Dimensionality Reduction

David Mareček

December 17, 2024

EUROPEAN UNION
European Structural and Investment Fund
Operational Programme Research,
Development and Education

Charles University
Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics

Word vectors

For example, in the modern NLP tools, we work with the token vectors:

the 0.06706389 -0.02177303 0.01558930 0.02813231 0.03983797 0.04102217 -0.01329948 0.06286515 ...
" -0.04291666 0.02249542 -0.05009500 -0.06409580 -0.02206697 -0.00230843 0.02658171 0.04260673 ...
and 0.01757652 0.04324145 0.04058476 -0.03789425 0.06251122 0.05438545 -0.02366439 -0.05841338 ...
to -0.00532257 -0.05932277 0.01951243 -0.07709766 0.06911869 -0.02562860 0.08434074 0.04198023 ...
of 0.07348609 -0.08701739 0.01530370 0.01435481 0.07839862 -0.00587812 0.04881313 0.00704123 ...
a 0.04633269 0.02065392 -0.00007563 0.05094988 -0.01087748 0.09417078 0.01552414 0.06899974 ...
he -0.04827927 -0.04000403 -0.01654946 -0.02061500 -0.02676081 -0.04896818 0.03844265 0.05645940 ...
I 0.01558954 -0.01225288 -0.00119785 -0.02509643 0.02247903 0.01052496 0.03110695 -0.00235095 ...
was 0.01874722 0.05523073 -0.01489090 0.02162263 -0.01896353 -0.01322574 0.06215377 -0.01381461 ...
in -0.09954044 -0.05709903 0.11018854 -0.04675781 -0.01999592 -0.02787013 0.10401208 -0.01038842 ...
it -0.01069798 -0.01771499 -0.06531385 -0.00164481 -0.03059055 -0.00858863 0.07427775 -0.00638900 ...
that 0.10402621 0.07182080 0.06131404 0.00397065 -0.09825946 0.06330189 0.02241343 0.07013817 ...
you -0.07310216 0.08821464 -0.06123695 0.02440572 0.05764723 -0.00493006 0.02435281 0.12307153 ...
his -0.02382327 -0.04097176 -0.05015393 -0.06228361 -0.00908141 -0.06637910 0.01996890 0.08685388 ...
' -0.05308782 0.03133746 -0.04824331 -0.01246430 -0.06197543 0.02828505 -0.02937937 0.02694001 ...
had -0.00874879 0.04951908 0.03042142 0.07764163 -0.08997355 0.01246094 0.05392662 -0.09660292 ...
? 0.06881841 -0.01309363 0.04830608 -0.00015038 -0.07948416 0.01428877 0.09173763 -0.10053114 ...

...

Word vectors - motivation

- How are the words organized in the vector space?
- Are the synonyms close to each other?
- Are there subspaces representing some properties of the words?

Dimensionality Reduction

Generally, we often have:

- Big and high-dimensional data
- A lot of features
- Many of them may be redundant / correlated / linearly dependent

Dimensionality reduction algorithms map high-dimensional data to a lower dimension while preserving structure.

Motivation:

- Visualization
- More efficient use of resources (e.g., time, memory, communication)
- Statistical: fewer dimensions $\rightarrow$ better generalization (curse of dimenzionality)
- Noise removal (improving data quality)

Dimensionality Reduction

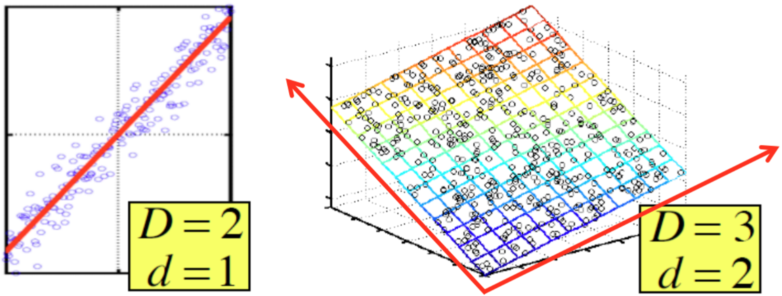
Feature selection:

- Select a subset of features.
- If a feature is almost irrelevant, we can omit it.

Dimensionality Reduction

Feature extraction:

- more general
- not limited to the original features
- Assumption: data (approximately) lies on a lower dimensional space



t-SNE

t-SNE

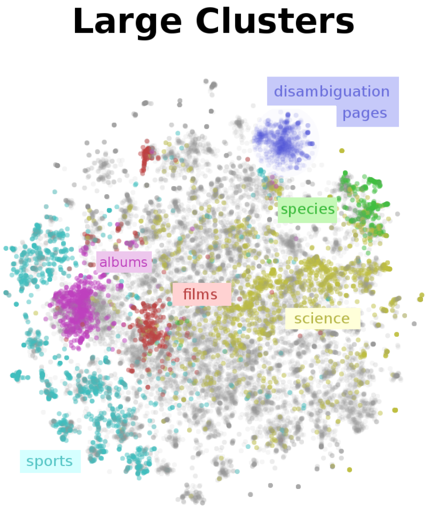
t-distributed Stochastic Neighbor Embedding
developed by Laurens van der Maaten and Geoffrey Hinton in 2008

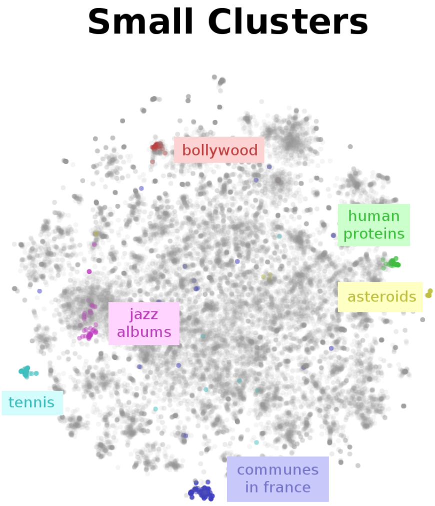
- a non-linear dimensionality reduction technique
- for visualization of high dimensional data in 2D (3D)
- it keeps very similar data points close together in lower-dimensional space
- it preserves the local structure of the data, not the global structure
- it preserves well-separated clusters

In this part, I am using illustrations by Kemal Erdem.

See `https://towardsdatascience.com/t-sne-clearly-explained-d84c537f53a`

t-SNE on Wikipedia articles



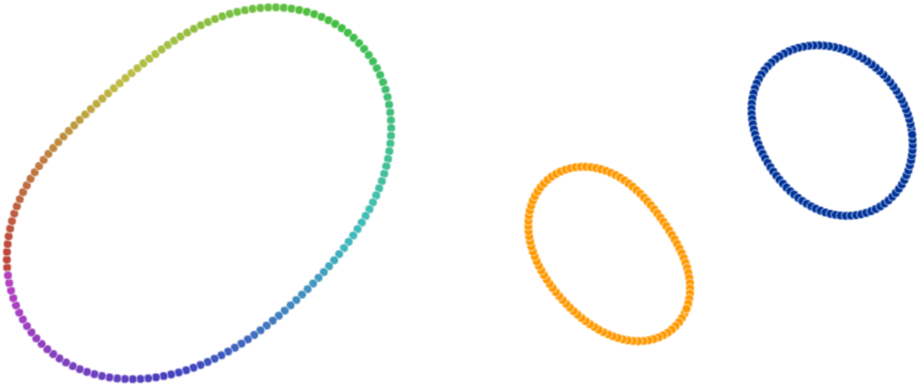


How you would preserve the local structure in 2D?

Original datasets in 3D

How you would preserve the local structure in 2D?

Their t-SNE visualization in 2D



Similarity of two points

Create a probability distribution that represents similarities between neighbors

For each pair of data points (i, j), compute

$$p_{j|i} = \frac{exp(-||x_i - x_j||^2/2\sigma_i^2)}{\sum_{k \neq i} exp(-||x_i - x_k||^2/2\sigma_i^2)},$$

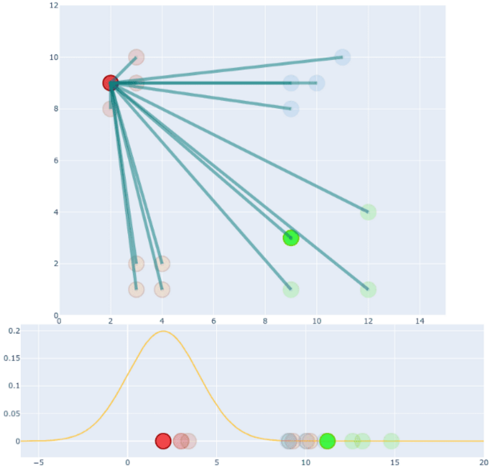
The denser the part of the data spaces at x_i is, the smaller the σ_i (Gaussian kernel bandwidth).

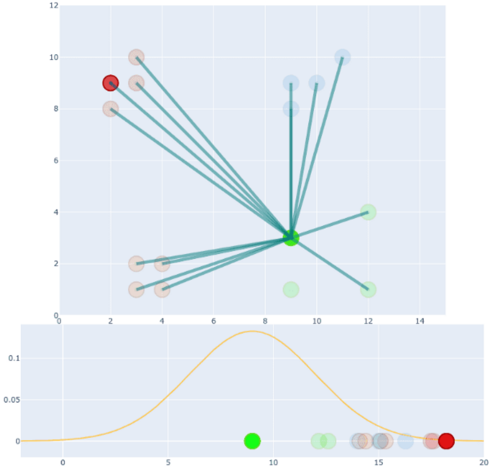
The similarity of the data point x_j to the data point x_i is the conditional probability $p_{j|i}$, that x_i would choose x_j as its neighbor.

The two asymetric distributions are then joined into a symetric one:

$$p_{ij} = \frac{p_{i|j} + p_{j|i}}{2N}$$

Similarity of two points





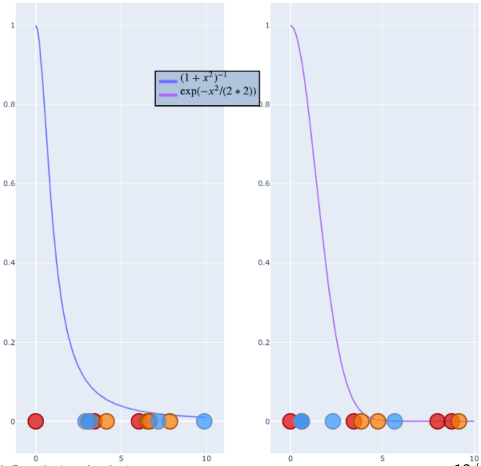
Similarity of two points in the low-dimensional space

As similarity measure in the target low-dimensional space, we will use Student t-distribution instead of the Gaussian

$$q_{ij} = \frac{(1 + ||y_i - y_j||^2)^{-1}}{\sum_{k \neq l} (1 + ||y_k - y_l||^2)^{-1}}$$

Student t-distribution "falls" more quickly and has longer tail than the Gaussian distribution

Therefore, we will not get similar points squashed into a single point.



Gradient descent

t-SNE starts with all points y_i randomly distributed in the target 2D (or 3D) space.

It uses Gradient descent optimization using the Kullback-Leibler divergence between p_{ij} and q_{ij} as a cost function.

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

In each step, a gradient is calculated for each point and describes how “strongly” it should be pulled and what direction it should choose.

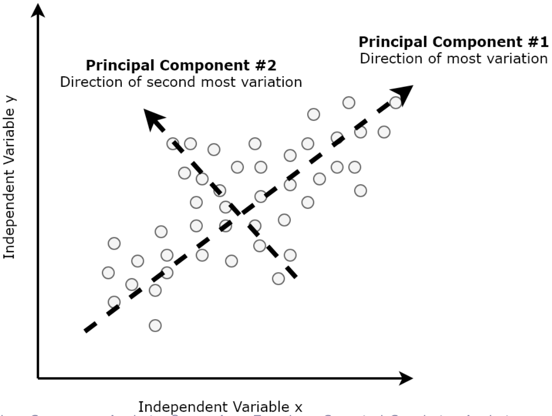
Demo: `projector.tensorflow.org`

Principal Component Analysis

Principal Component Analysis

Principal components (PC) are orthogonal directions that capture most of the variance in the data.

- 1st PC – direction of the greatest variability in data
- 2nd PC – next orthogonal (uncorrelated) direction of greatest variability



Principal Component Analysis

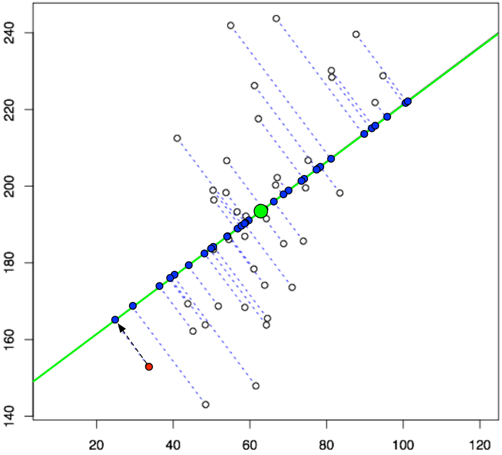
Given the centered data $[x_1, x_2, \ldots, x_n]$, the first principal vector is:

$$w_1 = \arg\max_w \frac{1}{m}\sum_{i=1}^{m}(w^T x_i)^2 = \arg\max_w w^T X X^T w, \quad w^T w = 1$$

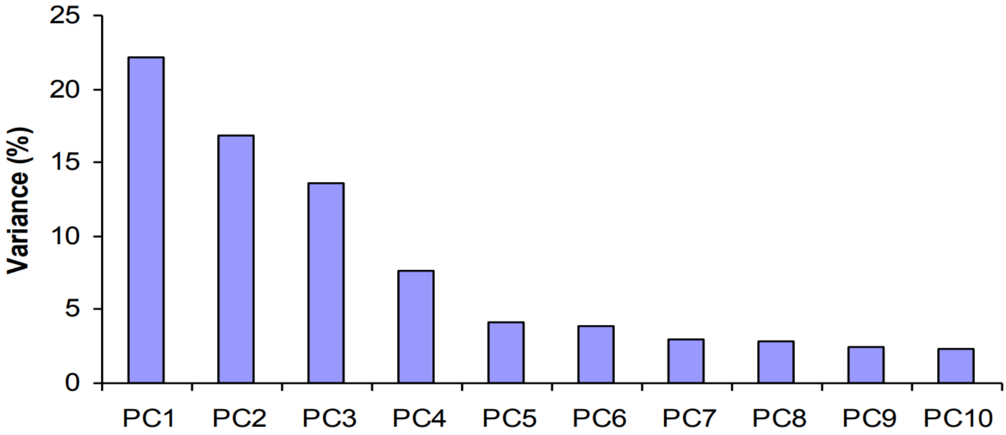
We maximize the variance of projection of x to w.
$\rightarrow$ we maximize the covariance between x and w (the data set is centered)

To calculate the k-th principal vector, we first remove all variability from the previous $k-1$ PC directions and find the next direction of the greatest variability.

Principal Component Analysis



Principal Component Analysis



Principal Component Analysis

1. Standardize the original high-dimensional dataset.
2. Take the standardized data and compute a covariance matrix A that provides a means to measure how all our features relate to each other.

$$A_{xy} = cov(x, y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})$$

3. Find its eigenvectors w and corresponding eigenvalues λ. Eigenvectors represent the principal components and provide a means to understand the direction of the data. The corresponding eigenvalues represent the amount of variance in the data in that direction.

$$Aw = \lambda w$$

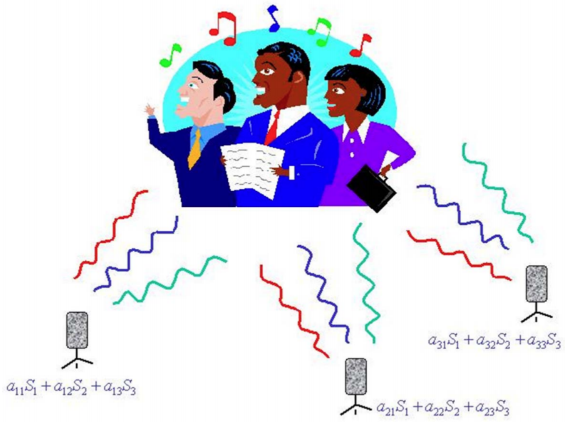
Principal Component Analysis

4. The eigenvectors are then sorted in descending order based on their corresponding eigenvalues, after which the top k eigenvectors are selected representing the most important representations found in the data.
5. A new matrix is then constructed with these k eigenvectors, thereby reducing the original n-dimensional dataset into reduced k dimensions.

Independent Component Analysis

Independent Component Analysis

- The classical "cocktail party" problem
- Separate the mixed signal into sources
- Assumption: different sources are independent



Independent Component Analysis

ICA: find the directions in the vector space so that the data projected onto these directions have maximum statistical independence

How to actually maximize independence?

- Minimize the mutual information
- Maximize the non-Gaussianity

$$x_i = a_{i1}s_1 + a_{i2}s_2 + \cdots + a_{in}s_n, \forall i = 1, \dots, n$$

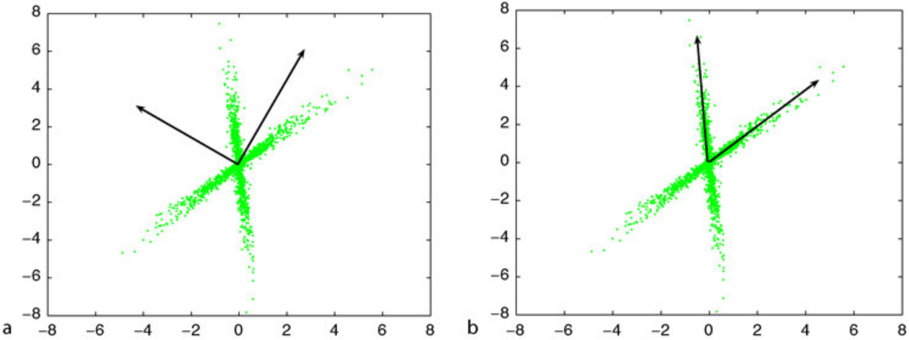
- x_i is the point we observe
- $[s_1, s_2, \dots, s_i]$ are the independent components
- a_{ij} is the associated linear combination

PCA versus ICA

Both PCA and ICA reduce dimensions.

Differences:

- PCA are ordered from the strongest one to the weakest, ICA components have all the same importance
- PCA vectors are orthogonal, ICA vectors are not orthogonal

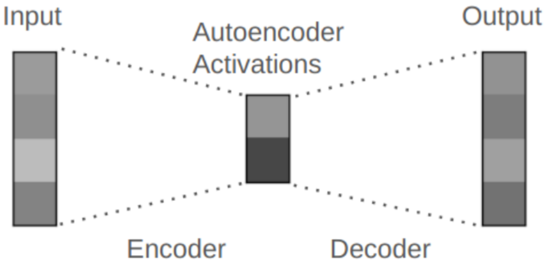


Sparse Auto-Encoders

Auto-Encoders

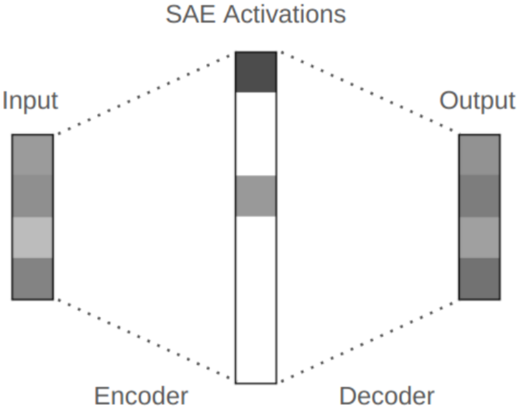
A regular autoencoder is a neural network designed to compress and then reconstruct its input data.

The reconstruction is typically imperfect because the compression makes the task challenging.

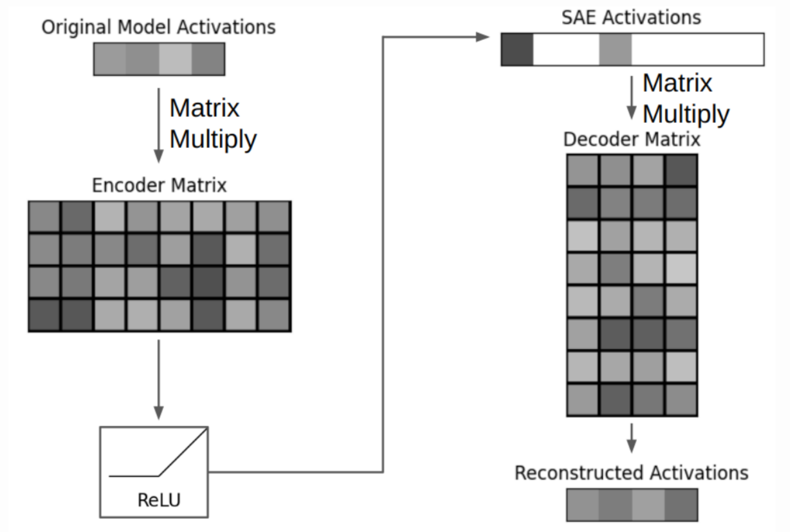


Sparse Auto-Encoders

- the intermediate vector's dimension is typically larger than the input's
- without additional constraints the task would be trivial (perfect reconstruction)
- additional constraint: sparsity (e.g. we want only 10% non-zero elements)



Sparse Auto-Encoders



Sparse Auto-Encoders - Loss Function

The loss function L is the combination of an L2 penalty on the reconstruction loss and an L1 penalty on feature activations.

$$L = || X - \hat{X}||_2^2 + \lambda \sum_i f_i(X) \cdot ||W_{\cdot,i}^{dec}||_2$$

The multiplication $f_i(X) \cdot ||W_{\cdot,i}^{dec}||_2$ prevents the SAE from "cheating" the L1 penalty by making $f_i(X)$ small and $||W_{\cdot,i}^{dec}||_2$ large in a way that leaves the reconstructed activations unchanged.

Examples on big LLM models

https://transformer-circuits.pub/2024/scaling-monosemanticity/

Canonical Correlation Analysis

Canonical Correlation Analysis

Now consider two sets of variables x and y

- x is a vector of p variables
- y is a vector of q variables
- Basically, two feature spaces

Example: consider variables related to exercise and health
X: climbing rate on a stair stepper, how fast you can run a certain distance, the amount of weight lifted on bench press, the number of push-ups per minute, ...
Y: blood pressure, cholesterol levels, glucose levels, body mass index, ...
How to find the connection between two set of variables (or two feature spaces)?

Canonical Correlation Analysis

- CCA: find a projection direction u in the space of x, and a projection direction v in the space of y, so that projected data onto u and v has max correlation
- Note: CCA simultaneously finds dimension reduction for two feature spaces

Example: We can find that a certain linear combination of bench press and running time very well correlates with a certain linear combination of blood presure and body mass index.

Canonical Correlation Analysis

CCA formulation:

$$\arg\max_{u,v} \frac{u^T X^T Y v}{\sqrt{(u^T X^T X u)(v^T Y^T Y v)}},$$

- X is n by p: n samples in p-dimensional space
- Y is n by q: n samples in q-dimensional space
- The n samples are paired in X and Y

How to solve? ... kind of complicated ...