A Formal Approach to Anomaly Detection

André Eriksson and Hedvig Kjellström
Computer Vision and Active Perception Lab, KTH Royal Institute of Technology, Stockholm, Sweden

Keywords: Anomaly Detection, Formal Methods, Model Selection.

Abstract: While many advances towards effective anomaly detection techniques targeting specific applications have been made in recent years, little work has been done to develop application-agnostic approaches to the subject. In this article, we present such an approach, in which anomaly detection methods are treated as formal, structured objects. We consider a general class of methods, with an emphasis on methods that utilize structural properties of the data they operate on. For this class of methods, we develop a decomposition into *sub-methods*—simple, restricted objects, which may be reasoned about independently and combined to form methods. As we show, this formalism enables the construction of software that facilitates formulating, implementing, evaluating, as well as algorithmically finding and calibrating anomaly detection methods.

1 INTRODUCTION

Anomaly detection tasks are encountered in many areas of science, technology, and business, and automated anomaly detection methods are indispensable in many applications, such as intrusion detection and fraud detection (Lazarevic et al., 2003; Phua et al., 2010). As manual analysis of the ever growing datasets encountered in many application domains becomes increasingly difficult, the need for such methods can be expected to grow.

For this need to be effectively met, approaches that enable researchers and organizations to effectively develop and implement appropriate methods are required.

While there are excellent tools available for certain applications (Twitter, 2015; Etsy, 2015), there is a notable lack of application-agnostic tools and approaches.

Considering the disparate nature of data encountered in applications, and the often subjective notion of what constitutes an anomaly, it seems unlikely that specific methods that work well across a majority of applications can be found. A more viable approach might be to instead focus on developing application-agnostic tools that facilitate formulating, implementing, evaluating, or calibrating methods.

We believe that taking a formal, high-level approach to the subject—where the focus is on what can be said about anomaly detection methods *in general*, rather than in the context of any specific application or task—is a vital step towards this goal.

The aim of this article is to present a particular such approach—where methods are treated as formal objects, which map datasets to *solutions* (i.e. a collection of anomaly scores, or a set of 'most anomalous' items), and which may be decomposed into *sub-methods* that may in turn be recombined into methods—and to demonstrate the utility of this approach in reaching the goals outlined above.

We target a general class of methods—with a focus on methods that utilize the structure of the dataset to find *contextual* or *collective* anomalies—for which we develop a formalism for decomposing methods into a collection of such sub-methods, amenable to being shared between applications involving similar types of data.

This enables an approach to developing methods where the principal consideration is the collection of applicable sub-methods (as constrained by the targeted task). These sub-methods may then be combined to form methods (either manually or algorithmically) until one that accurately solves the task at hand is found.

To demonstrate the utility of this approach, we apply it to a number of tasks involving sequences, as well as to finding and calibrating methods for such tasks (given a collection of sub-methods and labeled testing data).

2 RELATED WORK

Throughout the years, many anomaly detection

Eriksson, A. and Kjellström, H.
A Formal Approach to Anomaly Detection.
DOI: 10.5220/0005710803170326
In *Proceedings of the 5th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2016)*, pages 317-326
ISBN: 978-989-758-173-1

methods and applications has been studied. Plenty of surveys and books which discuss these in detail have been published (Hodge and Austin, 2004; Agyemang et al., 2006; Chandola et al., 2009; Fu, 2011).

To our knowledge, the formal, method-centric approach we take to anomaly detection is unique. However, there have been a few attempts to provide a general treatment of anomaly detection in relation to specific applications. For instance, in (Chandola, 2009), a high-level, formal discussion of common anomaly detection problems for sequences is presented. We build on this approach, taking it further and generalizing it to other types of tasks and data.

Our discussion of anomaly detection in sequences shows how diverse applications and methods related to sequences can be reconciled (Chandola et al., 2012; Chandola, 2009; Fu, 2011) and treated coherently.

We discuss a few specific tasks, including the detection of *point anomalies* (individual anomalous elements, also referred to as *outliers*) (Fox, 1972; Abraham and Chuang, 1989; Abraham and Box, 1979; Galeano et al., 2006; Tsay et al., 2000), *novelties* (previously unseen elements) (Markou and Singh, 2003a; Markou and Singh, 2003b; Ma and Perkins, 2003), elements anomalous with regard to nearby elements (Basu and Meckesheimer, 2007) and *anomalous subsequences* (Keogh et al., 2005; Keogh et al., 2007; Fu et al., 2006).

3 GENERALITY

When attempting to provide a formal basis for a concept as broad as *anomaly detection methods*, it is vital that care is taken to ensure that the breadth of the concept is captured by the resulting formalism.

In a widely cited survey of the subject, Chandola et al. (Chandola et al., 2009) discuss a few key aspects of anomaly detection tasks: the nature of the data and the types of anomalies involved, the expected solution format, and the type of supervision employed. Our aim is to provide a formalism which captures or generalizes these three aspects.

First, we formulate our formalism in a data- and solution-agnostic manner, so its applicability is independent of the nature of data and solutions.

Second, we target a general type of anomalies. Most methods are focused on detecting *point anomalies*—individual elements anomalous compared to the rest of the rest of the data. Such methods are appropriate for unstructured data (i.e. data in which individual elements are not related). However, the datasets encountered in applications (e.g. sequences, graphs, and spatial data) often have structure that can be exploited to better detect anomalies.

Chandola et al. discuss two other categories of anomalies: *contextual anomalies*—elements anomalous compared to a *context* (some subset of the data; typically 'nearby' elements)—and *collective anomalies*—collections of elements anomalous compared to the rest of the data. These can both be seen as generalizations of the concept of point anomalies.

Our formalism targets a fourth such category: *collective contextual anomalies*—collections of elements anomalous compared to a context—which naturally generalizes the other three[1].

An illustration of these four anomaly types is shown in Figure 1.

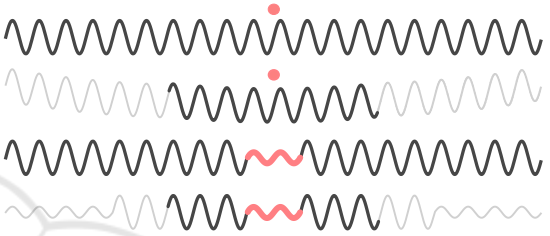


Figure 1: Examples of a *point anomaly* (top), a *contextual anomaly* (above center), a *collective anomaly* (below center), and a *collective contextual anomaly* (bottom) in univariate real-valued sequences. Anomalies are shown in light red; appropriate contexts in black.

Third, Chandola et al. classify methods as unsupervised, semi-supervised, or supervised based on whether they incorporate zero, one, or two classes of labeled training data. We formalize methods as maps from datasets to solutions; an approach naturally suited to expressing unsupervised methods. However, semi-supervised and supervised methods may also be expressed by replacing the input dataset with the disjoint union of the evaluation data and one or two sets of training data.

4 SUB-METHODS

Formally, an anomaly detection method may be treated as a mapping $m : D \to S$, that associates with each potential input dataset $d \in D$ a solution $s \in S$, where:

- D is an application-dependent set of well-formed datasets (e.g. all real-valued sequences, or all potential sets of users of a social network). We will

[1]Contextual and point anomalies correspond to single-element collections; for collective and point anomalies, the context is the entire dataset.

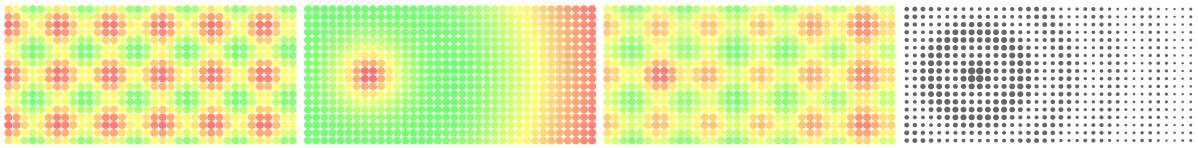


Figure 2: A dataset (center right) in $D = \mathcal{P}(C \times B) = \mathcal{P}(\mathbb{N}^2 \times \mathbb{R})$, constructed by linearly combining periodic data (far left) and data containing an anomaly (center left). We use the pattern on the far right to indicate contextual data.

assume that any dataset d is a set of items in some application-specific set X, so[2] $D \subseteq \mathcal{P}(X)$.

- S is a corresponding application-dependent set of potential solutions (e.g. all sequences of real-valued anomaly scores, or all potential sets of anomalous clusters of users in a social network).

For any given application, the set $M = D \to S$ corresponds to all potential methods.

When designing a method targeting point anomalies, there are two aspects to consider: what *anomaly measure* should be used to compare each item to the rest of the data, and how the results of these comparisons should be aggregated to form a solution.

Targeting collective anomalies means an additional aspect must be considered: how the set of *candidate anomalies* should be selected.

When targeting contextual anomalies, one must instead consider how a context should be associated with each candidate anomaly.

Since our formalism targets collective contextual anomalies, it must capture all four of these aspects.

This may be achieved by decomposing any $m \in M$ into four *sub-methods*, each responsible for one aspect. We will encode these as functions:

- The selection of candidate anomalies may be encoded as a function

$$\alpha : D \to \mathcal{P}(D),$$

which maps any dataset to a set of candidate anomalies[3] (subsets of that data; i.e. $\forall x, \forall y \in f(x) : y \subset x$). For methods targeting point and contextual anomalies, α produces singleton sets; i.e. $\alpha(d) = \{\{x\} | x \in d\}$.

- The selection of contexts may be encoded as a function

$$\beta : D \times D \to D,$$

which maps any dataset and one of its candidate anomalies to the context of that candidate anomaly (a subset of the dataset, disjoint with the candidate anomaly; i.e. $\forall x, \forall y \subset x : \beta(x,y) \subseteq x \setminus y$). For methods targeting point and collective anomalies, $\beta(x,y) = x \setminus y$.

- The comparison of candidate anomalies and contexts may be encoded as a function

$$\gamma : D \times D \to A,$$

which assigns a dissimilarity score $a \in A$ (where A is some method-specific set) to any candidate anomaly-context pair.

- The aggregation of anomaly scores may be encoded as a function

$$\delta : \mathcal{P}(D \times A) \to S,$$

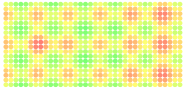
which maps any collection of candidate anomaly-dissimilarity score pairs to a solution.

Any tuple $(\alpha, \beta, \gamma, \delta)$ (for given S, D, and A) may be combined[4] to form an $m \in M$. Conversely, any $m \in M$ may be defined [5] as a tuple $m = (D, S, A, \alpha, \beta, \gamma, \delta)$.

For any given application, appropriate methods may be designed by reasoning about which choices of these sub-methods are applicable.

As an illustration of this approach, consider an application involving grids of real-valued data containing collective contextual anomalies (regions of the grid, anomalous with regard to their surroundings), as illustrated in Figure 2. Assume that the desired solution format is a grid of real-valued anomaly scores; i.e. $S = D$. What choices of sub-methods might be suitable?

First, α should produce candidate anomalies roughly on the scale of the anomalies we wish to capture. For instance, an α that produces non-overlapping square regions of size 6-by-6 may be employed:

$D \ni$  $\overset{\alpha}{\longmapsto} \{$ $, \ldots\} \in \mathcal{P}(D)$

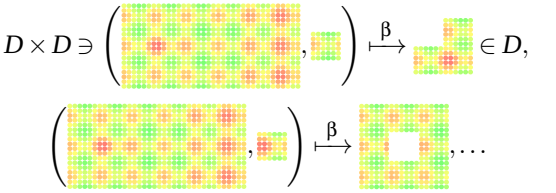
[2] We denote the power set of a set X by $\mathcal{P}(X)$.

[3] Here, and in the remainder of this section, we assume that information about the 'original' position of data items in the dataset is implicitly preserved when the data is rearranged or transformed. This issue is resolved in Section 5 through the requirement that the contextual data of each item is unique.

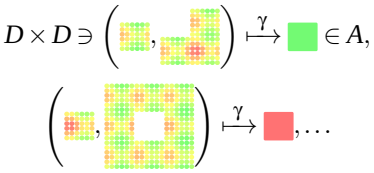
[4] Specifically by—for $d \in D$—letting $X = \alpha(d)$, $Y = \{(x, \beta(x,d)) \mid x \in X\}$, $Z = \{(x, \gamma(x,y)) \mid (x,y) \in Y\}$, and $m(d) = \delta(Z)$.

[5] Note that this implies no loss of generality; any $m : D \to S$ may be encoded by e.g. letting $A = S$, $\alpha(d) = \{d\}$, $\gamma(d,y) = m(d)$, and $\delta(\{s\}) = s$.

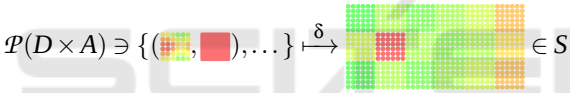
Second, β should produce as context some appropriately sized neighborhood of the candidate anomaly, such as the union of all adjacent such square regions:



Third, γ may be selected to compute the mean value of the items in the candidate anomaly, and the mean values in each 6-by-6 region of the context, and produce as anomaly score the mean absolute difference between the former and the latter (this means that $A = \mathbb{R}$)[6]:



Finally, δ should be selected to associate with each element the anomaly score of the candidate anomaly to which it belongs:



It would be an easy task to construct software that takes implementations of α, β, γ, and δ and combines them into a corresponding method implementation. Such software might be useful in constructing, calibrating, or evaluating methods.

However, its utility would be limited by the fact that α, β, γ, and δ are all formulated in terms of D, and would thus have to be implemented anew for each new application.

If the sub-methods could be defined such that implementations could be shared between applications with similar (rather than identical) data, software could then be coupled with a library of implemented sub-methods, drastically increasing its utility.

5 CONTEXTUAL AND BEHAVIORAL ATTRIBUTES

To accomplish this, we may instead define our sub-methods to operate on either *behavioral* or *contextual* attributes of the data.

[6]We illustrate values in $\mathbb{R}$ as colored squares. To improve the clarity of the presentation, we normalize anomaly scores so that the most and least anomalous values of our example are colored red and green, respectively.

By *contextual attributes*, we mean attributes which identify and relate individual items of a dataset, such as the position in spatial data, the index in sequential data, or the vertex in graph data. These may be thought of as 'tags' for each item in a dataset, and must be unique. *Behavioral attributes*, are any other attributes. These are ideally relevant only to the anomaly measure.

Accordingly, we will henceforth assume that D may be decomposed as $D = \mathcal{P}(C \times B)$, where C and B are (application-specific) sets of contextual and behavioral data, respectively. In our example application, these may be represented as $C = \mathbb{N}^2$ (capturing the two-dimensional nature of the data) and $B = \mathbb{R}$.

The sub-methods may then be replaced as follows:

- Behavioral data should be irrelevant when selecting candidate anomalies, so α may be replaced by
$$\alpha' : \mathcal{P}(C) \to \mathcal{P}(\mathcal{P}(C)),$$
which operates on contextual data rather than on the full dataset.
- Similarly, β may be replaced by
$$\beta' : \mathcal{P}(C) \times \mathcal{P}(C) \to \mathcal{P}(C).$$
- When targeting point or contextual anomalies, γ may be replaced with a function
$$\gamma' : B \times \mathcal{P}(B) \to A,$$
that maps behavioral attributes of the candidate item and the context to an anomaly score.
When targeting collective or collective contextual anomalies, however, both contextual and behavioral aspects are likely to be relevant when computing the anomaly measure. Thus, replacing γ with sub-methods operating on either C or B is not feasible.
A better approach would be to break γ apart into smaller sub-methods, isolating the relation of contextual and behavioural considerations to a single, constrained sub-method.
Many anomaly measures compare one feature to a set of similar features, and are not formulated to operate on contextual data. For these, γ may be seen as encoding two responsibilities: extracting features from candidate anomalies and contexts, and comparing these to form anomaly scores.
These responsibilities may be encoded as (where F is some method-specific set of features)
$$\varepsilon : D \times D \to F \times \mathcal{P}(F), \text{ and}$$
$$\zeta : F \times \mathcal{P}(F) \to A.$$
Note that the anomaly measure ζ is not coupled to either C or B, so it is independent of D (as long as the features it operated on can be extracted).

In turn, ε may be seen as encoding two responsibilities: breaking the context up into a set of items, and extracting a single feature from each such item.

These responsibilities may be encoded separately as

$$\eta : \mathcal{P}(C) \to \mathcal{P}(\mathcal{P}(C))$$

(note the similarity to α'), and

$$\theta : D \to F.$$

- Behavioral data should be irrelevant when aggregating anomaly scores, so δ may be replaced by

$$\delta' : \mathcal{P}(\mathcal{P}(C) \times A) \to S.$$

If S is known, δ' may in turn be replaced further.

Reasonably, any S should involve assigning labels or scores either to individual items or to subsets of the data, so we may assume that $S = \mathcal{P}(G \times L)$, where either $G = C$ or $G = \mathcal{P}(C)$, and L is some set of labels.

When $S = \mathcal{P}(C \times L)$, and all candidate candidate anomalies are singleton sets (i.e. when point or contextual anomalies are targeted) δ' may be set to

$$\delta'(\{(\{c_1\}, a_1), \dots\}) = \{(c_1, \iota(a_1)), \dots\}$$

for some function $\iota : A \to L$.

Typically, either $A = L = \mathbb{R}$, in which case ι may be set as the identity function, or $A = \mathbb{R}$ and $L = \{0,1\}$, in which case ι may be set as a threshold function.

Analogously, when $S = \mathcal{P}(\mathcal{P}(C) \times L)$, δ' may be set to

$$\delta'(\{(C_1, a_1), \dots\}) = \{(C_1, \iota(a_1)), \dots\}.$$

Finally, when $S = \mathcal{P}(C \times L)$ and there are non-singleton candidate anomalies, δ' may be set to

$$\delta'(\{(C_1, a_1), \dots\}) = \left\{ (c_j, \kappa(A_j)) \mid c_j \in \bigcup C_i \right\},$$

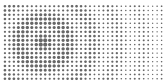
where $A_j = \{a_k \mid c_j \in C_k\}$, i.e. for each data item, the anomaly scores for all candidate anomalies to which it belongs are aggregated through some function $\kappa : \mathcal{P}(A) \to L$.

The above sub-methods allow for decomposing methods to various degrees; i.e. a method may be specified as $(D, S, A, \alpha', \beta', \gamma', \delta')$, or $(D, S, F, A, \alpha', \beta', \eta, \theta, \zeta, \delta')$, et cetera. Crucially, it is an easy task to write software that constructs a corresponding method for any such combination[7], and by extension, software that can algorithmically find appropriate methods given a set of potentially applicable sub-methods.

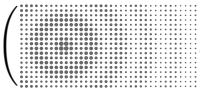
For a given choice of C, the number of interesting sub-methods can be rather limited (as we will see in Section 8). Thus, implementations of a few sub-methods may be used to handle a wide range of tasks.

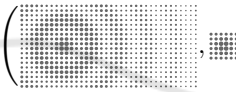
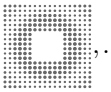
As an illustration of the sub-methods proposed above, we may consider how they may be used to replace the α, β, γ, and δ we applied to our example data. To indicate contextual data, we will use the pattern to the far right in Figure 2.

Our choice of α corresponds to an analogous α':

$\mathcal{P}(C) \ni$  $\overset{\alpha'}{\longmapsto} \{$ $, \dots\} \in \mathcal{P}(\mathcal{P}(C))$

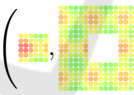
The same goes for our choice of β:

$\mathcal{P}(C)^2 \ni \left($  $, $ $\right) \overset{\beta'}{\longmapsto}$  $\in \mathcal{P}(C)$

$\left($  $, $  $\right) \overset{\beta'}{\longmapsto}$  $, \dots$

Our choice of γ corresponds to an ε that produces as features the mean value of each 6-by-6 region (so $F = \mathbb{R}$), and a ζ that computes the mean absolute difference between the feature extracted from the candidate anomaly and the features extracted from the context:

$D^2 \ni \left($  $\right) \overset{\varepsilon}{\longmapsto} ($ $, \{$ $, $  $, $  $\}) \in F \times \mathcal{P}(F)$

$\left($  $\right) \overset{\varepsilon}{\longmapsto} ($ $, \{$ $, $  $, \dots\}), \dots$

$F \times \mathcal{P}(F) \ni ($  $, \{$  $, $  $, $  $\}) \overset{\zeta}{\longmapsto}$  $\in A,$

$($  $, \{$  $, $  $, \dots\}) \overset{\zeta}{\longmapsto}$  $, \dots$

In turn, this ε corresponds to an η that extracts disjoint such square regions, and a θ that computes the mean value of its inputs:

$\mathcal{P}(C) \ni$  $\overset{\eta}{\longmapsto} \{$  $, $  $, $  $\} \in \mathcal{P}(\mathcal{P}(C)),$

 $\overset{\eta}{\longmapsto} \{$  $, $  $, $  $, \dots\}, \dots$

$D \ni$  $\overset{\theta}{\longmapsto}$  $\in F,$  $\overset{\theta}{\longmapsto}$  $, \dots$

Our δ may be replaced with an analogous δ':

$\mathcal{P}(\mathcal{P}(C) \times A) \ni \{($  $, $  $), \dots\} \overset{\delta'}{\longmapsto}$ $\in S$

[7] In the interest of saving space, we elide a precise formulation of how the sub-methods would be composed.

Finally, since the solution format is $S = \mathcal{P}(C \times L)$ for $L = A = \mathbb{R}$, and we are dealing with collective contextual anomalies, we may utilize κ. The candidate anomalies are disjoint, so κ should produce the single elements of the sets it receives:

$$\mathcal{P}(A) \ni \{\blacksquare\} \overset{\kappa}{\longmapsto} \blacksquare \in L, \quad \{\blacksquare\} \overset{\kappa}{\longmapsto} \blacksquare, \ldots$$

6 PARAMETRIC SUB-METHODS

Assuming that $D = \mathcal{P}(C \times B)$ and $S = \mathcal{P}(G \times L)$, the construction of a $m \in M = D \to S$ from e.g. some α', β', γ', and δ' may be seen as the application of a function

$$f(\alpha', \beta', \gamma', \delta') : A'_C \times B'_C \times \Gamma'_{C,B,A} \times \Delta'_{C,A,G,L} \to M,$$

where $A'_C = \mathcal{P}(C) \to \mathcal{P}(\mathcal{P}(C))$, et cetera.

Likewise, the construction of e.g. a γ' from some ε and ζ may be seen as the application of a function

$$g(\varepsilon, \zeta) : E_{C,B,F} \times Z_{F,A} \to \Gamma'_{C,B,A}.$$

Taking this approach one step further, we may consider *parametric sub-methods*—functions that take some tuple of parameters and produce a sub-method.

For instance, our choice of an α' that produces regions of size 6-by-6 may be seen as a special case of a parametric sub-method

$$\alpha'_{rect}(w,h) : \mathbb{N} \times \mathbb{N} \to (\mathcal{P}(\mathbb{N}^2) \to \mathcal{P}(\mathcal{P}(\mathbb{N}^2))) = A'_{\mathbb{N}^2}$$

that produces regions of width w and height h.

As we will see in Sections 8 and 9, parametric sub-methods naturally arise in applications, and are very helpful when formulating methods as well as when heuristically searching for appropriate methods.

7 HIGHER ORDER METHODS

Similarly, we may consider *higher order methods*, which map methods to methods.

For instance, consider the function

$$\tau : T_{C',B',C,B} \times M_{C,B,G,L} \times T_{G,L,G',L'} \to M_{C',B',G',L'},$$

defined by

$$\tau(t_i, m, t_o) = t_o \circ m \circ t_i,$$

which takes an *input transform*

$$t_i \in T_{C',B',C,B} = \mathcal{P}(C' \times B') \to \mathcal{P}(C \times B),$$

for some C', B', a method

$$m \in M_{C,B,G,L} = \mathcal{P}(C \times B) \to \mathcal{P}(G \times L),$$

and an *output transform*

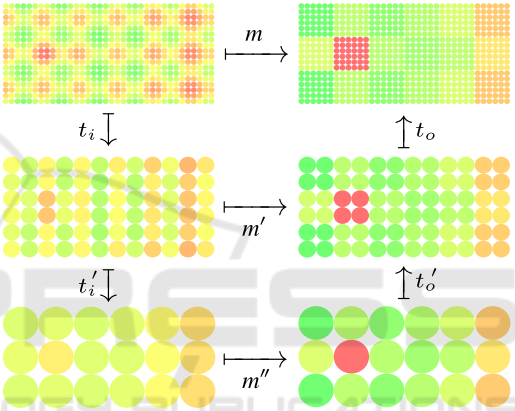
$$t_o \in T_{G,L,G',L'} = \mathcal{P}(G \times L) \to \mathcal{P}(G' \times L'),$$

for some G', L', and produces a method

$$m' \in \mathcal{P}(C' \times B') \to \mathcal{P}(G' \times L').$$

Many methods found in the literature involve a pre-analysis transformation of the data into some format more amenable to analysis, either through dimensionality reduction (Ding et al., 2008) or a change of data representation (i.e. of C or B) (Lin et al., 2007). Such methods may be accommodated through the use of τ together with appropriate t_i and t_o.

For instance, dimensionality reducing transformations may be applied to our example method to obtain equivalent methods that operate on a lower dimensionality:



Here, $t_i, t'_i, t_o, t'_o \in T_{\mathbb{Z}^2,\mathbb{R},\mathbb{Z}^2,\mathbb{R}}$ and

$$m = \tau(t_i, m', t_o) = \tau(t_i, \tau(t'_i, m'', t'_o), t_o).$$

Another interesting higher order method, which may be used to combine methods into ensembles, is $\mu : \mathcal{P}(M_{C,B,G,L}) \times U_{G,L} \to M_{C,B,G,L}$, given by

$$\mu(m,u)(d) = u(\{m_i(d) \mid m_i \in m\}),$$

where $u \in U_{G,L} = \mathcal{P}(\mathcal{P}(G \times L)) \to \mathcal{P}(G \times L)$ is some function that combines solutions.

Crucially, t_i, t_o and u may be used analogously to sub-methods to construct methods, either manually or algorithmically.

8 AN APPLICATION TO SEQUENCES

Anomaly detection tasks involving sequences[8] are commonly encountered in applications, and have

[8]We here consider only *regular* sequences, as opposed to irregular time series, for which $C = \mathbb{R}$, and which may be considered a type of one-dimensional spatial data.

been extensively studied. For sequences, we may let $C = \mathbb{N}$.

We will now illustrate how our approach may be used to formulate methods through an application to sequences. In the interest of saving space, we will restrict our attention to $S = \mathcal{P}(G \times L) = \mathcal{P}(\mathbb{N} \times \mathbb{R})$ (solutions which consist of real-valued per-element anomaly scores).

First, consider the following real-valued sequence (in $D = \mathcal{P}(\mathbb{N} \times \mathbb{R})$):

This sequence consists of a sinusoid with added noise, and two abnormalities: two extrema (in its latter half) and a trend of stray elements (beginning near its middle). Either abnormality may be considered an anomaly with regard to the (hypothetical) underlying application, so detecting either or both might be valuable.

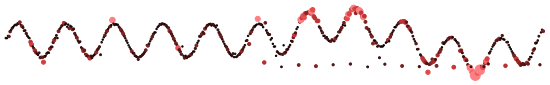
To detect the extrema, methods targeting point anomalies may be employed. As previously discussed, when point anomalies are targeted, and $G = C$, it suffices to specify γ' (an anomaly measure) and ι (a method of aggregating anomaly scores). We will restrict our consideration to $A = L = \mathbb{R}$, and may thus let $\iota(x) = x$.

A common choice of anomaly measures are k-nearest neighbor-based measures, which for any given candidate anomaly compute the mean distance to its k nearest elements (for some distance measure d). We may capture such measures through a parametric sub-method

$$\gamma'_{kNN}(k,d) : \mathbb{N} \times (\mathbb{R} \times \mathbb{R} \to \mathbb{R}) \to (\mathbb{R} \times \mathcal{P}(\mathbb{R}) \to \mathbb{R}),$$

where $\gamma'_{kNN}(k,d)(x,y)$ produces the mean of the k smallest values in $\{d(x,y_i) \mid y_i \in y\}$.

Applying $\gamma'_{kNN}(3,d_E)$ (for $d_E(x,y) = |x-y|$) to our sequence gives the following result (the anomaly score is indicated through color and point size; large, bright points indicate anomalous items):



As expected, it captures the extrema but not the trend of stray elements. The elements of this trend are contextual anomalies with regard to a *local context*, which consists of all elements within some distance m of a candidate anomaly (with respect to the sequence ordering). This may be captured through an appropriate β':

$$\beta'_{local}(m) \in \mathbb{Z} \to (\mathcal{P}(\mathbb{N}) \times \mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathbb{N})).$$

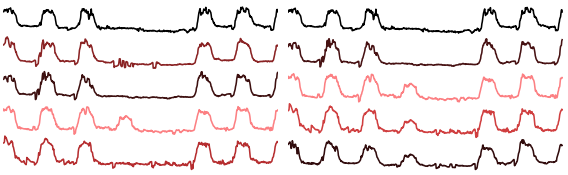
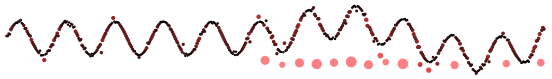


Figure 3: Left: results obtained by applying $\gamma'_{kNN}(3,d_{DTW})$ to the UCR power usage dataset (Chen et al., 2014). Right: results obtained by applying $\gamma'_{kNN}(3,d_{DTW})$ and $\beta'_{novelty}$ to a variant of the same data, where at a certain point, an artificial anomaly has been superimposed on subsequent sequences.

Applying $\beta'_{local}(10)$ together with $\gamma'_{kNN}(3,d_E)$ to the sequence gives the following result:



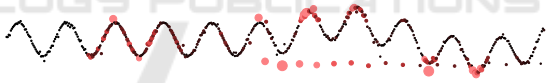
Capturing the entirety of this trend might not be desirable; in some applications, *novelties*—such as the onset of such trends—are more interesting.

Novelties can be captured through a *novelty context*

$$\beta'_{novelty} : \mathcal{P}(\mathbb{N}) \times \mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathbb{N}),$$

where $\beta'_{novelty}(d,c)$ produces all elements in d that come before the elements of c (with respect to the sequence ordering).

Replacing $\beta'_{local}(10)$ with $\beta'_{novelty}$ gives the following result:



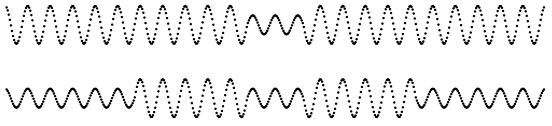
It should be noted that β'_{local} and $\beta'_{novelty}$ are both special cases of a more general parametric $\beta'_{asym}(b,a)$, which produces as context the b and a elements before and after the candidate anomaly.

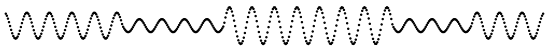
The sub-methods illustrated above may just as well be applied to sequences of other types of elements. For instance, consider an application involving sequences of real-valued vectors of some fixed length (i.e. $D = \mathcal{P}(\mathbb{N} \times \mathbb{R}^n)$), as illustrated in Figure 3. Here, point and contextual anomalies may be captured through e.g. $\gamma'_{kNN}(3,e_{DTW})$, where

$$e_{DTW} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$$

is the *dynamic time warp* distance (Berndt and Clifford, 1994).

Now consider the following three sequences:





Here, the top sequence contains a collective anomaly (at its center), the middle sequence contains a (local) contextual collective anomaly (also at its center), and the bottom sequence contains a few change points (which may be considered contextual collective anomalies with respect to a novelty context).

For these anomalies to be detectable, the candidate anomalies under consideration should be subsequences of the original sequence. To this end, we may employ an appropriate α', e.g.

$$\alpha'_{win}(w,s) : \mathbb{N} \times \mathbb{N} \to (\mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathcal{P}(\mathbb{N})))$$

where $\alpha'_{win}(w,s)$ selects every sth subsequence of width w:

$$\alpha'_{win}(w,s)(\{c_1,\dots\}) = \{\{c_1,\dots,c_w\},\{c_{1+s},\dots\}\dots\}.$$

To form an appropriate anomaly measure, we may employ

$$\eta_{win}(w,s) : \mathbb{N} \times \mathbb{N} \to (\mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathcal{P}(\mathbb{N}))),$$

defined identically to α'_{win}, together with

$$\theta_{vec}(n) : \mathbb{N} \to (\mathcal{P}(\mathbb{N} \times \mathbb{R}) \to \mathbb{R}^n)$$

defined by $\theta_{vec}(n)(\{(i,x_i),\dots\}) = [x_i,\dots,x_{i+n-1}]$, and

$$\zeta_{kNN}(k,d) : \mathbb{N} \times (\mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}) \to (\mathbb{R}^n \times \mathcal{P}(\mathbb{R}^n) \to \mathbb{R})$$

defined analogously with $\gamma'_{kNN}(k,d)$.

Finally, since $S = \mathcal{P}(\mathbb{N} \times \mathbb{R})$, some κ must be employed, e.g. $\kappa_{mean}(x) = \sum x_i/|x|$.

Applying e.g. $\alpha'_{win}(40,5)$, $\eta_{win}(40,5)$, $\theta_{vec}(40)$, $\zeta_{kNN}(3,e_{DTW})$, and κ_{mean} to our first sequence gives the following result:

Combining the above sub-method choices with $\beta'_{local}(75)$ results in a method that captures the anomaly in the middle sequence:

Finally, using $\beta'_{novelty}$ gives a method that captures novel change points in the last sequence:

While there are countless potentially interesting anomaly measures (i.e. γ' or ζ) to apply to sequences, the choices of other sub-methods are rather limited.

For methods that involves contiguous subsequences and contexts (likely a vast majority of interesting methods) it seems that the only reasonable approach would be to employ α'_{win}, β'_{asym} and η_{win} (when applicable).

While θ_{vec} handles behavioral data, and is thus technically dependent on B, its results are not affected by the individual behavioral values, and it could be extracted into a more portable sub-method, independent of C. This would likely be involved in most interesting methods involving sequences.

Finally, there are only a few interesting choices of κ (e.g. it could produce the mean, median, maximum, minimum of its input values).

Thus, these sub-methods may be considered to fairly exhaustively cover anomaly detection tasks in sequences (with the exemption of γ'/ζ, transforms, and ensemble methods). It should further be noted that since γ' and ζ are formulated independently of C, the same implementation of γ' or η may be used for sequences, grids, graphs, et cetera, as long as an appropriate θ is provided.

9 OPTIMIZATION

The application-agnostic and modular nature of our formalism enables the construction of software that heuristically searches for appropriate methods. Given any collection of sub-method implementations, together with some means of assessing its associated methods—e.g. a function $e : M \to \mathbb{R}$—we may find optimal methods by iteratively constructing and evaluating sub-method combinations.

One way to construct such an e is to employ a set $T \subset D \times S$ of labeled training data, together with some dissimilarity measure for solutions $e' : S \times S \to \mathbb{R}$, to form

$$e(m) = \sum_{(d_i,s_i)\in T} e'(m(d_i),s_i).$$

This function provides us with a convenient means of evaluating methods. Software that implements e can be used to easily evaluate and compare methods (especially if bundled with a collection of sub-method implementations) for novel applications.

Furthermore, it gives rise to a supervised, application-agnostic optimization problem—minimizing e given a set of (potentially parametric) sub-methods. If some means of efficiently solving this problem could be found, the task of finding appropriate methods for any given application could be reduced to selecting appropriate sets of candidate sub-methods and training data.

To illustrate this approach, we implemented a rudimentary solver for the optimization problem. This solver takes a collection of parametric sub-methods (with an ordered or unordered set of candi-

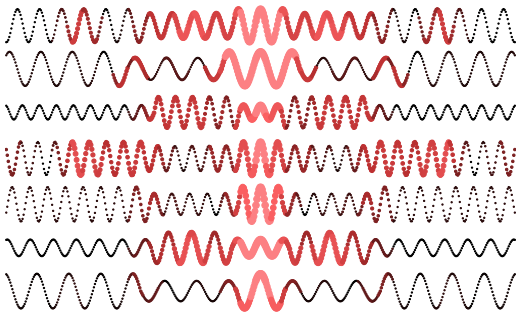


Figure 4: A few sample sequences from the evaluation data, with anomaly scores taken from a method with a low evaluation error (6.4), corresponding to $\alpha'_{win}(30,10)$, $\beta'_{local}(100)$, $\eta_{win}(30,5)$, $\theta_{vec}(30)$, $\zeta_{kNN}(\delta_{DTW},1)$, and κ_{mean}.

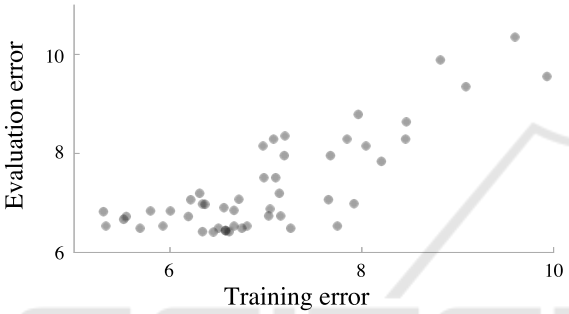


Figure 5: Average training vs evaluation error for 50 solver runs, with 10 training items and 100 and evaluation items.

date values for each parameter) and a set $T \subset \mathcal{P}(C \times [0,1])$ of training data. It uses the Euclidean distance (with a prior rescaling of the anomaly scores to $[0,1]$) as e'.

The solver employs a naïve, two-phase optimization heuristic: In the first phase, the solver evaluates all valid combinations of sub-methods. For each such combination, it randomly samples the parameter space (the product of the sets of sub-method parameter values) a fixed number of times, and evaluates each resulting method on the training data.

In the second phase, the solver uses hill climbing to calibrate the sub-method combination that produced the lowest error in the first phase[9].

We applied this solver to a procedurally generated data set consisting of real-valued sequences with collective contextual anomalies[10], as illustrated in Figure 4.

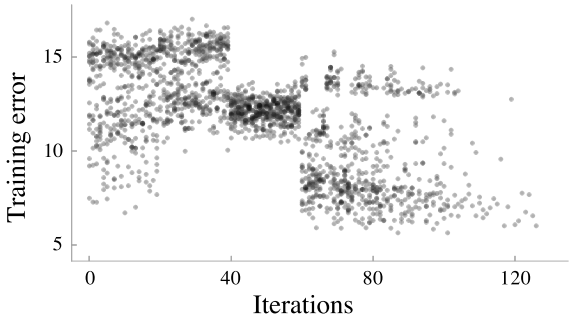


Figure 6: The training error at each iteration for a set of solver executions. Each sub-method combination is sampled 20 times before the solver switches to hill climbing.

We used the sub-methods presented in the previous section[11] and a set of 20 randomly sampled training items, let the solver take 20 random samples of each valid sub-method combination, and repeated the experiment 50 times. The resulting methods were then evaluated on a set of 100 items.

As seen in Figure 5, a large share of the resulting methods seem to perform close to optimally. The solver occasionally gets stuck in local minima, producing poorly performing methods. Considering the simplistic nature of the solver, this is hardly surprising, and it is likely that a more sophisticated solver would have performed better. The per-iteration training data error for 20 experiments is shown in Figure 6, and a few solutions produced by one method with a low evaluation error is shown in Figure 4.

10 CONCLUSIONS

We have introduced an application-agnostic approach to anomaly detection, in which anomaly detection methods are treated as formal objects that may be decomposed and recombined.

We have applied this formalism to sequences, showing that it may be used to easily express a wide range of anomaly detection tasks for this type of data.

Finally, we have demonstrated that our approach may be used to construct application-agnostic soft-

[9]Specifically, by starting at the point (out of those sampled) with the smallest error, and iteratively—until a (local) minimum is found—evaluating all adjacent points (changing one parameter at a time) and moving to the one with the lowest error.

[10]Specifically: 500-element, sinusoidal real-valued sequences with an angular frequency $\omega \in \mathcal{U}(1,2)$ and two distinct amplitudes a_1, a_2 (where $a_1 = 1$, and a_2 is $\mathcal{U}(1.3,1.7)$ with probability 0.5 and $\mathcal{U}(0.3,0.7)$ with probability 0.5), arranged in a *a-b-c-b-a* pattern, where the width of the *c* region is $\mathcal{U}(15,30)$ (rounded so that the amplitude transition happens at the nearest sign change), and the width of the *b* regions is $\mathcal{U}(80,100)$ (also rounded).The labels were set to 1 in the anomalous regions and 0 elsewhere.

[11]Specifically, $\alpha'_{win}(w,s)$, $\beta'_{local}(m)$, $\beta'_{novelty}$, the trivial β' used for collective anomalies, $\eta_{win}(w,s')$, $\theta_{vec}(w)$, and $\zeta_{kNN}(d,k)$, for $s,s' in \{5,10,\ldots,25\}$, $w \in \{30,35,\ldots,60\}$, $k \in \{1,2,\ldots,5\}$, $m \in \{80,90,\ldots,130\}$, $d \in \{d_E, d_{DTW}\}$

ware that facilitates implementing and evaluating methods, and that can be used to automatically find appropriate methods (given labeled training data and a set of candidate sub-methods).

Future Work. We foresee several venues for future work.

First, there are plenty of interesting sub-methods, transforms, ensemble methods, and non-sequence types of data (e.g. graphs, spatial data) to which our formalism could be extended. There is work to be done both in terms of studying these and in terms of creating flexible and efficient implementations.

There is also work to be done on efficiently solving the optimization problem outlined in Section 9; we have demonstrated that it may solved for simple tasks, but it remains to be seen if it can be effectively solved for real-world tasks.

Finally, modifying or extending our formalism could be valuable. For instance, associating additional information with sub-methods could enable algorithms that can optimize or approximate the resulting methods.

REFERENCES

Abraham, B. and Box, G. E. (1979). Bayesian analysis of some outlier problems in time series. *Biometrika*, 66(2):229–236.

Abraham, B. and Chuang, A. (1989). Outlier detection and time series modeling. *Technometrics*, 31(2):241–248.

Agyemang, M., Barker, K., and Alhajj, R. (2006). A comprehensive survey of numeric and symbolic outlier mining techniques. *Intelligent Data Analysis*, 10(6):521–538.

Basu, S. and Meckesheimer, M. (2007). Automatic outlier detection for time series: an application to sensor data. *Knowledge and Information Systems*, 11(2):137–154.

Berndt, D. J. and Clifford, J. (1994). Using dynamic time warping to find patterns in time series. In *KDD workshop*, volume 10, pages 359–370.

Chandola, V. (2009). *Anomaly detection for symbolic sequences and time series data*. PhD thesis, University of Minnesota.

Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys (CSUR)*, 41(3):15.

Chandola, V., Banerjee, A., and Kumar, V. (2012). Anomaly detection for discrete sequences: A survey. *Knowledge and Data Engineering, IEEE Transactions on*, 24(5):823–839.

Chen, Y., Keogh, E., Hu, B., Begum, N., Bagnall, A., Mueen, A., and Batista, G. (2014). The ucr time series classification archive. www.cs.ucr.edu/ẽamonn/time_series_data/. Accessed: 2014-09-13.

Ding, H., Trajcevski, G., Scheuermann, P., Wang, X., and Keogh, E. (2008). Querying and mining of time series data: experimental comparison of representations and distance measures. *Proceedings of the VLDB Endowment*, 1(2):1542–1552.

Etsy (2015). Etsy Skyline. github.com/etsy/skyline. Accessed: 2015-02-10.

Fox, A. J. (1972). Outliers in time series. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 350–363.

Fu, A. W.-C., Leung, O. T.-W., Keogh, E., and Lin, J. (2006). Finding time series discords based on Haar transform. In *Advanced Data Mining and Applications*, pages 31–41. Springer.

Fu, T.-c. (2011). A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24(1):164–181.

Galeano, P., Peña, D., and Tsay, R. S. (2006). Outlier detection in multivariate time series by projection pursuit. *Journal of the American Statistical Association*, 101(474):654–669.

Hodge, V. J. and Austin, J. (2004). A survey of outlier detection methodologies. *Artificial Intelligence Review*, 22(2):85–126.

Keogh, E., Lin, J., and Fu, A. (2005). Hot sax: Efficiently finding the most unusual time series subsequence. In *Data mining, fifth IEEE international conference on*.

Keogh, E., Lin, J., Lee, S.-H., and Van Herle, H. (2007). Finding the most unusual time series subsequence: algorithms and applications. *Knowledge and Information Systems*, 11(1):1–27.

Lazarevic, A., Ertöz, L., Kumar, V., Ozgur, A., and Srivastava, J. (2003). A comparative study of anomaly detection schemes in network intrusion detection. In *SDM*, pages 25–36.

Lin, J., Keogh, E., Wei, L., and Lonardi, S. (2007). Experiencing sax: a novel symbolic representation of time series. *Data Mining and Knowledge Discovery*, 15(2):107–144.

Ma, J. and Perkins, S. (2003). Online novelty detection on temporal sequences. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 613–618.

Markou, M. and Singh, S. (2003a). Novelty detection: a review, part 1: statistical approaches. *Signal processing*, 83(12):2481–2497.

Markou, M. and Singh, S. (2003b). Novelty detection: a review—part 2:: neural network based approaches. *Signal processing*, 83(12):2499–2521.

Phua, C., Lee, V., Smith, K., and Gayler, R. (2010). A comprehensive survey of data mining-based fraud detection research. *arXiv preprint arXiv:1009.6119*.

Tsay, R. S., Peña, D., and Pankratz, A. E. (2000). Outliers in multivariate time series. *Biometrika*, 87(4):789–804.

Twitter (2015). AnomalyDetection R Package. github.com/twitter/anomalydetection. Accessed: 2015-02-10.