Learning Homophily Couplings from Non-IID Data for Joint Feature Selection and Noise-Resilient Outlier Detection

Guansong Pang[†] and **Longbing Cao**[†] and **Ling Chen**[†] and **Huan Liu**[‡]
[†]School of Software, University of Technology Sydney, Australia
[‡]Computer Science and Engineering, Arizona State University, USA
guansong.pang@student.uts.edu.au, {longbing.cao;ling.chen}@uts.edu.au, Huan.Liu@asu.edu

Abstract

This paper introduces a novel wrapper-based outlier detection framework (WrapperOD) and an instance (HOUR) for identifying outliers in noisy data (i.e., data with noisy features) with strong couplings between outlying behaviors. Existing subspace or feature selection-based methods are significantly challenged by such data, as their search of feature subset(s) is independent of outlier scoring and thus can be misled by noisy features. In contrast, HOUR takes a wrapper approach to iteratively optimize the feature subset selection and outlier scoring using a top-k outlier ranking evaluation measure as its objective function. HOUR learns homophily couplings between outlying behaviors (i.e., abnormal behaviors are not independent - they bond together) in constructing a noise-resilient outlier scoring function to produce a reliable outlier ranking in each iteration. We show that HOUR (i) retains a 2-approximation outlier ranking to the optimal one; and (ii) significantly outperforms five state-of-the-art competitors on 15 real-world data sets with different noise levels in terms of AUC and/or $P@n$. The source code of HOUR is available at https://sites.google.com/site/gspangsite/sourcecode.

1 Introduction

Outliers are rare or inconsistent objects, compared to the majority of objects. In recent applications such as insider trading, network intrusion detection and fraud detection, a key task is to detect unexpected objects in a sophisticated environment with noise and complex feature relations.

Unsupervised outlier detection methods assign each object an outlier score and report the top-ranked objects as outliers without using class labels. They have been receiving great attention due to the high cost of obtaining class labels in real-world applications. However, they face big challenges in handling data with a mixture of relevant and noisy features (such data is referred to as *noisy data* hereafter). In such data, outliers can be detected in relevant features while they are masked as normal objects with the inclusion of *noisy features* - features in which outliers may contain normal behaviors while normal objects may contain abnormal behaviors.

Subspace and feature selection are two major approaches to handle outlier detection in noisy data. Subspace outlier detection methods (e.g., FPOF [He *et al.*, 2005] and COMP [Akoglu *et al.*, 2012]) first identify a set of relevant feature subspaces/patterns and then apply outlier scoring functions to combine the *outlierness* (i.e., outlying degree) of objects in these subspaces. These methods separate subspace/pattern search from outlier scoring to facilitate modular design and the application of existing subspace/pattern discovery techniques into outlier detection. However, such search can be misled by noisy features and produce faulty subspaces/patterns, resulting in high false positives.

Outlying feature selection is to select relevant features for subsequent outlier detection. Limited work has been done in this area. Moreover, the existing work (i.e., [Pang *et al.*, 2016a; 2016b]) on *filter*-based approaches [Li *et al.*, 2016] selects a feature subset independently from subsequent learning methods. Consequently, the relevant features they retain can be noisy w.r.t. subsequent outlier detection methods. In contrast to filter-based approaches, *wrapper*-based approaches choose an optimal feature subset w.r.t. the learning methods [Kohavi and John, 1997]. However, although wrapper-based feature selection is popular for classification and clustering [Li *et al.*, 2016], as far as we know, no such work has been reported on outlier detection.

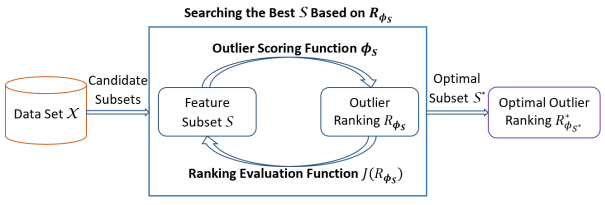

Figure 1: The Proposed WrapperOD Framework

This paper proposes a novel Wrapper-based Outlier Detection framework (*WrapperOD*) to detect outliers in noisy data. As shown in Figure 1, WrapperOD first defines *an outlier scoring function* to rank objects based on their outlierness in a given feature subset, and then designs *an outlier ranking evaluation function* to measure the relevance of the feature subset

by the outlier ranking quality. These two steps are iteratively performed until the best feature subset (alternatively the best outlier ranking) is obtained. Essentially, WrapperOD unifies the outlier ranking quality with the feature subset relevance into one objective function and makes a joint optimization.

We further instantiate WrapperOD to a Homophily cOupling-based oUtlieR detection method, called *HOUR*, for categorical data which has been insufficiently explored. Many real-world data often demonstrates strong *homophily coupling* between outlying behaviors (i.e., feature values) [Chau *et al.*, 2011; Pang *et al.*, 2016a]. That is, outlying behaviors are not independent and they tend to be concurrent. As a result, the outlierness of a behavior is dependent on its coupled behaviors, i.e., a behavior has large outlierness if it has strong linkage to many outlying behaviors and vice versa. HOUR treats such data as data with non-independent and identically distributed (non-IID) behaviors [Cao, 2014] and specifies a homophily coupling-based outlier scoring function to capture such non-IID behaviors. It further specifies the outlier ranking evaluation function to guide the joint optimization by maximizing the margin between the top-ranked k objects and the other objects. A heuristic search is used to generate reliable feature subsets.

This work makes the following two major contributions.

- We propose a novel WrapperOD framework to identify outliers in noisy data. In contrast to existing solutions that search feature subset(s) independently from outlier scoring, WrapperOD simultaneously optimizes its outlier scoring and feature selection, which enables its outlier scoring function to produce a much more reliable outlier ranking in noisy data.
- The performance of WrapperOD is verified by an instance HOUR. HOUR models homophily couplings between outlying behaviors to construct a fast and noise-resilient outlier scoring function that empowers the joint optimization in WrapperOD. HOUR is guaranteed to obtain a 2-approximation outlier ranking w.r.t. a given outlier ranking evaluation measure.

Extensive experiments show that HOUR (i) significantly outperforms three state-of-the-art outlier detectors and their combination with two of the latest outlying feature selection methods in terms of AUC and/or $P@n$ on 15 real-world data sets with a diverse range of noise levels; (ii) performs stably w.r.t. k in most cases; and (iii) obtains good scalability: it is linear to data size and quadratic to the number of features.

2 Related Work

Subspace outlier detection is a popular direction recently proposed to handle data with many noisy/irrelevant features [Zimek *et al.*, 2012]. Traditional methods (e.g., distance- and density-based methods [Chandola *et al.*, 2009]) identify outliers in *original* feature space and fail to work well in those data due to the meaningless distance with the presence of noisy/irrelevant features [Zimek *et al.*, 2012]. In contrast, subspace-based methods [He *et al.*, 2005; Lazarevic and Kumar, 2005; Angiulli *et al.*, 2009; Keller *et al.*, 2012; Akoglu *et al.*, 2012; Pang *et al.*, 2016c] compute outlier scores in subspaces. Most of these methods use heuristic search to identify outlying subspaces/patterns. This kind of search ignores subsequent outlier scoring functions. Consequently, noisy features may mislead the search, resulting in many faulty subspaces/patterns. The other methods work on a set of randomly generated subspaces and thus do not involve subspace search, but they include many noisy features into subspaces during the random generation of subspaces.

Feature selection has shown effective in removing noisy/irrelevant features for classification and clustering [Li *et al.*, 2016], but limited work has been done on outlier detection. Some work has been conducted on semi-supervised/supervised outlying feature selection [Azmandian *et al.*, 2012; Jeong *et al.*, 2012; Lorena *et al.*, 2015]. In contrast, very limited unsupervised methods are available in the literature due to the challenges brought by extreme class imbalance and the unavailability of class labels. The earliest related work is the PALM method for unsupervised minority class analysis [He and Carbonell, 2010], which assumes the minority class objects are strongly self-similar. However, this assumption is opposite to that of outlier detection, in which many outliers are isolated objects and far away from each other. CBRW_FS (denoted as CBFS) [Pang *et al.*, 2016a] computes the weights of features for weighted outlier scoring, which can also determine outlying feature selection. As far as we know, DSFS [Pang *et al.*, 2016b] is the first work specifically designed for unsupervised outlying feature selection. These two methods are filter-based approaches that evaluate feature subsets independently from subsequent outlier scoring functions and may retain many noisy features.

CBRW [Pang *et al.*, 2016a] also models similar homophily couplings to estimate value outlierness for handling noisy features, but HOUR is fundamentally different from CBRW in two aspects below: (i) HOUR is a joint optimization of outlier scoring and feature selection while CBRW involves no optimization and works on full feature space; and (ii) CBRW requires iteration algorithms to compute value and object outlierness, whereas the outlier scoring function in HOUR has a closed-form solution and runs substantially faster.

Great effort has been made on non-IID learning in recent years [Cao *et al.*, 2012; Cao, 2014; 2015; Cinbis *et al.*, 2016], while limited work [Pang *et al.*, 2016a; Chen *et al.*, 2016] has been reported on non-IID outlier detection. This work exploits the non-IID outlying behaviors to enable noise-resilient outlier detection.

3 HOUR for Joint Outlier Detection and Outlying Feature Selection

Given a set of data objects $\mathcal{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N\}$ with size N, described by D features $\mathcal{F} = \{f_1, f_2, \cdots, f_D\}$, WrapperOD first defines an outlier scoring function $\phi_{\mathcal{S}}$ to compute object outlierness in a given feature subset $\mathcal{S} \subseteq \mathcal{F}$ and then sorts the objects based on their outlierness to obtain an outlier ranking $R_{\phi_{\mathcal{S}}}$. WrapperOD further defines an outlier ranking evaluation function J to compute the quality of $R_{\phi_{\mathcal{S}}}$ and uses this ranking quality as the relevance indicator of the subset $\mathcal{S}$. This means that the task of finding the best feature subset is equivalent to finding the best outlier ranking. WrapperOD

iteratively performs function $\phi_{\mathcal{S}}$ and function J to obtain the best feature subset $\mathcal{S}^*$ and outlier ranking $R^*_{\phi_{\mathcal{S}^*}}$.

WrapperOD is fundamentally different from existing outlier detection and outlying feature selection frameworks in that: WrapperOD unifies the two correlated tasks, outlier detection and outlying feature selection, to simultaneously obtain the optimal outlier ranking and feature subset, while existing solutions treat these two tasks independently and are very sensitive to noisy features.

We further instantiate WrapperOD for categorical data by proposing HOUR. HOUR specifies its three components by a homophily coupling-based outlier scoring function $\phi_{\mathcal{S}}$, a score margin-based outlier ranking evaluation function J, and a heuristic feature subset search method:

$$\max_{\mathcal{S}} J(R_{\phi_{\mathcal{S}}}, k). \tag{1}$$

3.1 Specifying $\phi_{\mathcal{S}}$ with Homophily Couplings

The scoring function $\phi_{\mathcal{S}}$ has to meet at least the two requirements: (i) being sufficiently resilient to noisy features, and it may opt for noisy features other than relevant features otherwise; and (ii) being very efficient as it will be repeatedly performed to evaluate a large number of feature subsets. Unfortunately, most outlier detectors are sensitive to noisy features and/or are computationally costly [Pang *et al.*, 2016a].

HOUR exploits the homophily couplings between feature values to construct a fast and robust function $\phi_{\mathcal{S}}$. Let $dom(f) = \{v_1, v_2, \cdots\}$ be the domain of a feature $f \in \mathcal{S}$, which consists of a finite set of unordered feature values, and $\mathcal{V}$ be the whole set of feature values in $\mathcal{S}$: $\mathcal{V} = \cup_{f\in\mathcal{S}} dom(f)$, where $dom(f) \cap dom(f') = \emptyset, \forall f \neq f'$.

Definition 1 (Value Influence)**.** *The outlierness influence of a feature value $v \in \mathcal{V}$ is defined as follows.*

$$\tau(v) = \frac{\sum_{u\in\mathcal{N}_v} \delta(v)\delta(u)}{\sum_{v\in\mathcal{V}}\sum_{u\in\mathcal{N}_v} \delta(v)\delta(u)}, \tag{2}$$

where $\mathcal{N}_v$ denotes a set of values that co-occur with v and $\delta(\cdot) : \mathcal{V} \mapsto (0, 1)$ is an initial outlierness influence estimation of a value based on intra-feature frequency distribution.

We use $\delta(v) = \frac{1}{2}\left(\frac{freq(m)-freq(v)}{freq(m)} + \frac{1}{freq(m)}\right), \forall v \in dom(f)$, m is a value that occurs most frequently in f (i.e., the mode) and $freq(\cdot)$ is a frequency counting function. Such mode absolute deviation helps $\delta(\cdot)$ address features with imbalanced frequency distributions.

Essentially, δ estimates the value outlierness influence independently from the values of other features. τ further utilizes the homophily couplings between values from different features to have a better estimation of value influence. This value influence is then used to infer the value outlierness based on the coupling strength between feature values.

Definition 2 (Value Outlierness)**.** *The outlierness of a feature value $v \in \mathcal{V}$ is defined as follows.*

$$\psi(v) = \sum_{u\in\mathcal{N}_v} \rho(u, v)\tau(u), \tag{3}$$

where $\rho(u, v) = log\frac{p(u,v)}{p(u)p(v)}$ is pointwise mutual information to measure the coupling strength between two values.

We further define the outlierness of an object below.

Definition 3 (Object Outlierness)**.** *The outlierness of an object $\boldsymbol{x}$ is defined as a weighted product of value outlierness.*

$$\phi_{\mathcal{S}}(\boldsymbol{x}) = 1 - \prod_{f\in\mathcal{S}} [1 - \psi(x_f)]^{\omega(f)}, \tag{4}$$

where x_f is the value contained by $\boldsymbol{x}$ in feature f and $\omega(f) = 1 - \prod_{v\in dom(f)}[1 - \psi(v)]$ computes the weight of f.

Section 4.1 will discuss how this outlier scoring models the homophily couplings and why it is fast and noise-resilient.

3.2 Specifying J with Average Score Margin

The function J requires an internal evaluation measure for a given outlier ranking, i.e., evaluating outlier rankings without class labels. Internal evaluation measure has been extensively studied for clustering tasks, while very little work has been on outlier detection [Marques *et al.*, 2015]. One related work is [Marques *et al.*, 2015], which uses pseudo binary classification to evaluate the ranking quality. However, this method has $O(N^3)$ time complexity, which is computationally prohibitive to be used here. Below we introduce a linear-time outlier ranking evaluation measure based on the distribution of object outlierness:

$$J(R_{\phi_{\mathcal{S}}}, k) = \frac{\Delta_{\mathcal{S}}}{|\mathcal{S}|} = \frac{1}{k|\mathcal{S}|}\sum_{\boldsymbol{x}\in\mathcal{O}} [\phi_{\mathcal{S}}(\boldsymbol{x}) - \phi_{\mathcal{S}}(\boldsymbol{x}')], \tag{5}$$

where $\mathcal{O}$ is a set of top-ranked k objects and $\phi_{\mathcal{S}}(\boldsymbol{x}')$ is the median outlierness in the remaining objects. $\Delta_{\mathcal{S}} = \frac{1}{k}\sum_{\boldsymbol{x}\in\mathcal{O}}[\phi_{\mathcal{S}}(\boldsymbol{x}) - \phi_{\mathcal{S}}(\boldsymbol{x}')]$ is the average score margin between the top-k objects and the center of the other objects, which also indicates the relevance of feature subset $\mathcal{S}$. So maximizing J finds an outlier ranking that jointly maximizes the object outlierness margin and the feature subset relevance.

3.3 Recursive Search of Feature Subset $\mathcal{S}$

Feature subset search methods includes complete search, sequential search, and random search [Li *et al.*, 2016]. Although complete search outputs an optimal subset, it has exponential time complexity. Sequential search and random search may produce a suboptimal subset, but they are more practical than complete search as they run substantially faster.

A sequential search method, namely Recursive Backward Elimination (RBE), is used with the functions $\phi_{\mathcal{S}}$ and J to search for an approximately best subset. RBE recursively eliminates one feature at a time until no feature remains, and only retains the feature subset that results in the largest J. RBE is used because it can guarantee a 2-approximate J to the optimal one (see Section 4.2).

3.4 The Algorithm and Its Time Complexity

Algorithm 1 presents the procedure of HOUR. Steps (1-3) evaluates the outlier ranking in the full feature set, followed by the evaluation of outlier rankings in feature subsets generated by RBE in Steps (4-14).

Steps (1-2) require one database scan to perform ψ and $\phi_{\mathcal{S}}$ respectively, which is linear w.r.t. N. Step (3) needs to

Algorithm 1 *HOUR($\mathcal{X}$, k)*

Input: $\mathcal{X}$ - data objects, k - the number of targeted outliers
Output: R - an outlier ranking of objects, $\mathcal{S}$ - a feature subset

1: $\psi(v) \leftarrow \sum_{u \in \mathcal{N}_v} \rho(u,v)\tau(u), \forall v \in \mathcal{V}$
2: Compute $\phi_{\mathcal{F}}(\boldsymbol{x}), \forall \boldsymbol{x} \in \mathcal{X}$
3: $r \leftarrow J(R_{\phi_{\mathcal{F}}}, k)$
4: **while** $|\mathcal{F}| > 0$ **do**
5: **for** $i = 1$ to $|\mathcal{F}|$ **do**
6: Compute $\phi_{\mathcal{F} \setminus f_i}(\boldsymbol{x}), \forall \boldsymbol{x} \in \mathcal{X}$
7: Compute $J_i(R'_{\phi_{\mathcal{F}}}, k)$
8: **end for**
9: Find feature f_i with the largest $J_i(R'_{\phi_{\mathcal{F}}}, k)$
10: $\mathcal{F} \leftarrow \mathcal{F} \setminus f_i$ and update $\psi(v)$ for all v contained in $\mathcal{F}$
11: **if** $J_i(R'_{\phi_{\mathcal{F}}}, k) \geq r$ **then**
12: $R \leftarrow R'$, $\mathcal{S} \leftarrow \mathcal{F}$ and $r \leftarrow J_i(R'_{\phi_{\mathcal{F}}}, k)$
13: **end if**
14: **end while**
15: **return** R and $\mathcal{S}$

rank $\mathcal{X}$, which has $O(NlogN)$ in the worst case, and thus they have $O(NlogN)$. The two loops in Steps (4-14) result in $O(D^2)$ in the worst case, and the core computation within the loops performs outlier scoring and ranking, which has the same time complexity as the first three steps. Hence, the worst time complexity of HOUR is $O(D^2NlogN)$.

4 Theoretical Analysis

4.1 Robustness w.r.t. Noisy Features

We analyze the robustness of HOUR from the value level to the feature level. At the value level, per definition of outliers, *outlying values* are infrequent values contained by outliers, while *noisy values* are also infrequent but contained by normal objects. In contrast, *normal values* are frequent values contained by both outliers and normal objects. Below we discuss how the outlier scoring function in HOUR can efficiently distinguish outlying values from normal and noisy values.

Theorem 1 (Homophily Coupling Modeling)**.** *The value influence estimation $\tau(v)$ in Eqn.(1) is equivalent to the stationary probability of visiting v in random walks on a strongly connected undirected value-value graph $G =< \mathcal{V}, \mathcal{E}, \eta(\cdot,\cdot) >$, where a feature value v represents a graph node, $e(u,v) \in \mathcal{E}$ denotes an edge between two nodes u and v, and $\eta(u,v) = \delta(u)I(u,v)\delta(v)$ ($I(u,v) = 1$ if u and v have occurrences, and $I(u,v) = 0$ otherwise) is the weight of edge $e(u,v)$, $\forall u, v \in \mathcal{V}$.*

Proof. Let $\pi^*(v)$ be the stationary probability, $P(u,v)$ be the transition probability from u to v, $d(v) = \sum_{u \in \mathcal{N}_v} \eta(v,u)$ be the weighted degree of v and $vol(G) = \sum_{v \in \mathcal{V}} d(v) = \sum_{v \in \mathcal{V}} \sum_{u \in \mathcal{N}_v} \delta(v)\delta(u)$ be the graph volume. Then we have:

$$\pi^*(v) = \sum_{u \in \mathcal{V}} \pi^*(u)P(u,v) = \sum_{u \in \mathcal{V}} \frac{d(u)}{vol(G)} \frac{\delta(u)I(u,v)\delta(v)}{d(u)},$$

and we obtain:

$$\pi^*(v) = \frac{\sum_{u \in \mathcal{N}_v} \delta(v)\delta(u)}{\sum_{v \in \mathcal{V}} \sum_{u \in \mathcal{N}_v} \delta(v)\delta(u)} = \tau(v),$$

which completes the proof. □

Theorem 1 indicates that given $\forall u, v \in \mathcal{V}$, if value u has lower frequency and stronger couplings with other infrequent values compared to value v, i.e., $d(u) > d(v)$, then $\tau(u) > \tau(v)$. This essentially models the homophily couplings between outlying values. However, this homophily coupling modeling does not take account of the coupling strength between values. We further enhance the modeling by adding pointwise mutual information in Eqn. (3). There exist other ways to model homophily couplings. We use such a two-stage modeling because it has a closed-form solution which guarantees the efficiency of outlier scoring.

Outlying Values vs. Noisy Values. Noisy values have similarly low frequencies as outlying values, but they are supposed to co-occur randomly or follow a Gaussian distribution. Their homophily couplings are therefore weaker than that of outlying values. As a result, HOUR assigns smaller outlierness ψ to noisy values than outlying values.

Outlying Values vs. Normal Values. Normal values have much lower δ and τ than outlying values due to their high occurrence frequencies. Their high frequencies also result in weak couplings with infrequent values. As a result, they obtain substantially smaller outlierness ψ than outlying values.

At the feature level, HOUR prefers features that contain values of higher outlierness to maximize its objective function. Since outlying values have higher outlierness than normal or noisy values, the features HOUR iteratively eliminates are those containing normal and/or noisy values, resulting in a cleaned feature subset for its outlier scoring function.

4.2 Theoretical Bound

This section shows that HOUR is guaranteed to obtain an outlier ranking with the margin of at least half of the optimum value, provided that features are dependent on each other as in the homophily coupling modeling.

Theorem 2 (2-Approximation)**.** *Let R and $\mathcal{S}$ be the outlier ranking and feature subset returned by HOUR. Assume Θ_f be the contribution of feature $f \in \mathcal{S}$ to the outlier ranking R by integrating its conjunctive functions with other features $\theta(f \wedge f')$, i.e., $\Theta_f = \sum_{f' \in \mathcal{S}} \theta(f \wedge f')$, and $\Delta_{\mathcal{S}} = \frac{1}{2}\sum_{f \in \mathcal{S}} \Theta_f$. Then we have $J(R_{\phi_{\mathcal{S}}}, k) \geq \frac{1}{2} J_{opt}$, where J_{opt} is the optimum value of J.*

Proof. Since J_{opt} is the optimum value of J, we have

$$J_{opt} = \frac{\Delta_{\mathcal{S}^*}}{|\mathcal{S}^*|} \geq \frac{\Delta_{\mathcal{S}^*} - \Theta_f}{|\mathcal{S}^*| - 1}, \ \forall f \in \mathcal{S}^*.$$

We obtain $\Theta_f \geq J_{opt}$ after some replacements. Let $f \in \mathcal{S}^*$ be the feature that HOUR removes first among those contained in $\mathcal{S}^*$ during the iteration of RBE and $\mathcal{T}$ be the feature set before f is removed, i.e., $\mathcal{S}^* \subset \mathcal{T}$. Since HOUR removes the least contributive feature at a time, we have $\Theta_{f'} \geq \Theta_f \ , \forall f' \in \mathcal{T}$ when HOUR chooses to remove f, and thus $\Theta_{f'} \geq J_{opt}$. As a result, we obtain $\sum_{f' \in \mathcal{T}} \Theta_{f'} \geq J_{opt}|\mathcal{T}|$, and thus $2\Delta_{\mathcal{T}} \geq J_{opt}|\mathcal{T}|$, resulting in $J(R'_{\phi_{\mathcal{T}}}, k) = \frac{\Delta_{\mathcal{T}}}{|\mathcal{T}|} \geq \frac{J_{opt}}{2}$. Since HOUR retains $\mathcal{S}$ that results in the largest J and $\mathcal{T}$ is one of the candidates, we finally obtain $J(R_{\phi_{\mathcal{S}}}, k) \geq J(R'_{\phi_{\mathcal{T}}}, k) \geq \frac{J_{opt}}{2}$. □

5 Experiments and Evaluation

5.1 Competitors and Parameter Settings

HOUR is evaluated against three representative outlier detectors for categorical data: FPOF [He *et al.*, 2005], COMP [Akoglu *et al.*, 2012] and CBRW [Pang *et al.*, 2016a]. FPOF is chosen because it is the most popular pattern-based method. COMP is a state-of-the-art subspace method that captures arbitrary-length outlying behaviors. CBRW is a closely related value outlierness-based method. k in HOUR is set to the number of outliers by default. COMP is parameter-free. FPOF and CBRW are used with their default settings.

We also compare HOUR to the combination of outlier detectors with two of the most recently proposed outlying feature selection methods, CBFS [Pang *et al.*, 2016a] and DSFS [Pang *et al.*, 2016b]. CBFS returns a feature ranking. DSFS outputs a feature subset without any parameters. To have a fair comparison, CBFS selects the top-ranked $|\mathcal{S}|$ features so that CBFS and HOUR select the same number of features.

All methods are in Java in WEKA [Hall *et al.*, 2009] except COMP which is in MATLAB. All these methods are executed on a node in a 3.4GHz Phoenix Cluster with 32GB memory.

5.2 Performance Evaluation Method

Two of the most popular evaluation methods, the area under ROC curve (AUC) and precision at n, i.e., $P@n$ (where we set n as the number of outliers in a data set), are used. All the outlier detectors produce an ascending ranking based on outlier scores. AUC evaluates the global ranking quality, while $P@n$ considers the detection precision in the top n positions. Higher AUC or $P@n$ indicates better performance. The *Wilcoxon* signed-rank test is performed to check the significance of performance of HOUR against its competitors.

Two data indicators, *feature noise level* (*fnl*) and *outlier separability* (*sep*), are defined to evaluate data complexity before and after applying feature selection. *fnl* is defined by [Pang *et al.*, 2016a; 2016b] as the percentage of individual noisy features. Inspired by the indicator *feature efficiency* in [Ho and Basu, 2002; Leyva *et al.*, 2015], *sep* is defined as the maximum feature efficiency per data, in which the efficiency of a feature is quantified by the AUC performance of using frequency histogram to detect outliers on the single feature.

5.3 Data Sets

Fifteen publicly available real-world data sets with a range of feature noise levels are used, which cover diverse domains, e.g., image object recognition, intrusion detection and molecular screening, as shown in Table 1. These data sets are transformed from extremely imbalanced data, where the rare classes are treated as outliers versus the rest of classes as normal class [Lazarevic and Kumar, 2005; Pang *et al.*, 2016a].

5.4 Findings and Analysis

1) Obtaining Significantly Better Global or Top-n Outlier Ranking Than Other Outlier Detectors: We compare HOUR with CBRW, COMP and FPOF in terms of AUC and $P@n$ in Table 1. In terms of AUC, HOUR obtains the best performance on 11 data sets; and on average, it obtains about 2%, 7% and 21% improvement over CBRW, COMP and FPOF, respectively. HOUR significantly outperforms FPOF in AUC. In terms of $P@n$, HOUR performs significantly better than CBRW and COMP and obtains more than 30%, 37% and 90% improvements over CBRW, COMP and FPOF, respectively.

Table 1: AUC and $P@n$ Performance on 15 Data Sets. Data is sorted by *fnl*. '∇' indicates feature reduction rate of HOUR. FPOF runs out of memory in four high-dimensional data.

						AUC				$P@n$			
Data	N	$\|\mathcal{F}\|$	$\|\mathcal{S}\|(\nabla)$		*fnl*	HOUR	CBRW	COMP	FPOF	HOUR	CBRW	COMP	FPOF
SylvaA	14,395	172	16(91%)		91%	**0.9829**	0.9353	0.8855	NA	**0.7483**	0.5914	0.3770	NA
BM	41,188	10	5(50%)		90%	**0.6939**	0.6287	0.6267	0.5466	**0.3265**	0.2474	0.2565	0.1369
AID362	4,279	114	8(93%)		86%	0.5147	**0.6640**	0.6480	NA	**0.0833**	0.0500	0.0167	NA
APAS	12,695	64	13(80%)		81%	**0.9065**	0.8190	0.6554	NA	0.0000	0.0000	0.0000	NA
SylvaP	14,395	87	15(83%)		78%	**0.9725**	0.9715	0.9537	NA	**0.6907**	0.6151	0.5700	NA
Census	299,285	33	3(91%)		58%	0.4867	**0.6678**	0.6352	0.6148	0.0616	**0.0677**	0.0675	0.0637
CelebA	202,599	39	12(69%)		49%	**0.8879**	0.8462	0.7572	0.7380	**0.2085**	0.1748	0.1533	0.1256
CUP14	619,326	7	3(57%)		43%	**0.9833**	0.9420	0.9398	0.6041	**0.6730**	0.2671	0.2671	0.0000
Alcohol	1,044	32	3(91%)		38%	**0.9365**	0.9254	0.8919	0.5468	**0.3889**	0.3333	**0.3889**	0.0556
CMC	1,473	8	4(50%)		38%	**0.6647**	0.6339	0.5669	0.5614	0.0345	0.0345	0.0345	**0.1034**
CT	581,012	44	3(93%)		34%	0.9688	0.9703	**0.9772**	0.9770	0.0499	0.0386	**0.0688**	0.0644
Chess	28,056	6	3(50%)		33%	**0.8507**	0.7897	0.6387	0.6160	0.0000	0.0000	0.0000	0.0000
Turkiye	5,820	32	21(34%)		25%	**0.5256**	0.5116	0.5101	0.4746	**0.0776**	0.0746	0.0687	0.0597
Credit	30,000	9	6(33%)		11%	**0.7204**	0.5804	0.6543	0.6428	**0.4875**	0.2215	0.3502	0.3333
Probe	64,759	6	2(67%)		0%	0.9661	**0.9906**	0.9790	0.9867	0.8440	**0.8579**	0.7928	0.8548
Average	128,022	44	8(69%)		50%	0.8041	0.7918	0.7546	0.6644	0.3116	0.2383	0.2275	0.1634
				p-value			0.1876	0.0730	0.0322		0.0068	0.0068	0.1055

Using outlier scoring results to guide outlying feature selection enables HOUR to remove most, if not all, of the noisy features while having little or no loss in outlier separability on most data sets, e.g., the 11 data sets on which HOUR obtains the best AUC performance (see the *fnl* and *sep* results of HOUR in Table 3). Hence, although HOUR works with 69% less features than its competitors, it performs substantially better as it works on much cleaner data. Also, maximizing the margin of the top-k objects from the others helps rank more outliers in the top, resulting in significant improvement in $P@n$. On the other hand, HOUR opts for strongly relevant features that help rank outliers in the top, so it may remove weakly relevant features that distinguish outliers from normal objects in other positions. As a result, HOUR may obtain worse AUC performance while comparable $P@n$ compared to its competitors, e.g., the results on *AID362* and *Census*.

2) Defeating the Combination of Outlier Detectors with Outlying Feature Selection Methods: HOUR is compared with the combination of CBRW and COMP with outlying feature selection methods CBFS and DSFS in Table 2 [1]. The results show that, although the two feature selection methods largely improve CBRW and COMP in terms of AUC and/or $P@n$, HOUR remains as the best performer on most data sets. HOUR obtains significantly better performance than the combination of CBRW and COMP with CBFS (i.e., CBRW† and COMP† in Table 2) in AUC and significantly outperforms all the four different combinations in $P@n$.

The superiority of HOUR is because the wrapper-based feature selection scheme enables HOUR to remove substantially more truly noisy features than the filter-based methods CBFS and DSFS. This is verified by the *fnl* and *sep* differences between the full feature set and feature subsets selected by HOUR, CBFS and DSFS shown in Table 3. On average,

[1] The combination of FPOF with CBFS or DSFS underperforms that of CBRW and COMP and is omitted due to space limits.

Table 2: AUC and $P@n$ Performance Comparison between HOUR and the Combination of CBRW and COMP with CBFS (Denoted by †) and DSFS (Denoted by ‡).

	AUC					$P@n$				
Data	HOUR	CBRW†	CBRW‡	COMP†	COMP‡	HOUR	CBRW†	CBRW‡	COMP†	COMP‡
SylvaA	**0.9829**	0.8793	0.9381	0.8726	0.8858	**0.7483**	0.5327	0.5948	0.4831	0.3781
BM	**0.6939**	0.6104	0.6114	0.6239	0.6239	**0.3265**	0.2259	0.2269	0.2567	0.2575
AID362	0.5147	0.4659	**0.6518**	0.4982	0.6342	**0.0833**	0.0000	0.0500	0.0000	0.0167
APAS	**0.9065**	0.6621	0.8807	0.6532	0.8771	0.0000	0.0000	0.0000	0.0000	0.0000
SylvaP	**0.9725**	0.9582	0.9707	0.9307	0.9628	**0.6907**	0.5553	0.5609	0.6140	0.5892
Census	0.4867	0.4844	0.6999	0.4841	**0.7135**	0.0616	0.0604	0.0732	0.0635	**0.0991**
CelebA	**0.8879**	0.8865	0.8502	0.8855	0.7594	0.2085	0.2098	0.1698	**0.2142**	0.1482
CUP14	**0.9833**	0.9821	0.9358	0.9821	0.9618	**0.6730**	0.6686	0.2671	0.6686	0.3224
Alcohol	**0.9365**	0.9264	0.9294	0.8919	0.8595	0.3889	0.3889	**0.4444**	0.3889	0.0556
CMC	**0.6647**	0.6366	0.6444	0.6475	0.6586	0.0345	0.0345	0.0345	0.0345	0.0345
CT	**0.9688**	0.9192	0.9673	0.9187	0.9670	**0.0499**	0.0000	0.0386	0.0000	0.0386
Chess	**0.8507**	0.7268	0.7649	0.7529	0.6305	0.0000	0.0000	0.0000	0.0000	0.0000
Turkiye	**0.5256**	0.5161	0.5108	0.5145	0.5119	**0.0776**	0.0716	0.0716	0.0746	**0.0776**
Credit	**0.7204**	0.5712	0.5712	0.6566	0.6566	**0.4875**	0.2131	0.2131	0.3531	0.3531
Probe	0.9661	0.9591	0.9591	**0.9794**	**0.9794**	**0.8440**	0.8397	0.8397	0.7672	0.7672
Average	0.8041	0.7456	0.7924	0.7528	0.7788	0.3116	0.2533	0.2390	0.2612	0.2092
p-value		0.0001	0.0730	0.0006	0.1070		0.0029	0.0269	0.0098	0.0029

HOUR removes over 57% of the noisy features, which is about triple and double more than that of CBFS (17%) and DSFS (32%), respectively; while at the same time, it obtains a very comparable outlier separability. In addition, we observe that filter-based methods like DSFS generally retain many more features than HOUR. These extra features contain noisy features as well as relevant features. This is why DSFS obtains a smaller noise reduction level but a better outlier separability than HOUR in Table 3. The extra relevant features retained by DSFS enable CBRW and COMP to outperform HOUR in data sets where HOUR makes very aggressive feature reduction, e.g., on *AID362* and *Census*.

Table 3: Data Complexity Evaluation Results on $\mathcal{F}$, $\mathcal{S}$, $\mathcal{S}'$ and $\mathcal{S}''$. $\mathcal{F}$ is the original feature set. $\mathcal{S}$, $\mathcal{S}'$ and $\mathcal{S}''$ are feature subsets retained by HOUR, CBFS and DSFS, respectively.

	Feature Noise Level (*fnl*)				Outlier Separability (*sep*)			
Data	$\mathcal{F}$	$\mathcal{S}$ (∇)	$\mathcal{S}'$ (∇)	$\mathcal{S}''$ (∇)	$\mathcal{F}$	$\mathcal{S}$ (∇)	$\mathcal{S}'$ (∇)	$\mathcal{S}''$ (∇)
SylvaA	91%	13%(86%)	75%(18%)	91%(0%)	0.78	0.78(0%)	0.78(0%)	0.78(0%)
BM	90%	80%(11%)	80%(11%)	75%(17%)	0.63	0.63(0%)	0.63(0%)	0.63(0%)
AID362	86%	100%(-16%)	100%(-16%)	85%(1%)	0.60	0.49(19%)	0.47(23%)	0.60(0%)
APAS	81%	38%(53%)	85%(-4%)	50%(38%)	0.87	0.87(0%)	0.72(18%)	0.87(0%)
SylvaP	78%	0%(100%)	53%(32%)	71%(9%)	0.78	0.78(0%)	0.78(0%)	0.78(0%)
Census	58%	100%(-74%)	100%(-74%)	50%(13%)	0.76	0.49(35%)	0.49(35%)	0.76(0%)
CelebA	49%	0%(100%)	0%(100%)	50%(-3%)	0.80	0.78(2%)	0.78(2%)	0.80(0%)
CUP14	43%	0%(100%)	33%(22%)	50%(-17%)	0.92	0.92(0%)	0.92(0%)	0.92(0%)
Alcohol	38%	0%(100%)	0%(100%)	18%(53%)	0.91	0.91(0%)	0.91(0%)	0.91(0%)
CMC	38%	0%(100%)	0%(100%)	0%(100%)	0.66	0.66(0%)	0.66(0%)	0.66(0%)
CT	34%	0%(100%)	67%(-96%)	0%(100%)	0.97	0.97(0%)	0.97(0%)	0.97(0%)
Chess	33%	33%(0%)	6%(-100%)	25%(25%)	0.74	0.59(19%)	0.74(0%)	0.74(0%)
Turkiye	25%	14%(43%)	14%(43%)	21%(4%)	0.58	0.55(4%)	0.55(4%)	0.55(4%)
Credit	11%	0%(100%)	0%(100%)	0%(100%)	0.70	0.70(0%)	0.70(0%)	0.70(0%)
Probe	0%	0%(NA)	0%(NA)	0%(NA)	0.94	0.94(0%)	0.94(0%)	0.94(0%)
Average	50%	25%(57%)	44%(17%)	39%(32%)	0.78	0.74(5%)	0.74(5%)	0.77(0%)

3) Good Stability w.r.t. the Parameter k: We examine the stability of HOUR w.r.t. k in Figure 2. HOUR shows stable performance in most of the 15 data sets. Here we selectively illustrate representative and interesting trends in its AUC performance w.r.t. a wide range of k on four data sets due to space limits. HOUR performs very stably on *CelebA* and *CUP14*. It is very challenging to rank outliers in the top-k positions in data sets which contain only a very small proportion of outliers but have many noisy features (e.g., *CT*), as the outliers are easily masked as normal objects in those data. Due to these false negatives, HOUR requires a large k (e.g., 0.5% or 1.0%) to perform well on *CT*. On the other hand, HOUR can identify outliers more accurately using a smaller k in *Census* which contains a larger proportion of outliers, as the use of a large k in HOUR might lead to false positives. A general guideline is to set $k = 0.5\% \times N$ or $k = 1.0\% \times N$ to leverage the effect of false negatives and false positives.

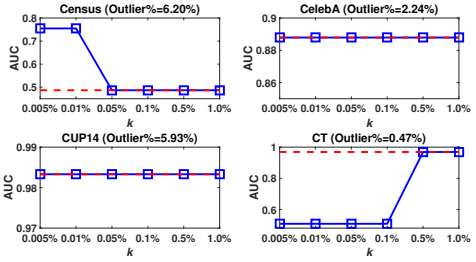

Figure 2: Representative AUC Performance of HOUR w.r.t. k. HOUR performs stably in most of the other data sets. The dashed line shows HOUR's performance with $k = outlier\%$.

4) Good Scalability w.r.t. Data Size and Dimensionality: The scale-up test results are presented in Figure 3. As expected, HOUR is linear w.r.t. data size and quadratic w.r.t. dimensionality. HOUR runs comparably fast to CBRW and FPOF w.r.t. different data sizes. In the right panel, HOUR runs over five orders of magnitude faster than FPOF, while the iterative optimization process makes HOUR run considerably slower than CBRW.

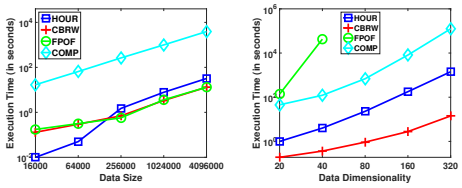

Figure 3: Scale-up Test w.r.t. Data Size and Dimensionality. FPOF runs out of memory when dimensionality reaches 80.

6 Conclusions and Future work

A wrapper-based outlier detection framework WrapperOD and its instance HOUR are introduced to joint top-k outlier detection with feature selection for handling data with noisy features. HOUR is more plausible than its competitors: (i) it guarantees the margin between the top-k outliers and the rest of the objects is a 2-approximation to the optimum value; (ii) it performs significantly better in global and/or local outlier ranking; and (iii) it obtains stable performance w.r.t. k and good scalability. The capability of returning the top k outliers with superior $P@n$ performance makes HOUR a good candidate for real-world applications, since investigation resources are often only sufficient for limited suspicious objects.

Acknowledgments

We would like to thank anonymous reviewers for their constructive comments. This work is partially supported by the ARC Discovery Grants DP130102691 and DP140100545.

References

[Akoglu *et al.*, 2012] Leman Akoglu, Hanghang Tong, Jilles Vreeken, and Christos Faloutsos. Fast and reliable anomaly detection in categorical data. In *CIKM*, pages 415–424. ACM, 2012.

[Angiulli *et al.*, 2009] Fabrizio Angiulli, Fabio Fassetti, and Luigi Palopoli. Detecting outlying properties of exceptional objects. *ACM Transactions on Database Systems*, 34(1):7, 2009.

[Azmandian *et al.*, 2012] Fatemeh Azmandian, Ayse Yilmazer, Jennifer G Dy, Javed Aslam, David R Kaeli, et al. GPU-accelerated feature selection for outlier detection using the local kernel density ratio. In *ICDM*, pages 51–60. IEEE, 2012.

[Cao *et al.*, 2012] Longbing Cao, Yuming Ou, and Philip S Yu. Coupled behavior analysis with applications. *IEEE Transactions on Knowledge and Data Engineering*, 24(8):1378–1392, 2012.

[Cao, 2014] Longbing Cao. Non-iidness learning in behavioral and social data. *The Computer Journal*, 57(9):1358–1370, 2014.

[Cao, 2015] Longbing Cao. Coupling learning of complex interactions. *Information Processing & Management*, 51(2):167–186, 2015.

[Chandola *et al.*, 2009] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41(3):15, 2009.

[Chau *et al.*, 2011] Duen Horng Polo Chau, Carey Nachenberg, Jeffrey Wilhelm, Adam Wright, and Christos Faloutsos. Polonium: Tera-scale graph mining and inference for malware detection. In *SDM*, pages 131–142. SIAM, 2011.

[Chen *et al.*, 2016] Ting Chen, Lu-An Tang, Yizhou Sun, Zhengzhang Chen, and Kai Zhang. Entity embedding-based anomaly detection for heterogeneous categorical events. In *IJCAI*, pages 1396–1403. AAAI, 2016.

[Cinbis *et al.*, 2016] Ramazan Gokberk Cinbis, Jakob Verbeek, and Cordelia Schmid. Approximate fisher kernels of non-iid image models for image categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(6):1084–1098, 2016.

[Hall *et al.*, 2009] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The WEKA data mining software: An update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18, 2009.

[He and Carbonell, 2010] Jingrui He and Jaime Carbonell. Coselection of features and instances for unsupervised rare category analysis. *Statistical Analysis and Data Mining*, 3(6):417–430, 2010.

[He *et al.*, 2005] Zengyou He, Xiaofei Xu, Zhexue Joshua Huang, and Shengchun Deng. FP-outlier: Frequent pattern based outlier detection. *Computer Science and Information Systems*, 2(1):103–118, 2005.

[Ho and Basu, 2002] Tin Kam Ho and Mitra Basu. Complexity measures of supervised classification problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):289–300, 2002.

[Jeong *et al.*, 2012] Young-Seon Jeong, In-Ho Kang, Myong-Kee Jeong, and Dongjoon Kong. A new feature selection method for one-class classification problems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6):1500–1509, 2012.

[Keller *et al.*, 2012] Fabian Keller, Emmanuel Muller, and Klemens Bohm. HiCS: High contrast subspaces for density-based outlier ranking. In *ICDE*, pages 1037–1048. IEEE, 2012.

[Kohavi and John, 1997] Ron Kohavi and George H John. Wrappers for feature subset selection. *Artificial intelligence*, 97(1):273–324, 1997.

[Lazarevic and Kumar, 2005] Aleksandar Lazarevic and Vipin Kumar. Feature bagging for outlier detection. In *SIGKDD*, pages 157–166. ACM, 2005.

[Leyva *et al.*, 2015] Enrique Leyva, Antonio González, and Raul Perez. A set of complexity measures designed for applying meta-learning to instance selection. *IEEE Transactions on Knowledge and Data Engineering*, 27(2):354–367, 2015.

[Li *et al.*, 2016] Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P Trevino, Jiliang Tang, and Huan Liu. Feature selection: A data perspective. *CoRR*, abs/1601.07996, 2016.

[Lorena *et al.*, 2015] Luiz HN Lorena, André CPLF Carvalho, and Ana C Lorena. Filter feature selection for one-class classification. *Journal of Intelligent & Robotic Systems*, 80(1):227–243, 2015.

[Marques *et al.*, 2015] Henrique O Marques, Ricardo JGB Campello, Arthur Zimek, and Jörg Sander. On the internal evaluation of unsupervised outlier detection. In *SSDBM*. ACM, 2015.

[Pang *et al.*, 2016a] Guansong Pang, Longbing Cao, and Ling Chen. Outlier detection in complex categorical data by modelling the feature value couplings. In *IJCAI*, pages 1902–1908. AAAI, 2016.

[Pang *et al.*, 2016b] Guansong Pang, Longbing Cao, Ling Chen, and Huan Liu. Unsupervised feature selection for outlier detection by modelling hierarchical value-feature couplings. In *ICDM*. IEEE, 2016.

[Pang *et al.*, 2016c] Guansong Pang, Kai Ming Ting, David Albrecht, and Huidong Jin. Zero++: Harnessing the power of zero appearances to detect anomalies in large-scale data sets. *Journal of Artificial Intelligence Research*, 57:593–620, 2016.

[Zimek *et al.*, 2012] Arthur Zimek, Erich Schubert, and Hans-Peter Kriegel. A survey on unsupervised outlier detection in high-dimensional numerical data. *Statistical Analysis and Data Mining*, 5(5):363–387, 2012.