Chapter 3

Unsupervised Anomaly Detection based on Representative Sequence Selection

In this chapter we present the second approach for unsupervised anomaly detection within non-numerical sequence data. The approach exploits the fact that certain kernel classes map the sequence data output by a stationary ergodic source to a spherical cluster in high-dimensional numerical space.

The algorithm calculates the matrix of pairwise distances of the sequences, and selects a sequence close to the center of the hypersphere of the normal data as a representative of the normal data. The sequences are classified according to their distance from the representative sequence. After stating the algorithm, we theoretically explain the choice and parameter setting of the kernel function used for our experiments, the so-called spectrum kernel. Using structural similarities between the kernel and a probabilistic suffix tree, we deduce an optimal setting of the dimensional parameter of the spectrum kernel, which regulates the subsequence length for mapping. We also explain the setting of the key parameter of the algorithm, and deduce bounds of the computational complexity of the algorithm.

Finally, we evaluate the performance of the algorithm using both real world data and artificial data, demonstrating the performance. We also point out practical limitations of the theoretical range of the cluster parameter.

3.1 Algorithm Statement

While the algorithms presented in the previous chapter used the index difference information of identical symbols, the subsequent algorithm follows the school of thought prevalent in unsupervised anomaly detection research by mapping the sequences $x^{b_i}\ i \in \{1,\ldots,n\}$ of $\mathcal{S}$ to points in a numerical space using a kernel, and using the pairwise distance of those points for classification. The pairwise distance of two sequences x^{l_1}, x^{l_2} of lengths l_1, l_2 is based on the kernel function $K\left(x^{l_2}, x^{l_2}\right)$, which is mostly defined as the dot product

$$K\left(x^{l_1}, x^{l_2}\right) = \langle \phi\left(x^{l_1}\right), \phi\left(x^{l_2}\right) \rangle, \tag{3.1}$$

where $\phi\left(x^{l_1}\right)$ is a transformation of the data to an inner product space. Thus, it computes a measure of similarity of the two data points x^{l_1}, x^{l_2} within the said numerical space, and a pseudo metric $d_K\left(x^{l_1}, x^{l_2}\right)$ of the two original data sequences x^{l_1}, x^{l_3} can be defined via

$$d_K\left(x^{l_1}, x^{l_2}\right) = \sqrt{K\left(x^{l_1}, x^{l_1}\right) - 2K\left(x^{l_1}, x^{l_2}\right) + K\left(x^{l_2}, x^{l_2}\right)} \tag{3.2}$$

Our algorithm utilizes the fact that within a feature space of appropriate dimension of certain kernel classes, the vectors representing sequences generated by a stationary distribution may be modeled by a hypersphere. This view is also applied by One-Class Support Vector Machines and Core Vector Machines [14]. The algorithm selects a sequence which is close to the center of the normal hypersphere in numerical space, and then uses the distance to this *representative* sequence for classification and anomaly detection.

This task is equivalent to calculating the median of a spatial dataset. Our algorithm is considerably more efficient than the standard algorithm called L_1 median, which has also been applied to kernel spaces recently [64]. This is because the L_1 median supposes the anomalous data points to be outliers of the normal data, which are scattered around the center of the normal data. We will show the effect of this supposition in the experimental section.

The algorithm consists of the following steps:

1. Calculate the mutual distance or dissimilarity matrix D of the sequence set $\mathcal{S}$ according to the chosen kernel function.

2. Rearrange the entries of every row $i \in \{1,\ldots,n\}$ of the distance matrix from smallest to largest such that for the entries of the rearranged matrix $\hat{D}$,

$$\hat{D}_{i,j} \quad \leq \quad \hat{D}_{i,j+1} \quad for \quad \forall j \in \{1,\ldots,n-1\} \tag{3.3}$$

   holds.

3. Find the row $i_\star$ which meets

$$\hat{D}_{i_\star,\theta} \quad < \quad \hat{D}_{i,\theta} \quad for \quad \forall i \in \{1,\ldots,n\} \setminus \{i_\star\} \tag{3.4}$$

for the given column parameter $\theta \in \{1,\ldots,n\}$. Resolve ties by repeatedly decreasing the column number θ by one and comparing the entries of sequences with identical entries during the last step. If $\theta = 0$, resolve by random selection.

4. The distances of the sequences within $\mathcal{S}$ from sequence number $i_\star$ are processed to discriminate the normal and the anomalous sequences.

3.2 Algorithm Parameter Setting

The algorithm searches for a sequence representative of the normal class. Because of the sequences contributed by the anomalous source and in order to combat noise, a parameter setting of

$$\theta \in [0.5n, 0.7n] \tag{3.5}$$

is used. Even if the radius of the enclosing sphere of normal vectors is much bigger than the radius of the sphere enclosing the anomalous vectors (in case of generation by a single anomalous source), we suppose that the properties of the kernel cause the center of the anomalous hypersphere to be located near the edge of the normal hypersphere. Thus, a setting of θ must obey the subsequent bounds:

$$\begin{aligned} \theta &\leq (1-\rho_{\max})n \\ \theta &\geq \rho_{\max} n \end{aligned} \tag{3.6}$$

The algorithm is stable for the range given by (3.5) for $\rho_{\max} = 0.3$, as we will show in the experimental section. Unless noted otherwise, we used a setting of $\theta = 0.6n$ for our experiments.

We illustrate the algorithm and the setting of the parameter θ by a simple example. Figure 3.1 shows a set $\mathcal{S} = \{s_1, s_2, s_3, s_4, s_5, s_6\}$ of $n = 6$ points in Euclidean space, with s_1, s_2 being anomalous ($\rho = \frac{2}{6} \approx 0.33$). The center of the normal class is the point s_4.

We calculate the subsequent Euclidean distance matrix

$$D = \begin{bmatrix} d_1^T \\ d_2^T \\ d_3^T \\ d_4^T \\ d_5^T \\ d_6^T \end{bmatrix} = \begin{bmatrix} 0 & 0.5 & 2 & 3 & 4.12 & 3.6 \\ 0.5 & 0 & 1.5 & 2.5 & 3.6 & 3.2 \\ 2 & 1.5 & 0 & 1 & 2.23 & 2.23 \\ 3 & 2.5 & 1 & 0 & 1.4 & 2 \\ 4.12 & 3.6 & 2.23 & 1.4 & 0 & 3.16 \\ 3.6 & 3.2 & 2.23 & 2 & 3.16 & 0 \end{bmatrix}, \tag{3.7}$$

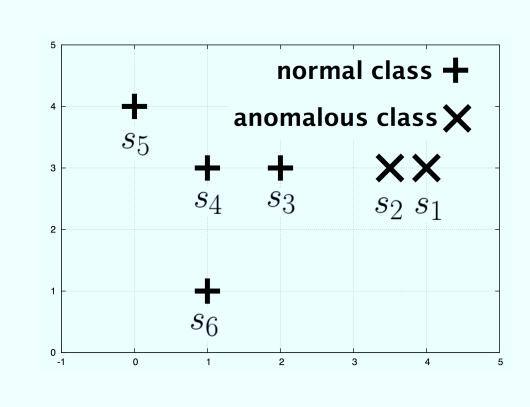

Figure 3.1: Example point set

the rows of which consist of the transposed distance vectors of the respective points, which are defined as

$$d_i^T = \begin{bmatrix} d_K\left(x^{b_i}, x^{b_1}\right) & d_K\left(x^{b_i}, x^{b_2}\right) & \cdots & d_K\left(x^{b_i}, x^{b_n}\right) \end{bmatrix} \tag{3.8}$$

Reordering the respective rows according to the first step of the algorithm, we get

$$\hat{D} = \begin{bmatrix} 0 & \overline{0.5} & 2 & 3 & 3.6 & 4.12 \\ 0 & \overline{0.5} & 1.5 & 2.5 & 3.2 & 3.6 \\ 0 & 1 & 1.5 & \underline{2} & 2.23 & \widehat{2.23} \\ 0 & 1 & \underline{1.4} & \underline{2} & 2.5 & 3 \\ 0 & 1.4 & 2.23 & 3.16 & 3.6 & 4.12 \\ 0 & 2 & 2.23 & 3.16 & 3.2 & 3.6 \end{bmatrix}. \tag{3.9}$$

Now we search for the row with the lowest entry in column θ. If θ is set to an overly low value of θ, say $\theta = \rho n = 2$, this will return $i_\star = 1$ or $i_\star = 2$ (entries overlined), because the variance within the anomalous sequences of our example is smaller than the variance of the normal class. On the other hand, an overly high setting of θ, e.g. $\theta = n = 6$ (minimum radius enclosing all sequences) will cause the selection of s_3 (hatted entry) because of its proximity to the anomalous s_1 and s_2. A setting of $\theta = 0.5n = 3$ or $\theta = 0.66n = 4$ returns the correct $i_\star = 4$ (entries underlined). The set of distances from point s_4 is then processed by means of robust statistics.

$$d_4{}^T = \begin{bmatrix} 3 & 2.5 & 1 & 0 & 1.4 & 2 \end{bmatrix} \tag{3.10}$$

3.3 Sequence Data Distance Matrix Generation

3.3.1 Definition of Kernels and Normalization

A kernel [12] is defined as a function $K(,)$, which for any possible distinct data set of n data points $\{x_1, \ldots, x_n\}$ (numerical or non-numerical), generates a positive semidefinite $n \times n$ matrix

$$\begin{bmatrix} K(x_1, x_1) & \cdots & K(x_1, x_n) \\ \vdots & & \vdots \\ K(x_n, x_1) & \ldots & K(x_n, x_n) \end{bmatrix}. \tag{3.11}$$

This is known as Mercer's condition. Note that although the above kernel specifies only two input data points, kernels may also process additional information on the data. One example is the Fisher kernel [65], which uses information about the underlying normal distribution to create a score. However, because of the unsupervised detection scenario, we will hereafter suppose that the kernel does not process information besides the two data points in question.

According to Kwong et al. [14], the kernelized data will form a hypersphere if the kernel function meets the condition

$$K\left(x^l, x^l\right) = \varrho \; \forall x^l \in \mathcal{X}^l \quad \forall l \in \mathcal{N} \tag{3.12}$$

with ϱ being a nonnegative constant. This condition is satisfied for any normalized kernel defined as

$$K_{\text{norm}}\left(x^{l_1}, x^{l_2}\right) = \frac{K\left(x^{l_1}, x^{l_2}\right)}{\sqrt{K\left(x^{l_1}, x^{l_1}\right)} \cdot \sqrt{K\left(x^{l_2}, x^{l_2}\right)}} \tag{3.13}$$

This covers most kernel functions used for non-numerical sequence data.

3.3.2 Kernel Functions for Non-Numerical Sequence Data and the Spectrum Kernel

The primary task of kernels processing non-numerical sequence data (which are usually referred to as sequence kernels or string kernels) [66] is to generate a numerical, non-negative, and symmetric similarity value, which may then be input to a support vector machine or a clustering algorithm. The standard approach calculates the inner product after separate processing of the sequences by the mapping function ϕ, with the dimensions of the numerical output vector reflecting the existence and/or number of subsequences of symbols, which may or may not be contiguous. This approach is non-localized, ignoring the start indices of the subsequences within the sequence. Contrary, localized approaches

use techniques like sequence alignment [67] in order to calculate a sum value according to the overall length of the matching parts of the two strings. In the following, we will review some of the standard inner product kernels, and justify our choice of the spectrum kernel.

A general inner product-based sequence kernel may be defined as follows:
For an alphabet $\mathcal{X}$, we define by $\mathcal{X}^\star$ the set of all finite strings.

$$\mathcal{X}^\star \stackrel{\text{def}}{=} \cup_{f=0}^{\infty} \mathcal{X}^f \tag{3.14}$$

Then the kernel $K\left(x^{l_1}, x^{l_2}\right)$ processing two sequences $x^{l_1}, x^{l_2} \in \mathcal{X}^\star$ may be expressed by

$$K\left(x^{l_1}, x^{l_2}\right) = \sum_{\forall z \in X^\star} I\left(x^{l_1}, z\right) \cdot w_z \cdot I\left(x^{l_2}, z\right) \cdot w_z = \sum_{\forall z \in X^\star} \phi\left(x^{l_1}\right)_z \cdot \phi\left(x^{l_2}\right)_z \tag{3.15}$$

Here, $I\left(x^{l_1}, z\right)$ represents a function outputting a numerical value according to the existence and/or number and diffusion of the occurrences of z within x^{l_1}. w_z represents the weight of the result of this subsequence for the overall evaluation of similarity.

- spectrum kernel (k):

  The entries of the feature vector of the spectrum kernel [51] consist of the occurrence numbers of all contiguous subsequences of length k. We present an example calculation, given an alphabet $\{a, b\}$ of size $Z = 2$ and two sample sequences $x^{l_1} = (a\ a\ b\ b\ b\ a)$ and $x^{l_2} = (a\ a\ a\ a\ b)$. Setting $k = 2$, the vectors consist of Z^k components $N_{x^{l_i}}\left(x^k\right)\quad x^k \in \mathcal{X}^k$, with $N_{x^{l_i}}\left(x^k\right)$ outputting the occurrence number of the subsequence x^k within the sequence x^{l_i}.

$$K(x,y) \quad = \quad \langle \phi\left(x^{l_1}\right), \phi\left(x^{l_2}\right) \rangle = \langle \begin{bmatrix} N_{x^{l_1}}(aa) \\ N_{x^{l_1}}(ab) \\ N_{x^{l_1}}(ba) \\ N_{x^{l_1}}(bb) \end{bmatrix}, \begin{bmatrix} N_{x^{l_2}}(aa) \\ N_{x^{l_2}}(ab) \\ N_{x^{l_2}}(ba) \\ N_{x^{l_2}}(bb) \end{bmatrix} \rangle = \langle \begin{bmatrix} 1 \\ 1 \\ 1 \\ 2 \end{bmatrix}, \begin{bmatrix} 3 \\ 1 \\ 0 \\ 0 \end{bmatrix} \rangle = 4 \tag{3.16}$$

  Using normalization according to the sequence length, the feature space is a space of probability distributions of sequences of length k. In case of two sources with expected values of the feature space vector located close to each other (i.e. two sources likely to be confused), the variance within feature space will be approximately equal if the length of the sequences is similar.

  A special variant of the spectrum kernel is the so-called full spectrum kernel, an extension of the spectrum kernel which processes all i-grams for $1 \leq i \leq k$. However in case of optimum $k \geq 3$, the performance of this kernel may deteriorate because of the noise added by small i. Therefore, we decided to use the original spectrum kernel.

- mismatch kernel (k, mis):

  The mismatch kernel[68] relaxes the conditions of the spectrum kernel by counting any occurrences of the subsequence of length k and any subsequences including up to mis mismatches. Reusing above example vectors $x^{l_1} = (a\ a\ b\ b\ b\ a)$ and $x^{l_2} = (a\ a\ a\ a\ b)$ with $k = 2$ and $mis = 1$, the vector entry for aa will also include the occurrences of ab and ba, but not bb.

$$K(x,y) = \langle \phi\left(x^{l_1}\right), \phi\left(x^{l_2}\right) \rangle = \langle \begin{bmatrix} N^m_{x^{l_1}}(aa) \\ N^m_{x^{l_1}}(ab) \\ N^m_{x^{l_1}}(ba) \\ N^m_{x^{l_1}}(bb) \end{bmatrix}, \begin{bmatrix} N^m_{x^{l_2}}(aa) \\ N^m_{x^{l_2}}(ab) \\ N^m_{x^{l_2}}(ba) \\ N^m_{x^{l_2}}(bb) \end{bmatrix} \rangle = \langle \begin{bmatrix} 3 \\ 4 \\ 4 \\ 4 \end{bmatrix}, \begin{bmatrix} 4 \\ 4 \\ 3 \\ 1 \end{bmatrix} \rangle = 44 \tag{3.17}$$

  Note that in contrast to the spectrum kernel, the sum of the entries of the feature vector is no longer a linear function of the sequence length.

- gap decay kernel (k, λ):

  The gap decay kernel [69] weights the occurrence of gapped subsequences of length k with a power of the decay factor λ according to the length of the gap. Reusing above example vectors $x^{l_1} = (a\ a\ b\ b\ b\ a)$ and $x^{l_2} = (a\ a\ a\ a\ b)$ with $k = 2$ and $\lambda = 0.5$, we calculate

$$\begin{aligned} K(x,y) &= \langle \phi\left(x^{l_1}\right), \phi\left(x^{l_2}\right) \rangle = \langle \begin{bmatrix} N^g_{x^{l_1}}(aa) \\ N^g_{x^{l_1}}(ab) \\ N^g_{x^{l_1}}(ba) \\ N^g_{x^{l_1}}(bb) \end{bmatrix}, \begin{bmatrix} N^g_{x^{l_2}}(aa) \\ N^g_{x^{l_2}}(ab) \\ N^g_{x^{l_2}}(ba) \\ N^g_{x^{l_2}}(bb) \end{bmatrix} \rangle \\ &= \langle \begin{bmatrix} 1+\lambda^3 \\ \lambda^1+1 \\ \lambda^2+\lambda^1+1 \\ 2 \end{bmatrix}, \begin{bmatrix} 3 \\ \lambda^3+\lambda^2+\lambda^1+1 \\ 0 \\ 0 \end{bmatrix} \rangle = \langle \begin{bmatrix} 1.125 \\ 1.5 \\ 1.75 \\ 2 \end{bmatrix}, \begin{bmatrix} 3 \\ 1.875 \\ 0 \\ 0 \end{bmatrix} \rangle = 6.1875 \end{aligned} \tag{3.18}$$

Although kernel functions taking into account gapped (non-contiguous) sequences or inexact sequence matching have shown good results, for evaluation of our algorithm, we chose the spectrum kernel. We did so because besides low computational effort, this kernel features only a single parameter k, the setting of which may be deduced theoretically, as we will show below. Also, the various derivatives mostly implicitly suppose a setting $k > 1$. However, as we will show below, depending on the ratio of the size of the dataset and the alphabet size Z, even for data generated by a source of strong memory, the optimum setting may be $k = 1$.

Note that the entries of the output vectors of the various inner product kernels may also be used for the calculation of different distance metrics like the Canberra metric or similarity coefficients like the Jaccard coefficient [70].

3.3.3 Model Selection Criteria

In order to prepare the derivation of the setting of the spectrum kernel parameter k in 3.3.4, we review the concept of model selection criteria [71] [72], and point out the connection to anomaly detection.

Presented with the task of fitting a model (statistical or deterministic) to a given training data set $\mathcal{T}$ in order to evaluate a yet unseen test data set $\mathcal{S}$, one finds that naively increasing the complexity of the model (degree of the polynomial, number of Gaussian distributions in the mixture) to achieve a perfect fit on the training data will result in suboptimal results on the test data. The phenomenon is caused by the model adapting to characteristics of the training data not shared by the test data. This is known as the problem of *overfitting*. Thus, the problem of model selection may be defined as follows: given a limited training data set $\mathcal{T}$, choose the optimum complexity of the model MOD, which minimizes the generalization error over the yet unseen test data $\mathcal{S}$.

In order to prevent overfitting, the optimization of model selection criteria is employed for setting the complexity of the model $MOD(\mathcal{T})$ trained using $\mathcal{T}$. A general definition may be written as

$$\text{MSC} \stackrel{\text{def}}{=} \min_{MOD(\mathcal{T})} F(\mathcal{T}|MOD(\mathcal{T})) + PEN(MOD(\mathcal{T})), \tag{3.19}$$

where F describes the deviation of the training data T with respect to a trained given model, while PEN is a strictly monotonic increasing penalty function of the complexity of the model. The task is to choose a MOD minimizing MSC i.e. yielding a low deviation value while keeping the penalty term low. For the calculation of both terms, various functions have been proposed. For statistical models, the deviation is usually calculated as the loglikelihood of the data according to the model. For deterministic models like polynomials, the deviaition may be expressed by the squared error.

The two most frequently used model selection criteria are the Akaike Information Criterion (AIC) [73] and the Bayesian Information Criterion (BIC) [74]. For statistical models, the AIC is defined as

$$\text{AIC} \stackrel{\text{def}}{=} \min_{\psi} -2\log P(\mathcal{T}|MOD(\mathcal{T},\psi)) + 2\psi, \tag{3.20}$$

while the BIC is defined as

$$\text{BIC} \stackrel{\text{def}}{=} \min_{\psi} -2\log P(\mathcal{T}|MOD(\mathcal{T},\psi)) + \psi \ln|\mathcal{T}|, \tag{3.21}$$

with ψ representing the number of free parameters in the model. The logarithm of the cardinality of the training data was included by the BIC in order to account for the precision necessary to build a model interpolating $|\mathcal{T}|$ data points. For small sample sizes $|\mathcal{T}|$, a corrected version of the Akaike Information Criterion, which is written as

$$\mathrm{AIC_c} \stackrel{\mathrm{def}}{=} \mathrm{AIC} + \frac{2\psi\left(\psi+1\right)}{|\mathcal{T}|-\psi-1}, \tag{3.22}$$

has been proposed [75].

Above criteria may also be deduced using the concept of Kolmogorov complexity $C\left(x\right)$ [76] [77], which has been introduced before as the minimum length of the program generating a sequence x, and the conditional Kolmogorov complexity $C\left(x|y\right)$, which is defined as the minimum length of the program generating x given y. The MSC may be interpreted as

$$\mathrm{MSC} \stackrel{\mathrm{def}}{=} \min_{MOD(\mathcal{T})} C\left(\mathcal{T}|MOD(\mathcal{T})\right) + C\left(MOD(\mathcal{T})\right). \tag{3.23}$$

With respect to supervised anomaly detection, the model selection task may be reformulated: given a sample of normal training data, choose a model complexity which is general enough to account for the normal data yet unseen, but specific enough to detect anomalies. In case of unsupervised anomaly detection, the goal is to choose a mapping which creates a coherent cluster of normal data while also achieving separability of normal data and anomalous data.

3.3.4 Unsupervised Probabilistic Suffix Tree Algorithm and Spectrum Kernel Parameter Setting

The unsupervised probabilistic suffix tree algorithm [48] creates a probabilistic suffix tree based on the whole of the data $\mathcal{S}$ given. While previous supervised approaches used a criterion based on the frequency of occurrence of the respective subsequences [50] for pruning single branches, the recent unsupervised approach referenced here favors the use of the Corrected Akaike Information Criterion introduced in the previous section for first setting a global maximum tree depth t, then optionally pruning single branches.

Based on the distribution $P_{\mathcal{X}}^{\mathcal{S}}\left(x_{t+1}|x_1,\ldots,x_t\right)\,\forall x^{t+1}\in\mathcal{X}^{t+1}$ of the set $\mathcal{S}$ contained in a suffix tree of depth t, a normalized dissimilarity measure for judging any sequence $x^b\in\mathcal{S}$ of b symbols can be defined as follows

$$\mathrm{DSIM}\left(x^b\right) \stackrel{\mathrm{def}}{=} -\frac{1}{b}\sum_{i=1}^{b}\log P_{\mathcal{X}}^{\mathcal{S}}\left(x_i|x_{i-t},\ldots,x_{i-1}\right) \tag{3.24}$$

The set of scalars calculated according to (3.24) is then processed by means of robust statistics and the receiver operating characteristic (ROC) curve is created.

For t fixed, $\rho = 0$ and $b \to \infty$, (3.24) converges towards the entropy rate of a Markov source of the normal data of memory t. The upper bound of the dissimilarity measure of the unsupervised case is determined by n and b, the lower bound being zero. A low dissimilarity measure is likely in case of normal data, while a block of high dissimilarity measure is likely to have been generated by an anomaly. However, the method is founded on the implicit assumption that the entropy rate of the normal source at the optimum tree depth is equal to or below the entropy rate of the anomalous source(s). If this assumption holds, the information criterion will indeed output a tree which will encode the normal source more efficiently than the anomalous source. An anomalous source of small entropy rate compared to the normal source, on the other hand, may not stand out as desired.

We illustrate our observation by the distribution of dissimilarity values output by (3.24) for various t processing two protein datasets with $n = 5,000$ and $\rho = 0.05$ (see experimental section). The first dataset, the results of which are shown by Fig. 3.2, features a normal class with entropy rate below that of the anomalous class, and therefore detection of the anomalous class can be easily done for $t = 3$, the global tree depth returned by the information criterion.

Contrary, the second dataset features the case of the normal class entropy rate being much higher than the entropy rate of the anomalous class. Figure 3.3 shows the results. The distribution for the depth $t = 3$ returned by the criterion does not allow for efficient detection. It is interesting to see, however, that although the scalar similarity measure is not able to separate the two classes, the optimum tree depth maximizes the respective intra-class variance of the dissimilarity measure.

Note that (3.24) can be approximated as

$$\mathrm{DSIM}\left(x^b\right) \approx -\sum_{x^{t+1}\in\mathcal{X}^{t+1}} \frac{N_{x^b}\left(x^{t+1}\right)}{b} \cdot \log P_{\mathcal{X}}^{\mathcal{S}}\left(x_{t+1}|x^t\right), \tag{3.25}$$

where $N_{x^b}\left(x^{t+1}\right)$ represents the number of occurrence of a certain sequence $x^{t+1} \in \mathcal{X}^{t+1}$ within a sample sequence x^b. We utilize this fact to determine the parameter of the spectrum kernel in the next section. The approximation given by (3.25) may be seen as a dot product of a vector with entries $\left[\frac{N_{x^b}\left(x^{t+1}\right)}{b}\right]_{x^{t+1}\in\mathcal{X}^{t+1}}$ and a vector consisting of entries $\left[\log P_{\mathcal{X}}^{\mathcal{S}}\left(x_{t+1}|x^t\right)\right]_{x^{t+1}\in\mathcal{X}^{t+1}}$, the former describing the distribution of subsequences of length $t+1$ based on x^b and the latter containing a weighting spectrum of how badly a predictor of memory t based on the entire set $\mathcal{S}$ emulates subsequences of length $t+1$. Because the spectrum kernel calculates an empirical estimate of the distribution of k-grams

based on the sequence, looking at (3.25), we understand that for $k = t + 1$, the entries of the feature vector of the spectrum kernel are equal to the occurrence numbers within the sum. Thus, the optimized depth t of the suffix tree may be used to set the parameter k to

$$k = t + 1. \tag{3.26}$$

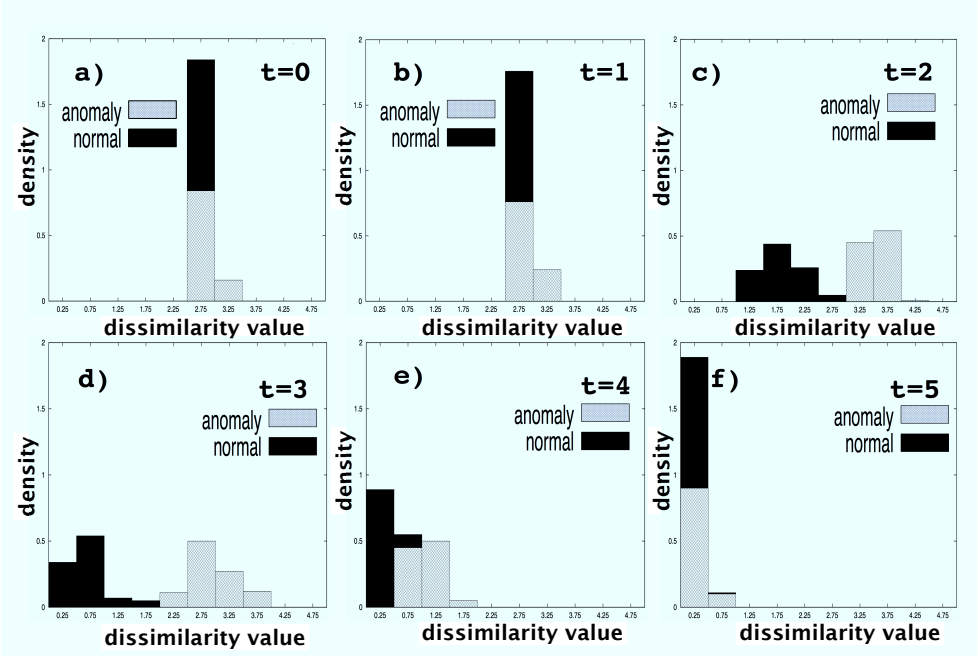

Figure 3.2: Example distribution of dissimilarity values for the case of the variance of the normal class being smaller than the variance of the anomalous class

3.4 Computational Cost of Algorithm Using Spectrum Kernel

It has been shown [78] that the computational cost of computing the spectrum kernel of two sequences of respective length l_1 and l_2 is upper bounded by a function of order

$$O(l_1 + l_2) \tag{3.27}$$

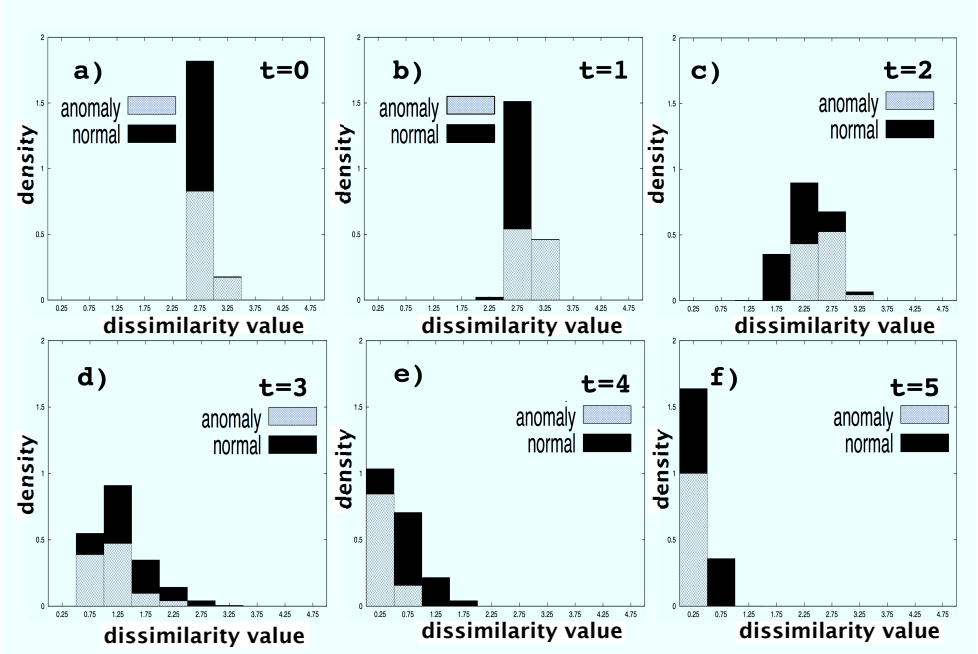

Figure 3.3: Example distribution of dissimilarity values for the case of the variance of the normal class exceeding the variance of the anomalous class

Thus, with $b_{\max}$ being the maximum sequence length within the n sequences of the set $\mathcal{S}$, the computational cost of creating the distance matrix is bounded by a function of order

$$O\left(n^2 \cdot b_{\max}\right) \tag{3.28}$$

On the other hand the computational cost of ordering a set of n nonnegative real values from smallest to largest without any further constraints can be bounded by a function of order

$$O\left(n \cdot \log n\right) \tag{3.29}$$

The computational cost of rearranging the n rows of the distance matrix and determining the representative sequence is thus upper bounded by a function of order

$$O\left(n^2 \cdot \log n\right) \tag{3.30}$$

3.5 Experimental Results

3.5.1 ROC Parameter Calculation

In order to calculate the receiver operating characteristic curve from the noisy set of scalars returned by the various algorithms, we use the median and the median absolute deviation for mean and deviation estimation. For further description see Section 2.4.2.

3.5.2 Artificial Data

We used i.i.d. symbol generation. While the first round of experiments used distributions featuring a uniform distribution of probabilities within a certain interval, thereby limiting the distinctiveness of the distributions, the second round of experiments used distributions featuring an exponential probability spectrum i.e. a few symbols have large generation probabilities, while the overwhelming majority features very small probabilities. For further explanation, see Section 2.4.3.

The overall sequence length g was set to $4,000$. Because we set the alphabet size $Z = 100$ to the same range as the block size $b = 200 = 0.05 \cdot g$ and because of i.i.d. generation, the subsequent simulations use zero maximum memory length $M_{\max}$.

Setting of the Representative Sequence Selection Algorithm

We used a setting of $\theta = 0.6n$. Experimental results presented in Fig. 3.4 show that for artificial data, within the bounds given by (3.5), the choice of θ has very little influence on the classification accuracy. The parameter k of the spectrum kernel is set to 1 according to the optimized depth of the probabilistic suffix tree.

Settings of the Previous Approaches

The cluster algorithm is set up with $k = 1$ and $w = 0.9$, with the latter setting retrieved via trial and error. The optimum depth of the suffix tree turned out to be $t = 0$, regardless of the setting of $S_{\min}$.

We simulate anomaly lengths l_{A} of $0.05 \cdot g$, $0.15 \cdot g$, and $0.25 \cdot g$ respectively, using uniform distribution (interval $[0.1, 1]$) and exponential generation for symbol distribution generation. The uniform generation limits the range of $h(x)$ to $\frac{1}{10} \leq h(x) \leq 10$ ($P(x) \in \left[\frac{1}{5.5 \cdot Z}, \frac{10}{5.5 \cdot Z}\right]$), thus posing are more difficult task than a symbol distribution generated

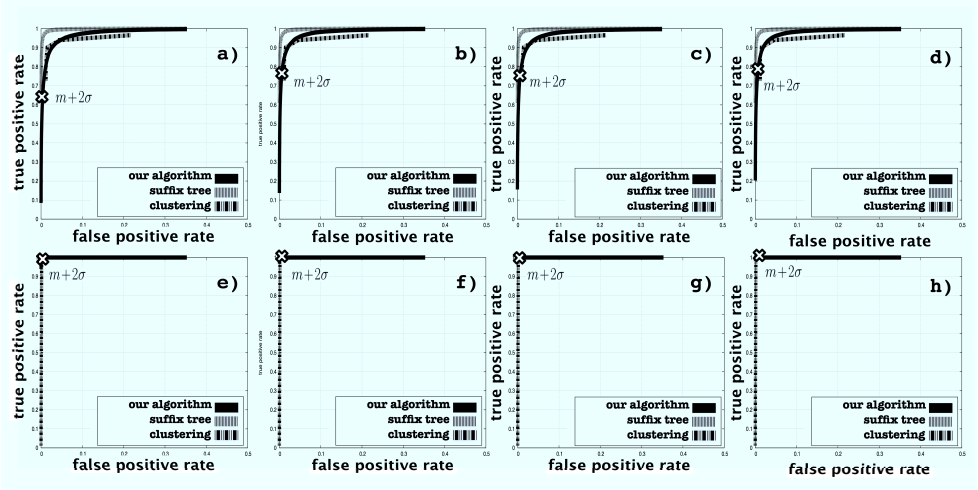

Figure 3.4: Representative sequence selection algorithm: Uniform distribution generation, 15% anomalous data: a) $\theta = 0.3n$ b) $\theta = 0.4n$ c) $\theta = 0.5n$ d) $\theta = 0.6n$; Exponential distribution generation, 15% anomalous data: a) $\theta = 0.3n$ b) $\theta = 0.4n$ c) $\theta = 0.5n$ d) $\theta = 0.6n$

by the exponential approach. Each simulation consists of $10,000$ repetitions of sequence generation, with the symbol distributions generated anew after every 100th run. The anomalous blocks are randomly rearranged after initial generation in case of the algorithm allowing for arbitrary distribution of anomalous blocks. Figure 3.5 shows the results of the simulation. Our algorithm performs very well for both uniform and exponential distribution generation, providing a stable performance in case of uniform distribution generation for various anomalous shares ρ, while the performance of the pervious methods decreases with rising ρ. The algorithm is slightly inferior to the previous algorithms in terms of the decrease of the false positive rate in case of no anomaly.

3.5.3 Computer Security Data

One possible application of the representative sequence selection algorithm is network masquerade attack detection, where the goal is to detect abuse of a valid network account by analyzing the session input. We use the well known data set created by Schonlau et al. [62] for their supervised experiments, discarding the training data. For more information on the background, see Section 2.4.4. The data set features $g = 10,000$ and $b = 100$. The parameters $M_{\max}$ and k are set using the depth t of the tree returned by the probabilistic suffix tree algorithm. In order to show the impact of $S_{\min}$ for data generated by a source with memory, we use the settings $S_{\min} = 1$ and $S_{\min} = Z$.

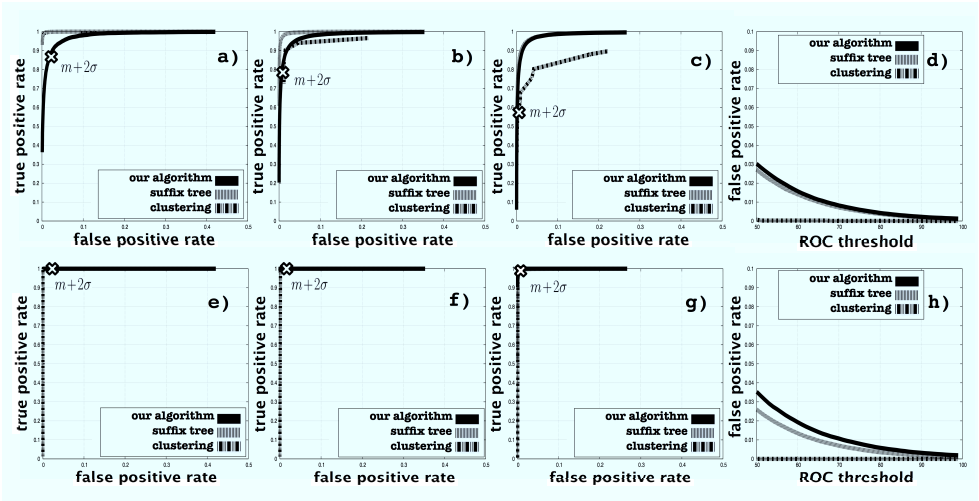

Figure 3.5: Representative sequence selection algorithm: Uniform distribution generation: a) 5% anomalous data b) 15% anomalous data c) 25% anomalous data d) Decrease of false positive rate; Exponential distribution generation: a) 5% anomalous data b) 15% anomalous data c) 25% anomalous data d) Decrease of false positive rate

Setting of the Representative Sequence Selection Algorithm

We used the optimum tree depths $t = 0$ and $t = 1$ returned by the suffix tree algorithm for respectively setting the parameter k of the spectrum kernel to $k = 1$ and $k = 2$. We tried for three different settings for θ: $\theta = 0.5n$, $\theta = 0.6n$ $\theta = 0.7n$

Settings of the Previous Approaches

The cluster algorithm uses a radius $w = 1.3$, again set using trial and error. For the suffix tree algorithm, $S_{\min} = 1$ and $S_{\min} = Z$ yields $t = 0 \rightarrow \ M_{\max} = 0,\ k = 1$ and $t = 1 \rightarrow \ M_{\max} = 1,\ k = 2$ respectively

The average ROC curve and the decrease of the false positive rate is shown by Fig. 3.6, with very good performance by our algorithm. The low false positive value returned for $m + 2\sigma$ (cross mark) hints that the distances of normal sequences from the center sequence may be modeled by a Gaussian distribution. The rather poor performance of the suffix tree algorithm may be explained by the combination of small n and b, a rather large Z (about 150), and the exponential distribution of generation probabilities, which causes an overly rough modeling of the data. The comparison of the results for $S_{\min} = 1$ and $S_{\min} = Z$ for a threshold of $m + 2\sigma$ show that the optimum depth returned by the suffix tree algorithm for

$S_{\min} = 1$ yields the best result for our algorithm. shows the average decrease of the false positive rate in case of $\rho = 0$ with rising threshold. The decrease of false positive rate of our algorithm, on the other hand, is notably improved for $S_{\min} = Z$. The most important difference compared to artificial data is the notable drop of performance for $\theta = 0.5n$. This may be explained by the fact that most real world data is rather generated by a combination of several closely related sources, creating a cluster of several hyperspheres of the normal data in the feature space.

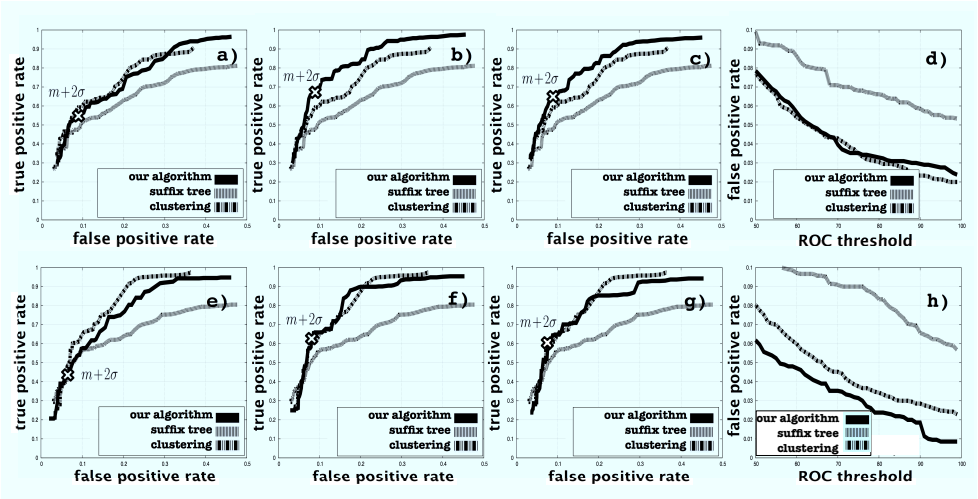

Figure 3.6: Real world masquerade data and $S_{\min} = 1$: a) $\theta = 0.5n$ b) $\theta = 0.6n$ b) $\theta = 0.7n$ d) decrease of false positive rate for $\theta = 0.6n$ and $\rho = 0$; Real world masquerade data and $S_{\min} = Z$: e) $\theta = 0.5n$ f) $\theta = 0.6n$ g) $\theta = 0.7n$ h) decrease of false positive rate for $\theta = 0.6n$ and $\rho = 0$

3.5.4 Protein Data

We follow the experiment of the paper introducing the unsupervised suffix tree algorithm by mixing data from the HCV_core protein family and the NADHdh protein family ($Z = 20$), both made available by the well-known Pfam database [79]. While the first family consists of $5,000$ sequences with an average sequence length of $b = 60$ derived from 6 seeds, the second family consists of $12,000$ sequences with an average length of $b = 128$ derived form 23 seeds. Thus, the entropy rate of the NADHdh family exceeds the entropy rate of the HCV_core family.

We created datasets of $n = 300$ sequences both with the HCV_core family and the NADHdh family as the major contributor.

Setting of the Representative Sequence Selection Algorithm

We used the optimum tree depths $t = 3$ returned by the suffix tree algorithm for setting the parameter k of the spectrum kernel to $k = 4$. We tried three different settings for θ: $\theta = 0.5n$, $\theta = 0.6n$, and $\theta = 0.7n$, which yielded no tangible difference of performance. The graph data given in below figure was retrieved for $\theta = 0.6n$.

Settings of the Previous Approaches

The cluster algorithm uses a radius $w = 1.1$ for the first and $w = 1.3$ for the second data set, again set using trial and error. For the suffix tree algorithm, $S_{\min} = 1$ and $S_{\min} = Z$ yield $t = 3$ and $t = 4$ respectively. Because of the sparseness of the data, we used $t = 3 \rightarrow k = 4$ for the subsequent experiments.

Figure 3.7 show the ROC curves and the decrease of the false positive rate for both data sets. While all algorithms show similar classification performance for a majority of HCV_core, for the difficult case of the entropy rate of the normal class NADHdh exceeding the entropy rate of the anomalous class, our algorithm clearly outperforms the other algorithms. Alike to the masquerade data, the low false positive value returned for $m + 2\sigma$ (cross mark) hints for a Gaussian distribution of normal sequence distances. Regarding the decrease of the false positive rate, our algorithm is more susceptible to noise in case of NADHdh because the intra-class variance hampers the selection of a central representative sequence. For a majority of HCV_core, the suffix tree algorithm does very well compared to the masquerade detection application. This is because of a more balanced distribution, a smaller alphabet, and a larger n.

We also compared our algorithm to detection based on the L_1 median. Figure 3.8 shows the results for $\rho = 0.05$ and $\rho = 0.15$. While the spatial median does well for small or zero ρ, the performance decreases significantly for higher ρ.

3.6 Concluding Remarks

In this chapter we presented an unsupervised anomaly detection algorithm based on the pairwise distance of data points, which may be used to process non-numerical sequence data given a suitable kernel function. We showed how the parameter of an example kernel can be set using an information theoretic criterion, yielding good experimental results.

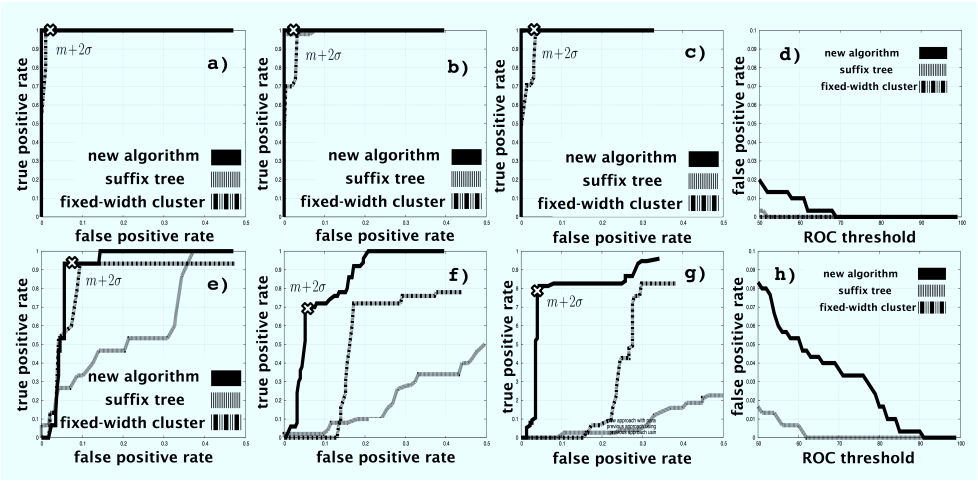

Figure 3.7: HCV_core dataset and $\theta = 0.6n$: a) $\rho = \mathbf{0.05}$ b) $\rho = \mathbf{0.15}$ c) $\rho = \mathbf{0.25}$ d) Decrease of false positive rate for $\rho = \mathbf{0}$; NADHdh dataset and $\theta = 0.6n$: e) $\rho = \mathbf{0.05}$ f) $\rho = \mathbf{0.15}$ g) $\rho = \mathbf{0.25}$ h) Decrease of false positive rate for $\rho = \mathbf{0}$

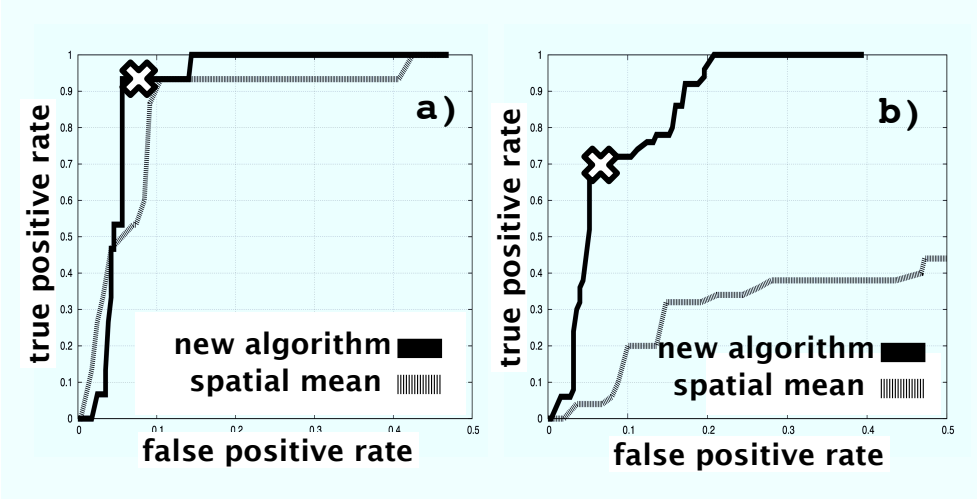

Figure 3.8: NADHdh dataset a) $\rho = \mathbf{0.05}$ b) $\rho = \mathbf{0.15}$

Chapter 4

Conclusion

In this thesis, we elaborated on two approaches to the problem of one-class unsupervised anomaly detection within a set $\mathcal{S}$ of non-numerical sequences.

In Chapter 1 we gave an introduction to the general problem of anomaly detection and to the particular problem of unsupervised anomaly detection, defining the problem scenario as follows:

1. We are given a set $\mathcal{S}$ of n sequences $x^{b_1}, x^{b_2}, \ldots, x^{b_{n-1}}, x^{b_n}$ of varying length, with $x \in \mathcal{X} = \{a_1, a_2, \ldots, a_{Z-1}, a_Z\}$,

2. We suppose that the majority of $1 - \rho$ ($0 \leq \rho \leq \rho_{\max} = 0.33$) of the sequences was generated by one stationary normal source N (one-class scenario), while the remaining share ρ of the sequences was generated by one or more stationary abnormal sources A.

3. The task is to derive a measure or score for the normality of each sequence in the set $\mathcal{S}$.

Having stated the scenario, we introduced two methods representative of previous research on the topic used for evaluation of our algorithm, the probabilistic suffix tree algorithm and the fixed-width clustering algorithm.

Our first approach, which we explained in Chapter 2, fuses together the set of sequences $\mathcal{S}$ into a single global sequence of length g. It uses a function called the average index difference to respectively generate a numerical value associated with every single symbol within the global sequence.

We introduced the average index difference function, which calculates the average of the index differences between a symbol or subsequence found at a particular index j and symbols or subsequence of identical value within the global sequence. We proved the convergence of the function to an expected value dependent only on the global index j but not on the symbol generation probability for the case of stationary ergodic symbol generation.

We presented two algorithms based on this function.

The first algorithm exploits the fact that, if the abnormal sequences happen to be clustered within a subsection of the global sequence, the output value of the average index difference function is reciprocally related to the likelihood of the symbol to be representative of the anomalous data compared to the normal data and to have been generated within an anomalous sequence. This is because most of the identical symbols will be generated close to the index j, thus significantly decreasing the average index difference function value. The percentage c of symbols within a sequence featuring an average index difference below a certain threshold τ_{th} is used for final classification.

Because the first algorithm is hampered by supposing the abnormal sequences to be consecutively clustered within the overall sequence, we conceived the second algorithm, which extends the original average index difference function, allowing for an arbitrary location of the anomalous sequences within the global sequence. Furthermore, the algorithm extends the function to subsequences of symbols, the maximum length $M_{\max}$ of which was set via an information theoretic criterion. It compares index differences between neighboring occurrences $\Delta\left(x^{M+1}\right)$ to a ξ multiple of the empirical mean value $\overline{\Delta}\left(x^{M+1}\right)$ of those index differences, in order to identify gaps between anomalous blocks prior to the calculation of the average index difference.

Besides conceiving the algorithms, our contribution consisted of showing how suitable settings for all the parameters both for the case of stationary ergodic generation and i.i.d. generation can be derived by theoretical considerations. We also deduced bounds for the computational cost of both algorithms, showing how the average index difference of every symbol within the sequence of length g can be computed in a time linear with g.

We evaluated the performance of the two algorithms using both computer security-related real world data and artificial data, comparing our results to those of the previous methods. Calculating the curves of the respective receiver operating characteristic, we deduced the thresholds from the set of scalar values returned for $\mathcal{S}$ by the algorithms by means of robust statistics. The experiments with the i.i.d. data showed the preference of the algorithms for symbol distributions featuring exponentially distributed symbol probabilities i.e. a small group of symbols features large generation probabilities, while the generation probability of the majority of the alphabet is very small. For those exponen-

tial distributions the two algorithms showed a performance equal to those of the previous methods, while for distributions featuring a uniform spectrum of generation probabilities, there was a notable gap of performance. Those findings were supported by the good results for computer-security related experiments, where the task consisted of finding anomalous records within a set of computer network session logs. The records feature a large alphabet of commands with exponentially distributed probabilities. Comparison with the previous methods showed that while the second algorithm is inferior for high false positive rates, in case of false positive rates below ten percent, it yields comparable or superior performance. Contrary, the first algorithm showed higher true positive rates at the price of an increased false positive rate.

Our second approach to the problem, which we explained in Chapter 3, computes the matrix of pairwise distances of the set of sequences $\mathcal{S}$ by mapping them into a numerical space via a suitable kernel function, turning the scenario into a spatial classification problem.

The algorithm conceived works as follows: First we map the sequences of $\mathcal{S}$ into a vector space using a suitable kernel, such that the vectors calculated from the sequences output by a stationary source will form a hypersphere. After calculating the matrix of pairwise distances of the vectors, we select a sequence close to the center of the hypersphere of the normal data as a representative of the normal data. This is done by using the distance matrix to calculate the radius β necessary to cover a share of θ of the n sequences for any sequence within $\mathcal{S}$. The sequence with minimum β is chosen as representative of the normal data. Finally, the sequences are classified according to their distance from the representative sequence.

Besides the algorithm, our contribution consisted of theoretically explaining the choice and parameter setting of the kernel function used for our experiments, the so-called spectrum kernel. Using the structural similarities between the kernel and a probabilistic suffix tree, we showed how the optimized depth of the tree may be used for setting the dimensional parameter of the spectrum kernel. This parameter regulates the subsequence length for mapping. Moreover, we explained the setting of the key parameter of the algorithm, θ. We also deduced bounds of the computational complexity of the algorithm.

We evaluated the performance of the algorithm using both real world data and artificial data, comparing our results to those of the previous methods. We used the same ROC calculation method as in Chapter 2. The experiments using i.i.d. generated artificial data showed that while for a uniform spectrum of generation probabilities our algorithm performs slightly worse than the previous approaches, this performance is stable for a wide range of anomalous shares ρ. Although the performance was stable for the theoretically deduced range of θ in case of artificial data, the results for computer-security related data showed that structural complexities of the data may decrease the performance for the lower

half of the theoretical range. Using data related to bioinformatics, we showed the superior performance of our algorithm for the difficult case of a combination of high entropy of the normal data and low entropy of the anomalous data.

Possible future research includes the subsequent problems:

- Extended investigation of the properties of the average index difference function for certain scenarios
  We showed the convergence of the expected value of the average index difference function for stationary ergodic symbol, but the speed of convergence could not be determined. Expressing the speed of convergence as a function on the memory length and alphabet size of the data source poses an interesting theoretical problem. Another point is the bounding of the variance of the function.

- Testing of representative sequence selection algorithm for various kernels
  So far, we only examined the combination of the representative sequence selection algorithm and the spectrum kernel. The good results of this combination invite experiments with other kernels, as well as experiments involving kernelized spatial data.

- Extension of one or both approaches to multi-class problems
  We only dealt with the problem of one-class unsupervised anomaly detection. However, many applications involve multiple normal classes. An extension of one or both approaches to this more general scenario would broaden the applicability of our approaches.

Appendix A

Proofs

Lemma A.1
For an infinite sequence of symbols $x \in \mathcal{X} = \{a_1, a_2, \ldots, a_{Z-1}, a_Z\}$ generated by a stationary ergodic source, given that the symbol $a_\star$ was found at index i, the expected value of the recurrence time of $a_\star$, which may be written as

$$\mathrm{E}(R) = \sum_{k=1}^{\infty} k \cdot P\left(x_i = a_\star \,|x_{i+1} \neq a_\star, \ldots, x_{i+k-1} \neq a_\star, x_{i+k} = a_\star\right), \tag{A.1}$$

is the inverse of the stationary generation probability of the symbol $a_\star$.

$$\mathrm{E}(R) = \frac{1}{P(x = a_\star)} \tag{A.2}$$

For the complete proof, see [80].

Lemma A.2
For a sequence of g symbols $x^g \in \mathcal{X}^g$ generated by a single stationary ergodic source, for the conditional average index difference of the symbol found at index $j = \lceil g \cdot y \rceil$ $(0 < y < 1)$, the subsequent expected value holds

$$\mathrm{E}\left(T_j(x) \,|C_\mathrm{b}, C_\mathrm{a}\right) = \frac{1}{2 \cdot P_X^\mathrm{N}(x)} \cdot \left(\frac{C_\mathrm{b}^2 + C_\mathrm{a}^2}{C_\mathrm{b} + C_\mathrm{a}} + 1\right) \tag{A.3}$$

Proof:
The index difference of consecutive identical symbols is a stationary ergodic process. We

first state

$$\begin{aligned}
\mathrm{E}\left(\sum_{o=1}^{C_\mathrm{b}}\left(j-j_o^\mathrm{b}\right)|C_\mathrm{b},C_\mathrm{a}\right) &= \mathrm{E}\left(\sum_{o=1}^{C_\mathrm{b}}(C_\mathrm{b}+1-o)\cdot\left(j_{o-1}^\mathrm{b}-j_o^\mathrm{b}\right)|C_\mathrm{b},C_\mathrm{a}\right)\\
&= \sum_{o=1}^{C_\mathrm{b}}(C_\mathrm{b}+1-o)\cdot\mathrm{E}\left(j_{o-1}^\mathrm{b}-j_o^\mathrm{b}|C_\mathrm{b},C_\mathrm{a}\right)\\
&= \sum_{o=1}^{C_\mathrm{b}}(C_\mathrm{b}+1-o)\cdot\frac{j}{C_\mathrm{b}+1} \qquad \text{(A.4)}\\
&= \frac{C_\mathrm{b}\cdot j}{2}, \qquad \text{(A.5)}
\end{aligned}$$

with

$$j_0^\mathrm{b}\stackrel{\mathrm{def}}{=}j \qquad \text{(A.6)}$$

(A.4) can be derived as follows. We suppose a hypothetical occurrence x identical to the one at index j at the imaginary index 0, the index of which we will annotate as $j_{C_\mathrm{b}+1}^\mathrm{b}$. Then the expected value of the sum of the index differences of neighboring occurrences of x with index equal to or below j may be expressed as

$$\mathrm{E}\left(\sum_{o=1}^{C_\mathrm{b}+1}\left(j_{o-1}^\mathrm{b}-j_o^\mathrm{b}\right)|C_\mathrm{b},C_\mathrm{a}\right) = j \qquad \text{(A.7)}$$

$$\text{(A.8)}$$

and

$$\sum_{o=1}^{C_\mathrm{b}+1}\mathrm{E}\left(j_{o-1}^\mathrm{b}-j_o^\mathrm{b}|C_\mathrm{b},C_\mathrm{a}\right) = j. \qquad \text{(A.9)}$$

$$\text{(A.10)}$$

Because of the stationarity of the sequence, the expected value is independent of the respective indices. Therefore,

$$\mathrm{E}\left(j_{o-1}^\mathrm{b}-j_o^\mathrm{b}|C_\mathrm{b},C_\mathrm{a}\right)=\frac{j}{C_\mathrm{b}+1}\quad\forall o\in\{1,\ldots,C_\mathrm{b}+1\} \qquad \text{(A.11)}$$

holds. A derivation similar to the above yields

$$\mathrm{E}\left(\sum_{q=1}^{C_\mathrm{a}}\left(j-j_q^\mathrm{a}\right)|C_\mathrm{b},C_\mathrm{a}\right) = \frac{C_\mathrm{a}\cdot(g-j)}{2}. \qquad \text{(A.12)}$$

We move on to

$$\begin{aligned}
\mathrm{E}\left(T_j\left(x\right)|C_{\mathrm{b}},C_{\mathrm{a}}\right) &= \mathrm{E}\left(\frac{\sum_{o=1}^{C_{\mathrm{b}}}\left(j-j_o^{\mathrm{b}}\right)+\sum_{q=1}^{C_{\mathrm{a}}}\left(j_q^{\mathrm{a}}-j\right)}{C_{\mathrm{b}}+C_{\mathrm{a}}}|C_{\mathrm{b}},C_{\mathrm{a}}\right)\\
&= \frac{\mathrm{E}\left(\sum_{o=1}^{C_{\mathrm{b}}}\left(j-j_o^{\mathrm{b}}\right)|C_{\mathrm{b}}\right)+\mathrm{E}\left(\sum_{q=1}^{C_{\mathrm{a}}}\left(j_q^{\mathrm{a}}-j\right)|C_{\mathrm{a}}\right)}{C_{\mathrm{b}}+C_{\mathrm{a}}}\\
&= \frac{1}{C_{\mathrm{b}}+C_{\mathrm{a}}}\cdot\frac{C_{\mathrm{b}}j+C_{\mathrm{a}}\cdot(g-j)}{2} \qquad \text{(A.13)}\\
&= \frac{C_{\mathrm{b}}j+C_{\mathrm{a}}\cdot(g-j)}{2\cdot(C_{\mathrm{b}}+C_{\mathrm{a}})}, \qquad \text{(A.14)}
\end{aligned}$$

where (A.13) follows from (A.5) and (A.12). This finishes the proof.

Note that the derivatives of (A.14) with respect to C_{b} and C_{a} read

$$\frac{d\mathrm{E}\left(T_j\left(x\right)|C_{\mathrm{b}},C_{\mathrm{a}}\right)}{dC_{\mathrm{b}}}=\frac{C_{\mathrm{a}}\cdot(2j-g)}{2\left(C_{\mathrm{b}}+C_{\mathrm{a}}\right)^2} \qquad \text{(A.15)}$$

and

$$\frac{d\mathrm{E}\left(T_j\left(x\right)|C_{\mathrm{b}},C_{\mathrm{a}}\right)}{dC_{\mathrm{a}}}=\frac{C_{\mathrm{b}}\cdot(g-2j)}{2\left(C_{\mathrm{b}}+C_{\mathrm{a}}\right)^2} \qquad \text{(A.16)}$$

A.1 Proof of Theorem 2.1: Expected Value of the Average Index Difference in Case of Stationary Ergodic Symbol Generation

For simplicity, we represent $\mathrm{E}\left(T_j\left(x\right)|l_{\mathrm{A}}=0\right)$ by $\mathrm{E}\left(T_j\left(x\right)\right)$. We first present some definitions

$$j=\lceil y\cdot g\rceil;\quad 0<y<\frac{1}{2} \qquad \text{(A.17)}$$

$$0<\delta<1 \qquad \text{(A.18)}$$

The limitation of j is valid because of the symmetry of the function.

$$\begin{aligned}
\mathcal{A} &\triangleq \left\{C_{\mathrm{b}}: P_X^{\mathrm{N}}\left(x\right)\cdot g\cdot y\cdot(1-\delta)\le C_{\mathrm{b}}\le P_X^{\mathrm{N}}\left(x\right)\cdot g\cdot y\cdot(1+\delta)\right\} \qquad \text{(A.19)}\\
\mathcal{B} &\triangleq \left\{C_{\mathrm{a}}: P_X^{\mathrm{N}}\left(x\right)\cdot g\cdot(1-y)\cdot(1-\delta)\le C_{\mathrm{a}}\le P_X^{\mathrm{N}}\left(x\right)\cdot g\cdot(1-y)\cdot(1+\delta)\right\} \qquad \text{(A.20)}
\end{aligned}$$

Rewriting the definition of the expected value of the average index difference using above definitions and dividing both sides by g, we get

$$\begin{aligned}\frac{\mathrm{E}\left(T_j\left(x\right)\right)}{g} &= \frac{\sum_{C_\mathrm{b}=0}^{y\cdot g}\sum_{C_\mathrm{a}=0}^{(1-y)\cdot g} P^\mathrm{N}\left(C_\mathrm{b},C_\mathrm{a}\right)\cdot \mathrm{E}\left(T_j\left(x\right)|C_\mathrm{b},C_\mathrm{a}\right)}{g} \\ &= \left(\sum_{C_\mathrm{b}\in\mathcal{A}\cap C_\mathrm{a}\in\mathcal{B}} P^\mathrm{N}\left(C_\mathrm{b},C_\mathrm{a}\right)\cdot \mathrm{E}\left(T_j\left(x\right)|C_\mathrm{b},C_\mathrm{a}\right)\cdot\frac{1}{g}\right. \\ &\quad \left. + \sum_{C_\mathrm{b}\notin\mathcal{A}\cup C_\mathrm{a}\notin\mathcal{B}} P^\mathrm{N}\left(C_\mathrm{b},C_\mathrm{a}\right)\cdot \mathrm{E}\left(T_j\left(x\right)|C_\mathrm{b},C_\mathrm{a}\right)\cdot\frac{1}{g}\right)\end{aligned} \tag{A.21}$$

The upper and the lower bound of (A.21) read as follows, with ϵ expressing the probability of C_a and/or C_b falling outside the range defined by (A.20) and (A.19):

$$\begin{aligned}\frac{\mathrm{E}\left(T_j\left(x\right)\right)}{g} &\leq \left(\sum_{C_\mathrm{b}\in\mathcal{A}\cap C_\mathrm{a}\in\mathcal{B}} P^\mathrm{N}\left(C_\mathrm{b},C_\mathrm{a}\right)\cdot \mathrm{E}\left(T_j\left(x\right)|C_\mathrm{b},C_\mathrm{a}\right)\cdot\frac{1}{g}\right. \\ &\quad \left. + \overbrace{\sum_{C_\mathrm{b}\notin\mathcal{A}\cup C_\mathrm{a}\notin\mathcal{B}} P^\mathrm{N}\left(C_\mathrm{b},C_\mathrm{a}\right)}^{\epsilon}\cdot g\cdot\frac{1}{g}\right) \\ &= \left(\sum_{C_\mathrm{b}\in\mathcal{A}\cap C_\mathrm{a}\in\mathcal{B}} P^\mathrm{N}\left(C_\mathrm{b},C_\mathrm{a}\right)\cdot \mathrm{E}\left(T_j\left(x\right)|C_\mathrm{b},C_\mathrm{a}\right)\cdot\frac{1}{g}+\epsilon\right) \\ &\leq \mathrm{E}\left(T_j\left(x\right)\Big|C_\mathrm{b}=P_X^\mathrm{N}\left(x\right)\cdot g\cdot y\cdot\left(1-\delta\right),\right. \\ &\qquad \left. C_\mathrm{a}=P_X^\mathrm{N}\left(x\right)\cdot g\cdot\left(1-y\right)\cdot\left(1+\delta\right)\right)\cdot\frac{1}{g}+\epsilon \qquad \text{(A.22)} \\ &\leq \frac{1}{2}\left(\frac{g^2y^2\left(1-\delta\right)+g^2\left(1-y\right)^2\left(1-\delta+2\delta\right)}{g\left(1-\delta\right)}\right)\frac{1}{g}+\epsilon \qquad \text{(A.23)} \\ &= \frac{1}{2}\left(\frac{j^2+\left(g-j\right)^2}{g}+\frac{\left(g-j\right)^2\cdot 2\delta}{g\left(1-\delta\right)}\right)\frac{1}{g}+\epsilon \qquad \text{(A.24)}\end{aligned}$$

Where inequality (A.22) holds because $\mathrm{E}\left(T_j\left(x\right)|C_\mathrm{b},C_\mathrm{a}\right)$ is a monotonically decreasing function of C_b and a monotonically increasing function of C_a for $0<y<\frac{1}{2}$, as can be seen

from (A.15) and (A.16).

$$\begin{aligned}
\frac{\mathrm{E}\left(T_j\left(x\right)\right)}{g} &\geq \sum_{C_\mathrm{b}\in\mathcal{A}\cap C_\mathrm{a}\in\mathcal{B}} P^\mathrm{N}\left(C_\mathrm{b},C_\mathrm{a}\right)\cdot \mathrm{E}\left(T_j\left(x\right)|C_\mathrm{b},C_\mathrm{a}\right)\cdot\frac{1}{g} \\
&\geq (1-\epsilon)\cdot \mathrm{E}\left(T_j\left(x\right)|C_\mathrm{b}=P_X^\mathrm{N}\left(x\right)\cdot g\cdot y\cdot(1+\delta)\right., \\
&\qquad\qquad \left. C_\mathrm{a}=P_X^\mathrm{N}\left(x\right)\cdot g\cdot(1-y)\cdot(1-\delta)\right)\cdot\frac{1}{g} \qquad \text{(A.25)} \\
&\geq (1-\epsilon)\frac{1}{2}\cdot\frac{g^2y^2\left(1+\delta\right)+g^2\left(1-y\right)^2\left(1+\delta-2\delta\right)}{g\left(1+\delta\right)}\cdot\frac{1}{g} \qquad \text{(A.26)} \\
&= (1-\epsilon)\frac{1}{2}\cdot\left(\frac{j^2+\left(g-j\right)^2}{g}-\frac{\left(g-j\right)^2\cdot 2\delta}{g\cdot\left(1+\delta\right)}\right)\frac{1}{g} \qquad \text{(A.27)}
\end{aligned}$$

Raising g to infinity, ϵ will converge to zero, and δ can be selected arbitrarily small, causing (A.22) and (A.25) to coincide. Hence, we have

$$\lim_{g\to\infty}\ \frac{1}{g}\cdot\left|\mathrm{E}\left(T_j\left(x\right)\ |l_\mathrm{A}=0\right)-\frac{1}{2}\left(\frac{j^2+\left(g-j\right)^2}{g}\right)\right|=0 \qquad \text{(A.28)}$$

This completes the proof of theorem 2.1.

A.2 Proof of Theorem 2.2: Expected Value of the Average Index Difference in Case of i.i.d. Symbol Generation

The general approximation (2.12) yields a rather demanding condition (2.14). However, in case of i.i.d. generation, we may derive an expression of expected value valid for any $P_X^\mathrm{N}\left(x\right)>0$. Consider a sequence of length g. With a generation probability of $P_X^\mathrm{N}\left(x\right)$, the expected value of (2.6) may be written as a sum of expected values:

$$\mathrm{E}\left(T_j(x)|l_\mathrm{A}=0\right)=\sum_{C_\mathrm{b}+C_\mathrm{a}=0}^{g-1}\mathrm{E}\left(T_j(x)|l_\mathrm{A}=0,C_\mathrm{b}+C_\mathrm{a}\right)\cdot P\left(C_\mathrm{b}+C_\mathrm{a}\right) \qquad \text{(A.29)}$$

The conditional expected value on the right side of (A.29) may be rewritten as follows in case of $C_\mathrm{b}+C_\mathrm{a}>0$

$$\begin{aligned}
&\mathrm{E}\left(T_j(x)|l_\mathrm{A}=0,C_\mathrm{b}+C_\mathrm{a}\right)= \\
&= \frac{1}{C_\mathrm{b}+C_\mathrm{a}}\cdot\mathrm{E}\left(\sum_{o=1}^{C_\mathrm{b}}\left|j-j_o^\mathrm{b}\right|+\sum_{q=1}^{C_\mathrm{a}}\left|j-j_q^\mathrm{a}\right|\ |C_\mathrm{b}+C_\mathrm{a}\right) \qquad \text{(A.30)} \\
&= \frac{1}{C_\mathrm{b}+C_\mathrm{a}}\cdot\mathrm{E}\left(\sum_{o=1}^{j-1}v\left(j-o\right)\cdot o+\sum_{q=1}^{g-j}v\left(j+q\right)\cdot q|C_\mathrm{b}+C_\mathrm{a}\right) \qquad \text{(A.31)}
\end{aligned}$$

Here, $v(p)$ represents a binary variable signalizing wether or not a symbol x has been generated at index p. Because of i.i.d. symbol generation, those variables obey an i.i.d. distribution *with respect to all possible occurrence combinations for the fixed value* $C_\mathrm{b}+C_\mathrm{a}$:

$$P(v(p)=1|C_\mathrm{b}+C_\mathrm{a}) = \frac{C_\mathrm{b}+C_\mathrm{a}}{g-1} \tag{A.32}$$

$$P(v(p)=0|C_\mathrm{b}+C_\mathrm{a}) = 1-\frac{C_\mathrm{b}+C_\mathrm{a}}{g-1} \tag{A.33}$$

Thus, (A.31) may be resolved to

$$\begin{aligned}
&\mathrm{E}(T_j(x)|l_\mathrm{A}=0, C_\mathrm{b}+C_\mathrm{a}) = \\
&= \frac{1}{C_\mathrm{b}+C_\mathrm{a}} \cdot \mathrm{E}\left(\sum_{o=1}^{C_\mathrm{b}} \left|j-j_o^\mathrm{b}\right| + \sum_{q=1}^{C_\mathrm{a}} \left|j-j_q^\mathrm{a}\right| | C_\mathrm{b}+C_\mathrm{a}\right) \\
&= \frac{1}{C_\mathrm{b}+C_\mathrm{a}} \cdot \left(\sum_{o=1}^{j-1} \mathrm{E}(v(j-o)|C_\mathrm{b}+C_\mathrm{a}) \cdot o + \sum_{q=1}^{g-j} \mathrm{E}(v(j+q)|C_\mathrm{b}+C_\mathrm{a}) \cdot q\right) \\
&= \frac{1}{C_\mathrm{b}+C_\mathrm{a}} \cdot \left(\sum_{o=1}^{j-1} \frac{C_\mathrm{b}+C_\mathrm{a}}{g-1} \cdot o + \sum_{q=1}^{g-j} \frac{C_\mathrm{b}+C_\mathrm{a}}{g-1} \cdot q\right) \\
&= \frac{1}{g-1} \left(\cdot \sum_{o=1}^{j-1} o + \sum_{q=1}^{g-j} q\right) \\
&= \frac{\frac{1}{2}\cdot(j-1)\cdot j + \frac{1}{2}\cdot(g-j)\cdot(g-j+1)}{g-1} \\
&= \frac{j^2+(g-j)^2+(g-2j)}{2(g-1)} \\
&= \left(\frac{j^2+(g-j)^2}{2(g-1)} + \frac{g-2j}{2(g-1)}\right)
\end{aligned} \tag{A.34}$$

Inserting (A.34) into (A.29), we get

$$\begin{aligned}
&\mathrm{E}\left(T_j(x)|l_\mathrm{A}=0\right) = \\
&= \sum_{C_\mathrm{b}+C_\mathrm{a}=0}^{g-1} \mathrm{E}\left(T_j(x)|l_\mathrm{A}=0, C_\mathrm{b}+C_\mathrm{a}\right)\cdot P\left(C_\mathrm{b}+C_\mathrm{a}\right) \\
&= \left(1-P\left(C_\mathrm{b}+C_\mathrm{a}=0\right)\right)\cdot\left(\frac{j^2+(g-j)^2}{2\left(g-1\right)}+\frac{g-2j}{2\left(g-1\right)}\right) \\
&= \left(1-\left(1-P_X^\mathrm{N}\left(x\right)\right)^{g-1}\right)\cdot\left(\frac{j^2+(g-j)^2}{2\left(g-1\right)}+\frac{g-2j}{2\left(g-1\right)}\right)
\end{aligned} \tag{A.35}$$

This completes the proof. With $P_X^\mathrm{N}\left(x\right)$ fixed, (A.35) will converge to (2.12) for $g\to\infty$.

A.2.1 Proof of Corollary 2.1: Bound of the Expected Value of the Average Index Difference in Case of i.i.d. Symbol Generation

We first present the following useful lower bound of (A.34):

$$\begin{aligned}
\mathrm{E}\left(T_j(x)|l_\mathrm{A}=0, C_\mathrm{b}+C_\mathrm{a}\right) &= \left(\frac{j^2+(g-j)^2}{2\left(g-1\right)}+\frac{g-2j}{2\left(g-1\right)}\right) \\
&= \frac{g}{g-1}\cdot\left(\frac{j^2+(g-j)^2}{2g}+\frac{g-2j}{2g}\right) \\
&> \frac{j^2+(g-j)^2}{2g}+\frac{g}{g-1}\cdot\frac{g-2j}{2g}
\end{aligned} \tag{A.36}$$

On the right side, we note the symmetry of the absolute value of the second term (which is negative for $j\geq\frac{g}{2}$ and positive otherwise) and the symmetry of the non-negative first term with respect to $j=\frac{g}{2}$. An upper bound of the absolute difference between (A.34)

and (2.12) may be deduced as follows:

$$
\begin{aligned}
&\left|\mathrm{E}\left(T_j(x)|l_\mathrm{A}=0,C_\mathrm{b}+C_\mathrm{a}\right)-\frac{j^2+(g-j)^2}{2g}\right| \\
&= \left|\left[\frac{g}{g-1}\cdot\left(\frac{j^2+(g-j)^2}{2g}+\frac{g-2j}{2g}\right)\right]-\frac{j^2+(g-j)^2}{2g}\right| \\
&= \left|\left[\frac{g}{g-1}\cdot\left(\frac{j^2+(g-j)^2}{2g}+\frac{g-2j}{2g}\right)\right]-\left[\frac{g-1}{g-1}\cdot\left(\frac{j^2+(g-j)^2}{2g}\right)\right]\right| \\
&\leq \frac{1}{g-1}\left|[g-(g-1)]\cdot\left(\frac{j^2+(g-j)^2}{2g}\right)\right|+\left|g\cdot\frac{g-2j}{2g}\right| \qquad \text{(A.37)} \\
&\leq \frac{1}{g-1}\cdot\left(\frac{g}{2}+\frac{g}{2}\right) \\
&= \frac{g}{g-1} \qquad \text{(A.38)} \\
&\leq 2 \qquad \text{(A.39)}
\end{aligned}
$$

The decomposition of (A.37) is valid because of the symmetry properties of the terms within (A.36). (A.38) converges to 1 for $g \to \infty$, while the relative difference is bounded by

$$
\begin{aligned}
\left|\frac{\mathrm{E}\left(T_j(x)|l_\mathrm{A}=0,C_\mathrm{b}+C_\mathrm{a}\right)-\frac{j^2+(g-j)^2}{2g}}{\frac{j^2+(g-j)^2}{2g}}\right| &\leq \frac{4}{g-1} \\
&\leq 4 \qquad \text{(A.40)}
\end{aligned}
$$

and will converge to zero.

Using above relative bound, the overall expected value of the average index difference may be bounded by

$$
\left|\frac{\mathrm{E}\left(T_j(x)|l_\mathrm{A}=0\right)-\frac{j^2+(g-j)^2}{2g}}{\frac{j^2+(g-j)^2}{2g}}\right| \leq \frac{4}{g-1}+\left(1-P_X^\mathrm{N}(x)\right)^{g-1}\cdot\left(\frac{g+3}{g-1}\right), \qquad \text{(A.41)}
$$

completing the proof. If $P_X^{\mathrm{N}}(x)$ is not fixed, for suitably large g, the approximation

$$\begin{aligned}
&\mathrm{E}\left(T_j(x)|l_{\mathrm{A}}=0\right) = \\
&= \left(1-\left(1-P_X^{\mathrm{N}}(x)\right)^{g-1}\right)\cdot\left[\frac{g}{g-1}\cdot\left(\frac{j^2+(g-j)^2}{2\cdot g}+\frac{g-2j}{2g}\right)\right] \\
&\approx \left(1-\left(1-P_X^{\mathrm{N}}(x)\right)^{g-1}\right)\cdot\left(\frac{j^2+(g-j)^2}{2g}\right) \\
&\approx \left(1-\exp^{-g\cdot P_X^{\mathrm{N}}(x)}\right)\cdot\left(\frac{j^2+(g-j)^2}{2g}\right)
\end{aligned} \tag{A.42}$$

may be used, with the second approximation being the Poisson approximation of the binomial distribution.

The preceding arguments show that for a fixed $C_{\mathrm{b}}+C_{\mathrm{a}}\neq 0$, every index within the overall sequence of length g (except j, of course) has the same chance of contributing to the sum of index differences of (A.30). If we select any of the $C_{\mathrm{b}}+C_{\mathrm{a}}$ occurrences at random, the index of this occurrence is approximately uniformly distributed within $[1,g]$. This is true because for i.i.d. generation, the probability distribution of the index j_i of the i^{th} occurrence of x counting from the start of a sequence of length $g-1$ for a fixed total number of occurrences $C_{\mathrm{b}}+C_{\mathrm{a}}\neq 0$ may be expressed as

$$\begin{aligned}
P(j_i) &= \frac{\binom{j_i-1}{i-1}\cdot\binom{g-1-j_i}{C_{\mathrm{b}}+C_{\mathrm{a}}-i}}{\binom{g-1}{C_{\mathrm{b}}+C_{\mathrm{a}}}} \\
&\quad j_i\geq i\ ;\quad j_i\leq g-1-(C_{\mathrm{b}}+C_{\mathrm{a}}-i)\,,
\end{aligned} \tag{A.43}$$

the so-called negative hypergeometric distribution [81]. The expected value of j_i is

$$\mathrm{E}(j_i) = i\cdot\frac{g}{C_{\mathrm{b}}+C_{\mathrm{a}}+1}. \tag{A.44}$$

This shows the correctness of the approximation.

The situation is analogous to the subsequent urn experiment without replacement. The urn is filled with $g-1$ balls numbered form 1 to g, excluding the index j. We draw $C_{\mathrm{b}}+C_{\mathrm{a}}$ balls without returning them. The numbers of the drawn balls are the occurrences of the particular symbol x found at index j within the sequence of length g. We do not care about the order of the drawn indices, only about their value. The index of each draw roughly obeys a uniform distribution between 1 and g. If we draw all the balls from the urn, the indices located within a subsection of the interval $[1,g]$ are uniformly distributed within the new sequence.

Using the approximation introduced above, the expected value of the average index difference for a fixed $C_\mathrm{b} + C_\mathrm{a}$ may be approximated by

$$\begin{aligned}
&\mathrm{E}\left(T_j(x)|l_\mathrm{A} = 0, C_\mathrm{b} + C_\mathrm{a}\right) = \\
&= \frac{1}{C_\mathrm{b} + C_\mathrm{a}} \cdot \mathrm{E}\left(\sum_{o=1}^{C_\mathrm{b}} \left|j - j_o^\mathrm{b}\right| + \sum_{q=1}^{C_\mathrm{a}} \left|j - j_q^\mathrm{a}\right|\right) \\
&\approx \frac{1}{C_\mathrm{b} + C_\mathrm{a}} \cdot \sum_{q=1}^{C_\mathrm{b}+C_\mathrm{a}} \mathrm{E}\left(\left|j - \mathrm{Uniform}\left(1, g\right)\right|\right) \\
&\approx \frac{1}{C_\mathrm{b} + C_\mathrm{a}} \sum_{q=1}^{C_\mathrm{b}+C_\mathrm{a}} \left(\frac{1}{2} \cdot j \cdot \frac{j}{g} + \frac{1}{2} \cdot (g - j) \cdot \frac{g - j}{g}\right) \\
&= \frac{j^2 + (g - j)^2}{2g}
\end{aligned} \tag{A.45}$$

While this expected value is identical to the one (A.34) converges to, the variance of this approximation provides an upper bound of the variance of the actual function, a fact that will be used for the subsequent proof.

A.3 Proof of Theorem 2.3: Upper Bound of the Variance of the Average Index Difference in Case of No Anomaly and i.i.d. Symbol Generation

While still dealing with small generation probabilities, we assume

$$P_X^{\mathrm{N}}(x) \geq \frac{5}{g}. \tag{A.46}$$

This enables the approximation the expected value by (2.12), as well as the bounding of the variance of (2.6) by a weighted sum of partial variances, ignoring the case of $C_{\mathrm{b}} + C_{\mathrm{a}} = 0$:

$$\begin{aligned}
&\mathrm{Var}\left(T_j(x)|l_{\mathrm{A}} = 0\right) = \\
&= \sum_{C_{\mathrm{b}}+C_{\mathrm{a}}=1}^{g-1} \mathrm{E}\left(\left(\left(T_j(x)|l_{\mathrm{A}} = 0, C_{\mathrm{b}} + C_{\mathrm{a}}\right) - \mathrm{E}\left(T_j(x)|l_{\mathrm{A}} = 0\right)\right)^2\right) \cdot P\left(C_{\mathrm{b}} + C_{\mathrm{a}}\right)
\end{aligned} \tag{A.47}$$

We start deducing an upper bound of the partial variance :

$$\begin{aligned}
&\mathrm{E}\left(\left[\left(T_j(x)|l_{\mathrm{A}} = 0, C_{\mathrm{b}} + C_{\mathrm{a}}\right) - \mathrm{E}\left(T_j(x)|l_{\mathrm{A}} = 0\right)\right]^2\right) \\
&= \mathrm{E}\left(\left[\frac{1}{C_{\mathrm{b}} + C_{\mathrm{a}}} \cdot \left(\sum_{o=1}^{C_{\mathrm{b}}} \left|j - j_o^{\mathrm{b}}\right| + \sum_{q=1}^{C_{\mathrm{a}}} \left|j - j_q^{\mathrm{a}}\right|\right) - \mathrm{E}\left(T_j(x)|l_{\mathrm{A}} = 0\right)\right]^2\right) \\
&\leq \mathrm{E}\left(\left[\frac{1}{C_{\mathrm{b}} + C_{\mathrm{a}}} \cdot \sum_{q=1}^{C_{\mathrm{b}}+C_{\mathrm{a}}} \left|j - \mathrm{Uniform}\left(1, g\right)\right| - \frac{C_{\mathrm{b}} + C_{\mathrm{a}}}{C_{\mathrm{b}} + C_{\mathrm{a}}} \cdot \mathrm{E}\left(T_j(x)| = 0\right)\right]^2\right) \\
&= \left(\frac{1}{C_{\mathrm{b}} + C_{\mathrm{a}}}\right)^2 \cdot \mathrm{E}\left(\left[\sum_{q=1}^{C_{\mathrm{b}}+C_{\mathrm{a}}} \left(\left|j - \mathrm{Uniform}\left(1, g\right)\right| - \mathrm{E}\left(T_j(x)|l_{\mathrm{A}} = 0\right)\right)\right]^2\right) \\
&= \left(\frac{1}{C_{\mathrm{b}} + C_{\mathrm{a}}}\right)^2 \cdot \sum_{q=1}^{C_{\mathrm{b}}+C_{\mathrm{a}}} \mathrm{E}\left(\left[\left|j - \mathrm{Uniform}\left(1, g\right)\right| - \mathrm{E}\left(T_j(x)|l_{\mathrm{A}} = 0\right)\right]^2\right),
\end{aligned} \tag{A.48, A.49}$$

which continues as

$$
\begin{aligned}
&\mathrm{E}\left(\left[(T_j(x)|l_\mathrm{A}=0, C_\mathrm{b}+C_\mathrm{a}) - \mathrm{E}\left(T_j(x)|l_\mathrm{A}=0\right)\right]^2\right) \\
&\quad\leq \quad \left(\frac{1}{C_\mathrm{b}+C_\mathrm{a}}\right)^2 \cdot \sum_{q=1}^{C_\mathrm{b}+C_\mathrm{a}} \mathrm{E}\left(\left[\left|j - \mathrm{Uniform}\left(1,g\right)\right| - \mathrm{E}\left(T_j(x)|l_\mathrm{A}=0\right)\right]^2\right) \\
&\quad\leq \quad \frac{1}{C_\mathrm{b}+C_\mathrm{a}} \cdot \frac{1}{g} \cdot \left(\frac{g^2}{4} + \sum_{q=1}^{j-1}\left[q - \mathrm{E}\left(T_j(x)|l_\mathrm{A}=0\right)\right]^2\right. \\
&\qquad \left. + \sum_{q=1}^{g-j}\left[q - \mathrm{E}\left(T_j(x)|l_\mathrm{A}=0\right)\right]^2\right) \qquad \text{(A.50)} \\
&\quad\leq \quad \frac{1}{C_\mathrm{b}+C_\mathrm{a}} \cdot \frac{1}{g} \cdot \left(\left[\frac{g}{2}+2\right]^2 + \sum_{q=1}^{j-1}\left[\left|q - \frac{j^2+(g-j)^2}{2g}\right| + 2\right]^2\right. \\
&\qquad \left. + \sum_{q=1}^{g-j}\left[\left|q - \frac{j^2+(g-j)^2}{2g}\right| + 2\right]^2\right) \qquad \text{(A.51)}
\end{aligned}
$$

Approximation (A.48) follows the line of argument used by the proof the expected value, treating the indices of the occurrences of x as independent identically distributed random variables, as described in (A.45). (A.49) is valid because the expected value of the average index difference given by (2.12) is equal to the expected value of the index difference between the j and an index randomly selected from $[1, g]$. (A.50) resolves the expected value and utilizes the fact that the $C_\mathrm{b} + C_\mathrm{a}$ sum terms are identical. As we continue the deduction, we restrict the range of j to $\left[0, \frac{g}{2}\right]$ for certain transformations. This does not

cause problems because of the symmetry of (A.51) with respect to $j = \frac{g}{2}$.

$$
\begin{aligned}
&\mathrm{E}\left(\left[T_j\left(x\right) - \mathrm{E}\left(T_j\left(x\right)\right)\right]^2 | l_{\mathrm{A}} = 0, C_{\mathrm{b}} + C_{\mathrm{a}}\right) \\
&\leq \frac{1}{C_{\mathrm{b}} + C_{\mathrm{a}}} \cdot \frac{1}{g} \cdot \left(\left[\frac{g}{2} + 2\right]^2 + \sum_{q=1}^{j-1} \left[\left| q - \frac{j^2 + (g-j)^2}{2g} \right| + 2 \right]^2 \right. \\
&\quad \left. + \sum_{q=1}^{g-j} \left[\left| q - \frac{j^2 + (g-j)^2}{2g} \right| + 2 \right]^2 \right) \\
&\leq \frac{1}{C_{\mathrm{b}} + C_{\mathrm{a}}} \cdot \frac{1}{g} \cdot \left(\frac{g^2}{2} + \sum_{q=1}^{j-1} \left[\left| q - \frac{g}{2} \right| + 2 \right]^2 + \sum_{q=1}^{g-j} \left[\left| q - \frac{g}{2} \right| + 2 \right]^2 \right) \qquad \text{(A.52)} \\
&\leq \frac{1}{C_{\mathrm{b}} + C_{\mathrm{a}}} \cdot \frac{1}{g} \cdot \left(\frac{g^2}{2} + \sum_{q=\frac{g}{2}-j+3}^{\frac{g}{2}+1} q^2 + \sum_{q=1}^{\frac{g}{2}+1} q^2 + \sum_{q=1}^{\frac{g}{2}-j+2} q^2 \right) \\
&= \frac{1}{C_{\mathrm{b}} + C_{\mathrm{a}}} \cdot \frac{1}{g} \cdot \left(g^2 + 2g + 2 + 2 \sum_{q=1}^{\frac{g}{2}} q^2 \right) \\
&= \frac{1}{C_{\mathrm{b}} + C_{\mathrm{a}}} \cdot \left(g + 2 + \frac{2}{g} + \left[\frac{2}{g} \cdot \frac{\frac{g}{2} \cdot \left(\frac{g}{2} + 1\right) \cdot (g+1)}{6} \right] \right) \\
&\leq \frac{1}{C_{\mathrm{b}} + C_{\mathrm{a}}} \cdot \left(g + 2 + \frac{2}{g} + \frac{g^2 + 3g + 2}{12} \right) \qquad \text{(A.53)} \\
& \qquad \text{(A.54)}
\end{aligned}
$$

The upper bound of (A.52) is founded upon the following argument. The function

$$
S(p) = \sum_{q=1}^{z} \left(|q - p| + y\right)^2 \quad z > 0,\ y > 0 \qquad \text{(A.55)}
$$

is symmetric with respect a single global minimum at $p = \frac{z}{2}$, and is monotonically decreasing for $p < \frac{z}{2}$ and monotonically increasing for $p > \frac{z}{2}$. This means that for two values p_1, p_2, $S(p_1) \geq S(p_2)$ holds if $\left|\frac{z}{2} - p_1\right| \geq \left|\frac{z}{2} - p_2\right|$ is met. One can easily show that the inequalities

$$
\begin{aligned}
\left| \frac{j}{2} - \frac{g}{2} \right| &\geq \left| \frac{j}{2} - \frac{j^2 + (g-j)^2}{2g} \right| \\
\left| \frac{g-j}{2} - \frac{g}{2} \right| &\geq \left| \frac{g-j}{2} - \frac{j^2 + (g-j)^2}{2g} \right| \\
\forall j \in [1, g] & \qquad \text{(A.56)}
\end{aligned}
$$

hold. Thus, the bounding is valid.

Inserting (A.54) into (A.47), we get

$$\begin{aligned}
\mathrm{Var}\left(T_j(x)|l_\mathrm{A}=0\right) &\leq \sum_{C_\mathrm{b}+C_\mathrm{a}=1}^{g-1} P\left(C_\mathrm{b}+C_\mathrm{a}\right)\cdot \mathrm{Var}\left(T_j(x)|l_\mathrm{A}=0, C_\mathrm{b}+C_\mathrm{a}\right) \\
&\leq \sum_{C_\mathrm{b}+C_\mathrm{a}=1}^{g-1} \left(P\left(C_\mathrm{b}+C_\mathrm{a}\right)\cdot \frac{1}{C_\mathrm{b}+C_\mathrm{a}}\cdot\left[g+2+\frac{2}{g}+\frac{g^2+3g+2}{12}\right]\right) \\
&= \left(1+\frac{2}{g}+\frac{2}{g^2}+\frac{g+3+\frac{2}{g}}{12}\right)\cdot \sum_{C_\mathrm{b}+C_\mathrm{a}=1}^{g-1}\left(P\left(C_\mathrm{b}+C_\mathrm{a}\right)\cdot\frac{g}{C_\mathrm{b}+C_\mathrm{a}}\right) \\
&\leq \frac{g+39+\frac{2}{g}}{12\cdot P_X^\mathrm{N}(x)} \qquad \text{(A.57)}
\end{aligned}$$

This completes the proof.

Bibliography

[1] Stefansky, W.: Rejecting outliers in factorial designs. Technometrics, Vol. 14, No. 2, pp. 469-479, 1972.

[2] Lazarevic, A., Kumar, V., Srivastava, J.: Intrusion Detection: A Survey. Massive Computing, Vol. 5, Chapter 2, Springer, 2005.

[3] Sudjianto, A., Nair, S., Yuan, M., Zhang, A., Kern, D., Cela-Diaz, F.: Statistical Methods for Fighting Financial Crimes. Technometrics, Vol. 52, No. 1, pp. 5-19, February 2010.

[4] Chang, L., Chen, C.: Detection of Ocean Surface Anomaly Using Optical Satellite Image. The 30^{th} Asian Conference on Remote Sensing, 2009.

[5] Minhas, A. S., Redd, M. R.: Neural Network based Approach for Anomaly Detection in the Lungs Region by Electrical Impedance Tomography. Physiological Measurements, Vol. 26, No. 4, pp. 489-502, August 2005.

[6] Grubbs, F. E.: Procedures for Detecting Outlying Observations in Samples. Technometrics, Vol. 11, No. 1, pp. 1-21, 1969.

[7] Hawkins, D. M.: Identification of Outliers. Chapman and Hall, 1980.

[8] Lian, D. et al.: Cluster-Based Outlier Detection. Annals of Operations Research, Vol. 168, No. 1, pp. 151-168, 2009.

[9] Chandola, V., Banerjee, A., Kumar, V.: Anomaly Detection - A Survey. ACM Computing Surveys, Vol. 41, No. 3, Article 15, July 2009.

[10] Schoelkopf, B., Platt, J. C., Shawe-Taylor, J. C., Smola, A. J., Williamson, R. C.: Estimating the Support of a High-Dimensional Distribution. Neural Computation, Vol. 13, No. 7, pp. 1443-1471, 2001.

[11] Vapnik, V. N.: The Nature of Statistical Learning Theory. Springer, 1995.

[12] Hoffman, T., Schoelkopf, B., Smola, A.: Kernel Methods in Machine Learning. The Annals of Machine Learning, Vol. 36, No. 3, pp. 1171-1220, January 2000.

[13] Rajasegarar, S., Leckie, C., Bezdek, J. C., Palaniswami, M.: Centered Hyperspherical and Hyperellipsoidal One-Class Support Vector Machines for Anomaly Detection in Sensor Networks. IEEE Transactions on Information Forensics and Security, Vol. 5, No. 3, pp. 518-533, September 2010.

[14] Tsang, I. V., Kwok, J. T., Cheung, P.: Core Vector Machines: Fast SVM Training on Very Large Data Sets. Journal of Machine Learning Research, Vol. 6, pp. 363-392, January 2005.

[15] Markou M., Singh, S.: Novelty Detection: A Review - Part 2: Neural network Based Approaches. Signal Processing, Vol. 83, No. 12, pp. 2481-2497, December 2003.

[16] Cohen, W. W.: Fast Effective Rule Induction. Proceedings of the 12^{th} International Conference on Machine Learning, pp. 115-123, 1995.

[17] Cohen, W. W.: Learning Trees and Rules with Set-Valued features. Proceedings of the 13^{th} National Conference on Artificial Intelligence and 8^{th} Innovative Applications of Artificial Intelligence Conference, Vol. 1, pp. 709-716, 1996.

[18] Shoemaker, C., Ruiz, C.: Association Rule Mining Algorithms for Set-valued Data. Proceedings of the 4^{th} International Conference on Intelligent Data Engineering and Automated Learning. Lecture Notes on Computer Science, Vol. 2690, Springer, 2003.

[19] Jolliffe, I.: Principal Component Analysis. Springer, 2002.

[20] Shyu, M., Chen, S., Sarinnapakorn, K., Chang, L.: A Novel Anomaly Detection Scheme Based on Principal Component Classifier. Proceedings of the IEEE Foundations and New Directions of Data Mining Workshop, in conjunction with the 3^{rd} IEEE International Conference on Data Mining (ICDM03), pp. 172-179, November 2003.

[21] Kwitt, R., Hofmann, U.: Unsupervised Anomaly Detection in Network Traffic by Means of Robust PCA. Proceedings of the International Multi-Conference on Computing in the Global Information Technology, pp. 37-41, 2007.

[22] Eskin, E., Arnold, A., Prereau, M., Portony, L., Stolfo, S.: A Geometric Framework for Unsupervised Anomaly Detection: Detecting Intrusions in Unlabeled Data. Applications of Data Mining in Computer Security. Kluwer Academic Publishers, 2002.

[23] Breunig, M. M., Kriegel, H.-P., Ng, R. T., Sander, J.: LOF: Identifying Density-Based Local Outliers. Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 93-104, 2000.

[24] Xu, R., Wunsch, D.C.: Clustering. IEEE Computational Intelligence Society, 2009.

[25] Gaddam, S., Phoha, V., Balagani, K.: K-Means+ID3: A Novel Method for Supervised Anomaly Detection by Cascading K-Means Clustering and ID3 Decision Tree Learning Methods. IEEE Transactions on Knowledge and Data Engineering, Vol. 19, No. 3, pp. 345-354, March 2007.

[26] Gath, I., Geva, A. B.: Unsupervised Optimal Fuzzy Clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 11, No. 7, pp. 773-780, July 1989.

[27] Moshtaghi, M., Rajasegarar, S., Leckie, C., Karunasekera, S.: Anomaly Detection by Clustering Ellipsoids in Wireless Sensor Networks. 5^{th} International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), pp. 331-336, December 2009.

[28] Leung, K., Leckie, C.: Unsupervised Anomaly Detection in Network Intrusion Detection Using Clusters. Proceedings of the 28^{th} Australasian Computer Security Conference, pp. 333-342, 2005.

[29] Veracini, T., Matteoli, S., Diani, M., Corsini, G.: Fully Unsupervised Learning of Gaussian Mixtures for Anomaly Detection in Hyperspectral Imagery. 9^{th} International Conference on Intelligent Systems Design and Applications, pp. 596-601, 2009.

[30] Dempster, S. P., Laird, N. M., Rubin, D. B.: Maximum Likelihood from Incomplete Data Via the EM Algorithm. Journal of the Royal Statistical Society Series B (Methodological), Vol. 39, No. 1, pp. 1-38, 1977.

[31] Yamanishi, K., Takeuchi, J.: Discovering Outlier Filtering Rules from Unlabeled Data: Combining a Supervised Learner with an Unsupervised Learner. Proceedings of the 7^{th} ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 389-394, 2001.

[32] Parzen, E.: On the Estimation of a Probability Density Function and Mode. Annals of Mathematical Statistics, Vol. 33, pp. 1065-1076, 1962.

[33] Yamanishi, K., Takeuchi, J., Williamas, G., Milne, P.: On-line Unsupervised Outlier Detection Using Finite Mixtures with Discounting Learning Algorithms. Data Mining and Knowledge Discovery Journal, Vol. 8, No. 3, pp. 275-300, May 2004.

[34] Chen, D., Shao, X., Hu, B., Su, Q.: Simultaneous Wavelength Selection and Outlier Detection in Multivariate Regression of Near-Infrared Spectra. Analytical Sciences, Vol. 21, No. 2, pp. 161-167, 2005.

[35] Kolmogorov, A. N.: Three Approaches to the Quantitative Definition of Information. Problems of Information Transmission, Vol. 1, No. 1, pp. 1-7, 1965.

[36] Callegari, C., Giordano, S., Pagano, M.: On the Use of Compression Algorithms for Network Anomaly Detection. IEEE International Conference on Communications 2009 (ICC '09), pp. 1-5, June 2009.

[37] Lee, W., Xiang, D.: Information-Theoretic Measures for Anomaly Detection. Proceedings of the 2001 IEEE Symposium on Security and Privacy (S&P 2001), pp. 130-143, 2001.

[38] Yamanishi, K., Maruyama, Y.: Dynamic Model Selection with Its Applications to Novelty Detection. IEEE Transactions on Information Theory , Vol. 53, No. 6, pp. 2180-2189, June 2007.

[39] He, Z., Xu, X., Deng, S.: An Optimization Model for Outlier Detection in Categorical Data. Proceedings of the International Conference on Intelligent Computing, Lecture Notes in Computer Science, Vol. 3644, Springer, 2005.

[40] Sculley, D., Brodley, C. E.: Compression and Machine Learning: A New Perspective on Feature Space Vectors. Proceedings of the Data Compression Conference 2006 (DCC 2006), pp. 332-341, March 2006.

[41] Li, M., Chen, X., Li, X., Ma, B., Vitanyi, P. M. B.: The Similarity Metric. IEEE Transactions on Information Theory, Vol. 50, No. 12, pp. 3250-3264, December 2004.

[42] Chandola, V., Mithal, V., Kumar, V.: A Comparative Evaluation of Anomaly Detection Techniques for Sequence Data. Technical Report of the University of Minnesota, 2008.

[43] Sun, P., Chawla, S., Arunasalam, B.: Mining for Outliers in Sequential Databases. Proceedings of the SIAM Conference in Data Mining, pp. 3-14, 2006.

[44] Srivastava, A. N., Eskin, E., Noble, W. S.: Discovering System Health Anomalies Using Data Mining Techniques. Proceedings of 2005 Joint Army Navy Nasa Air Force Conference on Propulsion, 2005.

[45] Maxion, R., Townsend, T.: Masquerade Detection Augmented With Error Analysis, IEEE Transactions on Reliability, Vol. 53, No. 1, pp. 124-147, 2004.

[46] Forrest, S., Hofmeyer, S., Somayai, A., Longstaff, T.: A Sense of Self for Unix Processes. Proceedings of the IEEE Symposium on Security and Privacy, pp. 120-128, 1996.

[47] Boriah, S., Chandola, V., Kumar, V.: Similarity Measures for Categorical Data: A Comparative Evaluation. Proceedings of the 8^{th} SIAM International Conference on Data Mining, pp. 243-254, 2008.

[48] Low-Kam, C., Laurent, A., Teisseire, M.: Detection of Sequential Outliers Using a Variable Length Markov Model. Seventh International Conference on Machine Learning and Applications, pp. 571-576, 2008.

[49] Ron, D., Singer, Y., Tishby, N.: The Power of Amnesia: Learning Probabilistic Automata with Variable Memory Length. Machine Learning, Vol. 25, No. 2, pp. 117-149, 1996.

[50] Agrawal, R., Srikant, R.: Mining Sequential Patterns. Eleventh International Conference on Data Engineering, pp. 3-14, 1995.

[51] Leslie, C. S., Eskin, E., Noble, W. S.: The Spectrum Kernel: A String Kernel for SVM Protein Classification. Proceedings of the Pacific Biocomputing Symposium, pp. 564-575, 2002.

[52] Kennel, M.: Statistical Test for Dynamical Nonstationarity in Observed Time-Series Data. Physical Review E, Vol. 56, No. 1, pp. 316-321, 1997.

[53] Rieke, C., Sternickel, K., Andrzejak, R., Elger, C., David, P., Lehnertz, K.: Measuring Nonstationarity by Analyzing the Loss of Recurrence in Dynamical Systems. Physical Review Letters, Vol. 88, No. 24, Article No. 244102, 2002.

[54] Rieke, C., Andrzejak, R., Mormann, F., Lehnertz, K.: Improved Statistical Test for Nonstationarity Using Recurrence Time Statistics. Physical Review E, Vol. 69, No. 4, Article No. 046111, 2004.

[55] Skudlarek S., Yamamoto, H.: Anomaly Detection Using Time Index Differences of Identical Symbols with and without Training Data. Proceedings of the 5^{th} International Conference on Advanced Data Mining and Applications, pp. 619-626, 2009.

[56] Skudlarek S., Yamamoto, H.: Representative Sequence Selection in Unsupervised Anomaly Detection using Spectrum Kernel with Theoretical Parameter Setting. Proceedings of the 2010 International Conference on Machine Learning and Cybernetics, pp. 2099-2104, 2010.

[57] Cover, T., Thomas, J.: Elements of Information Theory. Wiley & Sons, 2006.

[58] Hirata, M., Saussol, B., Vaienti, S.: Statistics of Return Times: A General Framework and New Applications. Communications in Mathematical Physics, Vol. 206, No. 1, pp. 33-55, 1999.

[59] Abadi, M.: Exponential Approximation for Hitting Times in Mixing Processes. Mathematical Physics Electronic Journal, Vol. 7, No. 2, pp. 19, 2001.

[60] Fawcett, T.: An Introduction to ROC Analysis. Pattern Recognition Letters, Vol. 27, No. 8, pp. 861-874, June 2006.

[61] Huber, P. J.: Robust Statistics. Wiley, 1981.

[62] Schonlau, M., DuMouchel, W., Ju, W., Karr, A., Theus, M., Vardi,Y.: Computer Anomaly: Detecting Masquerades. Statistical Science, Vol. 16, No. 1, pp. 58-74, 2001.

[63] Bertachini, M., Fierens, P.I.: A Survey on Masquerader Detection Approaches. Proceedings of the Fifth Ibero-American Congress on Information Security, 2009.

[64] Chen, Y., Dang, X., Peng, H., Bart, H. L.: Outlier Detection with the Kernelized Spatial Depth Function. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 31, No. 2, pp. 288-305, February 2009.

[65] Jaakkola,T., Diekhans, M., Haussler, D.: A Discriminative Framework for Detecting Remote Protein Homologies. Journal of Computational Biology, Vol. 7, No. 1-2, pp. 95-114, 2000.

[66] Maetschke, S., Gallagher, M., Boden, M.: A Comparison of Sequence Kernels for Localization Prediction of Transmembrane Proteins. Proceedings of the CIBCB 2007 Symposium, pp. 367-372, April 2007.

[67] Saigo, H., Vert, J.-P., Ueda, N., Akutsu, T.: Protein Homology Detection Using String Alignment Kernels. Bioinformatics, Vol. 20, No. 11, pp. 1682-1689, July 2004.

[68] Leslie, C. S., Eskin, E., Cohen, A., Weston, J., Noble, W. S.: Mismatch String Kernels for Discriminative Protein Classification. Bioinformatics, Vol. 20, No. 4, pp. 467-476, 2004.

[69] Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., Watkins, C.: Text Classification Using String Kernels. Journal of Machine Learning Research, Vol. 2, pp. 419-444, 2002.

[70] Rieck, K., Laskov, P., Mueller, K. R.: Efficient Algorithms for Similarity Measures over Sequential Data: A Look Beyond Kernels. Proceedings. of 28^{th} DAGM Symposium on Pattern Recognition, Lecture Notes in Computer Science, pp. 374-383, September 2006.

[71] Burnham K. P., Anderson D. R.: Model Selection and Multimodel Inference. Springer, 2002.

[72] Konishi, S. and Kitagawa, G.: Information Criteria and Statistical Modeling. Springer, 2008.

[73] Akaike, H.: A New Look at the Statistical Model Identification. IEEE Transactions on Automatic Control, Vol. 19, No. 6, pp. 716-723, 1974.

[74] Schwarz, G.: Estimating the Dimension of a Model. The Annals of Statistics, Vol. 6, No. 2, pp. 461-464, 1978.

[75] Sugiura, N.: Further Analysis of the Data by Akaike's Information Criterion and the Finite Corrections. Communications in Statistics, Theory and Methods, Vol. 7, No. 1, pp. 13-26, 1978.

[76] Vereshchagin, N. K., Vitanyi, P.: Kolmogorov's Structure Functions and Model Selection. IEEE Transactions on Information Theory, Vol. 50, No. 12, pp. 3265-3290, December 2004.

[77] Adriaans, P., Vitanyi, P.: Approximation of the Two-Part MDL Code. IEEE Transactions on Information Theory, Vol. 55, No. 1, pp. 444-457, January 2009.

[78] Vishvanathan, S. V. N. and Smola, A. J.: Fast kernels for string and tree matching. Kernel Methods in Computational Biology. MIT Press, pp. 113-130, 2004.

[79] Bateman, A., Birney, E., Durbin, R., Eddy, S. R., Howe, K. L. and Sonnhammer, E. L.: The Pfam Protein Families Database. Nucleic Acids Research, Vol. 28, No. 1, pp. 263-266, January 2000.

[80] Kac, M.: On the Notion of Recurrence in Discrete Stochastic Processes. Bulletin of the American Mathematical Society, Vol. 53, No. 10, pp. 1002-1010, October 1947.

[81] Schuster, E. F., Sype, W. R.: On the Negative Hypergeometric Distribution. International Journal of Mathematical Education in Science and Technology, Vol. 18, No. 3, pp. 453-459, 1987.