Short overview of Dimensionality reduction techniques

Dmitrii Maksimov

15.07.2019, Berlin

The main goal of dimensionality reduction:

Discovering of the hidden relationships in the data that are not obvious in original feature space.

The main goal of this talk:

To show that nowadays it is not difficult to apply the methods to your own research.

Dimensionality reduction

Feature selection

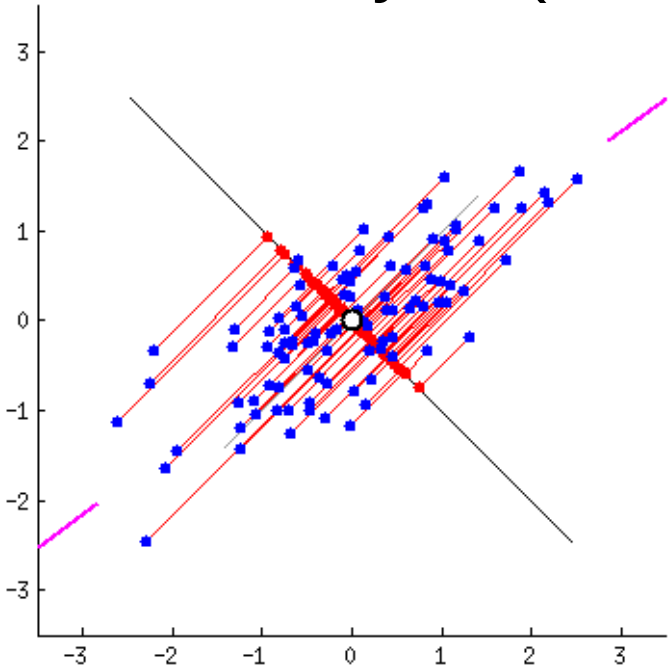
Searches for a relevant subset of existing variables.
(-) has combinatorial optimization problem
(+) features are easy to interpret

Feature extraction

Learns a new set of features
(-) features difficult to interpret
(+) unique solutions in polynomial time


SOME PICTURES ARE WORTH...
38
TEN THOUSAND WORDS
imgflip.com

Principal component analysis (PCA)



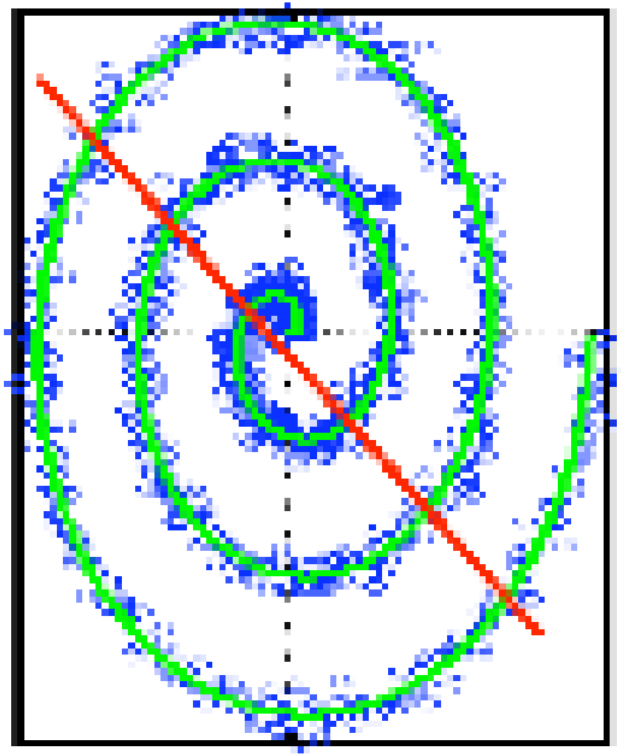
The most informative direction is where the data is most spread out

To play around: http://setosa.io/ev/principal-component-analysis/

Further reading how it is implemented: With Step-by-step explanation

https://github.com/llSourcell/Dimensionality_Reduction/blob/master/principal_component_analysis.ipynb

Manifolds



- PCA would not find the "correct" 1D manifold (green) because a) PCA is constrained to a linear mapping and b) PCA tries to preserve global features.
- Often, preserving local features, like neighborhoods, is more important than global properties.

Credit: https://events.uta.fi/mlvis2018/wp-content/uploads/sites/23/2018/06/mlvis18_dimred_jaakkopeltonen_ver2.pdf

Multidimensional scaling

The primary outcome of an MDS analysis is a spatial configuration, in which the points are arranged in such a way, that their distances correspond to the similarities of the objects: similar object are represented by points that are close to each other, dissimilar objects by points that are far apart.

$$S^2 = \sum_{ij} [F[D(A_i, B_j)] - f[d(x_i, x_j)]]^2$$

$$F(r) = 1 - (1 + (2^{a/b} - 1)(r/\sigma)^a)^{-b/a}$$

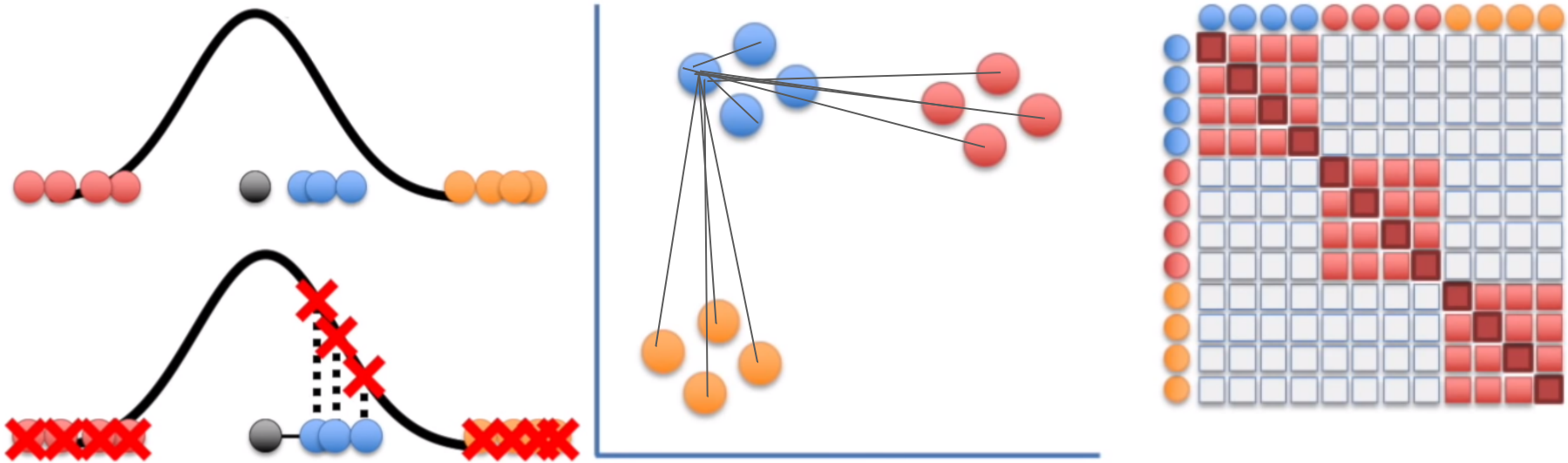
Wickelmaier, Florian. "An introduction to MDS." *Sound Quality Research Unit, Aalborg University, Denmark* (2003): 46

Credit : https://baoilleach.blogspot.com/2014/01/convert-distance-matrix-to-2d.html

Only distances (dissimilarities) between points are available

		1	2	3	4	5	6	7	8	9
		BOST	NY	DC	MIAM	CHIC	SEAT	SF	LA	DENV
		----	----	----	----	----	----	----	----	----
1	BOSTON	0	206	429	1504	963	2976	3095	2979	1949
2	NY	206	0	233	1308	802	2815	2934	2786	1771
3	DC	429	233	0	1075	671	2684	2799	2631	1616
4	MIAMI	1504	1308	1075	0	1329	3273	3053	2687	2037
5	CHICAGO	963	802	671	1329	0	2013	2142	2054	996
6	SEATTLE	2976	2815	2684	3273	2013	0	808	1131	1307
7	SF	3095	2934	2799	3053	2142	808	0	379	1235
8	LA	2979	2786	2631	2687	2054	1131	379	0	1059
9	DENVER	1949	1771	1616	2037	996	1307	1235	1059	0

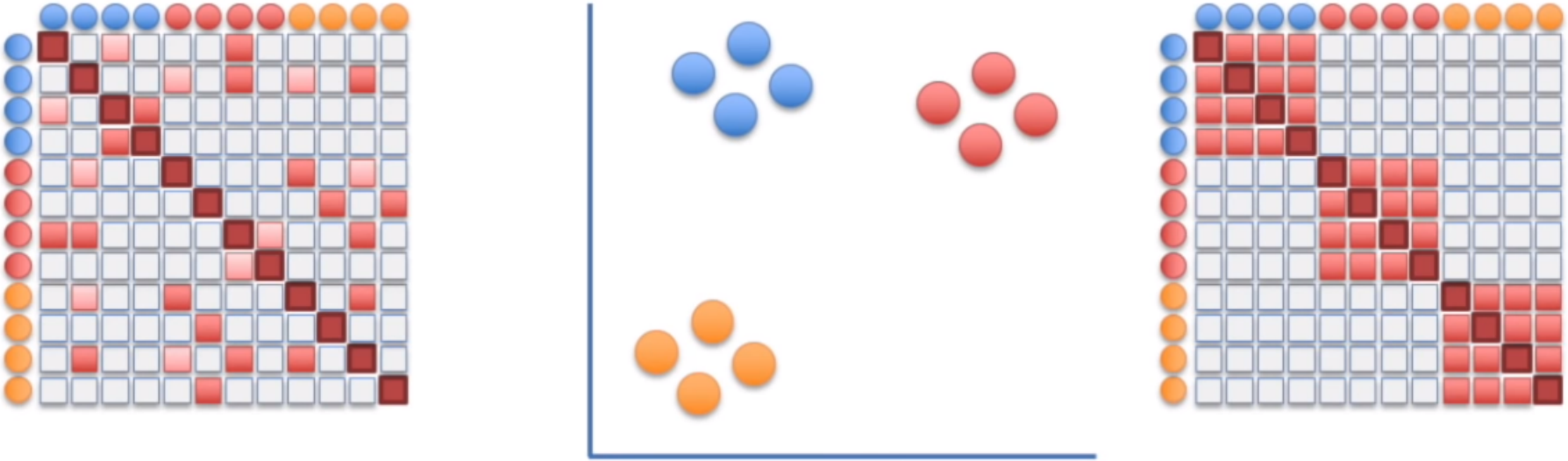
t-Distributed Stochastic Neighbour Embedding (t-SNE)



t-SNE moves the points a little bit at a time, and each step it chooses a direction that makes the matrix on the left more like the matrix on the right.

Credit: https://www.youtube.com/watch?v=NEaUSP4YerM

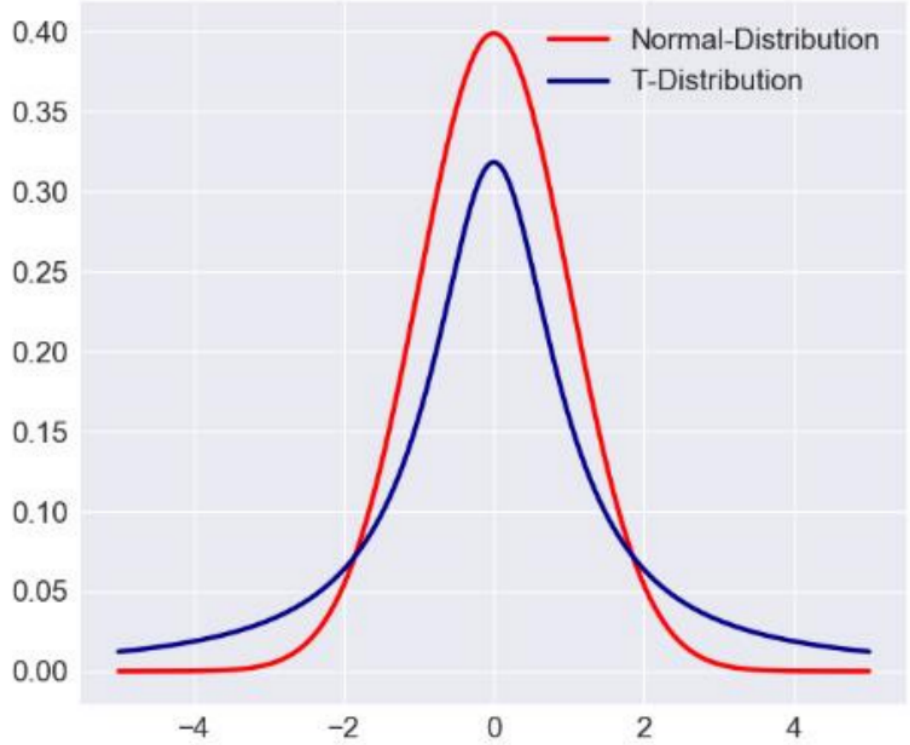
t-Distributed Stochastic Neighbour Embedding (t-SNE)



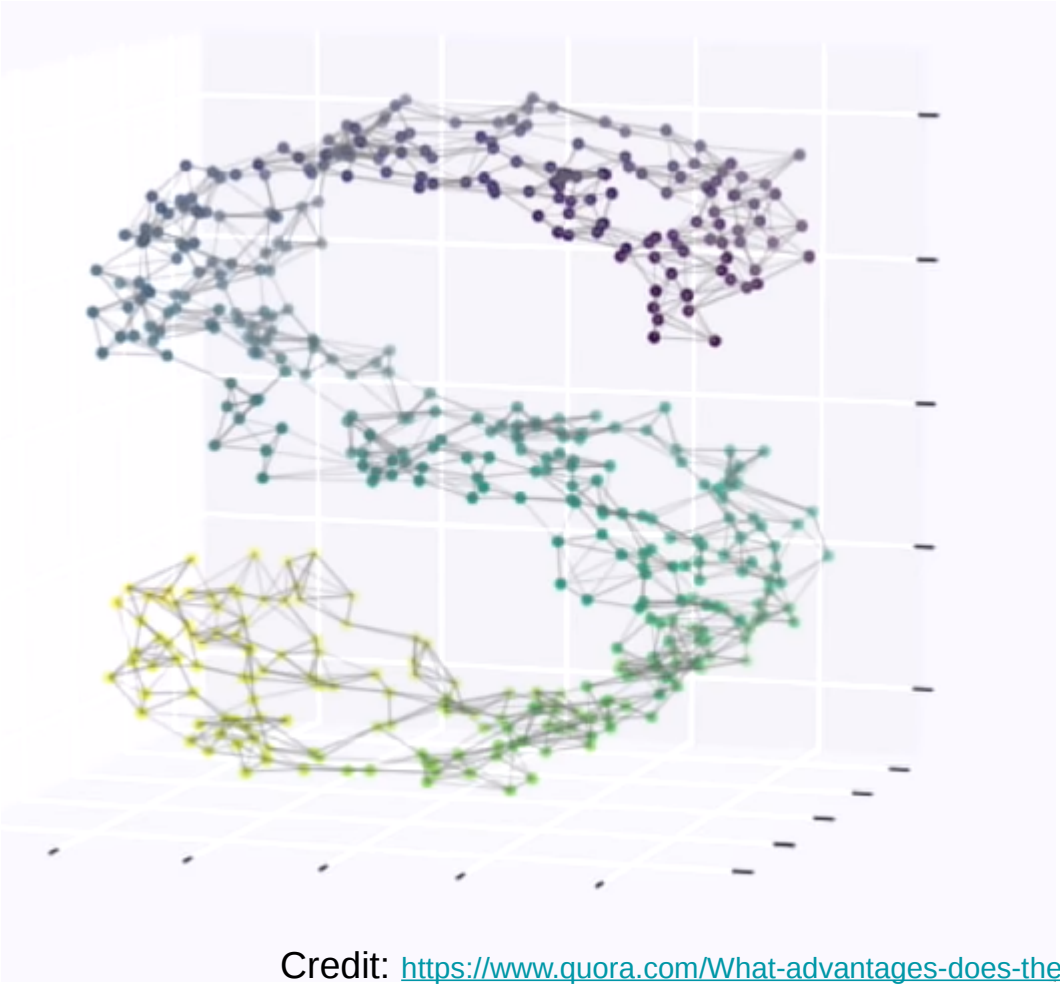
t-SNE moves the points a little bit at a time, and each step it chooses a direction that makes the matrix on the left more like the matrix on the right.

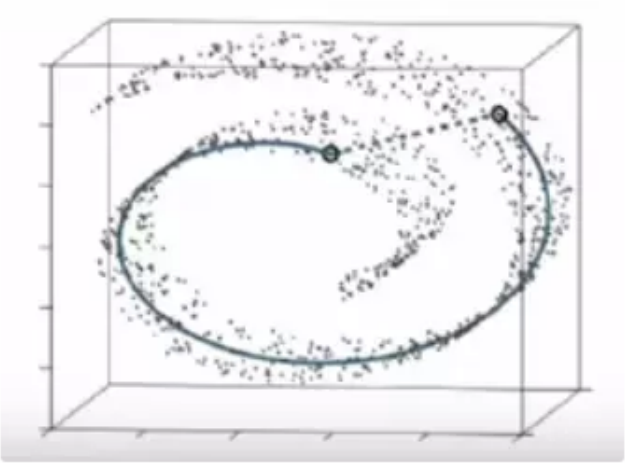
Original papers: https://lvdmaaten.github.io/tsne/

Why Student-t distribution?



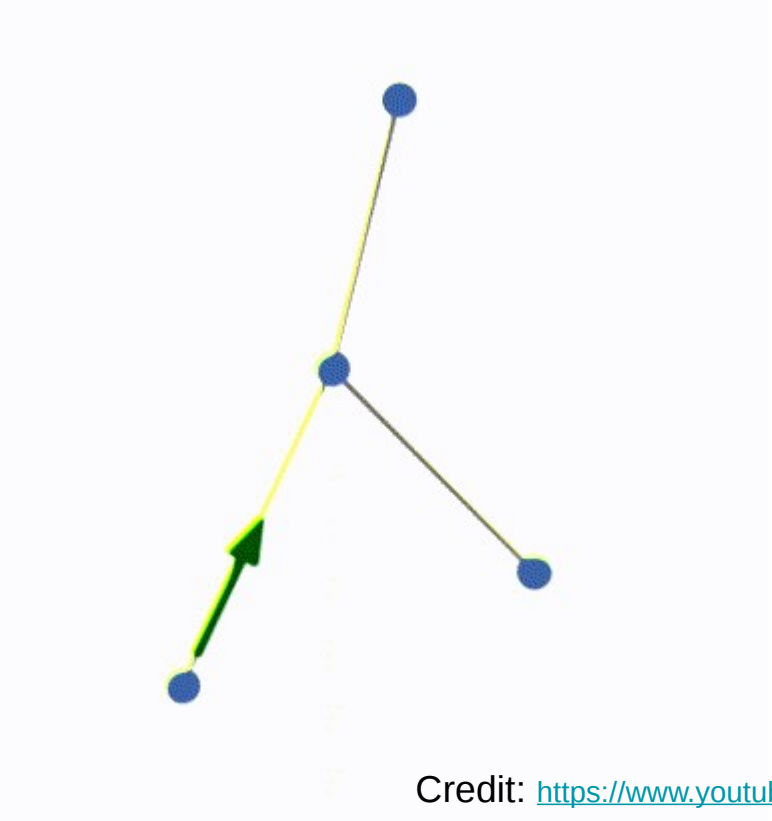
it uses a Student-t distribution rather than a Gaussian to compute the similarity between two points in the low-dimensional space, where heavy-tailed distribution in the low-dimensional space aimed to alleviate the crowding problem.

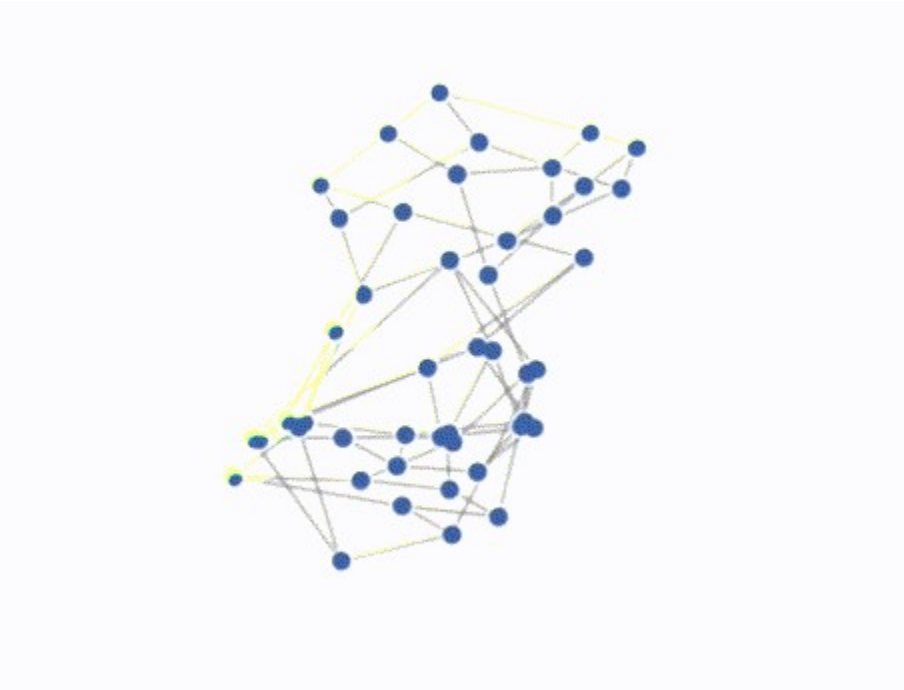




Credit: https://www.quora.com/What-advantages-does-the-t-SNE-algorithm-have-over-PCA

Force-directed graph layout





Credit: https://www.youtube.com/watch?v=9iol3Lk6kyU

Dimensionality Reduction toolbox in python

```python
import sklearn as sk
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import tensorflow as tf
from tensorflow.examples.tutorials.mnist import input_data
from sklearn.manifold import LocallyLinearEmbedding
from sklearn.decomposition import PCA
from sklearn.decomposition import IncrementalPCA
from sklearn.decomposition import KernelPCA
from sklearn.decomposition import SparsePCA
from sklearn.manifold import MDS
from sklearn.manifold import Isomap
from sklearn.manifold import TSNE
from sklearn.decomposition import TruncatedSVD
from sklearn.random_projection import GaussianRandomProjection
from sklearn.decomposition import FastICA
from sklearn.decomposition import MiniBatchDictionaryLearning
from sklearn.random_projection import SparseRandomProjection
import keras
from keras.models import Sequential, Model
from keras.layers import Dense
from keras.optimizers import Adam
```

Library.py hosted with ❤ by **GitHub** **view raw**

Credit: https://towardsdatascience.com/dimensionality-reduction-toolbox-in-python-9a18995927cd

Dimensionality reduction with t-SNE

Dataset: https://www.kaggle.com/oddrationale/mnist-in-csv

Good tutorials: https://github.com/oreillymedia/t-SNE-tutorial

https://github.com/gauss256/t-SNE/blob/master/t-SNE%20Explorations.ipynb

https://towardsdatascience.com/an-introduction-to-t-sne-with-python-example-5a3a293108d1

Perspectives

Periodic table from literature:

https://chemistrycommunity.nature.com/users/64392-john-dagdelen/posts/50785-unsupervised-word-embeddings-capture-latent-knowledge-from-materials-science-literature

Unsupervised word embeddings capture latent knowledge from materials science literature

https://www.nature.com/articles/s41586-019-1335-8.pdf

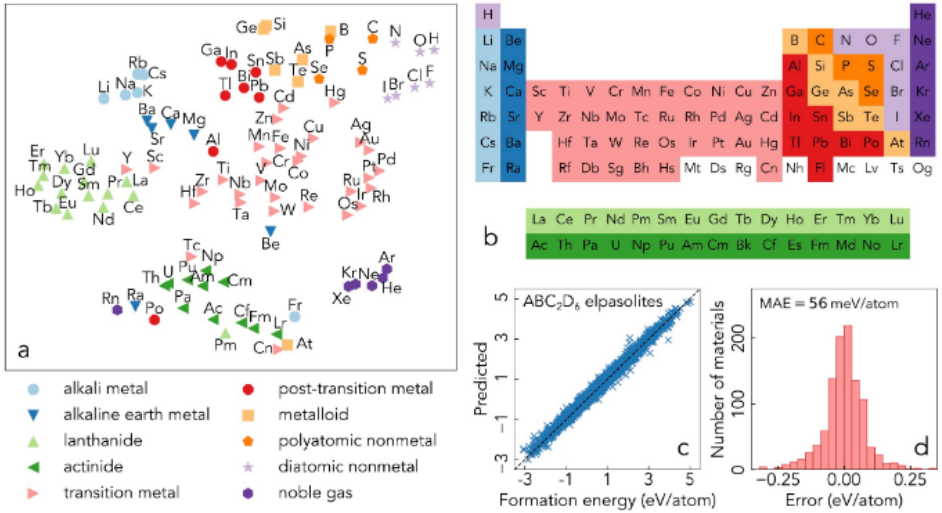


Figure 1: Chemistry is captured by word embeddings. a. *Two-dimensional t-SNE projection of the word embeddings of 100 chemical element names (e.g. "hydrogen", "helium") labelled with the corresponding element symbols and grouped according to their classification.* ***b.*** *The periodic table colored according to the classification shown in a.* ***c.*** *Predicted versus actual (density functional theory) values of formation energies of approximately 10,000* ABC_2D_6 *elpasolite compounds[1] using word embeddings of elements learned from text as features. The data points in the plot use 5-fold cross-validation.* ***d.*** *Error distribution for the 10% test set of elpasolite formation energies (MAE=56 meV/atom).*