Behavior Sensitive Hashing

Rohith Prakash

Abstract

In this paper, we study the problem of real-time anomaly detection of intelligent malware samples disguised within benign applications. We demonstrate that commonly used Dynamic Time Warping (DTW) distance is not suitable on time series of system resource traces when malware samples dynamically adapt their behavior to evade detection. To deal with malware samples in real-time which attempt to hide within benign behavior, we propose a new LSH-based scheme that has low hardware and complexity overhead, iteratively hashes time series based on *behavioral patterns*, triangulates hashing buckets to better categorize behavior, and is able to categorize new, previously unobserved behavior.

1 Introduction

Time series are used almost ubiquitously to represent time-based measurements in various fields. However, it is a well known problem that when these times series represent behavior-related observations of complex systems, unintended information about the system may be leaked through this channel [1, 2]. This problem of side channel leakage has been extensively studied in the past by, and others have proposed correlation-based measures to quantify the amount of leakage [3, 4].

In this paper, we consider the problem of efficiently learning and classifying the behavior of side channels through observed time series using probabilistic hashing techniques. We formally define the probability of information leakage on a set of such time series traces observed at a fixed granularity with respect to a distance measure. By expanding on that intuition, we propose a *ranked behavior-sensitive hashing* scheme based on previous locality-sensitive hashing schemes [5, 6, 7] that exploits the probability of information leakage for nearest neighbor computations in anomaly detection and behavior classification. We believe that these contributions lead to a novel, strong characterization of leakage over side channels, giving rise to a new notion of *dimensionality* of a channel. We show that this can determine the effectiveness of leakage prevention schemes as well as discuss possibilities of projecting onto higher and lower dimensional spaces to improve anomaly detection or improve the effectiveness of leakage prevention schemes.

2 Background and Definitions

Previous work in time series anomaly detection has largely focused on Euclidean distance and Dynamic Time Warping (DTW) distance [**?**, **?**, **?**, **?**, **?**]. However, we find that there are drawbacks of limiting evaluation to these two measures only. DTW is not a proper metric as it is not sub-additive; this is a result of DTW treating one time series as non-linear time-stretched of the other. In the context of anomaly detection, this limits the ability to mark a small, unexpected change in behavior when compared with other distance measures. Additionally, both of these measures operate under the assumption that observed time series which are similar will be of the same scale. Two time series which exhibit similar "behavior" but take values of a slightly different scale will not be marked as being similar by either measure. Changes in background system activity could therefore affect the ability of these two measure to detect true anomalous activity.

2.1 Information Leakage

Side channel exploitation, anomaly detection, and covert channel communication are problems of detecting or exploiting leakage over information channels. Side and covert channels exists when observable differences in system behavior occur as the result of actions performed by a *victim* or sending process. These attacks typically involve an adversary learning secret information over the channel, based on the behavior of a victim process. Anomaly detection, on the other hand, involves detectors running on a system analyzing and categorizing observed behavior in real time. In this setting, a malicious program leaks information about its behavior through an observed channel.

We consider a single information channel as a sequence of observations of a system resource — a *time series* of resource observations. For example, the *trace* of system calls on a system over time is an n-dimensional time series, where each observation determines the number of times each of the n system calls was invoked.

Our primary observation with regards to time series leakage is that information may only be learned from time series observations if the underlying distributions are distinguishable.

Definition 1 (Distribution-based behavior leakage)**:** Consider two distinct program behaviors x and x' and resulting time series for each behavior drawn from D_x and $D_{x'}$ respectively. Let $t(x)$ and $t(x')$ be two time series resulting from behaviors x, x', drawn from $D_x, D_{x'}$ respectively. Observing $t(x)$ and $t(x')$ can only leak information about x and x' if D_x, $D_{x'}$ are statistically distinguishable.

While Definition 1 is useful when one can carefully observe many samples from different distributions to assess the distinguishability of the underlying distributions, it is difficult to use in practice. Instead, we propose a slightly different definition of behavior dissimilarity:

Definition 2 (Time series behavior leakage)**:** Consider two distinct program behaviors x and x' with output distributions D_x and $D_{x'}$.

We say that program behaviors x and x' may **leak behavior information** regarding x and x' if there exists a distance measure $d(\cdot, \cdot)$ such that $\forall\ t_{x_i}, t_{x_j} \sim D_x$ and $t_{x'_i}, t_{x'_j} \sim D_{x'}$, the following hold:

$$\begin{aligned} &\mathbf{E}[d(t_{x_i}, t_{x'_i})] > \mathbf{E}[d(t_{x_i}, t_{x_j})], \\ &\mathbf{E}[d(t_{x_i}, t_{x'_i})] > \mathbf{E}[d(t_{x'_i}, t_{x'_j})]. \end{aligned} \tag{1}$$

Definition 2 describes time series behavior with respect to a specific distance function applied on observation points. If time series resulting from behavior x can be separated from time series from x' by a distance of more than r, there is potential information leakage through this channel.

2.2 Hashing

Hashing has long been used as a method of easing the curse of dimensionality for tasks such as clustering on a large set of high-dimensional data [8, 9, 10]. Exploiting the probabilistic nature and the computational efficiency of hashing enables approximations to difficult high-dimensional problems quickly and in real-time.

Consider the space of time series S and a distance function d on S. A LSH family is defined as such:

Definition 3 (Hash family)**:** A hash family $\mathcal{H} = \{h : S \to U\}$ is called (r_1, r_2, p_1, p_2)-sensitive w.r.t. $d(\cdot, \cdot)$ if for any $x, y \in S, h \in \mathcal{H}$

(i) If $d(x, y) \leq r_1$, then $\Pr[h(x) = h(y)] \geq p_1$

(ii) If $d(x, y) \geq r_2$, then $\Pr[h(x) = h(y)] \leq p_2$

Such a family is only interesting if $p_1 > p_2$. To increase the effectiveness of an LSH technique, the gap between p_1 and p_2 may be *amplified*:

Definition 4 (LSH amplification)**:** Consider a (r_1, r_2, p_1, p_2)-sensitive hash family $\mathcal{H}$ w.r.t $d(\cdot, \cdot)$. The LSH hash family can be amplified in the following ways:

(i) **AND construction:** Define $\mathcal{H}' = \{h' : S \to U^r\}$ such that $h' = [h_1, \ldots, h_r] \subset \mathcal{H}$. $h'(x) = h'(y)$ iff $h_i(x) = h_i(y)\ \forall\ h_i \in h'\ \mathcal{H}'$ is a (r_1, r_2, p_1^r, p_2^r)-sensitive LSH family

(ii) **OR construction:** Define $\mathcal{H}' = \{h' : S \to U^b\}$ such that $h' = [h_1, \ldots, h_b] \subset \mathcal{H}$. $h'(x) = h'(y)$ iff $h_i(x) = h_i(y)$ for any $h_i \in h'\ \mathcal{H}'$ is a $(r_1, r_2, 1 - (1 - p_1)^b, 1 - (1 - p_2)^b)$-sensitive LSH family

(iii) **AND-OR composition:** The composition of and with or constructions defines a $(r_1, r_2, 1 - (1 - p_1^r)^b, 1 - (1 - p_2^r)^b)$-sensitive LSH family

We consider hash families $\mathcal{H} = \{h : S \to \mathbb{R}_+^n\}$ such that $h \in \mathcal{H}$ approximates $d(\cdot, \cdot)$ on time series in S. Intuitively, time series which are "closer" to each other (as defined by d) will be harder to distinguish by any arbitrary classifier and thus present fewer possibilities for information leakage.

2.3 Distance Measure

Denote the space of a time series as S, and define a function d:

$$\begin{aligned} d\colon S^2 &\to \mathbb{R}_+ \\ (x, y) &\mapsto r \end{aligned} \tag{2}$$

where d maps two time series to a non-negative real number that represents some notion of distance between them. Additionally, we may require d to be sub-additive:

$$d(x, y) \leq d(x, z) + d(z, y) \tag{3}$$

The function $d(\cdot, \cdot)$ defines leakage in our threat model. To give intuition behind this, we consider an arbitrary classification attack on a set of time series. Applying Definitions 1 and 2, there is potential for leakage by observing the resulting time series if, for some function d, there exists separation by $d(\cdot, \cdot)$ between time series of differing classes.

2.4 Kernel Transforms

We have so far defined leakage with respect to an arbitrary, but fixed, distance measure. However, we now consider kernel transforms to define higher dimensional distance measures without explicitly defining the embedding space [**?**]. This methodology allows us to determine a leakage-sensitive distance measure with computational efficiency.

Kernel transforms have been used extensively in machine learning problems, especially in support vector machine (SVM) classifiers. For example, a kernel transform allows the use of user-specified similarity functions that may be computationally intractable to fully define. However, kernel transforms have also been recently applied to hashing problems in order to tackle even higher dimensional similarity problems [5, 14, 6]. A kernel function $\kappa(\cdot,\cdot)$ thus defines a new similarity measure on a higher dimensional space over which we would not otherwise be able to efficiently hash.

The following definitions let us formally define kernel transforms on time series:

Definition 5 (Hilbert space)**:** A vector space H over a field F with an inner product $\langle\cdot,\cdot\rangle_H : H \times H \to F$ that also defines a complete[1] metric space is called a Hilbert space.

The key property of Hilbert spaces we wish to leverage is the norm induced by the inner product $\langle\cdot,\cdot\rangle_H$. This inner product defines the higher order distance measure we wish to use on the raw observation space.

Definition 6 (Kernel transform)**:** Let X be an arbitrary space and H be a Hilbert space with inner product $\langle\cdot,\cdot\rangle_H$. $\kappa(\cdot,\cdot) : X \times X \to \mathbb{R}$ is a kernel transform if $\kappa(x,y) = \langle\phi(x),\phi(y)\rangle_H$ for some $\phi : X \to H$.

Note that in Definition 6, the mapping function ϕ need not be explicitly defined. In fact, κ being a positive-semidefinite function (or matrix over discrete spaces) implies the existence of a satisfactory function ϕ. Thus, we can consider arbitrary higher-order distance measures as any positive-semidefinite mapping $\kappa : X \times X \to H$ guarantees a similarity measure in H.

Definition 7 (Reproducing kernel Hilbert space (RKHS))**:** Let H be a Hilbert space of real-valued functions on an arbitrary set X. H is a reproducing kernel Hilbert space if there exists a **reproducing kernel**, $\kappa_x\ \forall x \in X$, where $f(x) = \langle f,\kappa_x\rangle_H\ \forall\ f \in H$.

Note that $\kappa(x,y) = \langle\kappa_x,\kappa_y\rangle_H$, and thus the kernel transform in Definition 6 defines a RKHS. This demonstrates that we can consider arbitrary higher-order similarity measures using kernel functions on the space of observed samples. Furthermore, we can construct proper distance measures from the norm induced by the inner product on a RKHS.

Definition 8 (Hilbert norm-induced distance)**:** Given an input set X and a reproducing kernel κ for RKHS H: $\kappa(\cdot,\cdot) : X \times X \to \mathbb{R}$, define a distance measure on X by:

$$\begin{aligned} d(x,y)^2 &= \|\phi(x)-\phi(y)\|_H^2 = \langle\phi(x)-\phi(y),\phi(x)-\phi(y)\rangle_H \\ &= \langle z,z\rangle_H = \kappa(\phi^{-1}(z),\phi^{-1}(z)) \end{aligned} \tag{4}$$

Note that we have still not explicitly defined the feature map $\phi(\cdot)$. Instead, we contend that such a function exists that allows our transform to take place, which we prove in

[1] A space X is complete if every Cauchy sequence converges in X. A Cauchy sequence is a sequence $\{x_n\}_{n\in\mathbb{N}}, x_n \in X$ with $\lim_{(m,n)\to\infty} |x_m - x_n| = 0$.

Section **??**. In certain instances, we may wish to actually define an explicit mapping $\phi(\cdot)$. In cases such as these, a distance measure induced by the kernel will arise much more simply.

2.5 (Weak) Derivatives on Time Series

Consider the following definitions of discrete time derivatives.

Definition 9 (Discrete-time derivatives)**:** Consider a discrete function $f\colon \mathbb{Z}^+ \to \mathbb{R}$ with samples h apart. For $n \in \mathbb{Z}$, the following derivatives are defined.

(i) **Forward difference:** $f'(n) = \frac{f(n+1)-f(n)}{h}$

(ii) **Backward difference:** $f'(n) = \frac{f(n)-f(n-1)}{h}$

(iii) **Central difference:** $f'(n) = \frac{f(n+1)-f(n-1)}{2h}$
This is the average of the forward and backward differences.

Note that these definitions imply continuous differentiability of a time series t represented as a function $f(n)$. We apply these definitions in Section 2.6, where it is necessary to consider derivatives of elements in a Hilbert space for application to distance measures.

2.6 Seminorms in Hilbert Spaces

We have so far defined machinery which will, given a reproducing kernel κ from the observation space, permit us to calculate higher order distances using the implicit feature map ϕ induced by the kernel function. Recall the kernel transform:

$$\kappa(x, y) = \langle \phi(x), \phi(y) \rangle_H \tag{5}$$

From this, the induced distance measure is:

$$d(x, y) = \|\phi(x) - \phi(y)\|_H \tag{6}$$

However, we may wish to understand behavior even more explicitly in this space by considering a functions derivatives. To this end, we consider the concept of *seminorms*.

Definition 10 (Seminorm)**:** A seminorm $\|\cdot\|_S$ on a space S satisfies:

(i) $\|ax\|_S = |a| \|x\|_S$ for scalar a

(ii) $\|x + y\|_S \leq \|x\|_S + \|y\|_S$ (triangle inequality)

This has some benefits for time series classification which we seek to exploit.

- **Shift-agnostic:** Let $f, g \in H$, a RKHS, and let $g = f + c$, where c is a constant, and let both f, g be n-differentiable. The norms $\|f\|_H, \|g\|_H$ will vary drastically depending on c. However, such shifting is not apparent with a derivative-based seminorm: $\|x\|_S = (\sum_{k=1}^{n} \left\|D^k x\right\|_H^p)^{\frac{1}{p}}$. Thus $\|g\|_S = (\sum_{k=1}^{n} \left\|D^k (f + c)\right\|_H^p)^{\frac{1}{p}} =$

$(\sum_{k=1}^{n} \left\|D^k f\right\|_H^p)^{\frac{1}{p}} = \|f\|_S$. Here, the operator D is understood in a weak sense (see Section 2.5).

- **Scaling behavior:** Consider the seminorm $\|x\|_S = \sum_{|\alpha|=1}^{n} (\|D^\alpha x\|_H^p)^{\frac{1}{p}}$. Let $f, g \in H$ be at least once differentiable with $g(x) = f(cx)$, c a scalar. Then, $\|g(x)\|_S = \|f(cx)\|_S = \sum_{|\alpha|=1}^{n} (\|D^\alpha f(cx)\|_H^p)^{\frac{1}{p}} = r^{p-n} \|f(x)\|_S$ where n is the dimension of H.

Expanding on Definition 8 to create a distance metric based on a seminorm, we consider the following derivative-based distance measure:

$$d(x, y) = \|\phi(x) - \phi(y)\|_S = \|D(\phi(x) - \phi(y))\|_H = \|\phi'(x) - \phi'(y)\|_H \tag{7}$$

If we are able to impose additional structure onto the feature map $\phi(\cdot)$, the seminorm definition has much greater value.

Suppose ϕ: $X \to H$ is a feature map from input space X to Hilbert space H, and that ϕ is surjective and once differentiable. Then, $\exists z$ such that $\phi'(x) - \phi'(y) = \phi(z)$ since H is a vector space closed under addition, and ϕ is surjective.

$$d(x, y) = \|\phi'(x) - \phi'(y)\|_H = \|\phi(z)\|_H = \sqrt{\langle \phi(z), \phi(z) \rangle_H} = \sqrt{\kappa_H(z, z)} \tag{8}$$

where κ_H is the reproducing kernel of H.

Note that while $d(\cdot, \cdot)$ as defined above is sub-additive [11], it is not identically zero and is thus not a proper metric: $g = f + c$ meets $d(f, g) = 0$ even when $f \neq g$.

Given a "nice" feature mapping ϕ from input space to a Hilbert space induced by a reproducing kernel, we can more intuitively and effectively categorize time series behavior and time series distance using seminorms in these spaces. However, this is not trivial, as Mercer's condition and theorem only guarantee the existence of ϕ given a positive semi-definite kernel and make no guarantees on its differentiability or surjectivity. We explore the potential for seminorm usage in the next section with an explicit kernel for anomaly detection.

3 Kernelized Hashing Model for Time Series

We now propose a specific hashing model for time series with the goal of anomaly detection.

Definition 11 (Property $\mathscr{A}$)**:** Let $\mathcal{H}$ be a $(2r, s, p, q)$-sensitive LSH family on X with distance measure $d(\cdot, \cdot)$. $\mathcal{H}$ has property $\mathscr{A}$ if the following holds: Fix $x \in X$ and construct $S = \{s \in X \mid d(x, s) \leq r\}$. Then, $\Pr[h(x_i) = h(x_j)] \geq p \ \forall x_i, x_j \in S, h \in \mathcal{H}$

Property $\mathscr{A}$ confers the notion that elements which lie within a ball of fixed radius should have high probability of hashing to the same value. Furthermore, such families may be *nested* with increasing values of n to form neighborhoods with different levels

of similarity. We now define what we need from a distance measure d to obtain this nesting property.

Claim 1: *An LSH family $\mathcal{H}$ with distance measure $d(\cdot,\cdot)$ on a set X has Property $\mathscr{A}$ if $d(\cdot,\cdot)$ is a sub-additive distance measure.*

Proof. Let $\mathcal{H}$ be an LSH family on X which is $(2r, s, p, q)$-sensitive and suppose $d : X \times X \to \mathbb{R}_+$ is sub-additive.

Fix $x \in X$ and construct $S = \{s \in X \mid d(x, s) \leq r\}$. Since d is a sub-additive distance measure, $d(x_i, x_j) \leq d(x_i, x) + d(x, x_j) = d(x, x_i) + d(x, x_j) \leq 2r \ \forall x_i, x_j \in S$. Thus by construction of $\mathcal{H}$, $\Pr[h(x_i) = h(x_j)] \geq p \ \forall x_i, x_j \in S, h \in \mathcal{H}$, and thus $\mathcal{H}$ has property $\mathscr{A}$. □

Property $\mathscr{A}$ additionally allows for a stratified LSH scheme using a set of LSH families index by a distance r which allows us to confer a notion of closeness between buckets of a lower LSH strata.

Definition 12 (Ranked LSH families)**:** A set of hash families $\{\mathcal{H}_r\}_{r\in R}$ is a ranked LSH family H_r has property $\mathscr{A}$ $\forall r \in R$. Denote such a set of families as (R, p_1, p_2)-sensitive LSH families $\mathcal{H}_R$.

This corroborates the notion of closeness to collision probabilities, which allows for the grouping of similar time series. Due to Property $\mathscr{A}$, we may apply iterative hashing scheme to *rank* the probabilities of closeness based on the varied parameter r. We discuss this in greater depth in Section 3.2

3.1 Kernel Model

Consider a space of samples $S = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$ with similarity measure $\kappa\colon S \times S \to \mathbb{R}$ defined by $\kappa(x, y) = \langle \phi(x), \phi(y) \rangle_H$, with H being a RKHS. We now consider the effects of a positive-semidefinite similarity measure $\kappa(\cdot,\cdot)$ as well as its corresponding distance measure $\tilde{\kappa}(x, y) = \|\phi(x) - \phi(y)\|_H$, the norm induced by the RKHS. Note that from Definition 6, we do not need to explicitly define the mapping $\phi(\cdot)$ to an RKHS.

Definition 13 (Kernelized LSH)**:** $\mathcal{H}_R = \{\mathcal{H}_r\}_{r\in R}$ is a (R, p, q)-sensitive ranked, kernelized LSH family if, for any $x, y \in S$:

(i) If $\tilde{\kappa}(x, y) \leq r, \Pr[h_r(x) = h_r(y)] > p$

(ii) If $\tilde{\kappa}(x, y) \geq r, \Pr[h_r(x) = h_r(y)] < q$

(iii) Property $\mathscr{A}$ holds for any LSH family $\mathcal{H}_{r_i} \in \mathcal{H}_R$

Denote such a family a **RKLSH** family.

We now consider how to apply a RKLSH family to time series for anomaly detection and construct such a family in Section 5.

3.2 Ranked Hashing by Iteration

Let $\mathcal{H}_R$ be a RKLSH family which approximates a measure on time series distance with R being a set of stratifying distance thresholds, and let S be the space of all possible time series.

Definition 14 (LSH neighborhood)**:** A **neighborhood** with respect to distance threshold $r_i \in R$ and LSH hash function $h \in \mathcal{H}_R$ is a set of points S such that $\forall s_i, s_j \in S, h(s_i) = h(s_j)$.

The goal of hashing the set of input points is to efficiently compute sets of *approximate nearest neighbors* (ANN) such that the behavior of each point can be classified. However, the construction of RKLSH we have previously defined provides extra structure which allows us to make even stronger similarity claims:

Definition 15 (Neighborhood rank)**:** Let X be an input space and $\mathcal{H}_R$ an RKLSH, with $R = \{r_1, r_2, \ldots\}$. Two data samples $x, y \in X$ have **rank** n if $n = \inf_i \{r_i \in R \mid \forall r_j \geq r_i,\ \exists h_{r_j} \in \mathcal{H}_{r_j}$ such that $h_{r_j}(x) = h_{r_j}(y)\}$.

We can thus apply the ranked hashing scheme to perform a stronger approximate nearest-neighbor calculation than with a single LSH family. The similarity of any two points within a neighborhood can be ranked to give an indication of the confidence a point truly falls within a given neighborhood and thus exhibits the given behavior.

Thus, by applying this tiered ANN calculation, we are able to *rank* the similarity of two elements by examining the threshold at which their hashes collide.

4 Anomaly Detection on Time Series

The above theory provides intuition that similar time series may be grouped together and ranked by their closeness via a kernel distance measure $\tilde{\kappa}$ [16]. In the context of anomaly detection, we can apply this scheme to a set of *normal* traces (time series of utilization, label encoded syscalls, etc.) to determine what thresholds and clusters constitute normal execution with a finer granularity.

Instead of a binary label of *normal* vs *anomalous*, we stratify applications by behavior exhibited during execution. Traces which do not match enough previously "normal" clusters over the set of thresholds may be considered to represent anomalous activity.

4.1 Distance Approximations in Real-Time

We have so far defined a kernelized hash family (Definition 13) over some arbitrary space S, which assumes that we treat an entire (finite) time series as a single element in this space and compute distances on these full time series. However, to compute distances between time series in real-time, we must make some adjustments to account for each time series being an incoming stream of data.

To determine the perceived behavior of time series in real-time with respect to a distance measure $\tilde{\kappa}(\cdot, \cdot)$, we *segment* the time series into overlapping windows of length k, with an overlap of $\lfloor \frac{k}{2} \rfloor$ points. We now construct a RKLSH family over the space of such k-length time series segments to satisfy the above theory.

In the context of real-time anomaly detection, this has the advantage of decomposing examined time series into finite time series of the same length, reducing computation by restricting time series length to a variable parameter, and allowing us to define *behavior over time* as we hash each segment independently.

Algorithm 1: Real-time Behavioral Hash for Time Series

Input : $t = (\ldots, t_i, \ldots) \in S$, real-time time series;
R, set of thresholds;
K, window sizes, w.r.t. R;
ϕ: $S \to H$, feature map;
a, number of hash functions;
$M \in \mathbb{N}$;
d: $\mathbb{R}^{\dim(H)} \times \mathbb{R}^{\dim(H)} \to \mathbb{R}^+$, distance metric;
$A_r(h)$, function on hash values for each $r \in R$. A_r maps known inputs to 1, unknown (anomalous) inputs to 0.

Output: Behavior hash value for time series at every $\lfloor \frac{k}{2} \rfloor$ points.
Signals anomalies when detected

for $k \in K$ **do**
  if `len`(t) mod $k = 0$ **then**
    Let r be the corresponding window size for k.
    Let $t_k = t[-k:]$
    $P = \{\pi_{h_a}(\phi(t_k))\}$ `/* project` $\phi(t_k)$ `onto` a `random hyperplanes` h_a `in` $\mathbb{R}^{\dim(H)}$ `*/`
    Define $v_P(r) = \min_{p \in P} [\frac{d(\mathbf{0}, p)}{r} \bmod M]$
    $D = \{v_P(r) \mid r \in R\}$ `/* min distance from origin to projected point in units` r `*/`
    Define $R_i = \{r_j \in R \mid j \geq i\}$
    if $\exists\ i$ *such that* $\forall r_j \in R_i, |\{A_r(v_P(r_j)) = 0\}| \geq |\frac{R_i}{2}|$ **then**
      Raise(*anomaly*)
    end
  end
end

5 Proposed Scheme

We now propose a specific distance measure and kernel transform to maximize the impact of our RKLSH algorithm for anomaly detection.

5.1 Seminorm Induced Space

Consider a Hilbert space H over finite-length, discrete-time time series equipped with the following norm:

$$\|x\|_H = \sqrt{\sum_i x_i^2 + \sum_i \left(D^1 x_i\right)^2} \tag{9}$$

Where $D^1 s$ represents the time series of point-wise first derivatives of s.

This formulation arises from discussion of seminorms (Section 2.6) for their scaling and shifting behavior. However, we show that this is in fact a proper norm and additionally explore the Hilbert space and inner product which induce this norm to derive an exact distance metric.

Claim 2: *The following seminorm is a proper norm:*

$$\|x\|_H = \sqrt{\sum_i x_i^2 + \sum_i \left(D^1 x_i\right)^2}$$

Proof.

$$\begin{aligned}
\|x+y\|_H^2 &= \sum_i \left(x_i + y_i\right)^2 + \sum_i \left(D^1(x_i + y_i)\right)^2 \\
&= \sum_i x_i^2 + \sum_i \left(D^1 x_i\right)^2 + \sum_i y_i^2 + \sum_i \left(D^1 y_i\right)^2 \\
&\quad + 2\sum_i (x_i y_i) + \sum_i \left(D^1 x_i D^1 y_i\right) \\
&\leq \|x\|_H^2 + \|y\|_H^2 + 2|\langle x, y\rangle_H| \\
&\leq \left(\|x\|_H + \|y\|_H\right)^2 \text{(by Cauchy-Schwartz)}
\end{aligned} \tag{10}$$

$$\begin{aligned}
\|ax\|_H &= \sqrt{\sum_i \left(a x_i\right)^2 + \sum_i \left(D^1 a x_i\right)^2} \\
&= \sqrt{a^2 \sum_i x_i^2 + a^2 \sum_i \left(D^1 x_i\right)^2} \\
&= |a| \, \|x\|_H
\end{aligned} \tag{11}$$

$$\|x\| = 0 \Leftrightarrow \sqrt{\sum_i x_i^2 + \sum_i \left(D^1 x_i\right)^2} = 0 \Leftrightarrow x = \mathbf{0} \tag{12}$$

□

Claim 3: *The norm in Equation 9 is naturally induced by an inner product of the following form:*

$$\langle x, y \rangle_H = \sum_i (x_i y_i) + \sum_i (D^1 x_i D^1 y_i) \tag{13}$$

Proof.

$$\sqrt{\langle x, x \rangle_H} = \sqrt{\sum_i x_i^2 + \sum_i \left(D^1 x_i\right)^2} = \|x\|_H$$

□

5.2 Kernel Transform

We now define a feature map from the raw observation space of time series S to the constructed Hilbert space H.

Claim 4 (Reproducing Kernel)**:** *Consider the feature map $\phi(\cdot)$ from S to H with norm as defined above:*

$$\begin{aligned} \phi \colon S &\to H \\ s &\mapsto (s, D^1 s) \end{aligned}$$

$\phi(\cdot)$ defines a reproducing kernel $\kappa \colon X \times X \to H$ with $\kappa(f, g) = \langle \phi(f), \phi(g) \rangle_H$ for $f, g \in X$ and $\|h\|_H = \sqrt{\sum_i h_i^2 + \sum_i \left(D^1 h_i\right)^2}$ for $h \in H$.

Proof. We first note that $\phi(\cdot)$ is a linear map:

$$\phi(ax + by) = (ax + by, D^1(ax + by)) = a(x, D^1 x) + b(y, D^1 y) = a\phi(x) + b\phi(y)$$

$\kappa(f, g) = \langle \phi(f), \phi(g) \rangle_H$ is an inner product on H, and therefore defines a unique positive-definite kernel (by the positive-definite property of the inner product and linearity of $\phi(\cdot)$). □

References

[1] Thomas Ristenpart, Eran Tromer, Hovav Shacham, and Stefan Savage. Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds. In Ehab Al-Shaer, Somesh Jha, and Angelos D. Keromytis, editors, *Proceedings of the 2009 ACM Conference on Computer and Communications Security, CCS 2009, Chicago, Illinois, USA, November 9-13, 2009*, pages 199–212. ACM, 2009.

[2] Yinqian Zhang, Ari Juels, Alina Oprea, and Michael K. Reiter. Homealone: Co-residency detection in the cloud via side-channel analysis. In *32nd IEEE Symposium on Security and Privacy, S&P 2011, 22-25 May 2011, Berkeley, California, USA*, pages 313–328. IEEE Computer Society, 2011.

[3] John Demme, Robert Martin, Adam Waksman, and Simha Sethumadhavan. Side-channel vulnerability factor: A metric for measuring information leakage. In *Proceedings of the 39th Annual International Symposium on Computer Architecture*, ISCA '12, pages 106–117, Washington, DC, USA, 2012. IEEE Computer Society.

[4] Tianwei Zhang, Fangfei Liu, Si Chen, and Ruby B. Lee. Side channel vulnerability metrics: The promise and the pitfalls. In *Proceedings of the 2Nd International Workshop on Hardware and Architectural Support for Security and Privacy*, HASP '13, pages 2:1–2:8, New York, NY, USA, 2013. ACM.

[5] B. Kulis and K. Grauman. Kernelized locality-sensitive hashing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(6):1092–1104, June 2012.

[6] Ke Jiang, Q. Que, and B. Kulis. Revisiting kernelized locality-sensitive hashing for improved large-scale image retrieval. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4933–4941, June 2015.

[7] Y. B. Kim, E. Hemberg, and U. M. O'Reilly. Stratified locality-sensitive hashing for accelerated physiological time series retrieval. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 2479–2483, Aug 2016.

[8] Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, STOC '98, pages 604–613, New York, NY, USA, 1998. ACM.

[9] Aristides Gionis, Piotr Indyk, and Rajeev Motwani. Similarity search in high dimensions via hashing. In *Proceedings of the 25th International Conference on Very Large Data Bases*, VLDB '99, pages 518–529, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

[10] Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab S. Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the Twentieth Annual Symposium on Computational Geometry*, SCG '04, pages 253–262, New York, NY, USA, 2004. ACM.

[11] M. Rosenlicht. *Introduction to Analysis*. Dover books on mathematics. Dover Publications, 1968.

[12] Young-Seon Jeong, Myong K. Jeong, and Olufemi A. Omitaomu. Weighted dynamic time warping for time series classification. *Pattern Recogn.*, 44(9):2231–2240, September 2011.

[13] Daniel Lemire. Faster retrieval with a two-pass dynamic-time-warping lower bound. *Pattern Recogn.*, 42(9):2169–2180, September 2009.

[14] D. C. Kale, D. Gong, Z. Che, Y. Liu, G. Medioni, R. Wetzel, and P. Ross. An examination of multivariate time series hashing with applications to health care. In *2014 IEEE International Conference on Data Mining*, pages 260–269, Dec 2014.

[15] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the Third Conference on Theory of Cryptography*, TCC'06, pages 265–284, Berlin, Heidelberg, 2006. Springer-Verlag.

[16] H. Hachiya and M. Matsugu. Nsh: Normality sensitive hashing for anomaly detection. In *2013 IEEE International Conference on Computer Vision Workshops*, pages 795–802, Dec 2013.

[17] R.L. Wheeden. *Measure and Integral: An Introduction to Real Analysis, Second Edition*. Chapman & Hall/CRC Pure and Applied Mathematics. Taylor & Francis, 2015.

[18] Gustavo E. Batista, Eamonn J. Keogh, Oben Moses Tataw, and Vinícius M. Souza. Cid: An efficient complexity-invariant distance for time series. *Data Min. Knowl. Discov.*, 28(3):634–669, May 2014.

[19] Rafael Giusti and Gustavo E. A. P. A. Batista. An empirical comparison of dissimilarity measures for time series classification. In *Proceedings of the 2013 Brazilian Conference on Intelligent Systems*, BRACIS '13, pages 82–88, Washington, DC, USA, 2013. IEEE Computer Society.