Unsupervised Representation Learning by Predicting Random Distances

Hu Wang[*] , **Guansong Pang**[*] , **Chunhua Shen**[†] , **Congbo Ma**
The University of Adelaide, Australia

Abstract

Deep neural networks have gained great success in a broad range of tasks due to its remarkable capability to learn semantically rich features from high-dimensional data. However, they often require large-scale labelled data to successfully learn such features, which significantly hinders their adaption in unsupervised learning tasks, such as anomaly detection and clustering, and limits their applications to critical domains where obtaining massive labelled data is prohibitively expensive. To enable unsupervised learning on those domains, in this work we propose to learn features without using any labelled data by training neural networks to predict data distances in a randomly projected space. Random mapping is a theoretically proven approach to obtain approximately preserved distances. To well predict these distances, the representation learner is optimised to learn genuine class structures that are implicitly embedded in the randomly projected space. Empirical results on 19 real-world datasets show that our learned representations substantially outperform a few state-of-the-art methods for both anomaly detection and clustering tasks. Code is available at: `https://git.io/RDP`

1 Introduction

Unsupervised representation learning aims to automatically extract expressive feature representations from data without any labelled data. Due to the remarkable capability to learn semantically rich features, deep neural networks have been becoming one widely-used technique to empower a broad range of machine learning tasks. One main issue with these deep learning techniques is that a massive amount of labelled data is typically required to successfully learn these expressive features. As a result, their transformation power is largely reduced for tasks that are unsupervised in nature, such as anomaly detection and clustering. This is also true to critical domains, such as healthcare and fintech, where collecting massive labelled data is prohibitively expensive and/or is impossible to scale. To bridge this gap, in this work we explore fully unsupervised representation learning techniques to enable downstream unsupervised learning methods on those critical domains.

In recent years, many unsupervised representation learning methods [Mikolov *et al.*, 2013; Le and Mikolov, 2014; Misra *et al.*, 2016; Lee *et al.*, 2017; Gidaris *et al.*, 2018] have been introduced, of which most are self-supervised approaches that formulate the problem as an annotation free pretext task. These methods explore easily accessible information, such as temporal or spatial neighbourhood, to design a surrogate supervisory signal to empower the feature learning. These methods have achieved significantly improved feature representations of text/image/video data. But they are often inapplicable to tabular data since it does not contain the required temporal or spatial supervisory information. We therefore focus on unsupervised representation learning of *high-dimensional tabular data*. Although many approaches, such as random projection [Li *et al.*, 2006], manifold learning [Donoho and Grimes, 2003; Hinton and Roweis, 2003] and autoencoder [Vincent *et al.*, 2010], are readily available for handling those data, many of them [Donoho and Grimes, 2003; Hinton and Roweis, 2003; Rahmani and Atia, 2017] are often too computationally costly to scale up to large or high-dimensional data. Approaches like random projection and autoencoder are very efficient but they often fail to capture complex class structures due to its underlying data assumption or weak supervisory signal.

In this paper, we introduce a Random Distance Prediction (RDP) model which trains neural networks to predict data distances in a randomly projected space. Particularly, as distances generally carry intrinsic class structure information in the data, the representation learner is optimised to learn the class structure to minimise the prediction error. We seek to obtain distances preserved in a projected space to be the supervisory signal because distances are concentrated and become meaningless in high dimensional spaces [Beyer *et al.*, 1999]. Random mapping is a highly efficient yet theoretical proven approach to obtain such approximately preserved distances. Therefore, we leverage the distances in the randomly projected space to learn the desired features. Intuitively, random mapping preserves rich local proximity information but

[*] Equal contribution.

[†] Corresponding author.

may also keep misleading proximity when its underlying data distribution assumption is inexact; by minimising the random distance prediction error, RDP essentially leverages the preserved data proximity and the power of neural networks to learn globally consistent proximity and rectify the inconsistent proximity information, resulting in a substantially better representation space than the original space. We show that this simple random distance prediction enables us to achieve expressive representations without using labelled data. In addition, some task-dependent auxiliary losses can be optionally added as a complementary supervisory source to the random distance prediction, so as to learn the feature representations that are more tailored to a specific task. In summary, we make the following three main contributions.

- We propose a random distance prediction formulation, which is very simple yet offers a highly effective supervisory signal for learning expressive feature representations that *optimise* the distance preserving in random projection. The learned features are sufficiently generic and work well for downstream prediction tasks.
- Our formulation is flexible to incorporate task-dependent auxiliary losses that are complementary to random distance prediction to further enhance the learned features, i.e., features that are specifically optimised for a downstream task while at the same time preserving the generic proximity as much as possible.
- As a result, we show that our instantiated model termed RDP achieves substantially better performance than state-of-the-art competing methods in two key unsupervised tasks, anomaly detection and clustering.

2 Random Distance Prediction Model

The Proposed Formulation and The Instantiated Model We propose to learn representations by training neural networks to predict distances in a randomly projected space without manually labelled data. The key intuition is that, given some distance information that faithfully encapsulates the underlying class structure in the data, the representation learner is forced to learn the class structure in order to yield distances that are as close as the given distances. Our proposed framework is illustrated in Figure 1. Specifically, given data points $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^D$, we first feed them into a weight-shared Siamese-style neural network $\phi(\mathbf{x};\Theta)$. $\phi : \mathbb{R}^D \mapsto \mathbb{R}^M$ is a representation learner with the parameters Θ to map the data onto a M-dimensional new space. Then we formulate the subsequent step as a distance prediction task and define a loss function as:

$$L_{rdp}(\mathbf{x}_i,\mathbf{x}_j) = l(\langle\phi(\mathbf{x}_i;\Theta),\phi(\mathbf{x}_j;\Theta)\rangle,\langle\eta(\mathbf{x}_i),\eta(\mathbf{x}_j)\rangle), \quad (1)$$

where η is an existing projection method and l is a function of the difference between its two inputs.

Here one key ingredient is how to obtain trustworthy distances via η. Also, to efficiently optimise the model, the distance derivation needs to be computationally efficient. In this work, we use the inner products in a randomly projected space as the source of distance/similarity since it is very efficient and there is strong theoretical support of its capacity in preserving the genuine distance information. Thus, our instantiated model RDP specifies $L_{rdp}(\mathbf{x}_i,\mathbf{x}_j)$ as follows[1]:

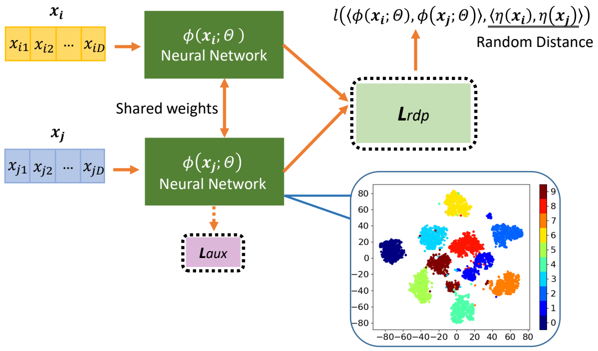
x_i
x_{i1} x_{i2} ··· x_{iD}
$\phi(x_i;\Theta)$ Neural Network
Shared weights
x_j
x_{j1} x_{j2} ··· x_{jD}
$\phi(x_j;\Theta)$ Neural Network
$l(\langle\phi(x_i;\Theta),\phi(x_j;\Theta)\rangle,\langle\eta(x_i),\eta(x_j)\rangle)$
Random Distance
L_{rdp}
L_{aux}
80 60 40 20 0 −20 −40 −60 −80
−80 −60 −40 −20 0 20 40 60 80
9 8 7 6 5 4 3 2 1 0

Figure 1: The proposed random distance prediction (RDP) framework. A weight-shared two-branch neural network ϕ first projects $\mathbf{x}_i$ and $\mathbf{x}_j$ into a new space, in which we aim to minimise the random distance prediction loss L_{rdp}, i.e., the difference between the learned distance $\langle\phi(\mathbf{x}_i;\Theta),\phi(\mathbf{x}_j;\Theta)\rangle$ and a predefined distance $\langle\eta(\mathbf{x}_i),\eta(\mathbf{x}_j)\rangle$ (η denotes an existing random mapping). L_{aux} is an auxiliary loss that is optionally applied to one network branch to learn complementary information w.r.t. L_{rdp}. The lower right figure presents a 2-D t-SNE visualisation of the features learned by RDP on a small dataset *optdigits* with 10 classes.

$$L_{rdp}(\mathbf{x}_i,\mathbf{x}_j) = (\phi(\mathbf{x}_i;\Theta)\cdot\phi(\mathbf{x}_j;\Theta) - \eta(\mathbf{x}_i)\cdot\eta(\mathbf{x}_j))^2, \quad (2)$$

where ϕ is implemented by multilayer perceptron for dealing with tabular data and $\eta : \mathbb{R}^D \mapsto \mathbb{R}^K$ is an off-the-shelf random data mapping function (see Sections 3.1 and 3.2 for detail). Despite its simplicity, this loss offers a powerful supervisory signal to learn semantically rich feature representations that substantially optimise the underlying distance preserving in η (see Section 3.3 for detail).

Flexibility to Incorporate Task-dependent Complementary Auxiliary Loss Minimising L_{rdp} learns to preserve pairwise distances that are critical to different learning tasks. Moreover, our formulation is flexible to incorporate a task-dependent auxiliary loss L_{aux}, such as reconstruction loss [Hinton and Salakhutdinov, 2006] for clustering or novelty loss [Burda *et al.*, 2019] for anomaly detection, to complement the proximity information and enhance the feature learning.

For clustering, an auxiliary reconstruction loss is used:

$$L_{aux}^{clu}(\mathbf{x}) = \left(\mathbf{x} - \phi'(\phi(\mathbf{x};\Theta);\Theta')\right)^2, \quad (3)$$

where ϕ is an encoder and $\phi' : \mathbb{R}^M \mapsto \mathbb{R}^D$ is a decoder. This loss may be optionally added into RDP to better capture global feature representations.

Similarly, in anomaly detection a novelty loss may be optionally added, which is defined as:

$$L_{aux}^{ad}(\mathbf{x}) = (\phi(\mathbf{x};\Theta) - \eta(\mathbf{x}))^2. \quad (4)$$

[1] Since we operate on real-valued vector space, the inner product is implemented by the dot product. The dot product is used hereafter to simplify the notation.

By using a fixed η, minimising L_{aux}^{ad} helps learn the frequency of underlying patterns in the data [Burda *et al.*, 2019], which is an important complementary supervisory source for the sake of anomaly detection. As a result, anomalies or novel points are expected to have substantially larger $(\phi(\mathbf{x};\Theta^{\star}) - \eta(\mathbf{x}))^2$ than normal points, so Eqn. (4) is used to define anomaly score for the anomaly detection task.

Note since L_{aux}^{ad} involves a mean squared error between two vectors, the dimension of the projected space resulted by ϕ and η is required to be equal in this case. Therefore, when this loss is added into RDP, the M in ϕ and K in η need to be the same. We do not have this constraint in other cases.

Overall, our loss function will then be:

$$L = L_{rdp} + L_{aux}, \tag{5}$$

where L_{aux} is L_{aux}^{clu} for clustering and L_{aux}^{ad} for anomaly detection.

3 Theoretical Analysis of RDP

This section shows the proximity information can be well approximated using inner products in two types of random projection spaces. This is a key theoretical foundation to RDP. Also, to accurately predict these distances, RDP is forced to learn the genuine class structure in the data.

3.1 When Linear Projection Is Used

Random projection is a simple yet very effective linear feature mapping technique which has proven the capability of distance preservation. Let $\mathscr{X} \subset \mathbb{R}^{N\times D}$ be a set of N data points, random projection uses a random matrix $\mathbf{A} \subset \mathbb{R}^{K\times D}$ to project the data onto a lower K-dimensional space by $\mathscr{X}' = \mathbf{A}\mathscr{X}^{\intercal}$. The Johnson-Lindenstrauss lemma [Johnson and Lindenstrauss, 1984] guarantees the data points can be mapped to a randomly selected space of suitably lower dimension with the distances between the points are approximately preserved. More specifically, let $\varepsilon \in (0, \frac{1}{2})$ and $K = \frac{20 \log n}{\varepsilon^2}$. There exists a linear mapping $f : \mathbb{R}^D \mapsto \mathbb{R}^K$ such that for all $\mathbf{x}_i, \mathbf{x}_j \in \mathscr{X}$:

$$(1-\varepsilon)||\mathbf{x}_i - \mathbf{x}_j||^2 \leq ||f(\mathbf{x}_i) - f(\mathbf{x}_j)||^2 \leq (1+\varepsilon)||\mathbf{x}_i - \mathbf{x}_j||^2. \tag{6}$$

Furthermore, assume the entries of the matrix $\mathbf{A}$ are sampled independently from a Gaussian distribution $\mathscr{N}(0,1)$. Then, the norm of $\mathbf{x} \in \mathbb{R}^D$ can be preserved as:

$$\Pr\left((1-\varepsilon)||\mathbf{x}||^2 \leq ||\frac{1}{\sqrt{K}}\mathbf{A}\mathbf{x}||^2 \leq (1+\varepsilon)||\mathbf{x}||^2\right) \geq 1 - 2e^{\frac{-(\varepsilon^2-\varepsilon^3)K}{4}}. \tag{7}$$

Under such random projections, the norm preservation helps well preserve the inner products:

$$\Pr\left(|\hat{\mathbf{x}}_i \cdot \hat{\mathbf{x}}_j - f(\hat{\mathbf{x}}_i) \cdot f(\hat{\mathbf{x}}_j)| \geq \varepsilon\right) \leq 4e^{\frac{-(\varepsilon^2-\varepsilon^3)K}{4}}, \tag{8}$$

where $\hat{\mathbf{x}}$ is a normalised $\mathbf{x}$ such that $||\hat{\mathbf{x}}|| \leq 1$.

The proofs of Eqns. (6-8) can be found in [Vempala, 1998].

Eqn. (8) states that the inner products in the randomly projected space can largely preserve the inner products in the original space, particularly when the dimension K is large.

3.2 When Non-linear Projection Is Used

Here we show that some non-linear random mapping methods are approximate to kernel functions which are a well-established approach to obtain reliable distance/similarity information. The key to this approach is the kernel function $k : \mathscr{X} \times \mathscr{X} \mapsto \mathbb{R}$, which is defined as $k(\mathbf{x}_i, \mathbf{x}_j) = \langle \psi(\mathbf{x}_i), \psi(\mathbf{x}_j) \rangle$, where ψ is a feature mapping function but needs not to be explicitly defined and $\langle \cdot, \cdot \rangle$ denotes a suitable inner product. A non-linear kernel function such as polynomial or radial basis function (RBF) kernel is typically used to project linear-inseparable data onto a linear-separable space.

The relation between non-linear random mapping and kernel methods is justified in [Rahimi and Recht, 2008], which shows that an explicit randomised mapping function $g : \mathbb{R}^D \mapsto \mathbb{R}^K$ can be defined to project the data points onto a low-dimensional Euclidean inner product space such that the inner products in the projected space approximate the kernel evaluation:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \langle \psi(\mathbf{x}_i), \psi(\mathbf{x}_j) \rangle \approx g(\mathbf{x}_i) \cdot g(\mathbf{x}_j). \tag{9}$$

Let $\mathbf{A}$ be the mapping matrix. Then to achieve the above approximation, $\mathbf{A}$ is required to be drawn from Fourier transform and shift-invariant functions such as cosine function are finally applied to $\mathbf{Ax}$ to yield a real-valued output. By transforming $\mathbf{x}_i$ and $\mathbf{x}_j$ in this manner, their inner product $g(\mathbf{x}_i) \cdot g(\mathbf{x}_j)$ is an unbiased estimator of $k(\mathbf{x}_i, \mathbf{x}_j)$.

3.3 Learning Class Structure By Random Distance Prediction

Our model using only the random distances as the supervisory signal can be formulated as:

$$\arg\min_{\Theta} \sum_{\mathbf{x}_i, \mathbf{x}_j \in \mathscr{X}} \left(\phi(\mathbf{x}_i;\Theta) \cdot \phi(\mathbf{x}_j;\Theta) - y_{ij}\right)^2, \tag{10}$$

where $y_{ij} = \eta(\mathbf{x}_i) \cdot \eta(\mathbf{x}_j)$. Let $\mathbf{Y}_\eta \in \mathbb{R}^{N\times N}$ be the distance/similarity matrix of the N data points resulted by η. Then to minimise the prediction error in Eqn. (10), ϕ is optimised to learn the underlying class structure embedded in $\mathbf{Y}$. As shown in the properties in Eqns. (8) and (9), $\mathbf{Y}_\eta$ can effectively preserve local proximity information when η is set to be either the random projection-based f function or the kernel method-based g function. However, those proven η is often built upon some underlying data distribution assumption, e.g., Gaussian distribution in random projection or Gaussian RBF kernel. Thus, the η-projected features can preserve misleading proximity when the distribution assumption is inexact. In this case, $\mathbf{Y}_\eta$ is equivalent to the imperfect ground truth with partial noise. Then optimisation with Eqn. (10) is to leverage the power of neural networks to learn consistent local proximity information and rectify inconsistent proximity, resulting in a significantly optimised distance preserving space. The resulting space conveys substantially richer semantics than the η projected space when $\mathbf{Y}_\eta$ contains sufficient genuine supervision information.

4 Experiments

This section evaluates our method through two typical unsupervised tasks: anomaly detection and clustering[2]. The nonlinear random projection is used in RDP by default throughout all our experiments.

4.1 Performance Evaluation in Anomaly Detection

Experimental Settings

Our RDP model is compared with five state-of-the-art methods, including iForest [Liu *et al.*, 2008], autoencoder (AE) [Hinton and Salakhutdinov, 2006], REPEN [Pang *et al.*, 2018], DAGMM [Zong *et al.*, 2018] and RND [Burda *et al.*, 2019]. iForest and AE are two of the most popular baselines. The other three methods learn representations specifically for anomaly detection. The RDP consists of one fully connected layer with 50 hidden units, followed by a leaky-ReLU layer[3]. It is trained using Stochastic Gradient Descent (SGD) as its optimiser for 200 epochs, with 192 samples per batch. The learning rate is fixed to 0.1.

Our RDP model uses the optional novelty loss for anomaly detection task by default. Note that the representation dimension M in the ϕ function and the projection dimension K in the η function are set to be the same to alleviate parameter tuning. This means that $M = K = 50$ is used here. Similar to RND, given a data point $\mathbf{x}$, its anomaly score in RDP is defined as the mean squared error between the two projections resulted by $\phi(\mathbf{x};\Theta^{\star})$ and $\eta(\mathbf{x})$. Also, a boosting process is used to filter out 5% likely anomalies per iteration to iteratively improve the modelling of RDP. This is because the modelling is otherwise largely biased when anomalies are presented. We repeated the boosting process 30 times to obtain statistically stable results. In order to have fair comparisons, we also adapt the competing methods AE, REPEN, DAGMM and RND into ensemble methods and perform the experiments using an ensemble size of 30.

As shown in Table 1, 14 publicly available datasets taken from the literature [Liu *et al.*, 2008; Pang *et al.*, 2018; Zong *et al.*, 2018], are used, which are from various domains, including network intrusion, credit card fraud detection, and disease detection. Many of the datasets contain real anomalies, including *DDoS*, *Donors*, *Backdoor*, *Creditcard*, *Lung*, *Probe* and *U2R*. Following [Liu *et al.*, 2008; Pang *et al.*, 2018; Zong *et al.*, 2018], the rare class(es) is treated as anomalies in the other datasets to create semantically real anomalies. The Area Under Receiver Operating Characteristic Curve (AUC-ROC) and the Area Under Precision-Recall Curve (AUC-PR) are used as our performance metrics. The reported performance is averaged over 10 independent runs.

Comparison to the State-of-the-art Competing Methods

The AUC-ROC and AUC-PR results are respectively shown in Tables 1 and 2. RDP outperforms all the five competing methods in both of AUC-ROC and AUC-PR in at least 12 out of 14 datasets. This improvement is statistically significant at the 95% confidence level according to the two-tailed sign test [Demšar, 2006]. Remarkably, RDP obtains more than 10% AUC-ROC/AUC-PR improvement over the best competing method on six datasets, including *Donors*, *Ad*, *Bank*, *Celeba*, *Lung* and *U2R*. RDP can be thought as a high-level synthesis of REPEN and RND, because REPEN leverages a pairwise distance-based ranking loss to learn representations for anomaly detection while RND is built using L^{ad}_{aux}. In nearly all the datasets, RDP well leverages both L_{rdp} and L^{ad}_{aux} to achieve significant improvement over both REPEN and RND. In very limited cases, such as on datasets *Backdoor* and *Census* where RND performs very well while REPEN performs less effectively, RDP is slightly downgraded due to the use of L_{rdp}. In the opposite case, such as *Probe*, on which REPEN performs much better than RND, the use of L^{ad}_{aux} may drag down the performance of RDP a bit.

Data	iForest	AE	REPEN	DAGMM	RND	RDP
DDoS	0.880 ± 0.018	0.901 ± 0.000	0.933 ± 0.002	0.766 ± 0.019	0.852 ± 0.011	**0.942** ± **0.008**
Donors	0.774 ± 0.010	0.812 ± 0.011	0.777 ± 0.075	0.763 ± 0.110	0.847 ± 0.011	**0.962** ± **0.011**
Backdoor	0.723 ± 0.029	0.806 ± 0.007	0.857 ± 0.001	0.813 ± 0.035	**0.935** ± **0.002**	0.910 ± 0.021
Ad	0.687 ± 0.021	0.703 ± 0.000	0.853 ± 0.001	0.500 ± 0.000	0.812 ± 0.002	**0.887** ± **0.003**
Apascal	0.514 ± 0.051	0.623 ± 0.005	0.813 ± 0.004	0.710 ± 0.020	0.685 ± 0.019	**0.823** ± **0.007**
Bank	0.713 ± 0.021	0.666 ± 0.000	0.681 ± 0.001	0.616 ± 0.014	0.690 ± 0.006	**0.758** ± **0.007**
Celeba	0.693 ± 0.014	0.735 ± 0.002	0.802 ± 0.002	0.680 ± 0.067	0.682 ± 0.029	**0.860** ± **0.006**
Census	0.599 ± 0.019	0.602 ± 0.000	0.542 ± 0.003	0.502 ± 0.003	**0.661** ± **0.003**	0.653 ± 0.004
Creditcard	0.948 ± 0.005	0.948 ± 0.000	0.950 ± 0.001	0.877 ± 0.005	0.945 ± 0.001	**0.957** ± **0.005**
Lung	0.893 ± 0.057	0.953 ± 0.004	0.949 ± 0.002	0.830 ± 0.087	0.867 ± 0.031	**0.982** ± **0.006**
Probe	0.995 ± 0.001	0.997 ± 0.000	0.997 ± 0.000	0.953 ± 0.008	0.975 ± 0.000	**0.997** ± **0.000**
R8	0.841 ± 0.023	0.835 ± 0.000	**0.910** ± **0.000**	0.760 ± 0.066	0.883 ± 0.006	0.902 ± 0.002
Secom	0.548 ± 0.019	0.526 ± 0.000	0.510 ± 0.004	0.513 ± 0.010	0.541 ± 0.006	**0.570** ± **0.004**
U2R	**0.988** ± **0.001**	0.987 ± 0.000	0.978 ± 0.000	0.945 ± 0.028	0.981 ± 0.001	0.986 ± 0.001

Table 1: AUC-ROC (mean±std) results of anomaly detection.

Ablation Study

This section examines the contribution of L_{rdp}, L^{ad}_{aux} and the boosting process to the performance of RDP. The experimental results in AUC-ROC are given in Table 3, where RDP\X means the RDP variant that removes the 'X' module from RDP. Similar observations can also be derived from AUC-PR results that are omitted due to page limits. In the last two columns, *Org_SS* indicates that we directly use the distance information calculated in the original space as the supervisory signal, while *SRP_SS* indicates that we use SRP to obtain the distances as the supervisory signal. It is clear that the full RDP model is the best performer. Using the L_{rdp} loss only, i.e., RDP\L^{ad}_{aux}, can achieve performance substantially better than, or comparably well to, the five competing methods in Table 1. This is mainly because the L_{rdp} loss alone can effectively force our representation learner to learn the under-

Data	iForest	AE	REPEN	DAGMM	RND	RDP
DDoS	0.141 ± 0.020	0.248 ± 0.001	0.300 ± 0.012	0.038 ± 0.000	0.110 ± 0.015	**0.301** ± **0.028**
Donors	0.124 ± 0.006	0.138 ± 0.007	0.120 ± 0.032	0.070 ± 0.024	0.201 ± 0.033	**0.432** ± **0.061**
Backdoor	0.045 ± 0.007	0.065 ± 0.004	0.129 ± 0.001	0.034 ± 0.023	**0.433** ± **0.015**	0.305 ± 0.008
Ad	0.363 ± 0.061	0.479 ± 0.000	0.600 ± 0.002	0.140 ± 0.000	0.473 ± 0.009	**0.726** ± **0.007**
Apascal	0.015 ± 0.002	0.023 ± 0.001	0.041 ± 0.001	0.023 ± 0.009	0.021 ± 0.005	**0.042** ± **0.003**
Bank	0.293 ± 0.023	0.264 ± 0.001	0.276 ± 0.001	0.150 ± 0.020	0.258 ± 0.006	**0.364** ± **0.013**
Celeba	0.060 ± 0.006	0.082 ± 0.001	0.081 ± 0.001	0.037 ± 0.017	0.068 ± 0.010	**0.104** ± **0.006**
Census	0.071 ± 0.004	0.072 ± 0.000	0.064 ± 0.005	0.061 ± 0.001	0.081 ± 0.001	**0.086** ± **0.001**
Creditcard	0.145 ± 0.031	**0.382** ± **0.004**	0.359 ± 0.014	0.010 ± 0.012	0.290 ± 0.012	0.363 ± 0.011
Lung	0.379 ± 0.092	0.565 ± 0.022	0.429 ± 0.005	0.042 ± 0.003	0.381 ± 0.104	**0.705** ± **0.028**
Probe	0.923 ± 0.011	0.964 ± 0.002	**0.964** ± **0.000**	0.409 ± 0.153	0.609 ± 0.014	0.955 ± 0.002
R8	0.076 ± 0.018	0.097 ± 0.006	0.083 ± 0.000	0.019 ± 0.011	0.134 ± 0.031	**0.146** ± **0.017**
Secom	0.106 ± 0.007	0.093 ± 0.000	0.091 ± 0.001	0.066 ± 0.002	0.086 ± 0.002	**0.096** ± **0.001**
U2R	0.180 ± 0.018	0.230 ± 0.004	0.116 ± 0.007	0.025 ± 0.019	0.217 ± 0.011	**0.261** ± **0.005**

Table 2: AUC-PR (mean±std) results of anomaly detection.

[2] See an extended version at https://arxiv.org/abs/1912.12186 for the data accessing and additional results on representation learning of raw image data, computational efficiency and classification task.

[3] We have also tried deeper network structures, but they worked less effectively than the shallow networks. This may be because the supervisory signal is not strong enough to support deeper networks.

	Decomposition				Supervision Signal	
Data	**RDP**	**RDP**$\backslash L_{rdp}$	**RDP**$\backslash L_{aux}^{ad}$	**RDP\Boosting**	**Org_SS**	**SRP_SS**
DDoS	**0.942** ± **0.008**	0.852 ± 0.011	0.931 ± 0.003	0.866 ± 0.011	0.924 ± 0.006	0.927 ± 0.005
Donors	**0.962** ± **0.011**	0.847 ± 0.011	0.737 ± 0.006	0.910 ± 0.013	0.728 ± 0.005	0.762 ± 0.016
Backdoor	0.910 ± 0.021	0.935 ± 0.002	0.872 ± 0.012	**0.943** ± **0.002**	0.875 ± 0.002	0.882 ± 0.010
Ad	**0.887** ± **0.003**	0.812 ± 0.002	0.718 ± 0.005	0.818 ± 0.002	0.696 ± 0.003	0.740 ± 0.008
Apascal	**0.823** ± **0.007**	0.685 ± 0.019	0.732 ± 0.007	0.804 ± 0.021	0.604 ± 0.032	0.760 ± 0.030
Bank	**0.758** ± **0.007**	0.690 ± 0.006	0.684 ± 0.004	0.736 ± 0.009	0.684 ± 0.002	0.688 ± 0.015
Celeba	**0.860** ± **0.006**	0.682 ± 0.029	0.709 ± 0.005	0.794 ± 0.017	0.667 ± 0.033	0.734 ± 0.027
Census	0.653 ± 0.004	**0.661** ± **0.003**	0.626 ± 0.006	0.661 ± 0.001	0.636 ± 0.006	0.560 ± 0.006
Creditcard	**0.957** ± **0.005**	0.945 ± 0.001	0.950 ± 0.000	0.956 ± 0.003	0.947 ± 0.001	0.949 ± 0.003
Lung	**0.982** ± **0.006**	0.867 ± 0.031	0.911 ± 0.006	0.968 ± 0.018	0.884 ± 0.018	0.928 ± 0.008
Probe	0.997 ± 0.000	0.975 ± 0.000	**0.998** ± **0.000**	0.978 ± 0.001	0.995 ± 0.000	0.997 ± 0.001
R8	0.902 ± 0.002	0.883 ± 0.006	0.867 ± 0.003	0.895 ± 0.004	0.830 ± 0.005	**0.904** ± **0.005**
Secom	**0.57** ± **0.004**	0.541 ± 0.006	0.544 ± 0.011	0.563 ± 0.008	0.512 ± 0.007	0.530 ± 0.016
U2R	0.986 ± 0.001	0.981 ± 0.001	0.987 ± 0.000	**0.988** ± **0.002**	0.987 ± 0.000	0.981 ± 0.002
#wins/draws/losses (RDP vs.)		13/0/1	13/0/1	12/0/2	10/2/2	6/0/8

Table 3: AUC-ROC ablation study results of anomaly detection.

Data Characteristics			NMI Performance					
Data	**N**	**D**	**Org**	**HLLE**	**SRP**	**AE**	**COP**	**RDP**
R8	7,674	17,387	0.524 ± 0.047	0.004 ± 0.001	0.459 ± 0.031	0.471 ± 0.043	0.025 ± 0.003	**0.539** ± **0.040**
20news	18,846	130,107	0.080 ± 0.004	0.017 ± 0.000	0.075 ± 0.002	0.075 ± 0.006	0.027 ± 0.040	**0.084** ± **0.005**
Olivetti	400	4,096	0.778 ± 0.014	**0.841** ± **0.011**	0.774 ± 0.011	0.782 ± 0.010	0.333 ± 0.018	0.805 ± 0.012
Sector	9,619	55,197	**0.336** ± **0.008**	0.122 ± 0.004	0.273 ± 0.011	0.253 ± 0.010	0.129 ± 0.014	0.305 ± 0.007
RCV1	20,242	47,236	0.154 ± 0.000	0.006 ± 0.000	0.134 ± 0.024	0.146 ± 0.010	N/A	**0.165** ± **0.000**
Data Characteristics			**F-score Performance**					
Data	**N**	**D**	**Org**	**HLLE**	**SRP**	**AE**	**COP**	**RDP**
R8	7,674	17,387	0.185 ± 0.189	0.085 ± 0.000	0.317 ± 0.045	0.312 ± 0.068	0.088 ± 0.002	**0.360** ± **0.055**
20news	18,846	130,107	0.116 ± 0.006	0.007 ± 0.000	0.109 ± 0.006	0.083 ± 0.010	0.009 ± 0.004	**0.119** ± **0.006**
Olivetti	400	4,096	0.590 ± 0.029	**0.684** ± **0.024**	0.579 ± 0.022	0.602 ± 0.023	0.117 ± 0.011	0.638 ± 0.026
Sector	9,619	55,197	**0.208** ± **0.008**	0.062 ± 0.001	0.187 ± 0.009	0.184 ± 0.010	0.041 ± 0.004	0.191 ± 0.007
RCV1	20,242	47,236	0.519 ± 0.000	0.342 ± 0.000	0.508 ± 0.003	0.514 ± 0.057	N/A	**0.572** ± **0.003**

Table 4: NMI and F-score performance of K-means.

lying class structure on most datasets so as to minimise its prediction error. The use of L_{aux}^{ad} and boosting process well complement the L_{rdp} loss on the other datasets.

In terms of supervisory source, RDP and SRP_SS perform substantially better than Org_SS on most datasets. This is because the distances in both the non-linear projection in RDP and the linear projection in SRP_SS is well preserved, enabling RDP to effectively learn much more faithful class structure than that working on the original space.

4.2 Performance Evaluation in Clustering

Experimental Settings

For clustering, RDP is compared with four state-of-the-art unsupervised representation learning methods in four different areas, including HLLE [Donoho and Grimes, 2003] in manifold learning, Sparse Random Projection (SRP) [Li *et al.*, 2006] in random projection, autoencoder (AE) [Hinton and Salakhutdinov, 2006] in data reconstruction-based neural network methods and Coherence Pursuit (COP) [Rahmani and Atia, 2017] in robust PCA. These representation learning methods are first used to yield the new representations, and K-means [Hartigan and Wong, 1979] is then applied to the representations to perform clustering. In this section RDP adds the reconstruction loss L_{aux}^{clu} by default.

RDP uses a similar network architecture and optimisation settings as the one used in anomaly detection. Compared to anomaly detection, more semantic information is required for clustering algorithms to work well, so the network consists of 1,024 hidden units and is trained for 1,000 epochs. AE is built and trained with the same settings as RDP. Similar to anomaly detection, $M = K$ is also used in clustering.

As shown in Table 4, five high-dimensional real-world datasets are used. Some of the datasets are image/text data. Since here we focus on the performance on tabular data, they are converted into tabular data using simple methods, i.e., by treating each pixel as a feature unit for image data or using bag-of-words representation for text data. Two widely-used clustering performance metrics, Normalised Mutual Info (NMI) score and F-score, are used. Larger NMI/F-score indicates better performance. The performance in the original feature space, denoted as Org, is used as a baseline. The reported NMI score and F-score are averaged over 30 times to address the randomisation issue in K-means clustering.

Comparison to the State-of-the-art Competing Methods

Table 4 shows the NMI and F-score performance of K-means clustering. Our method RDP enables K-means to achieve the best performance on three datasets and ranks second in the other two datasets. RDP-enabled clustering performs substantially and consistently better than that based on AE in terms of both NMI and F-score. This demonstrates that the random distance loss enables RDP to effectively capture some class structure in the data which cannot be captured by using the reconstruction loss. RDP also consistently outperforms the random projection method, SRP, and the robust PCA method, COP. It is interesting that K-means clustering performs best in the original space on *Sector*. This may be due to that this data contains many relevant features, resulting in no obvious curse of dimensionality issue. *Olivetti* may contain complex manifolds which require extensive neighbourhood information to find them, so only HLLE can achieve this goal in such cases. Nevertheless, RDP performs much more stably than HLLE across the five datasets.

Ablation Study

Similar to anomaly detection, this section examines the contribution of the two loss functions L_{rdp} and L_{aux}^{clu} to the performance of RDP, as well as the impact of different supervisory sources on the performance. The F-score results of this experiment are shown in Table 5, in which the notations have exactly the same meaning as in Table 3. Similar NMI results can also be observed but omitted due to page limits. The full RDP model that uses both L_{rdp} and L_{aux}^{clu} performs more favourably than its two variants, RDP$\backslash L_{rdp}$ and RDP$\backslash L_{aux}^{clu}$, but it is clear that using L_{rdp} only performs very comparably to the full RDP. However, using L_{aux}^{clu} only may result in large performance drops in some datasets, such as *R8*, *20news* and *Olivetti*. This indicates L_{rdp} is a more important loss function to the overall performance of the full RDP model. In terms of supervisory source, distances obtained by the non-linear random projection in RDP are much more effective than the two other sources on some datasets such as *Olivetti* and *RCV1*. Three different supervisory sources are very comparable on the other three datasets.

	Decomposition			Supervision Signal	
Data	**RDP**	**RDP**$\backslash L_{rdp}$	**RDP**$\backslash L_{aux}^{clu}$	**Org_SS**	**SRP_SS**
R8	0.360 ± 0.055	0.312 ± 0.068	0.330 ± 0.052	0.359 ± 0.028	**0.363** ± **0.046**
20news	**0.119** ± **0.006**	0.083 ± 0.010	0.117 ± 0.005	0.111 ± 0.005	0.111 ± 0.007
Olivetti	**0.638** ± **0.026**	0.602 ± 0.023	0.597 ± 0.019	0.610 ± 0.022	0.601 ± 0.023
Sector	0.191 ± 0.007	0.184 ± 0.010	**0.217** ± **0.007**	0.181 ± 0.007	0.186 ± 0.009
RCV1	**0.572** ± **0.003**	0.514 ± 0.057	0.526 ± 0.011	0.523 ± 0.003	0.532 ± 0.001

Table 5: F-score ablation study performance of K-means clustering.

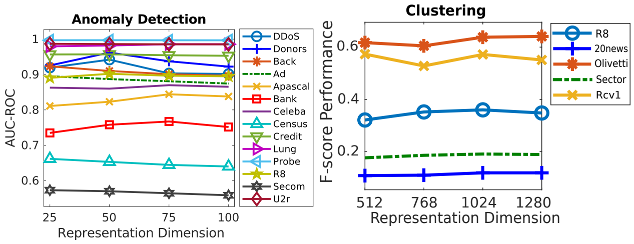

Figure 2: AUC-ROC and F-score performance of RDP using different representation dimensions in anomaly detection and clustering.

4.3 Sensitivity w.r.t. the Dimensionality of Representation Space

Figure 2 shows the AUC-ROC and F-score performance of RDP w.r.t. different representation dimensions in respective anomaly detection and clustering tasks. The results show RDP performs very stably w.r.t. the representation dimensionality on different datasets and downstream learning tasks. It is interesting to note that, the flat trends also indicate, as an unsupervised learning source, the random distance cannot provide sufficient supervision information to learn richer and more complex representations in a higher-dimensional space. This also explains the performance on a few datasets where the performance of RDP decreases when increasing the representation dimension. In general, the representation dimension 50 and 1024 are recommended for RDP to achieve effective anomaly detection and clustering respectively on datasets.

5 Related Work

Self-supervised Learning. Self-supervised learning has been recently emerging as one of the most popular and effective approaches for representation learning, especially in the scenarios where only a very limited amount of manually labelled data is available. Many of the self-supervised methods learn high-level representations by predicting some sort of 'context' information, such as spatial or temporal neighbourhood information. For example, the popular distributed representation learning techniques in NLP, such as CBOW/skip-gram [Mikolov *et al.*, 2013] and phrase/sentence embeddings in [Le and Mikolov, 2014] , learn the representations by predicting the text pieces (e.g., words/phrases/sentences) using its surrounding pieces as the context. In image processing, the pretext task can be the prediction of a patch of missing pixels [Pathak *et al.*, 2016] or the relative position of two patches [Doersch *et al.*, 2015]. Also, a number of studies [Misra *et al.*, 2016; Lee *et al.*, 2017; Oord *et al.*, 2018] explore temporal contexts to learn representations from video data, e.g., by learning the temporal order of sequential frames. Some other methods [Agrawal *et al.*, 2015; Zhou *et al.*, 2017; Gidaris *et al.*, 2018] are built upon a discriminative framework which aims at discriminating the images before and after some transformation, e.g., ego motion in video data [Agrawal *et al.*, 2015; Zhou *et al.*, 2017] and rotation of images [Gidaris *et al.*, 2018]. There have also been popular to use generative adversarial networks (GANs) to learn features [Radford *et al.*, 2015; Chen *et al.*, 2016]. The above methods have demonstrated powerful capability to learn semantic representations. However, most of them use the supervisory signals available in image/video data only, which limits their application to other types of data, such as tabular data. Though our method may also work on image/video data, we focus on handling high-dimensional tabular data to bridge this gap.

Other Approaches. There have been several well-established unsupervised representation learning approaches for handling tabular data, such as random projection [Bingham and Mannila, 2001; Li *et al.*, 2006], PCA [Schölkopf *et al.*, 1997; Rahmani and Atia, 2017], manifold learning [Donoho and Grimes, 2003; Hinton and Roweis, 2003] and autoencoder [Hinton and Salakhutdinov, 2006; Vincent *et al.*, 2010]. One notorious issue of PCA or manifold learning approaches is their prohibitive computational cost in dealing with large-scale high-dimensional data due to the costly neighbourhood search and/or eigen decomposition. Random projection is a computationally efficient approach, supported by proven distance preservation theories such as the Johnson-Lindenstrauss lemma [Johnson and Lindenstrauss, 1984]. We show that the preserved distances by random projection can be harvested to effectively supervise the representation learning. Autoencoder networks are another widely-used efficient feature learning approach which learns low-dimensional representations by minimising reconstruction errors. One main issue with autoencoders is that they focus on preserving global information only, which may result in loss of local structure information. Some feature learning methods are specifically designed for anomaly detection [Pang *et al.*, 2018; Zong *et al.*, 2018; Burda *et al.*, 2019]. By contrast, we aim at generic representations learning while being flexible to incorporate optionally task-dependent losses to learn task-specific semantically rich representations.

6 Conclusion

We have introduced a novel Random Distance Prediction (RDP) model which learns features in a fully unsupervised fashion by predicting data distances in a randomly projected space. The key insight is that random mapping is a theoretically proven approach to obtain approximately preserved distances, and to well predict these random distances, the representation learner is optimised to learn preserved proximity information while at the same time rectifying inconsistent proximity, resulting in representations with optimised distance preserving. Our idea is justified by thorough experiments in two unsupervised tasks, anomaly detection and clustering, demonstrating that RDP-based anomaly detectors and clustering substantially outperform their counterparts on real-world datasets. Our empirical results also demonstrate that RDP is flexible and very effective to incorporate task-dependent complementary auxiliary losses and learns more expressive representations. We are extending RDP to other types of data, such as image/text data.

References

[Agrawal *et al.*, 2015] Pulkit Agrawal, Joao Carreira, and Jitendra Malik. Learning to see by moving. In *Proc. IEEE Int. Conf. Comp. Vis.*, pages 37–45, 2015.

[Beyer *et al.*, 1999] Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. When is “nearest neighbor” meaningful? In *Proc. Int. Conf. Database Theory*, 1999.

[Bingham and Mannila, 2001] Ella Bingham and Heikki Mannila. Random projection in dimensionality reduction: applications to image and text data. In *Proc. ACM SIGKDD Int. Conf. Know. Disco. & Data Mining*, 2001.

[Burda *et al.*, 2019] Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. In *Proc. Int. Conf. Learn. Repre.*, 2019.

[Chen *et al.*, 2016] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 2172–2180, 2016.

[Demšar, 2006] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.*, 2006.

[Doersch *et al.*, 2015] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proc. IEEE Int. Conf. Comp. Vis.*, pages 1422–1430, 2015.

[Donoho and Grimes, 2003] David L Donoho and Carrie Grimes. Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *Proceedings of the National Academy of Sciences*, 100(10):5591–5596, 2003.

[Gidaris *et al.*, 2018] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *Proc. Int. Conf. Learn. Repre.*, 2018.

[Hartigan and Wong, 1979] John A Hartigan and Manchek A Wong. Algorithm as 136: A k-means clustering algorithm. *J. Royal Stat. Society*, 1979.

[Hinton and Roweis, 2003] Geoffrey E Hinton and Sam T Roweis. Stochastic neighbor embedding. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 857–864, 2003.

[Hinton and Salakhutdinov, 2006] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[Johnson and Lindenstrauss, 1984] William B Johnson and Joram Lindenstrauss. Extensions of lipschitz mappings into a hilbert space. *Contemporary Math.*, 1984.

[Le and Mikolov, 2014] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *Proc. Int. Conf. Mach. Learn.*, pages 1188–1196, 2014.

[Lee *et al.*, 2017] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *Proc. IEEE Int. Conf. Comp. Vis.*, pages 667–676, 2017.

[Li *et al.*, 2006] Ping Li, Trevor J Hastie, and Kenneth W Church. Very sparse random projections. In *Proc. ACM SIGKDD Int. Conf. Know. Disco. & Data Mining*, 2006.

[Liu *et al.*, 2008] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *Proc. IEEE Int. Conf. Data Mining*, pages 413–422. IEEE, 2008.

[Mikolov *et al.*, 2013] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv:1301.3781*, 2013.

[Misra *et al.*, 2016] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *Proc. Eur. Conf. Comp. Vis.*, pages 527–544. Springer, 2016.

[Oord *et al.*, 2018] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018.

[Pang *et al.*, 2018] Guansong Pang, Longbing Cao, Ling Chen, and Huan Liu. Learning representations of ultrahigh-dimensional data for random distance-based outlier detection. In *Proc. ACM SIGKDD Int. Conf. Know. Disco. & Data Mining*, pages 2041–2050. ACM, 2018.

[Pathak *et al.*, 2016] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pages 2536–2544, 2016.

[Radford *et al.*, 2015] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *Proc. Int. Conf. Learn. Repre.*, 2015.

[Rahimi and Recht, 2008] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Proc. Adv. Neural Inf. Process. Syst.*, 2008.

[Rahmani and Atia, 2017] Mostafa Rahmani and George Atia. Coherence pursuit: Fast, simple, and robust subspace recovery. In *Proc. Int. Conf. Mach. Learn.*, pages 2864–2873. JMLR. org, 2017.

[Schölkopf *et al.*, 1997] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Kernel principal component analysis. In *Proc. Int. Conf. Arti. Neur. Net.*, pages 583–588. Springer, 1997.

[Vempala, 1998] Santosh Vempala. Random projection: A new approach to VLSI layout. In *Proc. Ann. Symp. Found. Comp Sci.*, pages 389–395. IEEE, 1998.

[Vincent *et al.*, 2010] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.*, 2010.

[Zhou *et al.*, 2017] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, pages 1851–1858, 2017.

[Zong *et al.*, 2018] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *Proc. Int. Conf. Learn. Repre.*, 2018.