2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Resistant Learning on the Envelope Bulk for Identifying Anomalous Patterns

Shin-Ying Huang, Fang Yu, Rua-Huan Tsaih, Yennun Huang

***Abstract*—Anomalous patterns are observations that lie far away from the fitting function deduced from the bulk of the given observations. This work addresses the research issue to effectively identify anomalous patterns in both contexts of resistant learning, where there is no assumption about the fitting function form, and of changing environments. The resistant learning means that the learning procedure is not impacted significantly by the outlying observations. In literature, there is the resistant learning with searching a near-perfect fitting function for identifying the bulk of the majority of observations. However, the learning algorithm with searching a near-perfect fitting function suffers from time inefficiency. To effectively identify anomalous patterns in both contexts of resistant learning and changing environments, this study proposes a new resistant learning algorithm with envelope module that learns to evolve a nonlinear fitting function wrapped with a constant-width envelope for containing the majority of observations and thus identifying anomalous patterns. An illustrative experiment is set up to justify the effectiveness of the envelope module and the experimental result shows the positive promise.**

I. Introduction

In certain applications for which the input-output relationship is believed to be non-linear but is unknown, there can be occasional anomalous patterns that lie far from the bulk of the vast number of observations in input-output space. And the outlier detection is the key issue. Furthermore, the data nature is not only spatial dependent but also non-stationary and concept drifting. Thus the outlier detection usually relies on incremental learning techniques to constantly adjust the boundaries for identifying the majority that are evolved throughout the time and thus to recognize the anomalous ones. However, there are challenges to derive an algorithm for such detection of anomalous patterns.

Agyemang et al. [1] point out that outlier detection is a very complex task that is similar to finding a needle in a haystack. Chandola et al. [4] provide a comprehensive overview of the existing outlier detection techniques by classifying them along different dimensions. They mention that a key observation of outlier detection is that it is not a well-formulated problem. They conclude that every unique problem formulation entails a different approach, resulting in a huge literature on outlier detection techniques. Ngai et al. [16] state that, in the case of financial fraud detection, the detection of a fraud case could be regarded as recognizing outliers from the healthy majority, and data mining techniques for outlier detection have seen only limited use.

In practice, even given the independent variables, the function's form is usually unknown, and the conventional outlier detection studies do not appear to generalize to the resistant learning problems. Tsaih and Cheng [23] propose an outlier detection algorithm which can cope with the context of resistant learning; however, the algorithm is rather complicated and time-consuming when both sizes of the reference pool and the input dimensionality are large. Furthermore, the algorithm cannot directly cope with the outlier detection problem whose data nature is non-stationary and concept drifting.

T deal with the outlier detection problem in both contexts of resistant learning and changing environments, this study proposes a new resistant learning algorithm with the envelope module that is derived from changing the algorithm proposed by Tsaih and Cheng [23] by replacing a tiny pre-specified tolerance value ε of the envelope module with a non-tiny ε value that evolves a nonlinear fitting function f wrapped with an envelope whose width is 2ε.

This envelope module is distinct from the idea of Tsaih and Cheng, who want to evolve the fitting function to almost precisely fit all of the reference observations (because the corresponding ε value is tiny). Furthermore, the proposed envelope module can help to identify the anomalous patterns with less computational effort compared with the algorithm of Tsaih and Cheng. The envelope module helps to deal with the outlier detection problem in both contexts of resistant learning and changing environments.

The remainder of this paper is organized as follows: the literature reviews are introduced in Sections II and III, respectively. The proposed envelope module and its justification are introduced in Sections IV and V. An illustrative experiment is presented in Section VI. Finally, conclusions and future research directions are presented in Section VII.

* S. Y. Huang is with the Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan (email: smichelle19@citi.sinica.edu.tw).

F. Yu is with Department of Management Information Systems, National Chengchi University, Taipei, Taiwan (email: yuf@nccu.edu.tw)

R. H. Tsaih is with Department of Management Information Systems, National Chengchi University, Taipei, Taiwan (email: tsaih@mis.nccu.edu.tw)

Y. Huang is with the Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan (email: yennunhuang@citi.sinica.edu.tw).

We gratefully acknowledge the financial support from the National Science Council (Project No. NSC100-2410-H-004-013).

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

II. The Detection of Anomalous Patterns

In literature, there is an application in which DNA microarray technology is used to simultaneously probe thousands of gene expression profiles for disease classification, to identify outlier genes that are overexpressed in only a small number of disease samples. Tomlins et al. [22] propose cancer outlier profile analysis (COPA) to identify outlier genes. The COPA approach standardizes gene expression by centering at the median and scaling by the absolute deviation of the median. A kth percentile of the standardized expression value is then used as a cut-off point to identify the outlier gene. Tibshirani and Hastie [21] propose the outlier-sum statistic to improve on the COPA method. Compared to the traditional t-statistic, outlier-associated methods have the potential to detect a larger number of differentially expressed genes in heterogeneous data sets, at a lower false discovery rate [5]. However, these methods are less powerful than the approaches based on t-statistics when the differential expression is presented throughout the distribution or is concentrated at the center of the distribution as opposed to being concentrated in the tails [24].

Another issue of outlier detection is that fitting the observations with outliers could decrease the effectiveness of the fitting function because the outliers might have a large influence on model estimation, with their unusual high fitting deviances. Therefore, removing the outliers before the model-building process can help to fix this problem. For example, Connor and Martin [7] propose a robust learning algorithm that attempts to filter outliers from the training data first and then estimates the parameters from the filtered data. Windham [25] proposes a procedure to robustify any model fitting process by using weights from a parametric family from which the model is to be chosen, which is referred to as "robust model fitting." Methods for model selection and/or variable selection in the presence of outliers have been discussed in Hoeting et al. [8] and Atkinson and Riani [3]. Knorr and Ng [10][11] and Knorr et al. [12] focus on the development of algorithms for identifying the distance-based outliers in large data sets. Chuang et al. [6] propose a robust support vector regression method for the problem of function approximation with outliers. Sluban et al. [20] aim at detecting noisy instances for improved data understanding, data cleaning and outlier identification by presenting an ensemble-based noise ranking methodology for explicit noise and outlier identification. Specifically, they develop a methodology enabling the detection of noisy instances to be inspected by human experts in the phase of data cleaning, data understanding and outlier identification.

All of these methods are based on a family of parametric models or a given model form with several independent variables. Still, a substantial number of factors, such as the selection of variables and the function's form, make the identification of outliers very difficult because the outliers are model dependent. Even though the explanatory variables are known, the outlier detection is still a study issue.

III. The Resistant Learning

From a statistics perspective, outlier detection research can be conducted in either the context of model estimation or the context of resistant learning [23]. In the context of model estimation, the response y is modeled as $f(\mathbf{x}, \mathbf{w}) + \delta$, where $\mathbf{w}$ is the parameter vector and δ is the error term. The function form of the fitting function f is predetermined and fixed during the process of deriving values for its associated $\mathbf{w}$ from a set of N given observations $\{(\mathbf{x}^1, y^1), \cdots, (\mathbf{x}^N, y^N)\}$, with y^c being the observed response corresponding to the cth observation with explanatory variables $\mathbf{x}^c$. The least squares estimator (LSE) is one of the most popular methods for performing the estimation. The generalized delta rule proposed by Rumelhart et al. [19] is a kind of nonlinear LSE.

If $\hat{\mathbf{W}}$ denotes an estimate of $\mathbf{w}$, then LSE is defined to be the $\hat{\mathbf{W}}$ that minimizes $\sum_{c=1}^{N} (e^c)^2$, in which

$$e^c = y^c - f(\mathbf{x}^c, \mathbf{w}). \tag{1}$$

Intuitively, outliers are those observations with large error e^c and far away from the fitting function f. Of course, the errors depend on the fitting function, i.e., model-dependent, while the LSE, on the other hand, is very sensitive to outliers. In practice, the fitting function form and identification of outliers interact with each other which complicate identifying the true outliers.

In the context of resistant learning, the function form of f is adaptable during the process of derivation of its associated $\mathbf{w}$ from a set of N given observations $\{(\mathbf{x}^1, y^1), \cdots, (\mathbf{x}^N, y^N)\}$. Here, "resistant" is equivalent to "robust." The terms "robust" and "resistant" are often used interchangeably in the statistical literature, but sometimes have specific meanings [9]. Robust procedures are those whose results are not impacted significantly by violations of the model assumptions (such as when the errors are normally distributed). Resistant procedures are those whose numerical results are not impacted significantly by outlying observations.

Tsaih and Cheng [23] propose an algorithm with a tiny pre-specified ε value (say, 10^{-6}) that can deduce a proper nonlinear function form f and $\hat{\mathbf{W}}$ such that $|y^c - f(\mathbf{x}^c, \hat{\mathbf{w}})| \le \varepsilon$, for all c. These authors adopt both robustness analysis and deletion diagnostics to address the presence of outliers. Robustness analysis entails adopting the idea of a C-step [18] for deriving an (initial) subset of m+1 reference observations to fit the linear regression model, ordering the residuals of all N observations at each stage and then augmenting the reference subset gradually based upon the smallest trimmed sum of squared residuals principle. At the same time, the weight-tuning mechanism, the recruiting mechanism, and the reasoning mechanism allow the single-hidden layer feed-forward neural networks (SLFN) to adapt dynamically during the process and to be able to explore an acceptable nonlinear relationship between explanatory variables and the response in the presence of outliers. The deletion diagnostic

approach is employed with the diagnostic quantity as the number of pruned hidden nodes when one observation is excluded from the reference pool. This diagnostic quantity indicates whether the SLFN is stable.

IV. The Concept of Envelope Module

This study changes the algorithm proposed by Tsaih and Cheng [23], which uses a tiny ε value, into the envelope module with a non-tiny ε value that evolves a nonlinear fitting function f wrapped with an envelope whose width is 2ε. The setting of the ε value depends on the user's perception of the data and its associated outliers. For example, the perceptions are that the error is normally distributed, with a mean of 0 and a variance of 1, and the outliers are the points that have residuals that are greater than 1.96 (when the absolute value is taken). These perceptions are similar to the setting in the regression analysis that corresponds to a 5% significance level. Given that the error terms follow the normal distribution. Then, the user can set the ε value of the proposed envelope module to 1.96 and define the outliers as the points that have residuals that are greater than $\varepsilon*\gamma*\sigma$, where σ is the standard deviation of the residuals of the current reference observations and γ is a constant that is equal to or greater than 1.0, depending on the user's stringency in the outlier detection. The larger the γ value is, the more stern is the outlier detection.

The idea behind taking a larger rejection bound $\varepsilon*\gamma*\sigma$ is similar to the main concept in Repeated Significance Tests [15] and Group Sequential Tests [17]. Because we search for potential outliers sequentially (i.e., the problem of multiple testing), a type-I error will definitely increase, and we must re-adjust the significance level.

V. The Justification of Envelope Module

We shall first define the notations that are used to describe the proposed approach. The SLFN is defined in (2) and (3), where $tanh(x) \equiv \frac{e^x - e^{-x}}{e^x + e^{-x}}$; m is the number of explanatory variables x_j's; $\mathbf{x} \equiv (x_1, x_2, \ldots, x_m)^T$; p is the number of adopted hidden nodes; w_{i0}^H is the bias value of the i^{th} hidden node; the superscript H throughout the paper refers to quantities related to the hidden layer; w_{ij}^H is the weight between the j^{th} explanatory variable x_j and the i^{th} hidden node; w_0^o is the bias value of the output node; the superscript o throughout the paper refers to quantities related to the output layer; and w_i^o is the weight between the i^{th} hidden node and the output node. In this article, a character in bold represents a column vector, a matrix, or a *set*, and the superscript T indicates the transposition.

$$a_i(\mathbf{x}) \equiv tanh(w_{i0}^H + \sum_{j=1}^{m} w_{ij}^H x_j). \quad (2)$$

$$f(\mathbf{x}) \equiv w_0^o + \sum_{i=1}^{p} w_i^o \, tanh(w_{i0}^H + \sum_{j=1}^{m} w_{ij}^H x_j). \quad (3)$$

Furthermore, let $\mathbf{w}_i^H \equiv (w_{i0}^H, w_{i1}^H, w_{i2}^H, \ldots, w_{im}^H)^T$; $\mathbf{w}^o \equiv (w_0^o, w_1^o, w_2^o, \ldots, w_p^o)^T$; $\mathbf{w}^H \equiv \begin{pmatrix} \mathbf{w}_1^H \\ \mathbf{w}_2^H \\ \vdots \\ \mathbf{w}_p^H \end{pmatrix}$; and $\mathbf{w} \equiv \begin{pmatrix} \mathbf{w}^H \\ \mathbf{w}^o \end{pmatrix}$.

Here, we assume that $\mathbf{w}^o$ and $\mathbf{w}^H$ are non-zero variable vectors and p is an integer variable that is always positive. Note that because the value of p is adjustable, the (nonlinear) function form of f is adaptable, and the hyperbolic tangent function is used here as the base of f.

Through this SLFN, the input information $\mathbf{x}$ is first transformed into $\mathbf{a} \equiv (a_1, a_2, \ldots, a_p)^T$, and the corresponding value of f is generated by $\mathbf{a}$ rather than $\mathbf{x}$. In other words, given the observation $\mathbf{x}$, all of the corresponding values of hidden nodes are first calculated with $a_i = tanh(w_{i0}^H + \sum_{j=1}^{m} w_{ij}^H x_j)$ for all i and the corresponding value $f(\mathbf{x})$ is then calculated as $f(\mathbf{x}) = g(\mathbf{a}) \equiv w_0^o + \sum_{i=1}^{p} w_i^o a_i$.

Table I presents the proposed envelope module. Assume that there is a total of N observations, $N \geq m+1$, and $\mathbf{x}^i \neq \mathbf{x}^j$ when $i \neq j$. Let $\mathbf{I}(N)$ be the set of indices of all of the observations. Let the n^{th} stage of the corresponding process, $N \geq n > m+1$, be the stage of handling n *reference* observations, which are the observations with the smallest n squared residuals among N squared residuals, and let $\mathbf{I}(n)$ be the set of indices of these reference observations. At the n^{th} stage, we look for an *acceptable* SLFN estimate that leads to a set of $\{(\mathbf{x}^c, y^c): c \in \hat{\mathbf{I}}(n)\}$ with $(e^c)^2 \leq \varepsilon^2$ for all $c \in \hat{\mathbf{I}}(n)$ and $\mathbf{I}(n) \subseteq \hat{\mathbf{I}}(n)$, where e^c is defined in (1) and 2ε is equal to the pre-specified width of the envelope. At the n^{th} stage, $\hat{\mathbf{I}}(n)$ is the set of indices of the observations that are contained in the obtained envelope. $|\hat{\mathbf{I}}(n)| \geq n$ because $\mathbf{I}(n) \subseteq \hat{\mathbf{I}}(n)$. In other words, at the end of the n^{th} stage, the acceptable SLFN estimate presents a fitting function f around an envelope that contains at least n observations in $\{(\mathbf{x}^c, y^c): c \in \hat{\mathbf{I}}(n)\}$. Let $\bar{\mathbf{I}}(n)$ be the set of indices of $(\mathbf{x}^c, y^c)$ that have the smallest n squared residuals among N squared residuals at the end of the n^{th} stage. $\bar{\mathbf{I}}(n) \subseteq \hat{\mathbf{I}}(n)$, and $\bar{\mathbf{I}}(n)$ may not be equal to $\mathbf{I}(n)$.

TABLE I.
THE PROPOSED ENVELOPE MODULE

Step 1: Arbitrarily obtain the initial $m+1$ reference observations. Let $\mathbf{I}(m+1)$ be the set of indices of these observations. Set up an acceptable SLFN estimate with one hidden node regarding the reference observations $\{(\mathbf{x}^c, y^c): c \in \mathbf{I}(m+1)\}$. Set $n = m+2$.

Step 2: If $n > N$, STOP.

Step 3: Present the n reference observations $(\mathbf{x}^c, y^c)$ that are the ones with the smallest n squared residuals among the current N squared residuals. Let $\mathbf{I}(n)$ be the set of indices of these observations.

Step 4: If $(e^c)^2 \le \frac{(n-1)^2\varepsilon^2}{(N-1)^2}$ $\forall\, c \in \mathbf{I}(n)$, go to Step 7.

Step 5: Assume $\kappa \in \mathbf{I}(n)$, $(e^\kappa)^2 > \frac{(n-1)^2\varepsilon^2}{(N-1)^2}$, and $(e^c)^2 \le \frac{(n-1)^2\varepsilon^2}{(N-1)^2}$ $\forall$ $c \in \mathbf{I}(n)$-$\{\kappa\}$. Set $\tilde{\mathbf{w}} = \mathbf{w}$.

Step 6: Apply the gradient descent mechanism to adjust weights $\mathbf{w}$ until one of the following two cases occurs:

(1) If the deduced envelope (with the width $\varepsilon^*\gamma$) contains at least n observations, then go to Step 7.

(2) If the deduced envelope does not contain at least n observations, then set $\mathbf{w} = \tilde{\mathbf{w}}$ and apply the augmenting mechanism to add extra hidden nodes to obtain an acceptable SLFN estimate.

Step 7: Implement the pruning mechanism to delete all of the potentially irrelevant hidden nodes; $n + 1 \rightarrow n$; go to Step 2.

The proposed envelope module in Table I executes the following two procedures: (i) the ordering procedure implemented by Step 3 that determines the input sequence of reference observations and (ii) the modeling procedure implemented by Step 6 and Step 7 that adjusts the number of hidden nodes adopted in the SLFN estimate and the associated $\mathbf{w}$ to evolve the fitting function f and its envelope to contain at least n observations at the n^{th} stage. The details are explained as follows.

In Step 1, we arbitrarily obtain the initial m+1 reference observations $\{(\mathbf{x}^c, y^c): c \in \mathbf{I}(m+1)\}$. The number m+1 is also the number of the associated $\mathbf{W}_1^H$ and w_{10}^H. We can use the data set $\{(\mathbf{x}^c, \tanh^{-1}(\frac{y^c - \min_{c\in\mathbf{I}(N)} y^c + 1}{\max_{c\in\mathbf{I}(N)} y^c - \min_{c\in\mathbf{I}(N)} y^c + 2})): c \in \mathbf{I}(m+1)\}$ to set up the system (4), which is a system of m+1 linear equations in m+1 unknowns. Then, the values of w_0^o and w_1^o are assigned as $\min_{c\in\mathbf{I}(N)} y^c - 1$ and $\max_{c\in\mathbf{I}(N)} y^c - \min_{c\in\mathbf{I}(N)} y^c + 2$, respectively. Step 1 uses the weight values w_{10}^H and $\mathbf{w}_1^H$ that were obtained from solving system (4) and the assigned values of w_0^o and w_1^o to set up an acceptable SLFN estimate that renders $(e^c)^2 = 0 \le \frac{m^2\varepsilon^2}{(N-1)^2} = \frac{(m+1-1)^2\varepsilon^2}{(N-1)^2}$, for all $c \in \mathbf{I}(m+1)$.

$$w_{10}^H + \sum_{j=1}^{m} w_{1j}^H\, x_j^c = \tanh^{-1}\left(\frac{y^c - \min_{c\in\mathbf{I}(N)} y^c + 1}{\max_{c\in\mathbf{I}(N)} y^c - \min_{c\in\mathbf{I}(N)} y^c + 2}\right) \text{ for all } c \in \mathbf{I}(m+1). \tag{4}$$

At the n^{th} stage, Step 3 presents the n reference observations $\{(\mathbf{x}^c, y^c): c \in \mathbf{I}(n)\}$, which are the observations with the smallest n squared residuals among the current N squared residuals and are used to evolve the fitting function. Step 3 adopts the concept of forward selection [2], ordering the residuals of all N observations and then augmenting the reference subset gradually by including extra observations one by one to determine the input sequence of the reference observations. However, $\bar{\mathbf{I}}(n)$ might not equal $\mathbf{I}(n)$. Specifically, some of the reference observations at the early stages might not stay in the set of reference observations at the later stages, although most of them will.

Note that, at the end of the n-1$^{\text{th}}$ stage, the ones in $\{(\mathbf{x}^c, y^c): c \in \bar{\mathbf{I}}(n-1)\}$ are the smallest n-1 squared residuals among N squared residuals, and $(e^c)^2 \le \frac{(n-2)^2\varepsilon^2}{(N-1)^2}$ for all $c \in \bar{\mathbf{I}}(n-1)$. Thus, at Step 3, $\{(\mathbf{x}^c, y^c): c \in \mathbf{I}(n)\} = \{(\mathbf{x}^c, y^c): c \in \bar{\mathbf{I}}(n-1)\} \cup \{(\mathbf{x}^\kappa, y^\kappa)\}$, where $(e^c)^2 \le \frac{(n-1)^2\varepsilon^2}{(N-1)^2}$ for all $c \in \bar{\mathbf{I}}(n-1)$ and $(\mathbf{x}^\kappa, y^\kappa)$ is the observation with the n^{th} smallest squared residuals among the N squared residuals at the beginning of the n^{th} stage. $(\mathbf{x}^\kappa, y^\kappa)$ is named as the *next point* at the n^{th} stage. Therefore, at Step 4, to check if $(e^c)^2 \le \frac{(n-1)^2\varepsilon^2}{(N-1)^2}$ for all $c \in \mathbf{I}(n)$ is the same as checking if $(e^\kappa)^2 \le \frac{(n-1)^2\varepsilon^2}{(N-1)^2}$. If $(e^\kappa)^2 \le \frac{(n-1)^2\varepsilon^2}{(N-1)^2}$, then $(e^c)^2 \le \frac{(n-1)^2\varepsilon^2}{(N-1)^2}$ for all $c \in \mathbf{I}(n)$; only the pruning mechanism of Step 7 is involved, and the next stage can be implemented. If $(e^\kappa)^2 > \frac{(n-1)^2\varepsilon^2}{(N-1)^2}$, then we still have $(e^c)^2 \le \frac{(n-1)^2\varepsilon^2}{(N-1)^2}$ for all $c \in \mathbf{I}(n)$-$\{\kappa\}$, and Step 6 is executed.

The modeling procedure implemented by Step 6 to Step 7 requires proper values of $\mathbf{w}$ and p so that the obtained envelope contains at least n observations at the end of the n^{th} stage. Specifically, at the beginning of Step 6, the gradient descent mechanism is applied to adjust the weights $\mathbf{w}$. One of the gradient descent mechanisms proposed in the literature is the weight-tuning mechanism [23] for $\min_{\mathbf{w}} E_n(\mathbf{w}) \equiv \sum_{c\in\mathbf{I}(n)} (w_0^o + \sum_{i=1}^{p} w_i^o\, tanh(w_{i0}^H + \sum_{j=1}^{m} w_{ij}^H\, x_j^c) - y^c)^2 + \varepsilon_1\|\mathbf{w}\|^2$. However, a result of implementing the gradient descent mechanism might be getting stuck in a local optimum. Another possible scenario of getting stuck in a local optimum is when the SLFN estimate obtained at the previous stage is defective regarding the modeling job of the current stage, i.e., the current number of hidden nodes is not sufficient for the SLFN estimate to work well for the modeling job of the current stage. Both scenarios lead to an unacceptable SLFN estimate regarding the reference observations, and unfortunately, at present, there is no perfect optimization mechanism to simultaneously cope with both scenarios.

Step 6.2 restores the $\tilde{\mathbf{w}}$ that is stored in Step 5. Thus, we return to the previous SLFN estimate, which renders

$(e^{\kappa})^2 > \frac{(n-1)^2\varepsilon^2}{(N-1)^2}$ and $(e^{c})^2 \le \frac{(n-1)^2\varepsilon^2}{(N-1)^2}$ for all $c \in \mathbf{I}(n)$-$\{\kappa\}$. Then, the augmenting mechanism should recruit extra hidden nodes to render $(e^{c})^2 \le \frac{(n-1)^2\varepsilon^2}{(N-1)^2}$ for all $c \in \mathbf{I}(n)$. One of the augmenting mechanisms proposed in the literature is the recruiting mechanism of Tsaih and Cheng [23], which adds two extra hidden nodes to the previous SLFN estimate to render $(e^{c})^2 \le \frac{(n-1)^2\varepsilon^2}{(N-1)^2}$ for all $c \in \mathbf{I}(n)$.

To decrease the complexity of the fitting function f, the pruning mechanism is proposed in Step 7 to delete potentially irrelevant hidden nodes. At the n^{th} stage, a hidden node is potentially irrelevant if it is deleted, and the application of the gradient descent mechanism can accomplish the goal that the obtained envelope contains at least n observations. A defective SLFN estimate triggers the augmenting mechanism in Step 6.2, but the situation of leading to an undesired local minimum also triggers the augmenting mechanism. Thus, the augmenting mechanism could recruit excess hidden nodes that later become irrelevant. The irrelevant hidden nodes are useless with respect to the learning goal, and they could result in the overfitting likelihood of the fitting function f. One of the pruning mechanisms proposed in the literature is a reasoning mechanism given in [23].

Although Steps 4 to 7 presented in this study are similar to the steps presented by Tsaih and Cheng, the ε value adopted here is much larger than the value in Tsaih and Cheng [23]. Specifically, the proposed envelope module wants to evolve the fitting function around an envelope, to contain at least n observations at the n^{th} stage. This approach is distinct from the idea of Tsaih and Cheng, which attempts to evolve the fitting function to almost precisely fit all of the reference observations $\{(\mathbf{x}^c, y^c): c \in \mathbf{I}(n)\}$ at the n^{th} stage (because the corresponding ε value is tiny).

The proposed module would result in a fitting function with an envelope that includes all of the observations, and we would expect that the outliers will be included at later stages. In addition, we use the deviance information as the extra information to define the outliers. Specifically, here we adopt both the deviance information and the order information to identify the outliers. Regarding the deviance information for identifying the outliers, at Step 3 of the n^{th} stage, we further calculate the standard deviation of $(e^c)^2$ of $\{(\mathbf{x}^c, y^c): c \in \bar{\mathbf{I}}(n-1)\}$ and the deviation of the next point $(\mathbf{x}^{\kappa}, y^{\kappa})$ from the current fitting function f. Here, we define the *diagnostic* quantity for the outlier as the ratio of the residual of the next point in absolute value to the standard deviation of $(e^c)^2$ of $\{(\mathbf{x}^c, y^c): c \in \bar{\mathbf{I}}(n-1)\}$, assuming that the errors follow a normal distribution $N(0, \sigma^2)$ and the outliers are the points that have residuals greater than ε (in absolute value). Thus, if the diagnostic quantity is greater than $\varepsilon^*\gamma$, then the next point is treated as a potential outlier. Here, γ is a constant that is equal to or greater than one, depending on how stringent the threshold is for the outliers.

Regarding the order information for identifying the outliers, we propose two approaches, fixed and flexible. The fixed approach is to treat the last 5% of the observations as potential outliers. Namely, if $n \ge 0.95N$ AND the diagnostic quantity is greater than $\varepsilon^*\gamma$, then the next point is recorded as the *identified* outlier.

However, the flexible approach assumes that we know the number of outliers, say k, and treat the last k observations as potential outliers. Specifically, regarding the flexible approach, if $n \ge N$-k AND the diagnostic quantity is greater than $\varepsilon^*\gamma$, then the next point is recorded as the *identified* outlier.

VI. An Illustrative Experiment

A. The experiment design and the result

We apply the proposed envelope module to 100 simulation runs to evaluate the effectiveness of detecting the outliers. For each simulation run, we use the nonlinear model stated in (5) to generate a set of 100 observations for which the explanatory variable X is equally spaced from 0.0 to 20.0 and the error is normally distributed, with a mean of 0 and a variance of 1. Here, the *theoretical* fitting function f is the function stated in (5), and the *theoretical* outliers are the points with residuals, as compared to (5), that are greater than 1.96 in absolute value. This definition is similar to the setting in the regression analysis that corresponds to a 5% significance level given the normal distribution.

$$Y=0.5 + 0.4*X + 0.8*\text{Exp}(-X) + \text{Error}. \quad (5)$$

Here, we set the ε value of the proposed envelope module to $\sqrt{3}$, which is smaller than but close to 1.96, the threshold for the theoretical outliers. With a stringent threshold for the outlier in mind, the γ value of the proposed envelope module is set such that $\sqrt{3}*\gamma$ is equal to 2.5. The idea behind taking a larger rejection bound of 2.5 is similar to that in the Repeated Significance Tests [15] and Group Sequential Tests [17].

Table II shows the number of theoretical outliers in 100 simulated data sets. For example, there are 16 simulation runs, each of which has 3 theoretical outliers, and there are 14 runs that have 4 theoretical outliers. On average, there are 4.97 theoretical outliers in each observation set. As shown in Table II, there are 60 runs with 5 or fewer than 5 theoretical outliers. Among the data sets of these 60 runs, on average, there are 3.55 theoretical outliers. At the same time, there are 40 runs that have at least 6 theoretical outliers, and on average, there are 7.1 theoretical outliers. Note that the error terms were generated using the freeware R, version 2.12.0.

TABLE II.
Number of theoretical outliers in 100 simulation data sets

Number of theoretical outliers	Number of simulation runs
1	2
2	11
3	16
4	14

5	17
6	21
7	7
8	5
9	4
11	3

Without losing generality, the 10th data set shown in Fig. 1 is used to illustrate the model fitting and the outlier detection for using the proposed envelope module. Among 100 observations in the 10th data set, there are six theoretical outliers, marked with the red square shown in Fig. 1. Apparently, these outliers are located outside of the bulk of the majority of 100 observations.

Fig. 2 shows the graphs of $\{(\mathbf{x}^c, y^c): c \in \bar{\mathbf{I}}(n-1)\}$ and the corresponding next point $(\mathbf{x}^\kappa, y^\kappa)$ obtained at Step 3 of the 71st, 72nd, 96th, 98th, and 100th stage, as well as the graph of the final fitting function and its envelope regarding the 10th simulation run. The round circle denotes the next point. As shown in Fig. 2, the proposed algorithm evolves the fitting function and its envelope accordingly, to contain the corresponding next point $(\mathbf{x}^\kappa, y^\kappa)$.

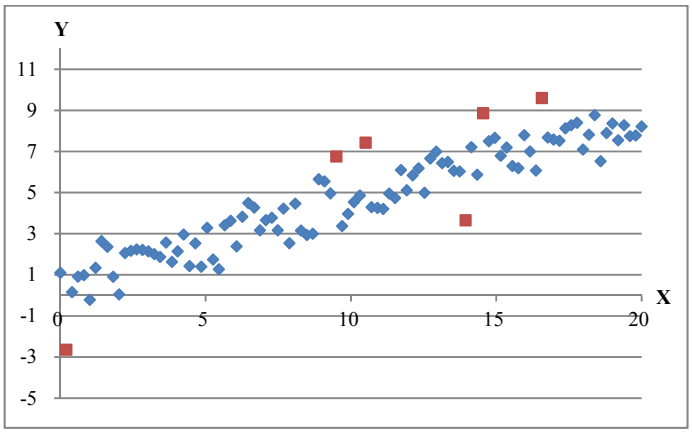

Fig. 1. The graph of $\{(\mathbf{x}^c, y^c)\}$ of the 10th data set.

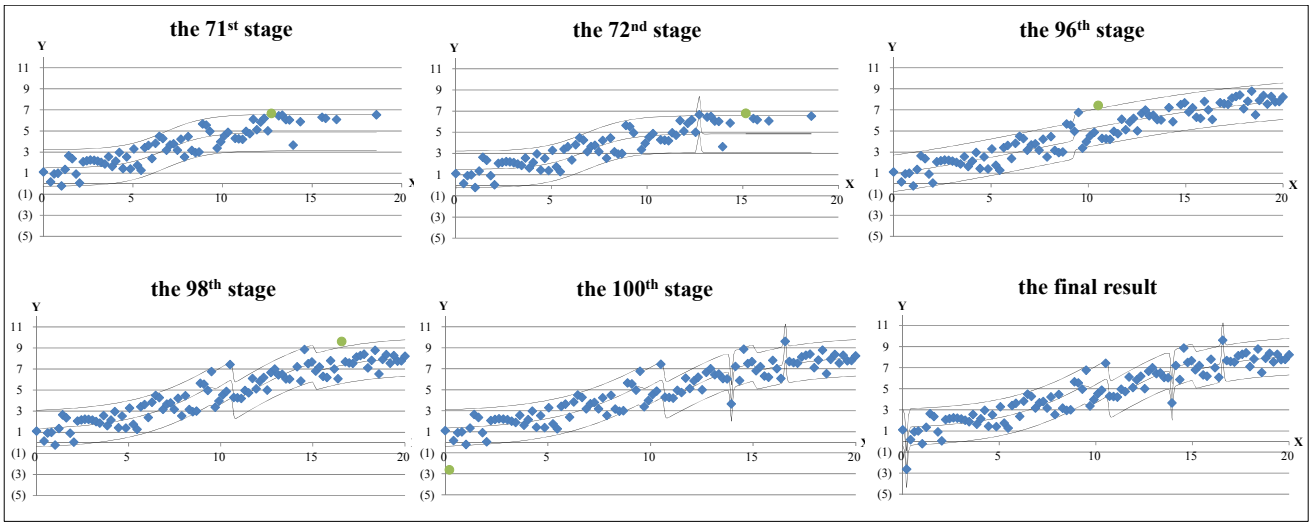

Fig. 2. The graphs of $\{(\mathbf{x}^c, y^c): c \in \bar{\mathbf{I}}(n-1)\}$ and the corresponding next point $(\mathbf{x}^\kappa, y^\kappa)$ obtained in Step 3 of the 71st, 72nd, 96th, 98th, and 100th stage and the graph of the final fitting function and its envelope regarding the 10th data set.

B. The evaluation

We use the non-linear regression method associated with the function form of equation (5) as the benchmark for evaluating the performance on outlier detection of our proposed algorithm.

The following two error measurements are also adopted to evaluate the performance of outlier detection: Type I and Type II errors. The Type I error is defined as the proportion of theoretical non-outliers that were mis-specified as identified outliers, and the Type II error is the proportion of theoretical

outliers that were mis-specified as identified non-outliers. In many applications, such as medical diagnosis, other criteria similar to these two errors are also used, as in Lalkhen and McCluskey [13] and Loong [14]. For example, Specificity and Sensitivity are two frequently used criteria, where Specificity = 1 − Type I error and Sensitivity = 1 − Type II error.

Table III lists the mean and standard deviation of Type I and II errors of the outlier detection regarding the envelope module and the benchmark. The last column indicates that 99.53% of the non-outliers are identified correctly and 80.49% of the outliers are identified correctly with respect to the benchmark. Surprisingly, the Type II error for the benchmark is fairly large. Approximately 20% of the outliers cannot be identified, even though we know the relationship between the response variable Y and the exploratory variable X.

TABLE III.
TYPE I AND II ERRORS.

	Envelope module		Benchmark
	Flexible	Fixed	
Type I error	1.19%	1.22%	0.47%
Type II error	52.52%	54.53%	19.51%

The type II errors regarding the envelope module with the flexible and fixed approaches are 52.52% and 54.53%, respectively. Note that, if we perform the outlier detection randomly without the knowledge of the fitting function form, the Type I and Type II errors are approximately 5% and 95%, respectively, because on average, there are 4.97 outliers in each set of 100 observations. In other words, the envelope module with the flexible approach contributes a 42.48% (= 95% − 52.52%) effect on the outlier detection, which is significantly large. This result is a large improvement on having no information, although the errors of the proposed module are still larger than those of the benchmark.

VII. DISCUSSION AND FUTURE RESEARCH

This study proposes an envelope module which adopts both the deviance information and the order information to identify the outliers. The proposed envelope module is based on data. In other words, at the n^{th} stage, the envelope has evolved to contain the reference observations of $\{(\mathbf{x}^c, y^c): c \in \bar{\mathbf{I}}(n-1)\} \cup \{(\mathbf{x}^\kappa, y^\kappa)\}$, and the identified outlier is the next point $(\mathbf{x}^\kappa, y^\kappa)$, whose deviation from the fitting function f is greater than $\varepsilon^*\gamma^*\sigma$, where σ is the standard deviation of the residuals of $\{(\mathbf{x}^c, y^c): c \in \bar{\mathbf{I}}(n-1)\}$.

In contrast with the algorithm proposed by Tsaih and Cheng [23], the envelope module uses a non-tiny ε value instead of a tiny ε value that results in a nonlinear fitting function f around the envelope whose width is 2ε. Also, the envelope should contain at least n observations at the n^{th} stage. This is different from the concept of searching the fitting functions of all the n reference observations at the n^{th} stage, which tends to result in overfitting to the noisy data.

In summary, this study has fulfilled the following two objectives: (1) Revise the algorithm of Tsaih and Cheng [23] to form an effective way of identifying outliers in the context of resistant learning. (2) Set up an illustrative experiment to justify the effectiveness of the envelope module in identifying outliers in the context of resistant learning.

Because of the complexity of outlier detection in the context of resistant learning, this study is the first study to derive an effective module for outlier detection. To deal with the outlier detection problem in both contexts of resistant learning and changing environments, future goals of this study are as follows:

(1) Integrate the moving window strategy with the envelope module into an outlier detection algorithm that can work in both contexts of resistant learning and changing environments. For instance, the new resistant learning algorithm is applied to the daily S&P 500 stock index futures price data from year 1984 to year 1993 for the outlier detection. The moving window strategy is as follows: the envelope module is applied to the 4-year daily data, from year 1984 to year 1987, to construct a proper envelope. This envelope is then applied to the outlier detection in year 1988. We replicate this strategy throughout the period from year 1989 to year 1993. In other words, the period of outlier detection in the simulation is from year 1988 to year 1993.
(2) Set up a real-world experiment (regarding security applications) to explore the effectiveness of the derived outlier detection algorithm.
(3) Explore the reality of identified outliers in a real-world experiment. In other words, we would like to explore some of the following questions: Are some of the identified outliers the real outliers? Are there any further analyses that can help to understand the method of detecting them and assessing their real-world impact?
(4) Apply the outlier detection algorithm to a real-work problem such as the detection of abnormal network behaviors and zero-day attacks.

REFERENCES

[1] M. Agyemang, K. Barker, and R. Alhajj, "A comprehensive survey of numeric and symbolic outlier mining techniques," *Intelligent Data Analysis*, vol. 10, no. 6, pp. 521-538, 2006.

[2] A. Atkinson, and T. Cheng, "Computing Least Trimmed Squares Regression with the Forward Search," *Statistics and Computing*, vol. 9, pp. 251-263, 1999.

[3] A. Atkinson, and M. Riani, Forward search added-variable t-test and the effect of masked outliers on model selection," *Biometrika*, vol. 89, pp. 939-946, 2002.

[4] V. Chandola, A. Banerjee, and V. Kumar, "Outlier detection: A survey," *ACM Computing Surveys*, 2007.

[5] L. A. Chen, D. T. Chen, and W. Chan, "The distribution-based p-value for the outlier sum in differential gene expression analysis," *Biometrika*, vol. 97, no. 1, pp. 246-253, 2010.

[6] C.C. Chuang, S. F. Su, J. T. Jeng, and C.C. Hsiao, "Robust support vector regression networks for function approximation with outliers," *IEEE Transactions on Neural Networks*, vol. 13, no. 6, pp. 1322-1330, 2002.

[7] J. T. Connor, and R. D. Martin, "Recurrent neural networks and robust time series prediction," *IEEE Transactions Neural Networks*, vol. 2, no. 5, pp. 240-253, 1994.

[8] J. Hoeting, A. Raftery, and D. Madigan, “A method for simultaneous variable selection and outlier identification in linear regression”, *Computational Statistics and Data Analysis*, vol. 22, pp. 251-270, 1996.

[9] J. Huber, *Robust Statistics*. New York: John Wiley, 1981.

[10] E. Knorr, and R. Ng, “A unified approach for mining outliers,” *Proc. KDD*, pp. 219-222, 1997.

[11] E. Knorr, and R. Ng, “Algorithms for mining distance-based outliers in large datasets,” *Proc. 24th Int. Conf. Very Large Data Bases*, pp. 392-403, 1998.

[12] E. Knorr, R. Ng, and V. Tucakov, “Distance-based outliers: Algorithm and applications,” *Very Large Data Bases*, vol. 8, pp. 237-253, 2000.

[13] A. G. Lalkhen, and A. McCluskey, “Clinical tests: sensitivity and specificity,” *Continuing Education in Anaesthesia, Critical Care & Pain*, vol. 8, no. 6, 221-223, 2008.

[14] T. Loong, “Understanding sensitivity and specificity with the right side of the brain,” *British Medical Journal*, vol. 327, no. 7417, pp. 716-719, 2003.

[15] C. K. McPherson, and P. Armitage, “Repeated significance tests on accumulating data when the null hypothesis is not true,” *Journal of the Royal Statistical Society*, Series A, vol. 134, pp. 15-25, 1971.

[16] E. W. Ngai, H. U. Yong, Y. H. Wong, Y. Chen, and X. Sun, “The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature,” *Decision Support Systems*, vol. 50, no. 3, pp. 559-569, 2011.

[17] S. J. Pocock, “Group sequential methods in the design and analysis of clinical trials,” *Biometrika*, vol. 64, no. 2, pp. 191-199, 1977.

[18] P. Rousseeuw, and K. Van Driessen, “Computing LTS Regression for Large Data Sets,” Tech. Rep. University of Antwerp, Belgium, 1999.

[19] D. Rumelhart, G. Hinton, and R. Williams, *Modeling Internal Representations by Error Propagation*. In: Rumelhart D, McClelland J (eds) Parallel Distributed Processing: Explorations in the Microstructure of Cognition 1. MIT Press, Cambridge, MA, pp. 318-362, 1986.

[20] B. Sluban, D. Gamberger, and N. Lavrač, "Ensemble-based noise detection: noise ranking and visual performance evaluation," Data Mining and Knowledge Discovery, vol. 28, no. 2, pp. 265-303, 2014.

[21] R. Tibshirani, and T.Hastie, “Outlier sums for differential gene expression analysis,” *Biostatistics*, vol. 8, no. 1, pp. 2-8, 2007.

[22] S. A. Tomlins, D. R. Rhodes, S. Perner, S. M. Dhanasekaran, and R. Mehra, “Recurrent Fusion of TMPRSS2 and ETS Transcription Factor Genes in Prostate Cancer,” *Science*, vol. 310, pp. 644-648, 2005.

[23] R. Tsaih, and T.Cheng, “A Resistant Learning Procedure for Coping with Outliers,” Annals of Mathematics and Artificial Intelligence, vol. 57, no. 2, pp. 161-180.

[24] H. Vuong, K. Shedden, Y. Liu, and D.M. Lubman, “Outlier-Based Differential Expression Analysis in Proteomics Studies,” *Bioinformatics*, vol. 4, no. 6, pp. 116-122, 2011.

[25] M. Windham, “Robustifying Model Fitting,” *Journal of the Royal Statistical Society*, Series B, vol. 57, pp. 599-609, 1995.