N2D: (Not Too) Deep Clustering via Clustering the Local Manifold of an Autoencoded Embedding

https://github.com/rymc/n2d.

1. Deep clustering has increasingly been demonstrating superiority over conventional shallow clustering algorithms. we propose to learn an autoencoded embedding and then search this further for the underlying manifold. For simplicity, we then cluster this with a shallow clustering algorithm, rather than a deeper network.
2. k-means [17], along with many conventional clustering algorithms such as Gaussian Mixture Models (GMMs) [23], DBSCAN [4], and hierarchical algorithms [12] typically require **hand engineered features to be created** for each dataset and task. Further, these features may then be **analysed using another process**, **feature selection**, in order to eliminate redundant or poor quality features. **This task is even more challenging in the unsupervised setting**. Additionally, it is a time-consuming and brittle process, with the choice of features having a large influence over the subsequent performance of the clustering algorithm.
3. Recent advances in deep learning have paved the way for algorithms which can effectively learn from raw data, **bypassing the need for manual feature extraction and selection**.
4. Autoencoders effectively seek to learn the intrinsic structure of the data with a deep neural network, and do so by learning to reconstruct the original data. This representation learned from the raw data is then typically used in a range of tasks, such as an input to a supervised classifier.
5. Deep clustering refers to the process of clustering with deep neural networks, typically with features automatically learned from the raw data by CNNs [30] or autoencoders [28] and clustered with a deep neural network.These algorithms have reported large performance gains on various benchmark tasks over conventional non-deep clustering algorithms.
6. We propose a simple approach, N2D, that effectively replaces the clustering network with a manifold learning technique on top of the autoencoded representation.
7. Given this updated embedding, we can then cluster it with conventional nondeep clustering algorithms. By doing so, N2D replaces the complexity of the clustering network with a manifold learning method and straightforward non-deep clustering algorithm,
8. One important question is which manifold learning technique to apply to the autoencoded representation. There are many possible methods, such as the well-known Principal
9. Component Analysis (PCA) [1].
10. More recently, UMAP [19] has been proposed, which while also local, has been shown to better preserve global structure. All of these methods seek to utilize the distances between points in order to better learn the underlying structure, and we posit that they will improve the clusterability of an autoencoded embedding.
11. Clustering algorithms can be broadly categorized into two different categories, hierarchical clustering and partitional clustering.
12. Deep clustering methods use deep neural networks to cluster, typically involving two different processes, one where a representation is learned, and one where the actual clustering occurs. This process may occur separately or jointly.
13. When used in the representation learning step, MLPs [28], Convolutional Neural Networks

(CNNs) [30] and Generative Adversarial Networks (GANs) [20]) these methods will optimize a specific loss, such as the reconstruction loss or generative adversarial loss. However, in addition, a clustering loss is added to guide the algorithm to find more cluster friendly features. These losses may include a k-means loss [29] or a cluster hardening loss [28]. These losses are then typically combined in some way, such as with joint training, where the clustering loss is usually given much lower weight than the non-clustering loss [7].

1. Along these lines, IDEC [7] and ASPC-DA [6] both use an autoencoder for their initial pre-training step. Based on this learned representation, these methods initialize the weights of a new clustering network with k-means. IDEC and ASPCDA then jointly trained this clustering network with the autoencoder.
2. An alternative to using two different losses is to use a single combined loss, such as DEC [28] or JULE [30].
3. The concept of manifold learning on embeddings has been explored by Hasan and Curry [8].
4. Our method relies primarily on the combination of two different manifold learning methods. The first is an autoencoder, which while learning a representation, does not explicitly take local structure into account. We will show that by augmenting the autoencoder with a manifold learning technique which explicitly takes local structure into account, we can increase the quality of the representation learned in terms of clusterability.
5. An autoencoder
   1. is a deep neural network consisting of two key components. The first is the encoder, which attempts to learn a function which maps the input x to a new feature vector (h = f(x)). The second component is the decoder, which attempts to learn a function which maps the learned feature space back to the original input space (r = g(h).
   2. The learning process can be described as minimizing the loss function L(x; g(f(x))), where L is a function which penalizes g(f(x)) for being dissimilar to x. One such loss may be the Mean Squared Error (MSE).
   3. While autoencoders have been shown to perform well at many feature representation tasks, they do **not explicitly preserve the distances of the data** in the representation that they learn.
6. Isomap
   1. There are a multitude of manifold learning techniques that explicitly seek to preserve distances within the data. Isomap [25] is a nonlinear method which extends multidimensional scaling (MDS) to incorporate geodesic distances imposed by a weighted graph.
   2. Geodesic distance is the distance between two points measured over the manifold, and thus by using the geodesic distance Isomap can learn the manifold structure.
   3. While Isomap is a global approach and our hypothesis is that a learning a local manifold on the autoencoded embedding will lead to better results, we will investigate the use of Isomap within N2D, specifically to understand how a global method performs and test our hypothesis.
   4. We consider Isomap to have two key parameters in our setting, the first is the number of components, which is the top n eigenvectors of the geodesic distance matrix which represent the co-ordinates in the new space. The next parameter of importance is the number of neighbours to consider, which is simply the number of k-nearest neighbours to consider as local to a point.
7. t-SNE
   1. t-SNE (t-distributed Stochastic Neighbor Embedding) [18] is a nonlinear method with a specific objective of optimizing local distances when creating the embedding. The first stage of the t-SNE algorithm is to construct a probability distribution over pairs within the data in such away that similar points will have a high probability of being chosen while dissimilar points have an extremely low probability of being chosen. In the second stage t-SNE defines a probability distribution over the mapped points, minimising the KullbackLeibler (KL) divergence between the two distributions.
8. UMAP
   1. A recently proposed manifold learning method is UMAP (Uniform Manifold Approximation and Projection) [19], which seeks to accurately represent local structure, but has been shown to also better incorporating global structure. UMAP better preserves global structure, while remaining focused on preserving distances within local neighbourhoods, it may inherit benefits from both local and global methods.
   2. UMAP relies on three assumptions, namely that the data is uniformly distributed on a Riemannian manifold, that the Riemannian metric is locally constant and that the manifold is locally connected. From these assumptions it is possible to model the manifold with a fuzzy topological structure.
   3. UMAP is similar to Isomap [25] in that it uses a k-neighbour based graph algorithm to compute the nearest neighbours of points. At a high level, UMAP first constructs a weighted kneighbour graph, and from this graph a low dimensional layout is computed. This low dimensional layout is optimized to have as close a fuzzy topological representation to the original as possible based on cross entropy.
9. N2D
   1. For all manifold learning methods, we set the number of components or dimensions to be the number of clusters in the data.
   2. For Isomap and UMAP we consider the number of neighbours to be an important parameter, and we set it to a sensible default value of 5 for Isomap, and 20 for UMAP.
   3. UMAP also has another parameter we believe will be influential, which is the minimum distance between points. We believe that a default minimum distance of 0 is ideal for

our method, as our prime motivation is not visualization and thus a more accurate representation of the true manifold is preferred.

* 1. We summarize the high level steps of our proposed method N2D as:
     1. Apply an autoencoder *FA* to the raw data to learn an initial representation.
     2. We re-embed the autoencoded embedding by searching for a more clusterable manifold with a manifold learning method *FM* which preserves local distances.
     3. Finally, given this new, more clusterable embedding, we apply a final shallow clustering algorithm *FC* to discover the clusters.

More concisely, we may also simply represent N2D as

*C = FC(FM(FA(X)))*

where C is the final clustering, *FC* is the clustering algorithm, *FM* is the manifold learner, *FA* is the autoencoder and X is the original data.

* 1. we conduct experiments on a range of diverse datasets, including standard datasets used to evaluate deep clustering algorithms.

1. Evaluation Metrics

We will use two standard evaluation metrics for validating the performance of unsupervised clustering algorithms.

* 1. Accuracy: In clustering, accuracy (ACC) is defined as the best match between the ground truth and the predicted clusters. where y are the ground truth labels, c are the cluster labels, and m enumerates mappings between clusters and labels.

Text

Description automatically generated with low confidence

* 1. Normalized Mutual Information: The Normalized Mutual Information (NMI) can be viewed as a normalization of the mutual information to scale the results between 0 and 1, where 0 has no mutual information and 1 is perfect correlation. where y are the ground truth labels, c are the cluster labels, H measures the entropy, and I is the mutual information between the ground truth labels and the cluster labels.

Text

Description automatically generated with low confidence

1. Experimental Settings
   1. We base our autoencoder on the architecture described by Xie et al. [28], which is a fully connected Multi-Layer Perceptron (MLP). The dimensions are inspired by those chosen by van der Maaten et al. in t-SNE [18], which are d-500-500-2000-c, where d is the dimensionality of the data and c is the number of clusters. All layers use ReLU activation [21]. The optimizer is Adam [15]. We train the autoencoder on for 1000 epochs for all datasets.
   2. We use UMAP with the following default parameter set across all datasets. The number of neighbours is 20, the number of dimensions is the number of clusters, and the minimum distance between each point in the manifold is 0.
   3. We use a GMM for the final clustering algorithm, where each component has its own general covariance matrix, and there are c components, where c is the number of clusters.
2. Results
   1. TABLE I: Evaluating the performance of each component of the proposed method. AE, UMAP and N2D each have a GMM clustering step post-manifold learning. Results for Isomap were sometimes not available (—) as it exhausted memory on our 64GB machine.

Table

Description automatically generated

* 1. This table shows that the performance of the non-deep clustering algorithm GMM is typically poorest across all datasets. When we introduce manifold learning methods, and cluster those embeddings, we see improvements in cluster accuracy and NMI.
  2. we use 3 different manifold learning methods (Isomap, tSNE and Umap) with different properties. The first is Isomap, which is a globally focused manifold learner. When applied to the autoencoded embedding, N2D with UMAP outperforms both Isomap and t-SNE on all datasets. This supports the hypothesis that a manifold learner,

1. A comparison of our method with both shallow clustering algorithms, along with the latest deep-clustering algorithms. Results were retrieved from the literature, or computed by us when not found and possible to compute.

Table

Description automatically generated

1. The most similar methods to N2D are IDEC and ASPCDA. Both of these approaches pre-train an autoencoder before jointly training a second deep network with a clustering and non-clustering (reconstruction) loss. The clustering network weights are initialized with a non-deep clustering algorithm such as k-means.
2. In contrast, we replace the second deep network with a manifold learning method, UMAP, and then use a nondeep clustering algorithm, a GMM, to cluster the resulting embedding. Hence, our less deep method, N2D, benefits from less complexity
3. On five of the six datasets tested, our approach is in the top 3 for at least one of the metrics.
4. We also include two non-image datasets, pendigits and HAR, to validate performance on different types of data. Many of the best-performing deep-clustering methods are intended for image clustering
5. We also note that one of the closest competitors, ASPC-DA, which typically slightly outperforms our method on several datasets, achieves this performance due to **data augmentation**.
6. N2D, which reduces the deepness of typical deep clustering algorithms by replacing the clustering network with an alternative framework which seeks to find the manifold within the autoencoder embedding, and clusters this new embedding with a shallow clustering architecture.