised machine learning, the goal is to learn the mapping from a feature space $F$ to a target variable $d$. To consider the target variable, we use the notion of decison systesm. In the definitions below, a decision system is an information system of the form $DS = (U, F, d)$, where d is called the decision feature and $U$ and $F$ represent the universe and the full set of conditional features, respectively. Moreover, $F - \{f\}$ represents the feature subset excluding the single feature $f$.

**Definition 1** (Noise resistant functional dependency)**.** *Let $D$ and $C$ be subsets of $F \cup \{d\}$. For $0 \leq k \leq 1$, it is said that $D$ is noise resistant functionally dependent on $C$ in the kth degree (denoted $C \Rightarrow_k^\rho D$), if*

$$k = \rho(C,D). \tag{13}$$

Table 1: An Example Dataset.

| $x \in U$ | $f_1$ | $f_2$ | $f_3$ | $d$ |
|---|---|---|---|---|
| 1 | 1 | 3 | 1 | 0 |
| 2 | 2 | 1 | 1 | 4 |
| 3 | 1 | 3 | 1 | 0 |
| 4 | 1 | 3 | 1 | 3 |
| 5 | 2 | 1 | 1 | 2 |
| 6 | 0 | 2 | 1 | 3 |
| 7 | 2 | 1 | 1 | 0 |
| 8 | 2 | 1 | 1 | 2 |
| 9 | 1 | 3 | 1 | 2 |
| 10 | 1 | 3 | 1 | 4 |

**Definition 2** (Noise resistantly redundant feature subset)**.** *A feature subset $C \subset F$ is a noise resistantly redundant subset for $DS = (A, F, d)$ iff $\exists F' \subseteq F - C$ s.t. $F' \Rightarrow^{\rho}_{1} C$, otherwise it is noise resistantly non-redundant subset.*

Noise resistantly redundant features can be described using some other features in the conditional feature set and therefore they can be eliminated without loosing useful information.

**Definition 3** (Noise resistantly irrelevant feature subset )**.** *A feature subset $F' \subseteq F$ is a noise resistantly irrelevant feature subset for $DS = (A, F, d)$ iff $\rho(F, d) - \rho(F - F', d) = 0$.*

Noise resistantly irrelevant features are dispensable and can be eliminated from the decision system.

### 4.1.1 An example

To illustrate the notations of noise resistant functional dependency, a small example decision system is considered (Table 1). This decision system contains ten discrete-valued objects.

Let $C = \{f_1\}$ and $D = \{d\}$, then based on equation (12),

$$\rho(C, D) = \frac{\nu(C, D) + \gamma(C, D)}{2}.$$

Using equation (7), $\gamma(f_1, d)$ can be calculated:

$$\begin{aligned}
\gamma(\{f_1\}, d) &= \frac{|POS_{\{f_1\}}(d)|}{10} = \frac{|\bigcup_{X \in U/d} \underline{\{f_1\}} X|}{10} \\
&= \frac{|\underline{\{f_1\}}\{1,3,5\} \cup \underline{\{f_1\}}\{5,8,9\} \cup \underline{\{f_1\}}\{2,10\} \cup \underline{\{f_1\}}\{4,6\}|}{10} \\
&= \frac{|\emptyset \cup \emptyset \cup \emptyset \cup \{6\}|}{10} = 0.1.
\end{aligned}$$

The noisy dependency of $d$ on $f_1$ can be calculated using equation (11),

$$\begin{aligned}\nu(\{f_1\},d) &= \sum_{Y\in U/d} \phi_{\{f_1\}}(Y)\\ &= \phi_{\{f_1\}}(\{1,3,7\}) + \phi_{\{f_1\}}(\{5,8,9\}) + \phi_{\{f_1\}}(\{2,10\}) + \phi_{\{f_1\}}(\{4,6\})\\ &= \frac{\xi\left(\{1,3,4,9,10\},\{1,3,7\}\right) + \xi\left(\{2,5,7,8\},\{1,3,7\}\right) + \xi\left(\{6\},\{1,3,7\}\right)}{3}\\ &+ \frac{\xi\left(\{1,3,4,9,10\},\{5,8,9\}\right) + \xi\left(\{2,5,7,8\},\{5,8,9\}\right) + \xi\left(\{6\},\{5,8,9\}\right)}{3}\\ &+ \frac{\xi\left(\{1,3,4,9,10\},\{2,10\}\right) + \xi\left(\{2,5,7,8\},\{2,10\}\right) + \xi\left(\{6\},\{2,10\}\right)}{3}\\ &+ \frac{\xi\left(\{1,3,4,9,10\},\{4,6\}\right) + \xi\left(\{2,5,7,8\},\{4,6\}\right) + \xi\left(\{6\},\{4,6\}\right)}{3}\\ &= \frac{0+0+0.5}{3} = \frac{1}{6}.\end{aligned}$$

Having calculated the two measures, the noisy dependency measure can be calculated:

$$\rho(\{f_1\},d) = \frac{\frac{1}{10}+\frac{1}{6}}{2} = 0.1333.$$

and therefore $\{f_1\} \Rightarrow^{\rho}_{0.1333} \{d\}$. Using similar calculations, we can see that $\{f_1\} \Rightarrow^{\rho}_{1} \{f_2\}$. This means that $f_2$ can be described using $f_1$ and therefore $f_2$ is a noise resistantly redundant feature for the decision system. Moreover, we can see that $\{f_1, f_3\} \Rightarrow^{\rho}_{0.1333} \{d\}$, hence, $f_3$ is an irrelevant feature for the decision system because $\rho(\{f_1, f_3\}, d) = \rho(\{f_1\}, d)$.

## 4.2 OSFS using noise resistant functional dependency

Suppose that $DS_t = (U_t, F_t, d)$ is a decision system at time $t$. In online streaming features (OSF), for every $t' > t$, $|F_{t'}| \geq |F_t|$ and $|U_{t'}| = |U_t|$. Because the full feature space is not accessible in OSF scenario, the selected subset must be gradually built over time based on features streamed so far. Algorithm 1 represents the proposed OSFS-NRFS algorithm. This algorihm keeps a best subset ($R$) from so far seen features and updates whenever a new feature streams in. Using Definitions 1-3, this algorithm contains two steps: (1) online noise resistantly relevance analysis that discards irrelevant features. (2) online noise resistantly redundancy analysis, which eliminates redundant features from the features selected so far. When a new feature $f$ streams in, the algorithm tests its relevance to decision feature $d$ in the first step. If $f$ is not noise resistantly relevant, the algorithm simply rejects $f$. Otherwise, the algorithm adds $f$ to selected feature subset $R$. If the new feature is not rejected, the algorithm executes the second step. This step sorts the features in $R$ according to their relevance and then starting from the least relevant feature, calculates the noise resistant functional dependency of each feature to the remaining features. If the feature is noise resistantly redundant, then the algorithm simply eliminates it.

### 4.2.1 The time complexity of OSFS-NRFS

The time complexity of OSFS depends on the number of $\rho$-tests. The time required by this test is $O(|R||U|^2)$ [9, 19]. Suppose that at time $t$ a new feature $f_t$ be present to the OSFS-NRFS

**Algorithm 1** OSFS-NRFS($d$).

```
$d$: The decision feature
$R = \emptyset$
while stopping criterion is not met do
    $f$ = GET-NEW-FEATURE()
    $added = False$
%First Phase: Online Noise Resistantly Relevance Analysis%
    if $\rho(R \cup \{f\}, d) - \rho(R, d) \neq 0$ then
        $R = R \cup \{f\}$
        $added = True$
    end if
%Second Phase: Online Noise Resistantly Redundancy Analysis%
    if added then
        $S = R$
        for $k = 1 : |R|$ do
            $g = argmin_{f' \in S}\{\rho(S, d) - \rho(S - \{f'\}, d)\}$
            if $S - \{g\} \Rightarrow_1^{\rho} \{g\}$ then
                $S = S - \{g\}$
            end if
        end for
        $R = S$
    end if
end while
```

algorithm and let $R_t$ be the selected feature subset at this time. At this time, the first phase of the algorithm will be triggered, which includes two $\rho$-tests for online noise resistantly relevance analysis. Therefore, the worst-case time complexity of this phase is $O(|R_t||U|^2)$. If $f_t$ is noise resistantly relevant feature, then the second phase of the algorithm will be triggered. This phase includes $|R_t|$ tests for removing noise resistantly redundant features. Therefore, the worst-case time complexity of this phase is $O(|R_t|^2|U|^2)$.

Although the worst-case time complexity of the proposed algorithm is square with respect to the number of selected features, in many real-world applications, only a small number of features in a large feature space are predictive and relevant to decision feature [9]. Therefore $|R_t|$ is so small that its square does not affect the time complexity of the OSFS algorithm, significantly.

## 5 Experimental results

In this section, we show the performance of the proposed method. To do this, the proposed OSFS-NRFS algorithm is compared with five state-of-the-art OSFS algorithms, information-investing [40], fast-OSFS [42], OSFSMI [32], OS-NRRSAR-SA [9] and OSFS-MRMS [19]. For information-investing, we set $W_0 = 0.5$ and $W_\Delta = 0.5$. For fast-OSFS, we used $G^2$ tests for all discrete (categorical and integer-valued) datasets and Fishers z-tests for all continues (real-valued) datasets. For both tests, we used 0.05 as the statistical significance level. For OS-

Table 2: Summary of the Benchmark High Dimensional Data sets.

| No | Dataset | # Attributes | # Train | # Test | Type | Source |
|---|---|---|---|---|---|---|
| 1 | dorothea | 100000 | 800 | 800 | Pharmacology | [6] |
| 2 | arcene | 10000 | 100 | 700 | Mass Spectrometry | [6] |
| 3 | dexter | 20000 | 300 | 2000 | Text classification | [6] |
| 4 | madelon | 500 | 2000 | 1800 | Artificial | [6] |
| 5 | VOC 2007 | 6096 | 5011 | 4952 | Image classification | [11] |
| 6 | VOC 2012 | 6096 | 11530 | 11001 | Image classification | [10] |
| 7 | mf | 649 | 2000 | – | Handwritten Digit Classification | [1] |
| 8 | arrhythmia | 279 | 452 | – | Health | [1] |

NRRSAR-SA, we adopted SBE3 implementation of the NON-SIGNIFICANT procedure and for OSFS-MRMS, we set the maximum subset size ($k$) in REDUNDANT routine to be 3. Kernel SVM with RBF kernel function [4] is employed for the classification of the data. For two class classification problems, average precision (AP%) is used as accuracy measure. For multi-class cases, we used the mean of the APs (mAP%) on different classes. Table 2 summarizes the eight datasets used in our experiments. For all the datasets, we considered features one by one to simulate OSF scenario. The dorothea, arcene, dexter, and madelon datasets are from the NIPS 2003 feature selection challenge [6]. The VOC2007 and VOC2012 are two image classification datasets from the PASCAL Visual Object Classes Challenge [10,11], and the mf and arrhythmia are from the UCI Repository of machine learning databases [1]. For VOC images, we used a combination of penultimate layers of three well-known convolutional neural networks: 1) VGG-VD [35] (4096 features), 2) GoogleNet [38] (1000 features) and 3) ResNet [15] (1000 features). The networks are pre-trained on ILSVRC [41].

Because of the fact that we do not have access to the full feature space, the streaming order of the features affects the final results. Therefore, in order to strengthen the comparison, we generated 30 different random streaming orders for each dataset.

## 5.1 Classification accuracy

The average SVM classification accuracy of selected subsets during features streaming are reported in Figure 1. The results are averaged over 30 random streaming orders. It should be noticed that the fast-OSFS algorithm failed to select a feature subset for arcene dataset. This is due to the fact that this algorithm uses conditional independence tests, which needs sufficiently large number of training instances.

As it can be seen, the proposed OSFS-NRFS algorithm performs very well and shows increase in classification accuracies for most of the tests. According to the recorded accuracy values for each data set (10 measurements on 30 streaming orders), OSFS-NRFS outperforms the OS-NRRSAR-SA, OSFS-MRMS, Information-Investing, fast-OSFS and OSFSMI in 71.25% and 67.5%, 97.3%, 97.25% and 98.5% of the cases, respectively. Moreover, considering all the records, the average accuracy of the OSFS-NRFS is 1.65%, 1.08%, 22.66%, 13.50% and 18.51% higher than OS-NRRSAR-SA, OSFS-MRMS, Information-Investing, fast-OSFS and OSFSMI, respectively.

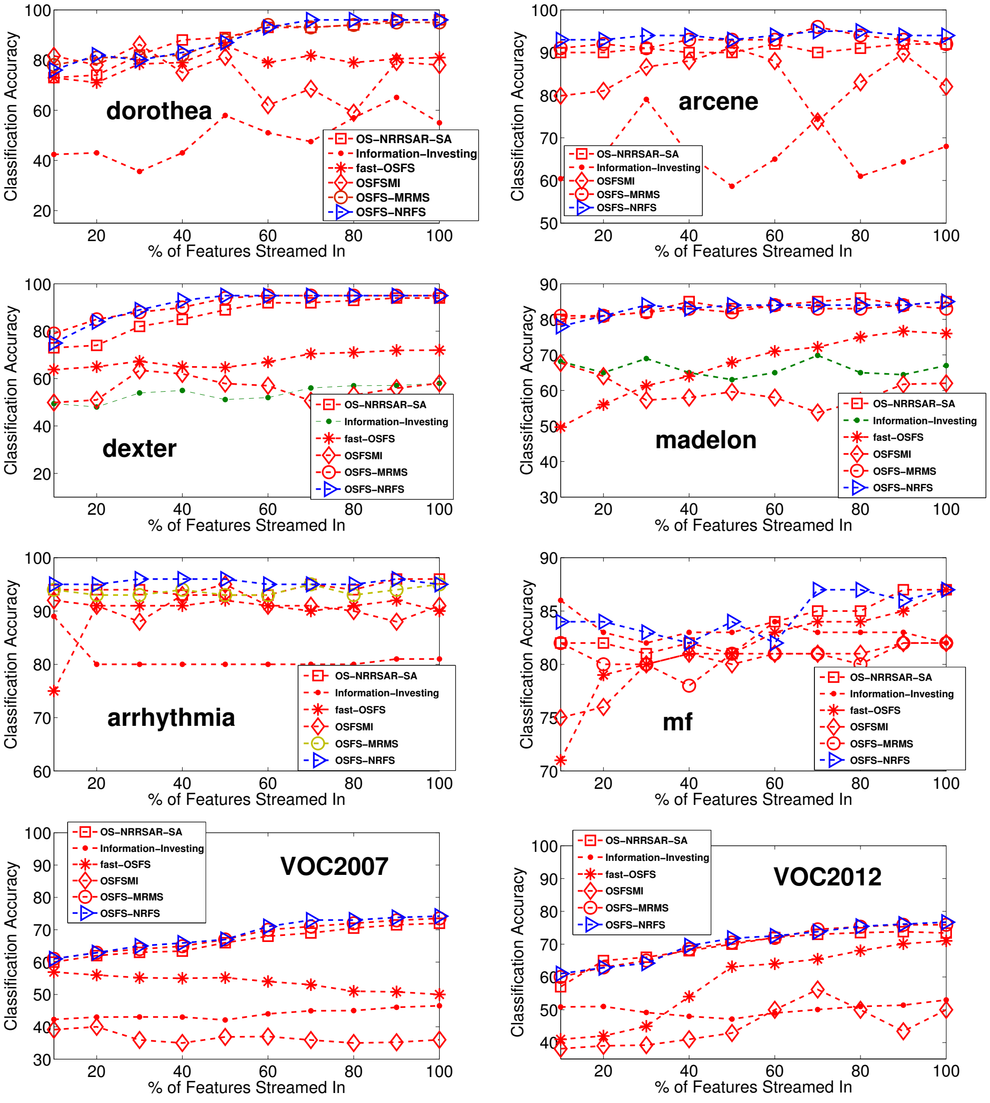

Figure 1: SVM classification accuracies during features streaming.

## 5.2 Running time

Table 3 reports the running times of the six algorithms at the end of the features streaming. As it can be seen, the Information-Investing, which is an embedded method, is the fastest algorithm. However, it should be noticed that this algorithm is the least accurate one. Comparing the filter methods (fast-OSFS, OSFSMI, OSFS-MRMS, OS-NRRSAR-SA and OSFS-NRFS), we see that the proposed OSFS-NRFS is superior for five cases, dorothea, arcene, dexter, VOC 2007 and mf. Although calculating the noise resistance measure in the proposed algorithm imposes an extra computational time, the smaller selected subsets during features streaming cause faster partitioning routine for calculating the elementary sets.

Table 3: Comparison of run times for OS-NRRSAR-SA, OSFS-MRMS and OSFS-NRFS.

| **Dataset** | **Information-Investing** | **fast-OSFS** | **OSFSMI** | **OS-NRRSAR-SA** | **OSFS-MRMS** | **OSFS-NRFS** |
|---|---|---|---|---|---|---|
| dorothea | 388.3 | 643.0 | 1283.9 | 509.7 | 486.9 | 477.3 |
| arcene | 11.3 | 77.5 | 93.9 | 82.2 | 80.6 | 71.3 |
| dexter | 101.9 | 502.8 | 638.5 | 572.7 | 511.0 | 475.5 |
| madelon | 9.7 | 79.3 | 89.4 | 87.0 | 123.8 | 113.8 |
| VOC 2007 | 482 | 3844.9 | 3392.1 | 3028.8 | 2472.9 | 2394.6 |
| VOC 2012 | 601 | 4832.9 | 5554.3 | 5421.8 | 3964.4 | 4216.0 |
| mf | 18.5 | 172.9 | 179.4 | 234.8 | 163.0 | 152.7 |
| arrhythmia | 19.4 | 72.9 | 133.5 | 123.1 | 118.3 | 118.9 |

## 6 Conclusions

This paper presented an OSFS method based on a newly proposed rough sets based functional dependency, called noise resistant functional dependency. The proposed method, which is called OSFS-NRFS, consists two major steps: 1) Online redundancy analysis that discards redundant features and, 2) Online noise resistantly relevance analysis that discards irrelevant features and (2) online noise resistanlty redundancy analysis, which eliminates redundant features. To show the efficiency and accuracy of the proposed algorithm, it was compared with five state-of-the-art OSFS algorithms OS-NRRSAR-SA, OSFS-MRMS, fast-OSFS, Information-Investing and OSFSMI. Eight high-dimensional data sets were used for comparisons, and their features considered one by one to simulate the true OSF scenarios. The running time and SVM classification accuracy during the features streaming were the comparison terms. The experiments demonstrate that the proposed algorithm achieves better results than existing algorithms.

## References

[1] C. Blake, *Uci repository of machine learning databases*, 1998, http://www.ics.uci.edu/~mlearn/MLRepository.htm.

[2] G. Brown, A. Pocock, M.J. Zhao, M. Luján, *Conditional likelihood maximisation: a unifying framework for information theoretic feature selection*, J. Mach. Learn. Res. **13** (2012) 27–66.

[3] G. Chandrashekar, F. Sahin, *A survey on feature selection methods*, Comput. Electr. Eng. **40** (2014) 16–28.

[4] C.C. Chang, C.J. Lin, *Libsvm: A library for support vector machines*, ACM Trans. Intell. Syst. Technol. **2** (2011) 1–27.

[5] G. Chen, J. Chen, *A novel wrapper method for feature selection and its applications*, Neurocomputing **159** (2015) 219–226.

[6] Clopinet, *Feature Selection Challenge, NIPS 2003*, 2003, http://clopinet.com/isabelle/Projects/NIPS2003. [Online; accessed 22-May-2019].

[7] M. Dash, H. Liu, *Consistency-based search in feature selection*, Artif. Intell. **151** (2003) 155–176.

[8] S. Eskandari, E. Akbas, *Supervised infinite feature selection*, arXiv:1704.02665, 2017, http://arxiv.org/abs/1704.02665.

[9] S. Eskandari, M.M. Javidi, *Online streaming feature selection using rough sets*, Int. J. Approx. Reasoning **69** (2016) 35–57.

[10] M. Everingham, S.M.A. Eslami, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, *The pascal visual object classes challenge: A retrospective*, Int. J. Comput. Vis. **111** (2015) 98–136.

[11] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, *The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results*, http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

[12] O. Gokalp, E. Tasci, A. Ugur, *A novel wrapper feature selection algorithm based on iterated greedy metaheuristic for sentiment classification*, Expert Syst. Appl. **146** (2020) 113176.

[13] J. González, J. Ortega, M. Damas, P. Martín-Smith, J.Q. Gan, *A new multi-objective wrapper method for feature selection–accuracy and stability analysis for bci*, Neurocomputing, **333** (2019) 407–418.

[14] I. Guyon, A. Elisseeff, *An introduction to variable and feature selection*, J. Mach. Learn. Res. **3** (2003) 1157–1182.

[15] K. He, X. Zhang, S. Ren, J. Sun, *Deep residual learning for image recognition*, Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern. Recognit. 2016, 770–778.

[16] Z. Hua, J. Zhou, Y. Hua, W. Zhang, *Strong approximate markov blanket and its application on filter-based feature selection*, Appl. Soft Comput. **87** (2020) 105957.

[17] M.M. Javidi, S. Eskandari, *Streamwise feature selection: A rough set method*, Int. J. Mach. Learn. Cybern. **9** (2016) 667–676.

[18] M.M. Javidi, S. Eskandari, *A noise resistant dependency measure for rough set-based feature selection*, J. Intell. Fuzzy Syst. **33** (2017) 1613–1626.

[19] M.M. Javidi, S. Eskandari, *Online streaming feature selection: a minimum redundancy, maximum significance approach*, Pattern Anal. Appl. **22** (2019) 949–963.

[20] R. Kohavi, G.H. Johnet al., *Wrappers for feature subset selection*, Artif. Intell. **97** (1997) 273–324.

[21] D. Lei, P. Liang, J. Hu, Y. Yuan, *New online streaming feature selection based on neighborhood rough set for medical data*, Symmetry **12** (2020) 1635.

[22] J. Liu, Y. Lin, Y. Li, W. Weng, S. Wu, *Online multi-label streaming feature selection based on neighborhood rough set*, Pattern Recognit. **84** (2018) 273–287.

[23] J. Ma, X. Gao, *A filter-based feature construction and feature selection approach for classification using genetic programming*, Knowl. Based Syst. **196** (2020) 105806.

[24] S. Maldonado, J. Lpez, *Dealing with high-dimensional class-imbalanced datasets: Embedded feature selection for svm classification*, Appl. Soft Comput. **67** (2018) 94–105.

[25] S. Maldonado, R. Weber, *A wrapper method for feature selection using support vector machines*, Inf. Sci. **179** (2009) 2208–2217.

[26] Y. Oh, S. Park, J.C. Ye, *Deep learning covid-19 features on cxr using limited training data sets*, IEEE Trans. Med. Imag., 2020.

[27] N. Parthalain, Q. Shen, R. Jensen, *A distance measure approach to exploring the rough set boundary region for attribute reduction*, IEEE Trans. Knowl. Data Eng. **22** (2009) 305–317.

[28] Z. Pawlak, *Rough sets*, Int. J. Comput. Inf. Sci. **11** (1982) 341–356.

[29] H. Peng, F. Long, C. Ding, *Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy*, IEEE Trans. Pattern Anal. Mach. Intell. **27** (2005) 1226–1238.

[30] B. Peralta, A. Soto, *Embedded local feature selection within mixture of experts*, Inf. Sci. **269** (2014) 176–187.

[31] S. Perkins, J. Theiler, *Online feature selection using grafting*, in Proceedings of the 20th International Conference on Machine Learning (ICML-03) 2003, 592–599.

[32] M. Rahmaninia, P. Moradi, *Osfsmi: Online stream feature selection method based on mutual information*, Appl. Soft Comput. **68** (2018) 733–746.

[33] G. Roffo, S. Melzi, M. Cristani, *Infinite feature selection*, Proc. IEEE Int. Conf. Comput. Vis. 2015, 4202–4210.

[34] A.T. Sahlol, D. Yousri, A.A. Ewees, M.A. Al-Qaness, R. Damasevicius, M. Abd Elaziz, *Covid-19 image classification using deep features and fractional-order marine predators algorithm*, Sci. Rep. **10** (2020) 1–15.

[35] K. Simonyan, A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, arXiv preprint arXiv:1409.1556, (2014).

[36] J. Stamper, A. Niculescu-Mizilm, S. Ritter, G. Gordon, K. Koedinger, *Data set from kdd cup 2010 educational data mining challenge*, https://pslcdatashop.web.cmu.edu/KDDCup/downloads.jsp, 2010.

[37] L. Sun, Z. Mo, F. Yan, L. Xia, F. Shan, Z. Ding, W. Shao, F. Shi, H. Yuan, H. Jiang, D. Wu, Y. Wei, Y. Gao, W. Gao, H. Sui, D. Zhang, D. Shen, *Adaptive feature selection guided deep forest for covid-19 classification with chest CT*, IEEE J. Biomed. Health Inform. **24** (2020) 2798–2805.

[38] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, *Going deeper with convolutions*, Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern. Recognit. 2015, 1–9.

[39] F. Thabtah, F. Kamalov, S. Hammoud, S.R. Shahamiri, *Least loss: A simplified filter method for feature selection*, Inf. Sci. **534** (2020) 1–15.

[40] L.H. Ungar, J. Zhou, D.P. Foster, B.A. Stine, *Streaming feature selection using IIC*, in AISTATS, 2005.

[41] A. Vedaldi, K. Lenc, *Matconvnet: Convolutional neural networks for matlab*, Proc. ACM Int. Conf. Multimed. 2015, 689–692.

[42] X. Wu, K. Yu, W. Ding, H. Wang, X. Zhu, *Online feature selection with streaming features*, IEEE Trans. Pattern Anal. Mach. Intell. **35** (2013) 1178–1192.

[43] B. Xue, M. Zhang, W.N. Browne, X. Yao, *A survey on evolutionary computation approaches to feature selection*, IEEE Trans. Evol. Comput. **20** (2015) 606–626.

[44] P. Zhao, B. Yu, *On model selection consistency of lasso*, J. Mach. Learn. Technol. **7** (2006) 2541–2563.

[45] J. Zhou, D. Foster, R. Stine, L. Ungar, *Streaming feature selection using alpha-investing*, in Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, ACM, 2005, 384–393.

[46] P. Zhou, X. Hu, P. Li, X. Wu, *Ofs-density: A novel online streaming feature selection method*, Pattern Recognit. **86** (2019) 48–61.