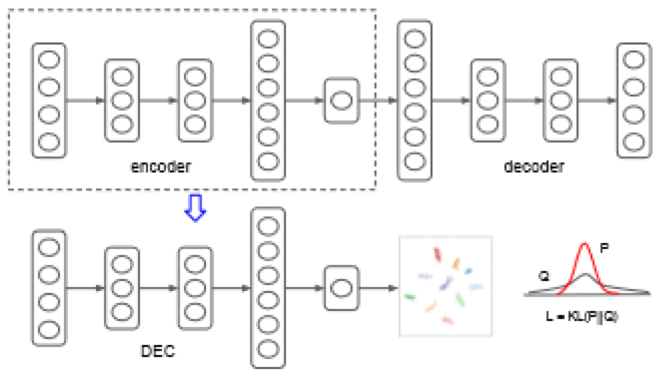
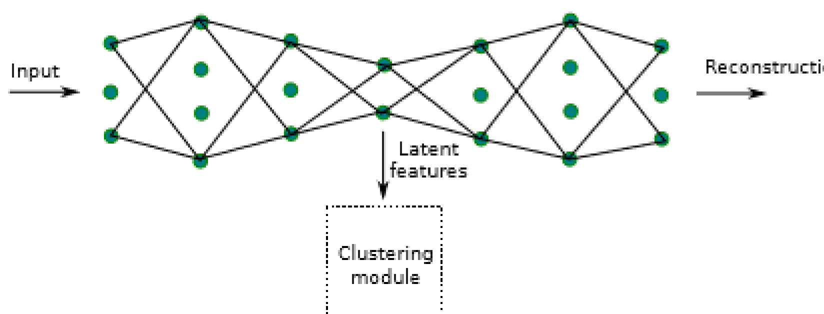
X: Deep Clustering

Deep Clustering Research Papers

Deep Embedding Clustering	Research Summary	
Unsupervised Deep Embedding for Clustering Analysis http://proceedings.mlr.press/v48/xieb16.pdf	This study focus on learning representations for clustering and proposes **Deep Embedded Clustering (DEC)**, a method that simultaneously learns feature representations (deep autoencoder) and cluster assignments (K-means). DEC learns a mapping from the data space to a lower-dimensional feature space in which it iteratively optimizes a clustering objective. DEC Steps: ▪ Parameter initialization with deep autoencoder, ▪ KL Divergence Minimization: compute an auxiliary target distribution and minimize the Kullback-Leibler (KL) divergence to it. ▪ Optimization: jointly optimize the cluster centers and Neural Network parameters using SGD with momentum. Experiment on image and text corpora.	 *Figure 1.* Network structure Evaluation Metric Unsupervised Clustering Accuracy (ACC)
Towards K-means-friendly Spaces: Simultaneous Deep Learning and Clustering https://arxiv.org/pdf/1610.04794.pdf	Researchers propose a joint dimension reduction (DR) and K-means clustering approach in which DR is accomplished via learning a deep neural network (DNN). K-means works well when the samples are evenly scattered around their centroids in the feature sapce Aim to avoid trivial solutions in DEC and propose new cost function to jointly optimize two tasks (DR and DNN), while exploiting the deep neural network's ability to approximate any nonlinear function.	 *Figure 3.* Proposed deep clustering network (DCN).
A Survey of Clustering With Deep Learning: From the Perspective of Network Architecture https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8412085	One systematic survey of clustering with deep learning in views of architecture. ▪ Introduce the preliminary knowledge for better understanding of this eld. ▪ Provide a taxonomy of clustering with deep learning and some representative methods. ▪ Propose interesting future opportunities of clustering with deep learning and give some conclusion remarks Classical clustering methods are usually categorized as partition-based methods, density-based methods, hierarchical methods and so on.	**Clustering Performance Evaluation Metrics:** Unsupervised Metrics: ▪ Unsupervised clustering accuracy (ACC) $ACC = \max_m \frac{\sum_{i=1}^{n} 1\{y_i = m(c_i)\}}{n}$ ▪ Normalized Mutual Information (NMI) $NMI(Y, C) = \frac{I(Y, C)}{\frac{1}{2}[H(Y) + H(C)]}$

Deep Clustering Categories in this paper:

- AE: use autoencoder to create nonlinear mapping function.
- CDNN: clustering feed-forward networks trained by specific clustering loss, referred as Clustering DNN.
- VAE: clustering based on Variational Autoencoder.
- GAN: clustering based on Generative Adversarial Network.

Neural Network Architecture for Deep Clustering:

- FCN - Feedforward fully-connected neural network
- CNNs - convolutional neural networks:
  (it can be trained with a specific clustering without any requirement on initialization. No theoretical explanation is given in any existing papers.)
- DBNs - Deep Belief Network: generative graphical models which learn to extract a deep hierarchical representation of the input data.
- AE - Autoencoder
- Generative Adversarial Network (GAN) and Variational Autoencoder (VAE)

Loss Function related to Clustering:

- **Principal Clustering Loss:** it contains cluster centroids and cluster assignments. Cluster loss directly guide training neural network to form clusters.
  K-means loss: *Towards K-means-friendly Spaces: Simultaneous Deep Learning and Clustering*
  Cluster assignment hardening loss: *Unsupervised deep embedding for clustering analysis*
  Agglomerative clustering loss: *A gglomerative clustering of a search engine query log*
  Nonparametric maximum margin clustering: *Deep learning with nonparametric clustering*
- **Auxiliary Clustering Loss:** use deep learning merely with auxiliary clustering loss to obtain clusters. It is a more feasible representation for clustering but doesn't output clusters straightforwardly.
  Locality-preserving loss: enforce the network to preserve the local property of embedding.
  Group sparsity loss: explots block diagonal similarity matrix for representation learning.
  Sparse subspace clustering loss: learns sparse code of data.

Loss functions are the optimizing objective of deep clustering and are typically composed Ln (loss of network) and Lc (loss of clustering).

$$L = \lambda L_n + (1-\lambda) L_c$$

TABLE 2. Comparison of algorithms based on network architecture and loss function.

Categories	Algorithms	Network Architecture	Network loss	Clustering loss: Principal	Clustering loss: Auxiliary
AE	DCN	AE	reconstruction loss	k-means loss	N
	DEN	AE	reconstruction loss	N	1) locality-preserving constraint 2) group sparsity constraint
	DSC-Nets	CAE	reconstruction loss	N	self-expressiveness term
	DMC	AE	reconstruction loss	proximity penalty term	locality-preserving loss
	DEPICT	CAE (Denoising)	reconstruction loss	unsupervised cross entropy loss	N
	DCC	AE/CAE	reconstruction loss	robust continuous clustering loss	N
CDNN	DNC	RBM	N	nonparametric maximum margin clustering loss	N
	DEC	FCN	N	cluster assignment hardening loss	N
	DBC	CNN	N	cluster assignment hardening loss	N
	CCNN	CNN	N	k-means	N
	IMSAT	FCN	N	1) regularized information maximization, 2) self-augmented training loss	N
	JULE	CNN	N	agglomerative clustering	N
	DAC[1]	CNN	N	pairwise-classification loss	N
VAE	VaDE	VAE	variational lower bound on the marginal likelihood, with a GMM priori		
	GMVAE	VAE	variational lower bound on the marginal likelihood, with a GMM priori		
GAN	DAC[2]	Adversarial autoencoder	reconstruction loss	1) GMM likelihood, 2) adversarial objective	N
	CatGAN	GAN	adversarial objective with a multi-classes priori		
	InfoGAN	GAN	adversarial objective with a multi-classes priori		

[1] Deep Adaptive Clustering
[2] Deep Adversarial Clustering

TABLE 3. Main contributions of the representative algorithms.

Categories	Algorithms	Main contributions to clustering
AE	DCN	perform k-means clustering and feature learning simultaneously, simple but effective
	DEN	learn a clustering-friendly representation
	DSC-Nets	improve the classical subspace clustering by AE
	DMC	improve the classical multi-manifold clustering by AE
	DEPICT	computational efficient, robust, perform well on image datasets
	DCC	avoid alternative optimization, require no prior knowledge of cluster number
CDNN	DNC	improve the classical NMMC clustering by DBN
	DEC	the first well-known deep clustering method, making this field popular
	DBC	improve DEC using CNN
	CCNN	computational efficient, deal with large-scale image datasets
	IMSAT	introduce self-augment training to deep clustering
	JULE	perform well on image datasets, but have high computational and memory cost
	DAC	well-designed clustering loss, achieve the-state-of-art performance on several datasets
VAE	VaDE	combine VAE with clustering
	GMVAE	combine VAE with clustering
GAN	DAC	combine AAE with clusteirng
	CatGAN	combine GAN with clustering
	InfoGAN	learn disentangled representations

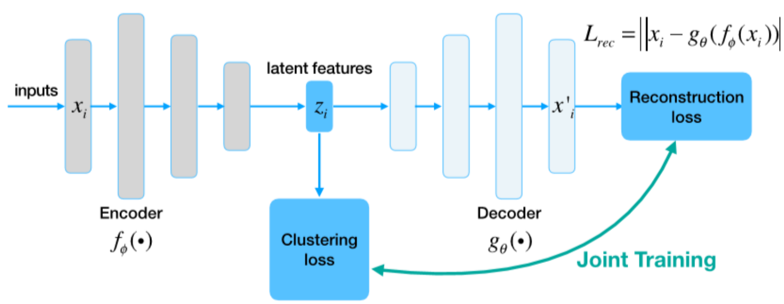
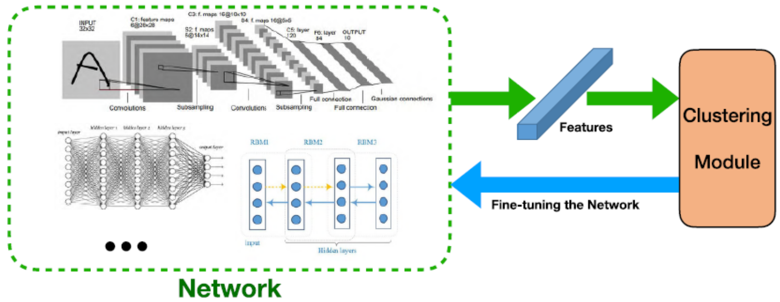
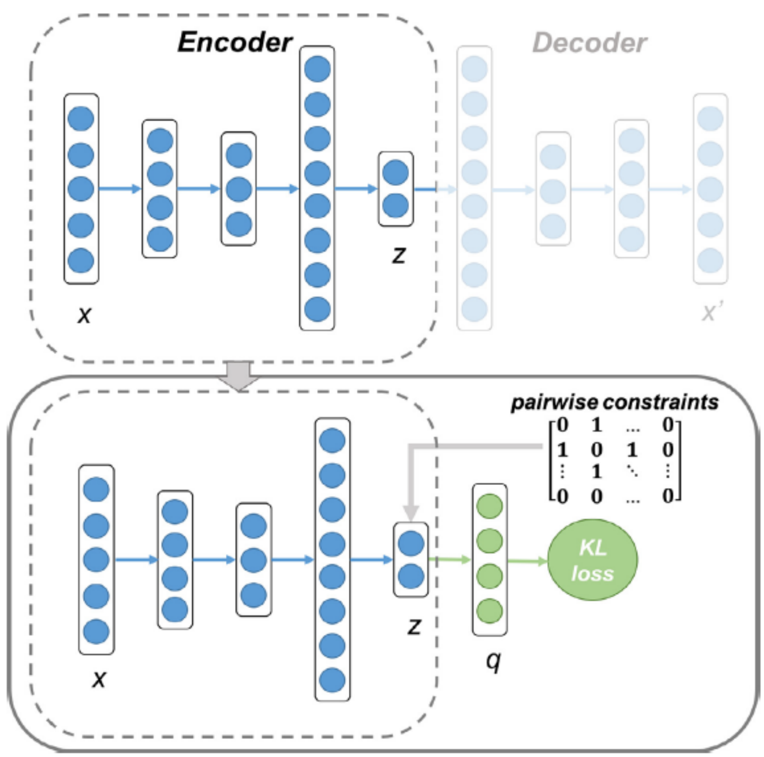
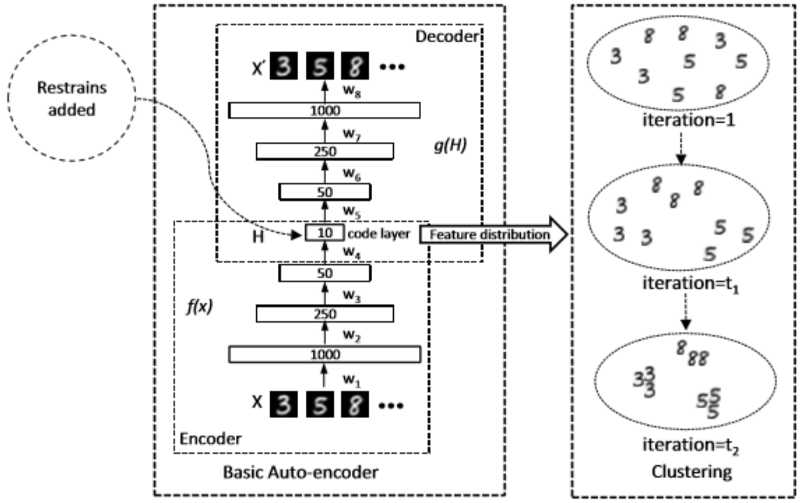
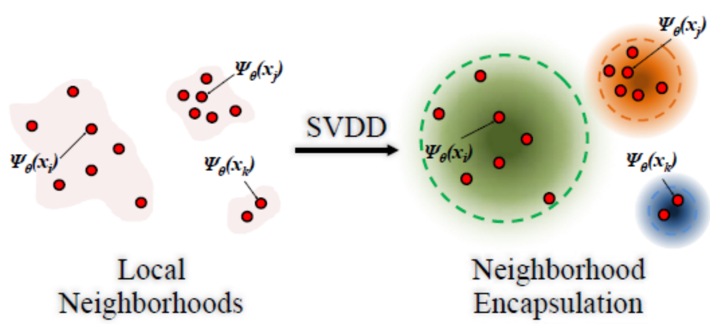


FIGURE 1. Architecture of clustering based on autoencoder. The network is trained by both clustering loss and reconstruction loss.

	If it is CDNN , only Lc will be considered.	 **FIGURE 2.** Architecture of CDNN-based deep clustering algorithms. The network is only adjusted by the clustering loss. The network architecture can be FCN, CNN, DBN and so on.
Summary of Deep Clustering Method and Code https://github.com/zhoushengisnoob/DeepClustering		
Semi-supervised Deep Embedded Clustering https://www.researchgate.net/publication/ 328300194_Semi-supervised_Deep_Embedded_Clustering	Deep embedded clustering (DEC) is one of the state-of-the-art deep clustering methods. However, DEC does not make use of prior knowledge to guide the learning process. This paper propose a new scheme of **semi-supervised deep embedded clustering (SDEC)** to overcome this limitation. - SDEC learns feature representations that favor the clustering tasks and performs clustering assignments simultaneously. - In contrast with DEC, SDEC incorporates pairwise constraints in the feature learning process such that data samples belonging to the same cluster are close to each other and data samples belonging to different clusters are far away from each other in the learned feature space.	Y. Ren et al./Neurocomputing 325 (2019) 121–130 

Auto-encoder Based Data Clustering http://cripac.ia.ac.cn/people/lwang/M-MCG/research/CFS-CIAPR13/paper.pdf	Based on the auto-encoder network, researchers create a highly non-linear mapping function to learn an eective representation in a low dimensional space. Then they propose a new objective function to make similar input data obtain the same representations in the code layer. The proposed objective ensures that the data representations in the code layer are close to their corresponding cluster centers, and meanwhile the reconstruction error is still under control, which is important to obtain stable non-linear mapping.	 **Fig. 2.** Framework of the proposed method

Unknown Number of Clusters		
Deep Density Clustering of Unconstrained Faces http://openaccess.thecvf.com/content_cvpr_2018/papers/ Lin_Deep_Density_Clustering_CVPR_2018_paper.pdf	**Deep Density Clustering (DDC)**: DDC is tested on clustering unconstrained face images without prior knowledge of the number of distinct subjects. DDC consists of three main steps: - Extracting deep features, - Computing density-based similarity, - Merging clusters. Specifically, DDC contains three steps (1) Build a nearest-neighbor graph for the entire dataset. (2) Represent each neighborhood in a compact form. (3) Compute a density-based similarity.	DDC first associates each data point with one E neighborhood. Points inside the neighborhood are then represented by a minimal covering sphere which encapsulates local information. Finally, DDC computes pairwise similarity by evaluating data points on the functional defined by the spheres.  Figure 3: Neighborhood encapsulation. (left) Pink regions are the local neighborhoods of the points x_i, x_j, and x_k in feature space. (right) Encapsulations are learned by solving (3). The encapsulation is density-aware. In the figure, regions closer to the centers of the spheres have higher density.