

Unsupervised Deep Embedding for Clustering Analysis

Bachelorseminar Data Mining

Lukas Mahr

Ludwig-Maximilians-Universität München

Roadmap

- 1 Clustern von Daten mit hohen Dimensionen
- 2 Einleitung zu Neuronalen Netzen
- 3 Autoencoders
- 4 Unsupervised Deep Embedding for Clustering Analysis
- 5 Deep embedded clustering
- 6 Ergebnisse

■ Problem

- Gaussian Mixture Models, KMeans
 - Abstandsmetriken sind beschränkt auf den ursprünglichen Datendimensionen
 - unwirksam wenn Datendimension hoch sind[1]
- Variationen von KMeans für Daten mit hohen Dimensionen
 - limitiert zu linearen Embeddings[2]
- Spectral clustering
 - Quadratische oder Super-quadratische Komplexität

Clustern von Daten mit hohen Dimensionen

■ Idee

- Neuronalesnetzwerk zum reduzieren der Dimensionen
 - nicht lineares mapping
- Clustern der reduzierten Daten
 - einfaches Clustering möglich, da Dimensionen reduziert
- Verbessern des NN und der Cluster durch Backpropagation

Neuronale Netze

Einleitung zu Neuronalen Netzen

- Nicht-lineare statistische Modelle zur Informationsverarbeitung
- Informationsverarbeitung umfasst hierbei unter anderem
 - Klassifikation
 - Prognosenerstellung
- Units der Neuronalen Netze angelehnt an Neuronen
 - Inputs zusammenfassen
 - Mit Schwellenwert vergleichen bzw. aktivieren
- Verbindungen zwischen Units angelehnt an Synapsen
 - Gewichtung mit verstärkender oder schwächender Wirkung

Einleitung zu Neuronalen Netzen

Künstlichen Neurons

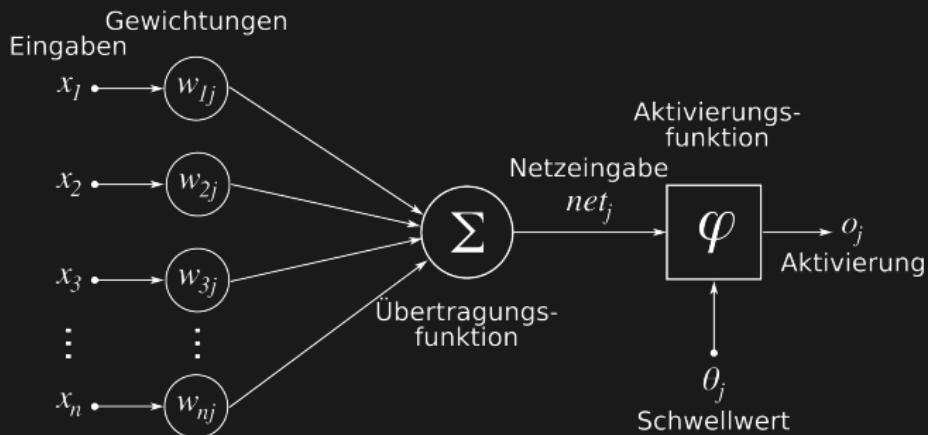


Figure: Darstellung eines künstlichen Neurons mit seinen Elementen

https://de.wikipedia.org/wiki/Datei:ArtificialNeuronModel_deutsch.png

Einleitung zu Neuronalen Netzen

Layer/Schichten

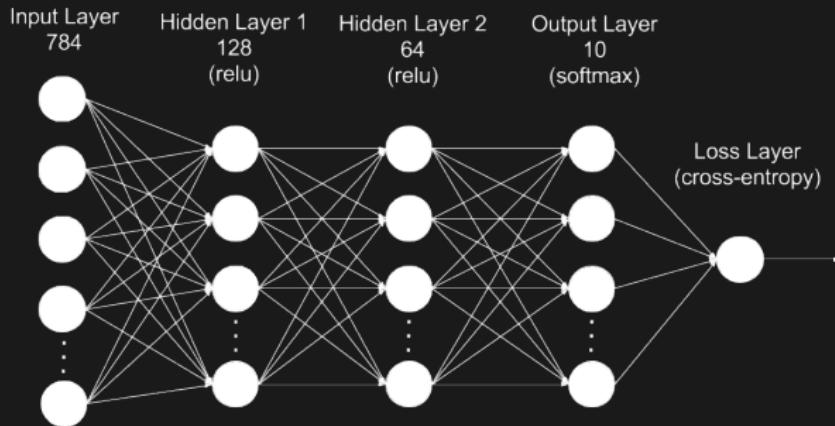


Figure: Deep learning Künstliches neuronales Netz maschinelles lernen
Apache MXNet - mehrschichtige PNG <https://de.cleanpng.com/png-x3zkr7/>

Einleitung zu Neuronalen Netzen

Aktivierungsfunktionen



Figure: Rectifier-Aktivierungsfunktion

https://de.wikipedia.org/wiki/Datei:Activation_rectified_linear.svg

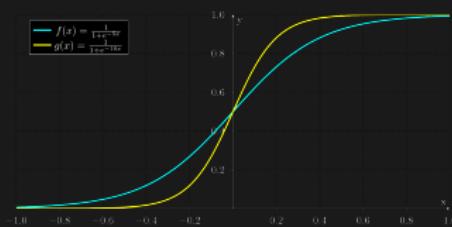


Figure: Sigmoid Funktion mit Steigungsmaß $a=5$ sowie $a = 10$

<https://de.wikipedia.org/wiki/Datei:Sigmoid-function.svg>

Einleitung zu Neuronalen Netzen

Loss/Kostenfunktion

Mean Squared Error

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$$

Mean absolute error

$$MAE = \frac{\sum_{i=1}^n |\hat{y}_i - y_i|}{n}$$

Binary Cross-Entropy

$$H(y, \hat{y}) = -\frac{1}{n} \sum_{i=1}^n y_i \cdot \log(\hat{y}_i) + (1 - y_i) \cdot \log(1 - \hat{y}_i)$$

Figure: https://en.wikipedia.org/wiki/Mean_squared_error#Predictor
https://en.wikipedia.org/wiki/Mean_absolute_error
<https://towardsdatascience.com/understanding-binary-cross-entropy-log-loss-a-visual-explanation-a3ac6025181a>

Einleitung zu Neuronalen Netzen

Backpropagation mit Gradient descent

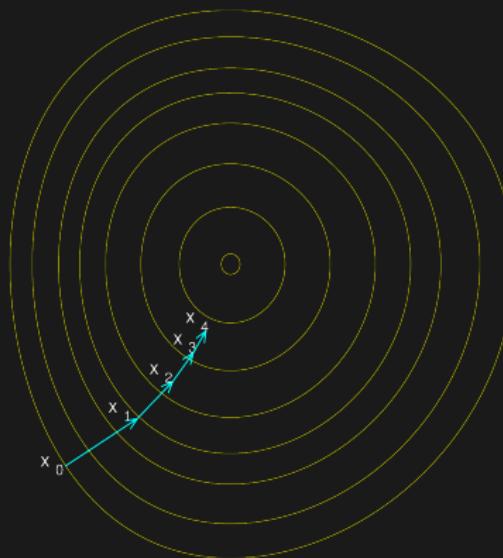


Figure: Illustration of gradient descent on a series of level sets

https://en.wikipedia.org/wiki/File:Gradient_descent.svg

Autoencoder

Autoencoders

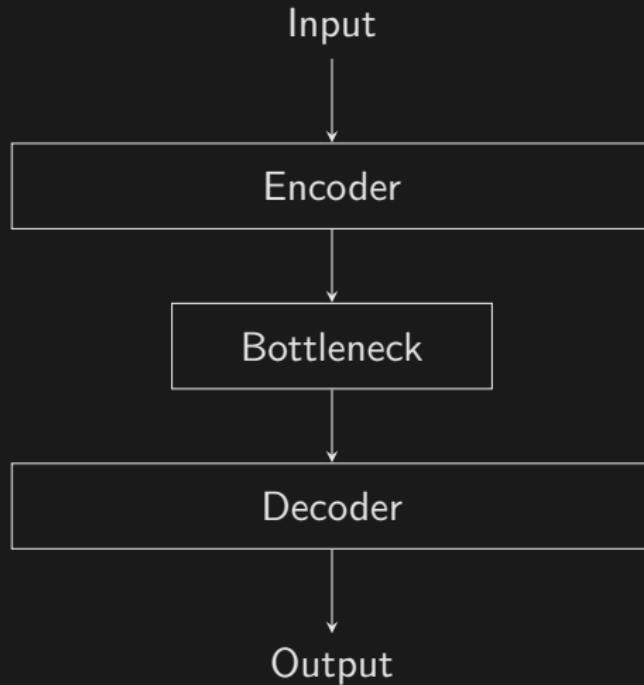


Figure: Einfaches Autoencoder Model

Autoencoders

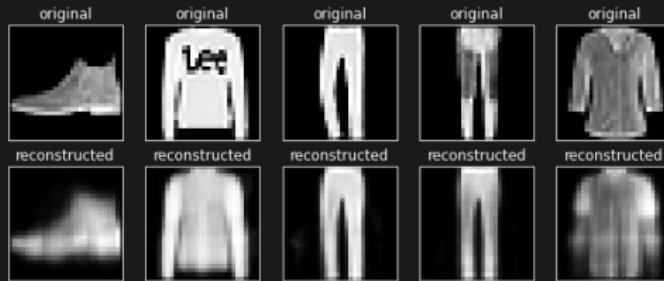


Figure: Orginal und Decoded Bilder

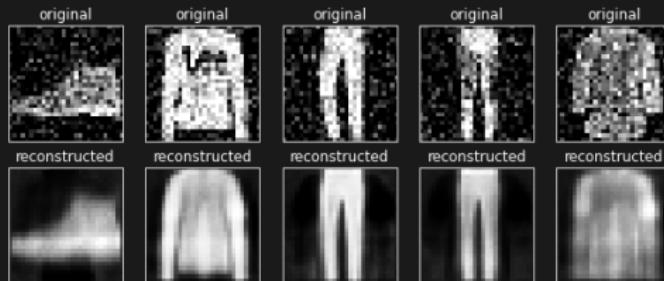


Figure: Noisy und Decoded Bilder

<https://github.com/Plutokekz/dec/blob/main/Autoencoders.ipynb>

DEC

Unsupervised Deep Embedding for Clustering
Analysis

Stacked Autoencoders

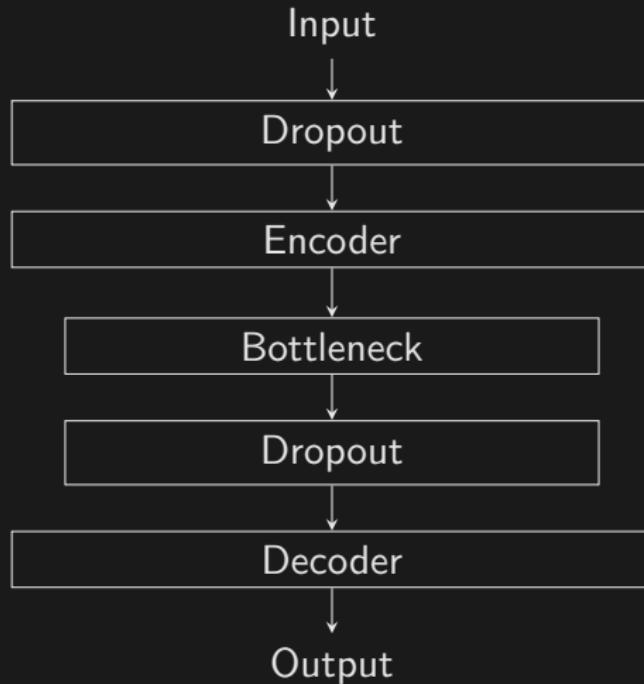


Figure: Autoencoder mit Dropout

Stacked Autoencoders

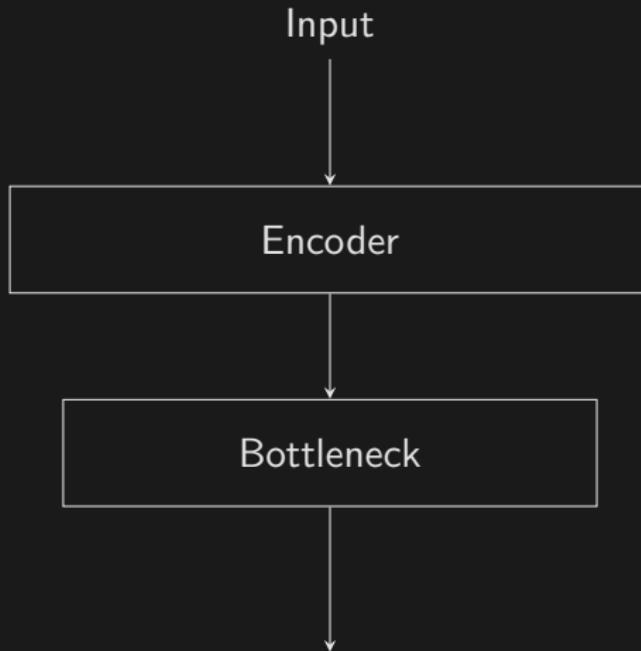


Figure: Encoder ohne Dropout

Stacked Autoencoders

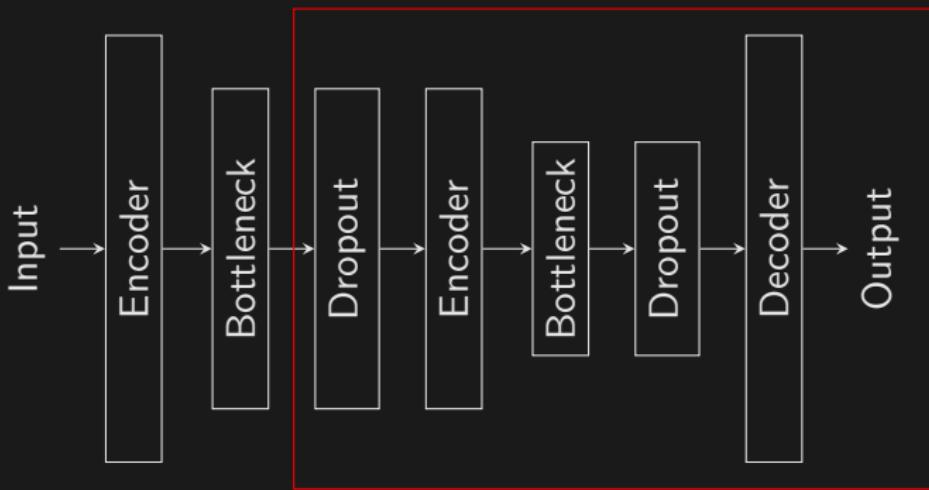


Figure: Stacked Autoencoder

Deep embedded clustering

Deep embedded clustering

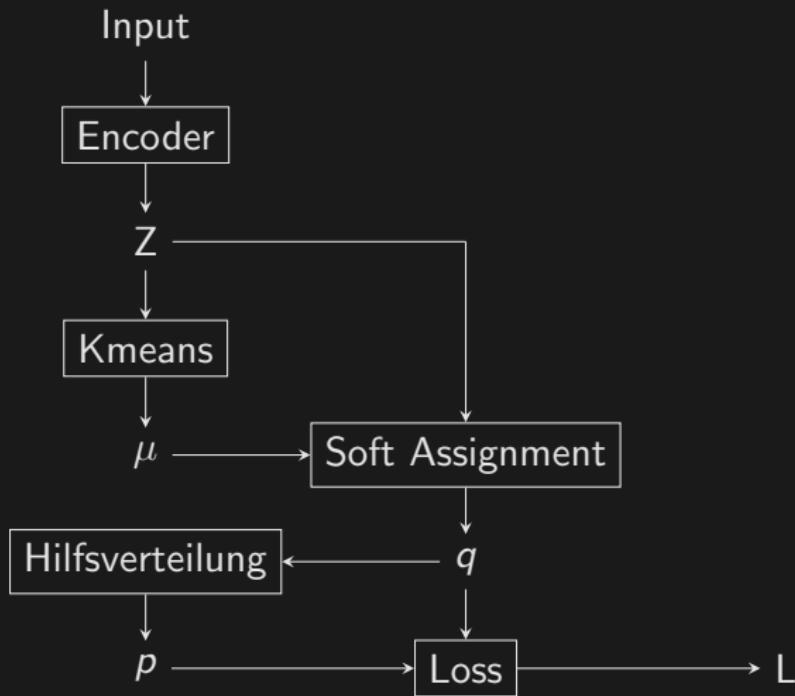
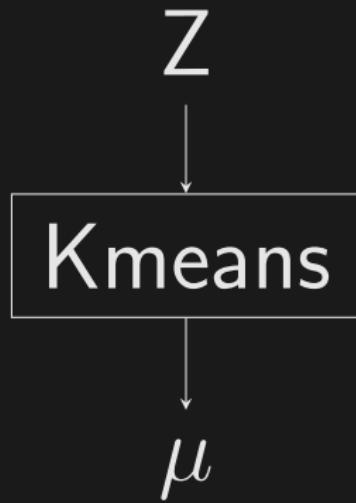
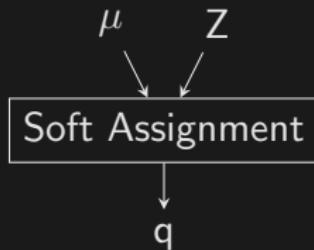


Figure: Abstrakter Aufbau des Clustering Layers

KMeans

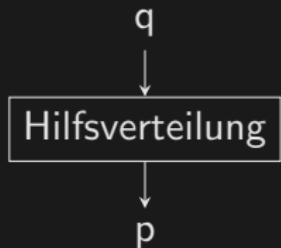


Soft Assignments



$$q_{ij} = \frac{(1 + \|z_i - \mu_j\|^2 / \alpha)^{-\frac{\alpha+1}{2}}}{\sum_{j'} (1 + \|z_i - \mu'_{j'}\|^2 / \alpha)^{-\frac{\alpha+1}{2}}}$$

Hilfsverteilung



$$p_{ij} = \frac{q_{ij}^2 / f_j}{\sum_{j'} q_{ij'}^2 / f_{j'}}$$

$$f_j = \sum_j q_{ij}$$

Loss

Kullback-Leibler-Divergenz



$$L = KL(P||Q) \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

Stochastic Gradient Descent

$$\frac{\partial L}{\partial z_i} = \frac{\alpha+1}{\alpha} \sum_j \left(1 + \frac{\|z_i - \mu_j\|^2}{\alpha}\right)^{-1} \times (p_{ij} - q_{ij})(z_i - \mu_j)$$

$$\frac{\partial L}{\partial \mu_j} = -\frac{\alpha+1}{\alpha} \sum_i \left(1 + \frac{\|z_i - \mu_j\|^2}{\alpha}\right)^{-1} \times (p_{ij} - q_{ij})(z_i - \mu_j)$$

MNIST

STL-10

REUTERS

Bewertungsmetrik

$$ACC = \max_m \frac{\sum_{i=1}^n 1\{l_i = m(c_i)\}}{n}$$

Ergebnisse

Methode	MNIST	STL-HOG	REUTERS-10K	REUTERS
k-means	53.49%	28.39%	52.42%	53.29%
LDMGI	84.09%	33.08%	43.84%	N/A
SEC	80.37%	30.75%	60.08%	N/A
DEC w/o	79.82%	34.06%	70.05%	69.62%
DEC	84.30%	35.90%	72.17%	75.63%

Clusters



Figure: MNIST

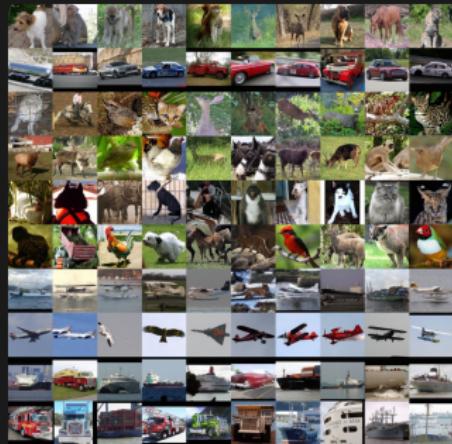


Figure: STL-10

Autoencoder Features

Methode	MNIST	STL-HOG	REUTERS-10K	REUTERS
AE+kmeans	81.84%	33.92%	66.59%	71.97%
AE+LDMGI	83.98%	32.04%	42.92%	N/A
AE+SEC	81.56%	32.29%	61.86%	N/A
DEC	84.30%	35.90%	72.17%	75.63%

Gradient Visualisierung

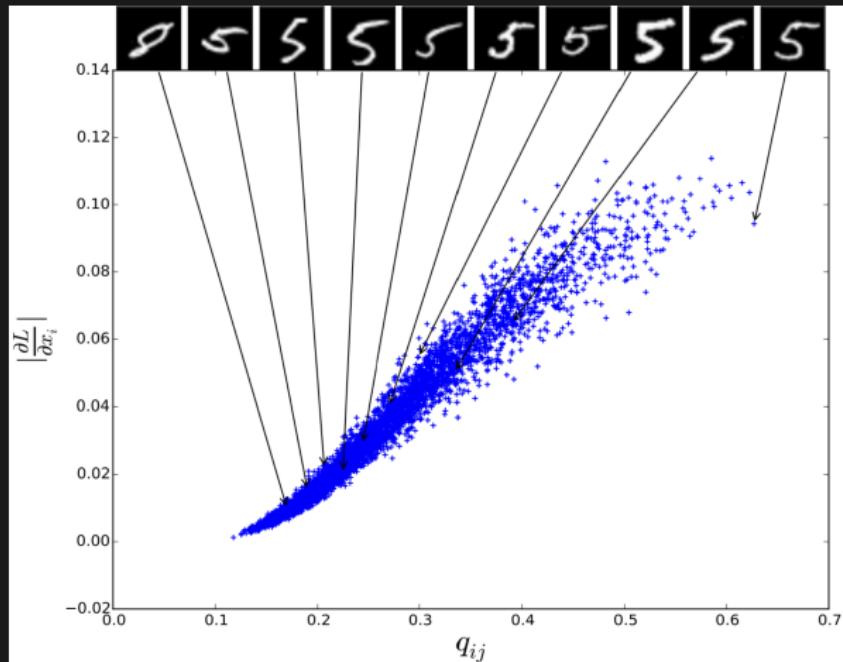
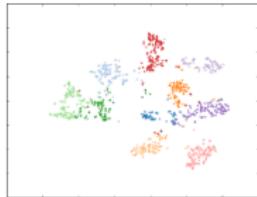
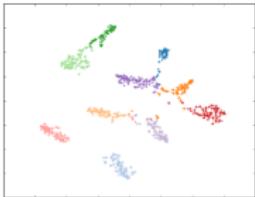


Figure: Cluster Soft Assignment vs Loss L

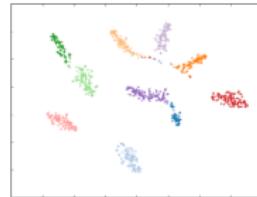
Cluster Überzeit



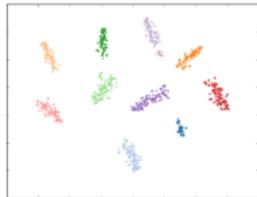
(a) Epoch 0



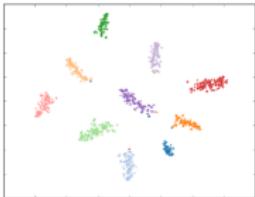
(b) Epoch 3



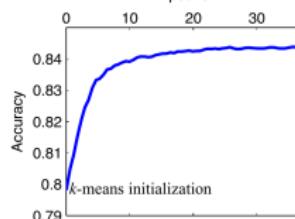
(c) Epoch 6



(d) Epoch 9



(e) Epoch 12



(f) Accuracy vs. epochs

Gut
aber
Aufwendig

Von wem ist das Paper

Junyuan Xie
University of Washington

Ross Girshick
Facebook AI Research (FAIR)

Ali Farhadi
University of Washington

Referenzen

-  Steinbach, Michael, Ertöz, Levent, and Kumar, Vipin. The challenges of clustering high dimensional data. In *New Directions in Statistical Physics*, pp. 273–309. Springer, 2004.
-  Ye, Jieping, Zhao, Zheng, and Wu, Mingrui. Discriminative k-means for clustering. In *NIPS*, 2008.
-  van der Maaten, Laurens. Learning a parametric embedding by preserving local structure. In *International Conference on Artificial Intelligence and Statistics*, 2009.