LN-SNE: Log-Normal Distributed Stochastic Neighbor Embedding for Anomaly Detection

Zahra Ghafoori, Sarah M. Erfani, James C. Bezdek, Life Fellow, IEEE, Shanika Karunasekera, Christopher Leckie

Abstract—We present a new unsupervised dimensionality reduction technique, called LN-SNE, for anomaly detection. LN-SNE generates a parametric embedding by means of Restricted Boltzmann Machines and uses a heavy-tail distribution to project data to a lower dimensional space such that dissimilarities between normal data and anomalies are preserved or strengthened. We compare LN-SNE to several benchmark dimensionality reduction methods on real datasets. The results suggest that LN-SNE for anomaly detection is less sensitive to the dimension of the latent space than the other methods and outperforms them in terms of accuracy. We empirically show that our technique scales near-linearly with respect to the number of dimensions and data size.

Index Terms—Unsupervised learning, dimensionality reduction, anomaly detection, outlier detection.

1 INTRODUCTION

Unsupervised Anomaly Detection (AD), also called outlier detection [1], refers to analysing unlabelled data to identify rare observations that are different from the majority of the data. Unsupervised techniques to AD have attracted a great deal of attention in recent years, because of the considerable costs of labelling [2], and the changing behaviour of data that may require relabelling over time. Exploratory data analysis in domains such as fraud and intrusion detection may involve processing large datasets with many dimensions and no ground truth [3], [4]. Aside from the storage costs, the complexity of analysing such data is proportional to its size and dimensionality. Moreover, visualising the data is always useful in the case of exploratory data analysis [3], which is difficult with large number of features [4]. Finally, differentiating between anomalies and normal data is inversely related to the data dimensionality [1].

These challenges can be alleviated using dimensionality reduction [5]. This technique maps the data to a lower dimensional latent space, which is ideally equal to its intrinsic dimensionality [3], [5]. Chandola et al. [1] refer to dimensionality reduction based AD methods as spectral AD, which assumes that there is a lower dimensional representation of the data that significantly differentiates anomalies from normal data. The two main steps of such methods are reducing the dimensionality, and then applying an AD technique. The initial step increases the accuracy and reduces the computational cost of the AD technique by finding a more useful representation of the features.

Although dimensionality reduction techniques are widely used in classification, there are still challenges when they are used for unsupervised AD. First, very few techniques provide a parametric mapping that can be applied to new unseen data. Recalculating the mapping is expensive, and in case a nonparametric out-of-sample extension exists, it may or may not result in a proper embedding of the new records [5]. Second, selecting an embedding that increases the performance of the AD, which we refer to as a good embedding, is important. In an unsupervised setting, no ground truth is available to distinguish between poor and good embeddings. Third, the intrinsic dimensionality of the data is often unknown. Although the error or information loss of dimensionality reduction methods, if computable, can be used to estimate the intrinsic dimensionality, it may fail to identify good mappings or result in selecting low reduction rates. Finally, some mappings may create undesirable properties that hinder the best performance of unsupervised AD techniques. To illustrate, unsupervised AD methods usually assume that (i) anomalies in the training set occur far less frequently than normal inputs, and (ii) they are dissimilar to the majority of the data [1]. If the detection phase is based on a dissimilarity measure like density, distance or a statistical measure, any projection that reduces the difference between the normal samples and anomalies in the latent space results in training poor models on the embedded data.

We mitigate the above challenges using a new unsupervised dimensionality reduction technique called Log-Normal distributed Stochastic Neighbor Embedding (LN-SNE). Inspired by the Parametric $t$-distributed Stochastic Neighbor Embedding (Pt-SNE) technique [6], LN-SNE is a two-step algorithm that pretrains an autoencoder using Restricted Boltzmann Machines (RBMs), then applies a fine-tuning step to obtain a proper embedding. LN-SNE provides an explicit parametric mapping to be used on future data. The fine-tuning step relies on distance and density based dissimilarity measures instead of ground truth to improve the data representation in the new space. To the best of our knowledge, LN-SNE is the first dimensionality reduction technique with an unsupervised fine-tuning step that is customised for AD. This step uses the log-normal distribution in modelling the conditional probabilities in the latent space to ensure that anomalies are mapped far from the normal samples, and their density is also lower than the density of the normal...
samples, which makes LN-SNE independent of a priori knowledge regarding the intrinsic dimensionality of the data. Our experiments show that LN-SNE is less sensitive to latent space dimensionality than Principal Component Analysis (PCA), autoencoders [7] and Pt-SNE [6], and its empirical computational cost is near-linear with respect to the data dimension and size.

Ruff et al. [8] recently proposed a deep learning based dimensionality reduction method that, similar to LN-SNE, uses an unsupervised fine-tuning step to build a compact hyper-spherical cluster for normal data in the latent space. However, unlike LN-SNE, they assumed that the normal data comes from a single distribution and forms a single cluster. In this paper, we consider a more general case, i.e., the normal data comprises an arbitrary number of clusters. For a more comprehensive survey on deep learning based AD methods, readers are referred to [9]. In the rest of the paper, we first discuss LN-SNE in Section 2, and then empirically evaluate its performance in Section 3. Section 4 concludes the paper.

2 LN-SNE

LN-SNE trains an RBM to define a parametric mapping \( \phi : X \subset \mathbb{R}^d \rightarrow \mathbb{R}^{d'} \) that projects \( X \) from the input space \( \mathbb{R}^d \) to the latent space \( \mathbb{R}^{d'} \), where \( d' \ll d \). The aim is to find a compact representation of the normal data while preserving the dissimilarity of anomalies from the normal data. Ground truth is not available to select the best embedding. Our technique, shown in Algorithm 1, comprises two steps: initialising and optimising the mapping.

\[
\begin{align*}
\text{Algorithm 1 LN-SNE initialisation and optimisation} \\
1: \text{Inputs: } X \subset \mathbb{R}^d, \beta, \psi, d', \text{ perplexity } \text{Perp}, \text{ iterations } T \\
2: \text{Output: } \phi(X; W) \quad \triangleright \text{ offsets and } a \text{ and } b \text{ are omitted for simplicity} \\
3: \text{W } \leftarrow \text{ RBM}(X, d') \quad \triangleright \text{ Step 1: initialisation (Section 2.1)} \\
4: \text{divide } X \text{ into batches } B_1, \ldots, B_T \text{ of size } \psi \quad \triangleright \text{ Step 2: optimisation (Section 2.2)} \\
5: \text{compute } \phi_p, \phi_a \text{ for all pairs in all batches } B_1, \ldots, B_T \text{ using (7) and the given } \text{Perp} \\
6: \text{for } t = 1 \text{ to } T \text{ do} \\
7: \text{for batch } b = 1 \text{ to } \beta \text{ do} \\
8: \text{compute } \phi_{p,b}, \phi_{a,b} \text{ values for all pairs in } \phi(B_{b,t}, \beta; W) \text{ using (8)} \\
9: \text{update } \text{W using (10)} \\
10: \text{W } \leftarrow \text{W } + \text{dW} \\
11: \text{end for} \\
12: \text{end for} \\
13: \text{compute the mapping } \phi(X; W) \text{ using (5)}
\end{align*}
\]

\( v' \) of the visible units given the hidden units \( h \), then resamples the hidden activations \( h' \) given \( v' \). The conditional probabilities given the hidden and visible layers are:

\[
P(h|v) = \prod_{j=1}^{k} L \left( b_j + \sum_{i=1}^{m} w_{ij} v_i \right) \\
P(v|h) = \prod_{i=1}^{m} L \left( a_i + \sum_{j=1}^{k} w_{ij} h_j \right)
\]

where \( L \) is the standard logistic function. Instead of using a stack of RBMs, we opt for an architecture with a single hidden layer to improve efficiency and reduce the number of model parameters, because no ground truth is available to tune the parameters. After training the RBM, the mapping for data point \( x_i \) is:

\[
\phi(x_i; W) = b + W^T x_i.
\]

Training an RBM does not require labelled data, but to guarantee that a good separation between the normal samples and anomalies is achieved, a further unsupervised fine-tuning step is proposed in the next section.

2.2 Optimising the Mapping

Let \( X_{nd} \) and \( X_{ad} \) denote the unknown subsets of normal and anomalous data in the dataset \( X \), and let \( \phi(X_{nd}; W) \) and \( \phi(X_{ad}; W) \) be the corresponding embeddings. Assume that there are \( M \) dissimilarity measures denoted by \( \{ D_m : m = 1..M \} \) on \( \mathbb{R}^d \) and \( \mathbb{R}^{d'} \). The objective of this step is to preserve or maximise the dissimilarities, which is the objective of the following optimisation problem:

\[
\min_W \sum_{m=1}^{M} \sum_{x_i \in X_{nd}} \sum_{x_j \in X_{ad}} |D_m(x_i, x_j) - D_m(\phi(x_i; W), \phi(x_j; W))|. 
\]

Here, we consider preservation of two properties of the data: pairwise distances and the degree of compactness, which we refer to as the density of the data. Given that the labels are unknown, substituting proper definitions that capture these properties is not trivial, so optimization problem (6) is intractable. However, we can effectively achieve the aim of (6) by using the following procedure. The high-dimensional Euclidean distances are converted into conditional probabilities using a Gaussian distribution parameterised by a variance \( s \) centered at each data point \( x_i \). The probability of picking \( x_j \) as \( x_i \)'s neighbor is given as follows:

\[
p_{ij} = \frac{\exp(-s \| x_i - x_j \|)}{\sum_{k \neq i} \exp(-s \| x_h - x_i \|)}
\]
where $||.||$ is the Euclidean norm. This definition of $p_{ij}$ is flexible; the application of the proposed method is limited to probabilities defined in this way. Pt-SNE [6] applies a binary search to assign a unique variance $s_i$ for each point $x_i$. Then to avoid the problem caused by anomalies, the $p_{ij}$ and $p_{ji}$ values are symmetrised. However, we opt for a fixed variance $s$ for all the data points, which results in symmetric probabilities for all the pairs automatically, and differentiates anomalies as well. This is done by applying a binary search to find a value for $s$ that preserves the perplexity of the conditional distribution corresponding to the data point $x_i$, where $i = \text{argmax}_i \frac{1}{n} \sum_j ||x_i - x_j||$. The value of $s$ should be updated until the perplexity of $x_i$, $\text{Perp}_i = - \sum_j p_{ij} \log(p_{ij})$, is close to the user-defined perplexity $\text{Perp}$. This fixed variance ensures at least the same perplexity for all the training samples, and also smaller normalised probabilities for anomalies. Moreover, LN-SNE terminates faster than Pt-SNE because it uses a fixed variance $s$ for all the points rather than computing it for each point individually.

To preserve the distance and density based dissimilarities, a heavy-tailed probability distribution is used to compute the conditional distributions in the latent space, which will be motivated later in this section. We select a log-normal distribution with a zero mean and a large enough scale parameter $\sigma$ for this aim, and hence call our proposed method Log-Normal(LN)-SNE. Let $d_{ij} = ||\phi(x_i; W) - \phi(x_j; W)|| + r$, and $r$ is a relocating parameter that controls the distribution peak. We define the similarity between these two samples in the latent space as:

$$q_{ij} = \frac{d_{ij}^{-1} \exp(-0.5 (\ln d_{ij} / \sigma)^2)}{\sum_{k \neq i} d_{kl}^{-1} \exp(-0.5 (\ln d_{kl} / \sigma)^2)}$$  \hspace{1cm} (8)

where $\sigma$ is the scale parameter that controls the heaviness of the distribution tail. Log-Normal is a well-behaved heavy-tailed distribution that can be parameterised to prevent what is called the crowding problem [11]. This problem arises because the distributions of the pairwise distances in the input and the latent spaces are very different. In the latent space, a considerably smaller volume (or area for 2D) is available to model the exact computed pairwise distances inherited from the higher dimensional space. As a result, strong repulsive forces may emerge between nearby data points in the latent space. On the other hand, distance-based similarities in a low-dimensional latent space can drop fairly quickly in proportion to the distance, which creates attractive forces between points that are far away in the input space. These problems create a mapping that collapses almost all the data points together [11]. To deal with this problem, we model similarities in the latent space so that the nearby points look less similar and farther points remain similar. A model with these features preserves the density of the normal data by keeping them close together in the latent space, and simultaneously preserves the large distances between the normal samples and anomalies.

The Kullback-Leibler divergence (9) can be used to measure the difference between the distributions $P$ and $Q$, and should be minimised to fine-tune the weights of the RBM.

$$C = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}$$  \hspace{1cm} (9)

The gradient of (9) based on the weight matrix $W$ is:

$$\frac{\partial C}{\partial W} = \left[ \frac{\partial C}{\partial \phi(x_i; W)} \right]_{d' \times 1} \left[ \frac{\partial \phi(x_i; W)}{\partial W} \right]_{1 \times d'}$$  \hspace{1cm} (10)

where the second term is computed using standard backpropagation [7], and $\frac{\partial C}{\partial \phi(x_i; W)}$ is given by:

$$\frac{\partial C}{\partial \phi(x_i; W)} = 2 \sum_{j} (p_{ij} - q_{ij}) \frac{\sigma^2 + \ln d_{ij}}{\sigma^2(d_{ij} - r)d_{ij}} \times \left( \phi(x_i; W) - \phi(x_j; W) \right)$$  \hspace{1cm} (11)

The training set is divided into $\beta$ batches of size $\psi$ to provide a considerable speed-up without sacrificing accuracy. The first step of training an RBM provides a compact representation of the normal data after a few iterations over these batches. As a result, the next step is devoted to separating anomalies from the normal data. In the fine-tuning step, the weights of the RBM are updated after computing the gradients for each batch (10). Moreover, a further subsampling is performed for each batch using a weighted random selection strategy, such that the probability of selecting a sample with a higher reconstruction error is increased. Since anomalies usually have higher reconstruction error due to their dissimilarity from the majority of the data, this ensures that the fine-tuning step focuses on separating anomalies from the normal data.

### 3 Empirical Evaluation

We compared LN-SNE to the benchmark unsupervised dimensionality reduction techniques that provide parametric embeddings, namely PCA, autoencoders (AE) [7] and Pt-SNE [6].

**Datasets and Data Preprocessing:** We used four real-life AD benchmark datasets of medium to high dimensionality, namely MNIST, Shuttle\(^1\), NSL-KDD\(^2\), and Credit Card Fraud Detection (Credit Card) [12]. The input dimensionality for the datasets are 784, 9, 28, and 30, respectively. Digit and Class 1 were selected as the normal concept, respectively for the MNIST and Shuttle datasets, and the rest of the classes were considered as anomalies. For the NSL-KDD dataset, the normal HTTP traffic and HTTP-based attacks were used as the normal and anomalous data. For each dataset, a subset of 10,000 records were picked at random and the test and training sets were randomly selected from the data with the ratio of 1 to 4. The proportions of anomalies in the training set were increases in the range [0.01, 0.05] by a step size of 0.01. For each proportion, we ran 10 independent experiments to measure the sensitivity of the methods to the ratio of anomalies in the training set. Therefore, each experimental setup for each dataset was run for 50 different scenarios.

**Experimental Setup:** To identify anomalies, we used a dimensionality reduction phase followed by a One-Class Support Vector Machine (OCSVM) [13], [14], [15]. OCSVMs identify anomalies by building a density-based normal profile that rejects anomalies. Consequently, *any* embedding

These reduction rates were selected based on the convergence of the performance measure. Although the reconstruction error and information loss of projecting the data into 2D were very high, we report the corresponding results for 2D as well to show that unlike the other methods, our proposed LN-SNE technique is insensitive to the choice of this parameter.

**Evaluation Metric**: The comparison is based on the average accuracy and its variation, which was measured using the Receiver Operating Characteristic (ROC) curve and the corresponding Area Under the Curve (AUC). For each experiment, the CPU-time of the methods was computed from the beginning of the dimensionality reduction phase to the end of training with the OCSVM.

### 3.1 Results and Discussion

Table 1 summarises the average variation and accuracy over the 50 runs with the different reduction rates. The best results seen for each statistical measure are shown in italic font, while the bold font results are the ones that are statistically better than the others in the following sense: a $t$-test with a level of significance of $\alpha = 0.05$ was used to measure the statistical improvement. LN-SNE+ showed a significant improvement over all the other methods in all the scenarios except PCA for MR and LR reduction on the MNIST dataset. LN-SNE+ also has lowest standard deviation in its accuracy, and its minimum accuracy over almost all of the 50 runs is considerably higher than the other methods. This improvement is highly important when no ground truth is available. As discussed before, LN-SNE preserves and even increases the distance and the density differences between the normal data and anomalies. Table 1 suggests that the accuracy of unsupervised AD via the OCSVM algorithm [16] applied in the latent space generated via LN-SNE is significantly higher than it is in the input space. Moreover, the performance of LN-SNE+, unlike the other methods, is almost independent of the number of dimensions of the latent space. The best performance of the OCSVM after dimensionality reduction in MR compression using AE, PCA, and LN-SNE is given in Table 2, assuming that the best parameter settings of the OCSVM algorithm are known and only normal data is used for training (denoted by a ++ sign after the name of the method). This study shows that if the architecture of the dimensionality reduction phase is somehow known, the AE+ and PCA+ methods isolate the anomalies in the latent space, but getting their best accuracy may still require ground truth labels. Training the model using ground truth increased the accuracy of LN-SNE as well; however, its accuracy was already higher than the other techniques in the unsupervised scenario. Due to limited space, in Figure 2 we depict only the 2D representations of the MNIST and Shuttle datasets, the ones with respectively the highest and lowest input dimensionality. The scatterplot shows how compact and more well separated the normal data is from the anomalies when LN-SNE is used compared to the other three methods. In both datasets, we can see that with LN-SNE, the density of anomalies (denoted by red crosses) compared to the normal data (denoted by blue dots) is less than the anomalies, which is not the case with the other methods. This property increases the accuracy of the OCSVM algorithm. Moreover, the 2D visualisation of

![Image](https://lvdmaaten.github.io/drtoolbox/)

**Fig. 1**: The performance of AE and PCA given the number of dimensions of the latent space of the training set that results in similar densities for normal and anomalous samples in the latent space can disrupt the performance of OCSVMs by increasing the misdetection rate. Thus, the performance of OCSVM models was used to compare the quality of embeddings generated using the four dimensionality reduction techniques. We used a hyperplane OCSVM [13] with an RBF kernel, which is denoted by a + sign after the name of each method. For example, $\text{PCA}+$ means that a OCSVM model was used to detect the anomalies after generating the mapping from PCA. We also apply the OCSVM directly on the input space to evaluate the improvement yielded by the dimensionality reduction phase. To set the hyper-parameters of the OCSVM in an unsupervised manner, we used the technique proposed by Ghafoori et al. [16]. The MATLAB LIBSVM toolbox [17] was used to implement the OCSVM method. We used the MATLAB dimensionally reduction toolbox$^3$. For LN-SNE, the values of $\sigma$ and $r$ were both fixed at 100 for all the datasets to obtain a heavy-tailed distribution with a low peak. We observed that as long as the tail of the distribution is heavier and the peak is lower or equal to the one produced by a student’s $t$-distribution with 1 degree of freedom, LN-SNE is insensitive to the choice of these parameters. Since the embedding was usually stable after a single iteration of LN-SNE over all batches, LN-SNE was run for a single iteration. For Pt-SNE, the heaviest tail corresponds to the embedding was usually stable after a single iteration. For PCA, the best parameter settings of the OCSVM algorithm are known and only normal data is used for training (denoted by a ++ sign after the name of the method). This study shows that if the architecture of the dimensionality reduction phase is somehow known, the AE+ and PCA+ methods isolate the anomalies in the latent space, but getting their best accuracy may still require ground truth labels. Training the model using ground truth increased the accuracy of LN-SNE as well; however, its accuracy was already higher than the other techniques in the unsupervised scenario. Due to limited space, in Figure 2 we depict only the 2D representations of the MNIST and Shuttle datasets, the ones with respectively the highest and lowest input dimensionality. The scatterplot shows how compact and more well separated the normal data is from the anomalies when LN-SNE is used compared to the other three methods. In both datasets, we can see that with LN-SNE, the density of anomalies (denoted by red crosses) compared to the normal data (denoted by blue dots) is less than the anomalies, which is not the case with the other methods. This property increases the accuracy of the OCSVM algorithm. Moreover, the 2D visualisation of

---

$^3$ https://lvdmaaten.github.io/drtoolbox/
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2019.2934450, IEEE Transactions on Knowledge and Data Engineering

### TABLE 1: Accuracy of the OCSVM on the input and the latent spaces

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stats</th>
<th>OCSVM</th>
<th>AE+</th>
<th>PCA+</th>
<th>Pt-SNE+</th>
<th>LN-SNE+</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>2D HR</td>
<td>MR</td>
<td>LR</td>
<td>2D HR</td>
<td>MR</td>
</tr>
<tr>
<td>AVG</td>
<td>0.789</td>
<td>0.779</td>
<td>0.788</td>
<td>0.799</td>
<td>0.805</td>
<td>0.774</td>
</tr>
<tr>
<td>STD</td>
<td>0.036</td>
<td>0.146</td>
<td>0.054</td>
<td>0.041</td>
<td>0.048</td>
<td>0.151</td>
</tr>
<tr>
<td>MIN</td>
<td>0.712</td>
<td>0.471</td>
<td>0.624</td>
<td>0.674</td>
<td>0.712</td>
<td>0.457</td>
</tr>
<tr>
<td>MAX</td>
<td>0.892</td>
<td>0.901</td>
<td>0.885</td>
<td>0.879</td>
<td>0.894</td>
<td>0.901</td>
</tr>
<tr>
<td>AVG</td>
<td>0.886</td>
<td>0.704</td>
<td>0.907</td>
<td>0.889</td>
<td>0.883</td>
<td>0.692</td>
</tr>
<tr>
<td>STD</td>
<td>0.058</td>
<td>0.179</td>
<td>0.049</td>
<td>0.059</td>
<td>0.059</td>
<td>0.173</td>
</tr>
<tr>
<td>MIN</td>
<td>0.765</td>
<td>0.482</td>
<td>0.788</td>
<td>0.772</td>
<td>0.734</td>
<td>0.481</td>
</tr>
<tr>
<td>MAX</td>
<td>0.951</td>
<td>0.975</td>
<td>0.930</td>
<td>0.951</td>
<td>0.949</td>
<td>0.979</td>
</tr>
</tbody>
</table>

1The + sign means that the OCSVM algorithm is used after the dimensionality reduction. Best results are in italic font, and statistically best results are in bold font.

### TABLE 2: Accuracy of LN-SNE+ compared to AE+ and PCA+ on MR compression given the best OCSVM setting

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stats</th>
<th>LN-SNE+</th>
<th>Pt-SNE+</th>
<th>PCA+</th>
<th>AE+</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AVG</td>
<td>STD</td>
<td>MIN</td>
<td>MAX</td>
<td>AVG</td>
</tr>
<tr>
<td></td>
<td>0.900</td>
<td>0.809</td>
<td>0.911</td>
<td>0.968</td>
<td>0.969</td>
</tr>
<tr>
<td>AVG</td>
<td>0.882</td>
<td>0.587</td>
<td>0.838</td>
<td>0.886</td>
<td>0.907</td>
</tr>
<tr>
<td>STD</td>
<td>0.011</td>
<td>0.057</td>
<td>0.032</td>
<td>0.039</td>
<td>0.040</td>
</tr>
<tr>
<td>MIN</td>
<td>0.836</td>
<td>0.478</td>
<td>0.744</td>
<td>0.772</td>
<td>0.760</td>
</tr>
<tr>
<td>MAX</td>
<td>0.941</td>
<td>0.975</td>
<td>0.930</td>
<td>0.951</td>
<td>0.949</td>
</tr>
</tbody>
</table>

Fig. 2: Sample embedded data in 2D for our LN-SNE technique compared to the others. LN-SNE separates the anomalies from the normal data better than the other three techniques in terms of their distance from normal data and they also have the lowest density.

![Sample embedded data in 2D for our LN-SNE technique compared to the others. LN-SNE separates the anomalies from the normal data better than the other three techniques in terms of their distance from normal data and they also have the lowest density.](image)

![Sample embedded data in 2D for our LN-SNE technique compared to the others. LN-SNE separates the anomalies from the normal data better than the other three techniques in terms of their distance from normal data and they also have the lowest density.](image)

Fig. 3: Sensitivity of the methods to the ratio of anomalies

(a) Credit Card (b) NSL-KDD (c) MNIST (d) Shuttle

![Sensitivity of the methods to the ratio of anomalies](image)

![Sensitivity of the methods to the ratio of anomalies](image)
LN-SNE helps us understand unlabeled datasets and may also help us identify their anomalies.

To measure the sensitivity to the ratio of anomalies in the training set, we averaged the accuracy of the methods over each 10 runs for the different anomaly proportions. The results are depicted in Figure 3 for the MR compression scenario. Compared to the other methods, LN-SNE+ is less sensitive to the proportion of anomalies. KDD-NSL and Shuttle include clusters of anomalies. Thus, a higher ratio of anomalies in the training set makes them less detectable even in the input space. However, LN-SNE solves this problem for KDD-NSL and mitigates it for Shuttle. This is due to the compact and isolated representation that LN-SNE creates for the normal data compared to anomalies (Figure 2).

Table 3 summarises the empirical computational cost of the different methods. The reported CPU-time for each method was computed from the dimensionality reduction phase to the end of training with the OCSVM. The average CPU-time for LN-SNE+ ranks the second in this table after PCA and is independent of the number of dimensions. However, the experiments were run only on a randomly selected subset of size 10,000 samples. To give a fair comparison between the best rivals of the previous step, the Credit Card dataset with the largest number of records was used. The number of training samples was increased from 10,000 to 260,000 and the CPU-times of the methods were measured. Figure 4 shows that the CPU-time when a dimensionality reduction technique is used is nearly linear with respect to the number of training samples, and considerably lower than using the OCSVM directly on the input space. While LN-SNE+ ranks between the AE+ and the PCA+ techniques, its accuracy is considerably higher and its sensitivity to the proportion of anomalies in the training set is less than both of them.

### 4 Conclusion

We proposed a dimensionality reduction technique called LN-SNE for unsupervised anomaly detection. The performance of LN-SNE on the datasets examined in this paper is less sensitive to the choice of the latent space dimensionality than PCA, AE, and PT-SNE. It creates a compact representation of the normal data that, for the datasets we have used, resides far away from anomalies in the latent space, and enhances the performance of the OCSVM algorithm for anomaly detection. Our new technique yields a statistical improvement over three benchmark dimensionality reduction techniques that are used for unsupervised anomaly detection. Unifying the feature and model learning phases by RBMs, LN-SNE, and OCSVs will be the main direction of our future work.

### References


Zahra Ghafoori is a research fellow in the Academic Centre of Cyber Security Excellence (ACCSE), the Department of Computing and Information Systems at the University of Melbourne. She is currently working on developing robust and efficient techniques for unsupervised anomaly detection in complex and dynamic environments. Her research interests include data mining, pattern recognition, cyber-security and anomaly detection.

Sarah M. Erfani is a lecturer in the School of Computing and Information Systems at The University of Melbourne. Her research interests include machine learning, large-scale data mining, cyber security, and data privacy.


Shanika Karunasekera received the BSc degree in electronics and telecommunications engineering from the University of Moratuwa, Sri Lanka, in 1990, and the PhD degree in electrical engineering from the University of Cambridge, UK, in 1995. From 1995 to 2002, she was a Software Engineer and a Distinguished Member of Technical Staff at Lucent Technologies, Bell Labs Innovations, USA. In January 2003, she joined the Department of Computer Science and Software Engineering, University of Melbourne as a Senior Lecturer. Her current research interests include decentralised algorithms for event and anomaly detection, and decentralised data aggregation.

Christopher Leckie received the B.Sc. degree, the B.E. degree in electrical and computer systems engineering (with first class honors), and the Ph.D. degree in computer science from Monash University, Australia, in 1985, 1987, and 1992, respectively. In 1988, he joined Telstra Research Laboratories, where he conducted research and development into artificial intelligence techniques for various telecommunication applications. In 2000, he joined The University of Melbourne, Australia, where he is currently a Professor with the School of Computing and Information Systems. His research interests include scalable data mining and security analytics.