DIMENSIONALITY REDUCTION

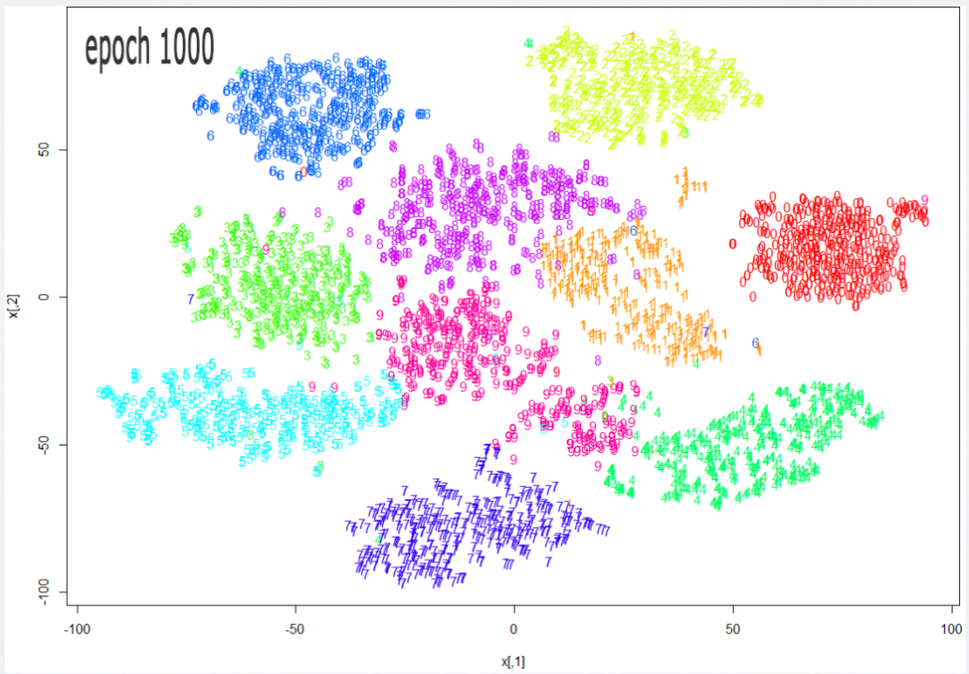
Low dimensionality embedding

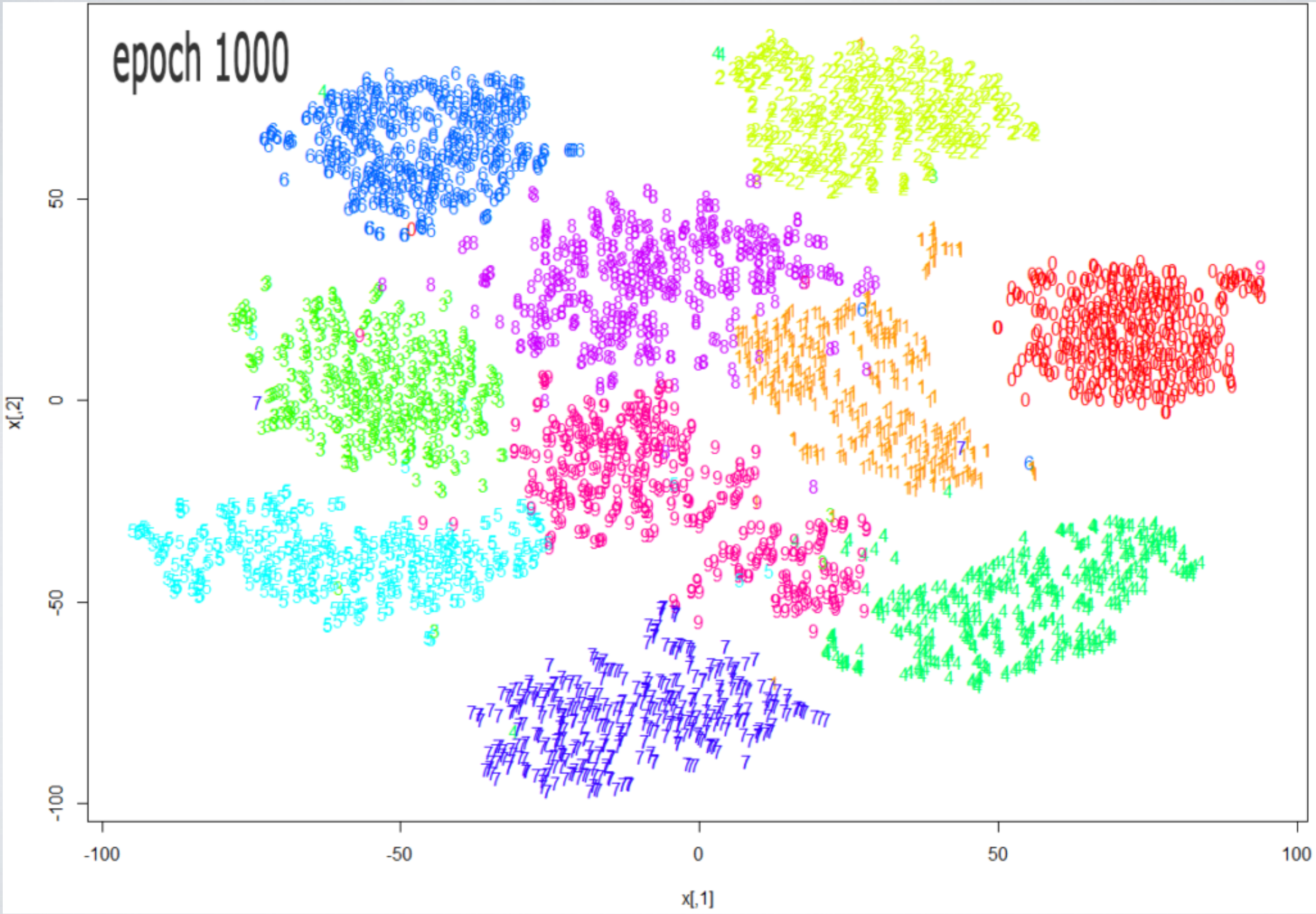
DIMENSIONALITY REDUCTION

- Data Mining objective: understand our data
  - We get a dataset composed of many features
    - Or worst, complex object (image, sound, graph…)
  - How to understand the organization of our data?
  - How to perform clustering?

VISUALIZATION

- Your data is perfectly fine, but you want to intuitively understand how it is organized
  - Are there groups of similar objects?
  - Are my clusters meaningful?
  - Is my classification/clustering on some types of elements and not others.




epoch 1000
x[,2]
50
0
-50
-100
-100
-50
0
50
100
x[,1]

CURSE OF DIMENSIONALITY

- Having hundreds/thousands of attributes is a problem for data analysis.
  - e.g.: medicine: blood analysis, genomics….
  - e.g.: cooking recipes: each column an ingredient…
- We want to reduce number of attributes while keeping most of the information
- Scalability

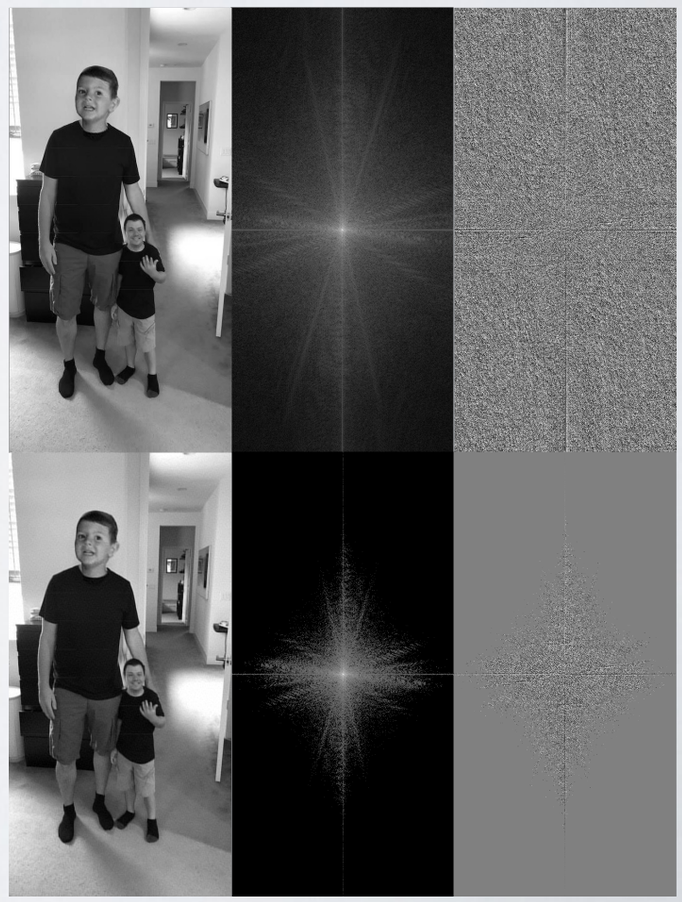
CORRELATION

- Assume that you have correlated features such as age, height and weight.
  - Linear regression will attribute the coefficients somewhat randomly between them
  - Decision tree will spend a lot of time choosing between them for no reason
- Dimensionality reduction can create a single variable to capture what is common
  - The rest can be lost or captured by another feature,
    - i.e., height - average height for that age, "residuals"

PCA

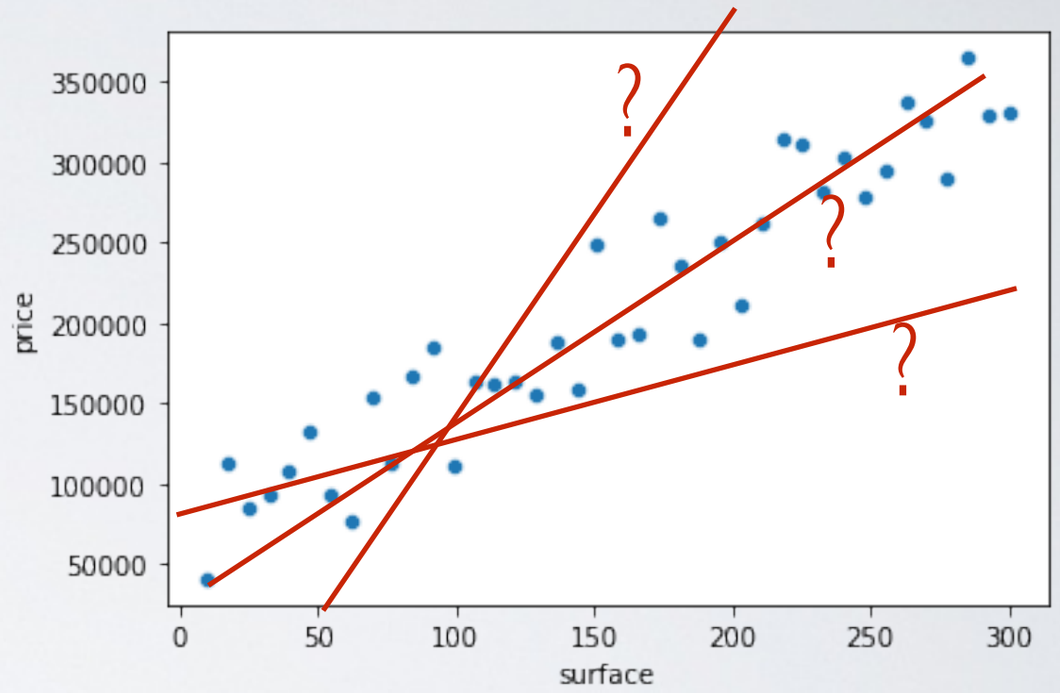
PCA

- PCA: Principal Component Analysis
- Defines new dimensions that are linear combinations of initial dimensions
  - Objective: concentrate the **variance** on some dimensions
    - So that we can keep only these ones.
    - Those we remove contain low variance, thus low information
- Similar principle than the Fourier transform technique for image compression

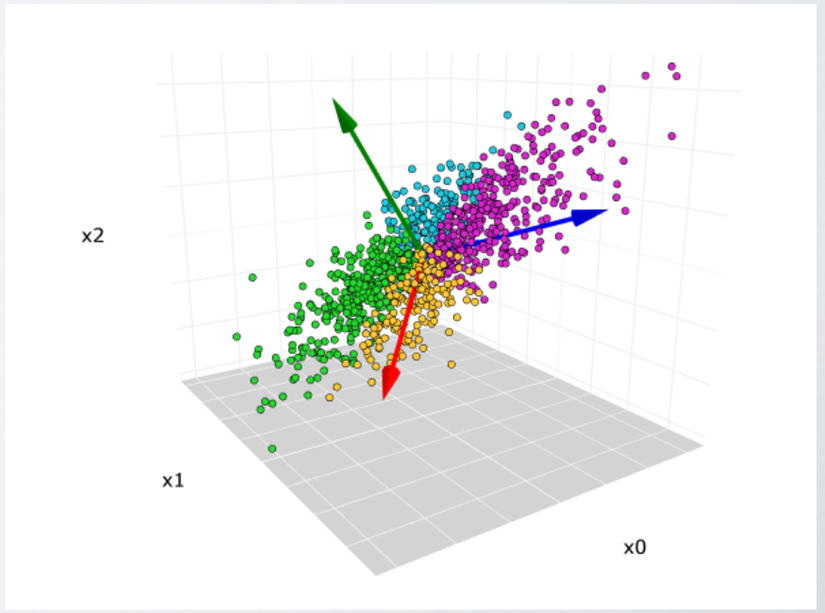


PCA

- Algorithm:
  - 1)Find an "axis", a unit vector defining a line in the space
    - That minimizes the variance=>the squared distance from all points to that line

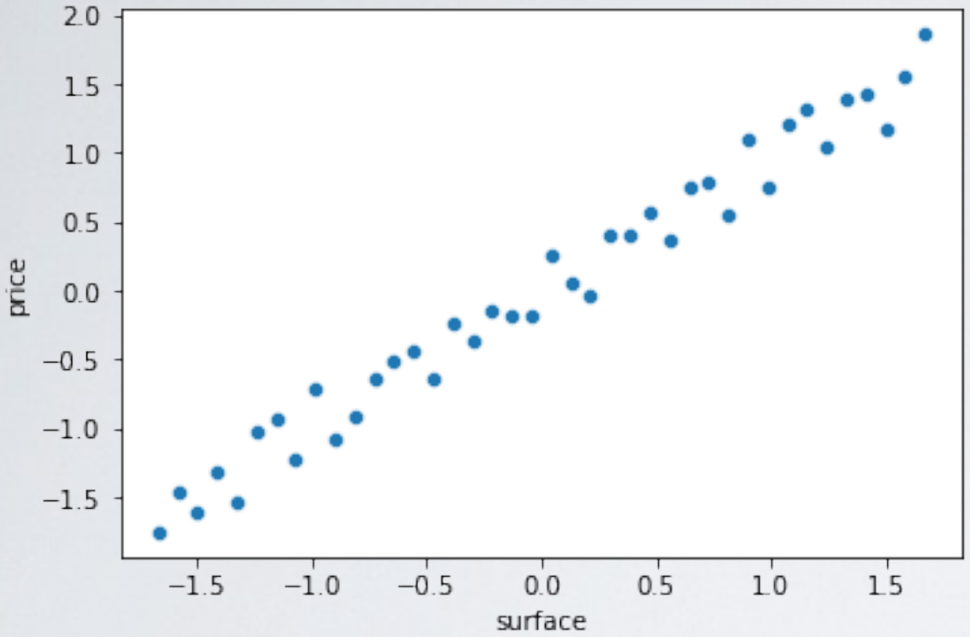


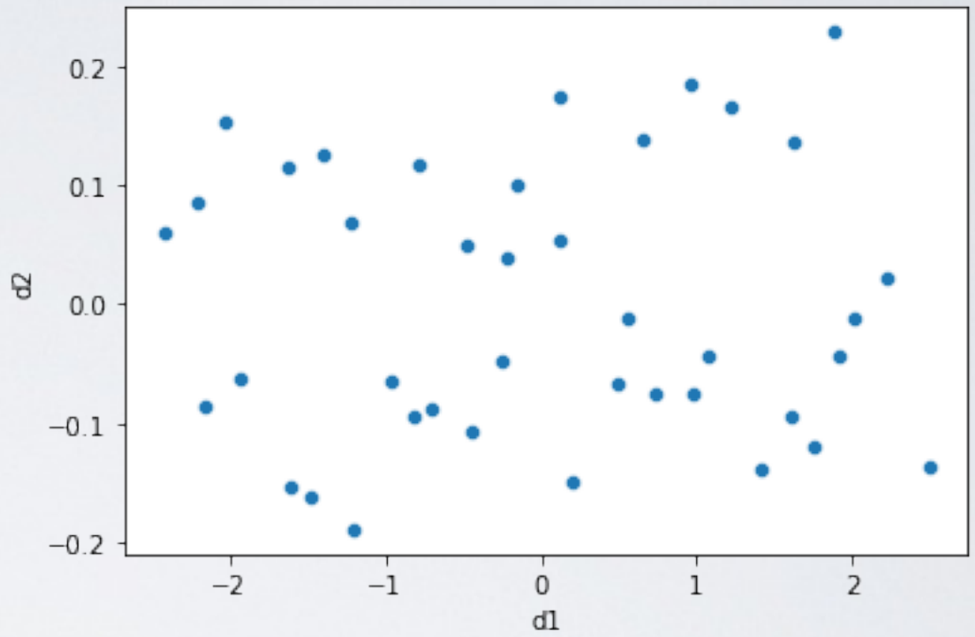
- 2)For d in (initial_d-1)
  - Find another axis, with two constraints:
    - Orthogonal to all previous axis
    - Among those, minimize the variance



- 3)At the end, keep the first k dimensions
  - Some information is lost

EXAMPLE PCA 2D





Covariance matrix (original)

```
[1.         0.98675899],
[0.98675899 1.        ]
```

Covariance matrix (pca)

```
[ 1.98675899e+00, 0],
[0,  1.32410092e-02]
```

Sum of variance

2

Sum of variance

2

Variance by dimension

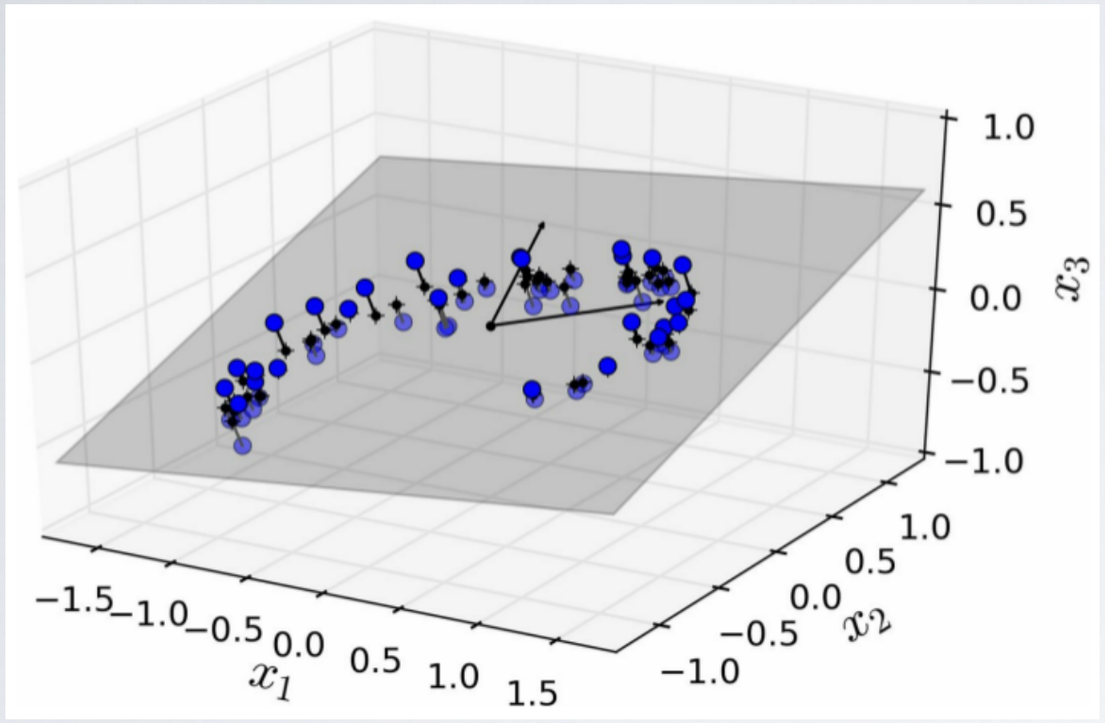
1 1

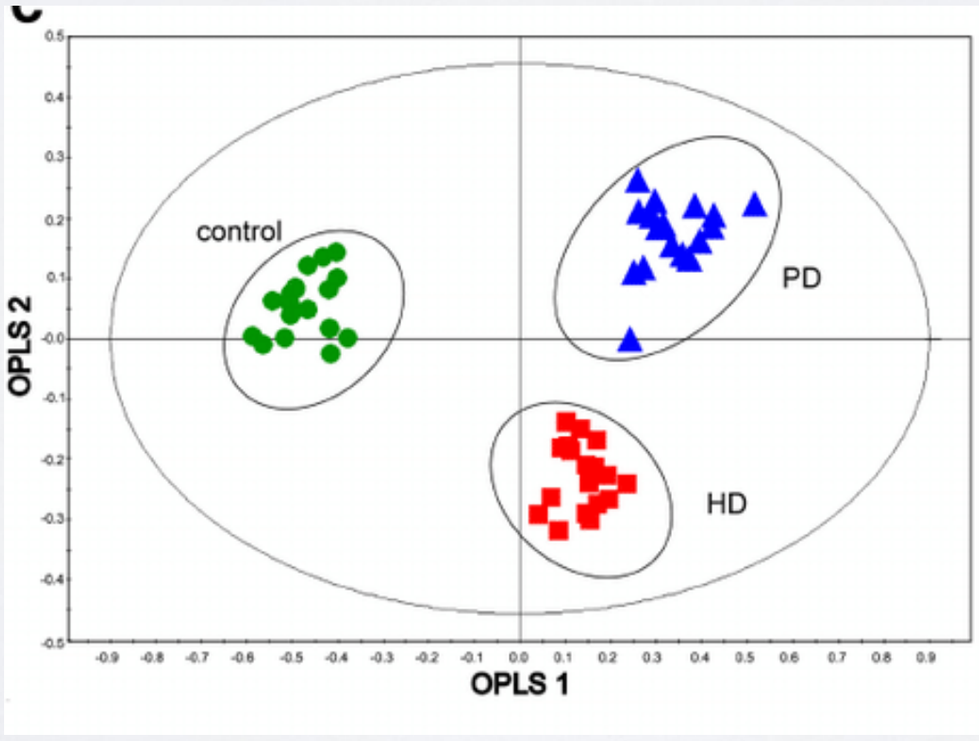
Variance by dimension

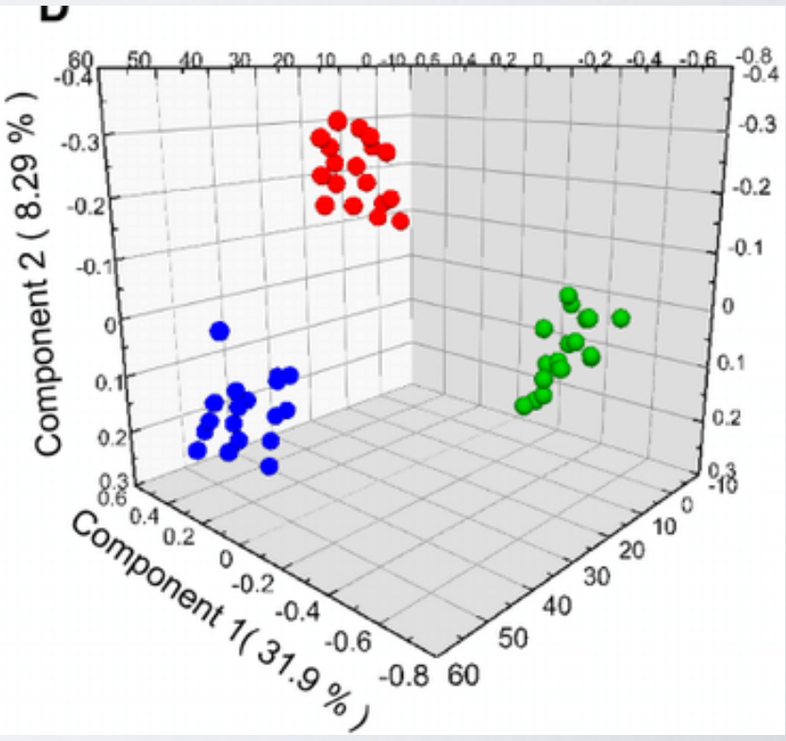
1.98675899 0.01324101

Explained variance(ratio) [0.9933795, 0.0066205]

3D=>2D


1.0
0.5
0.0
−0.5
−1.0
x_3
1.0
0.5
0.0
−0.5
−1.0
x_2
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
x_1

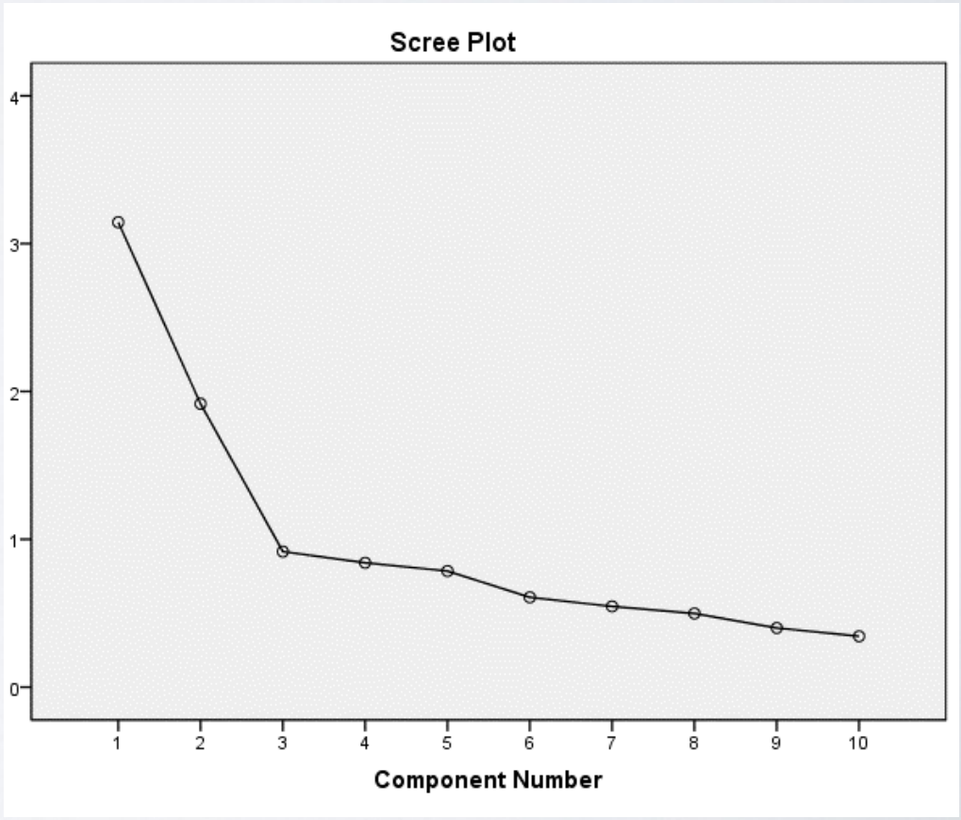

control
PD
HD
OPLS 2
OPLS 1


Component 2 (8.29 %)
Component 1(31.9 %)

CHOOSING COMPONENTS

- How to choose k?
  - Elbow method
  - OR fix beforehand a min threshold of explained variance, e.g.: 80%
    - We are fine with losing 20% of information

Explained variance



COMPUTATION IN PRACTICE

- Find the eigenvectors of the covariance matrix of centered data

- If you want k new dimensions, pick the k eigenvectors associated with the k largest eigenvalues
  - Eigenvalues = explained variance

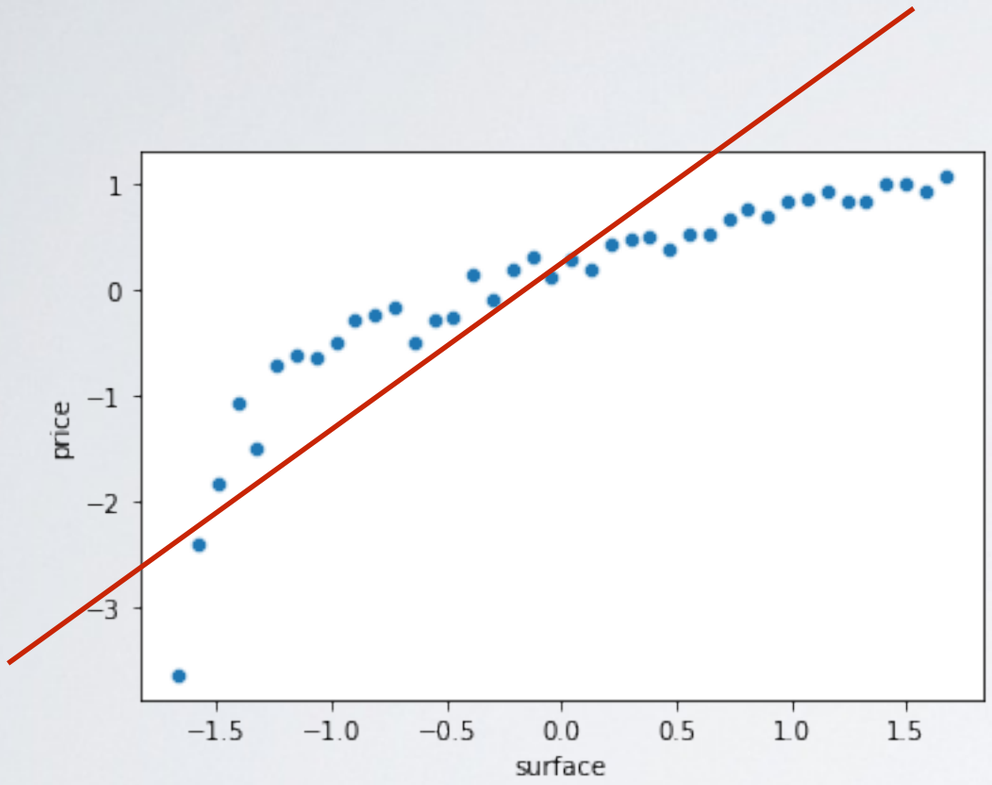
- The eigenvectors corresponding to the top eigenvalues are coefficients of the linear transformation

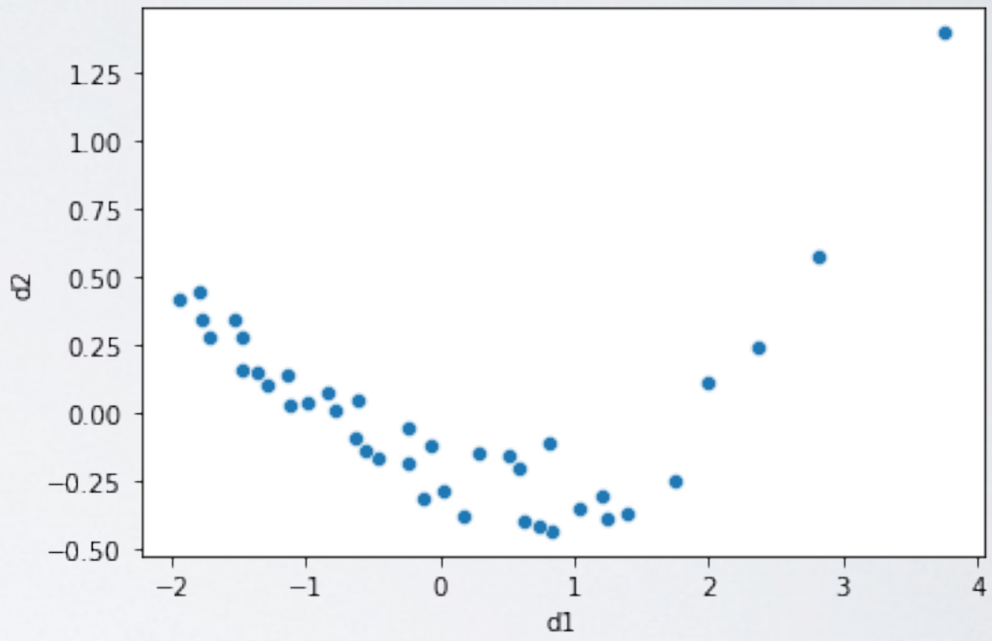
PCA POPULARITY

- Why is PCA popular?
- Similar reasons than linear regression:
  - Historically important
  - Analytical solutions
    - Guarantee to find the global minimum of the objective
    - Could be done before modern computers
  - Interpretable solution
  - Intuitively pleasant

- No reason to consider it "better" than other methods…

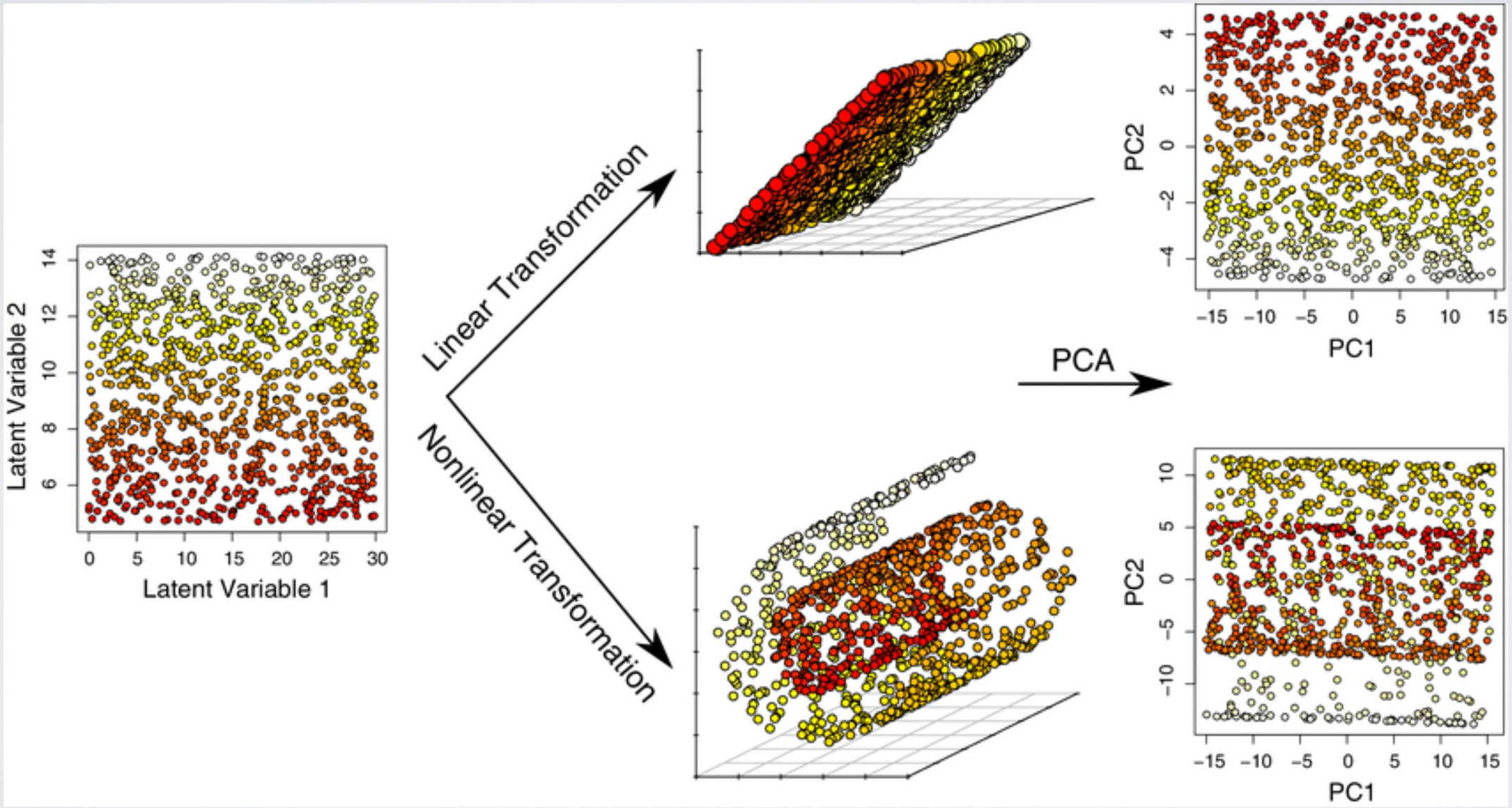
NON-LINEAR SITUATIONS






```
Pearson correlation(d1,d2): 0
```

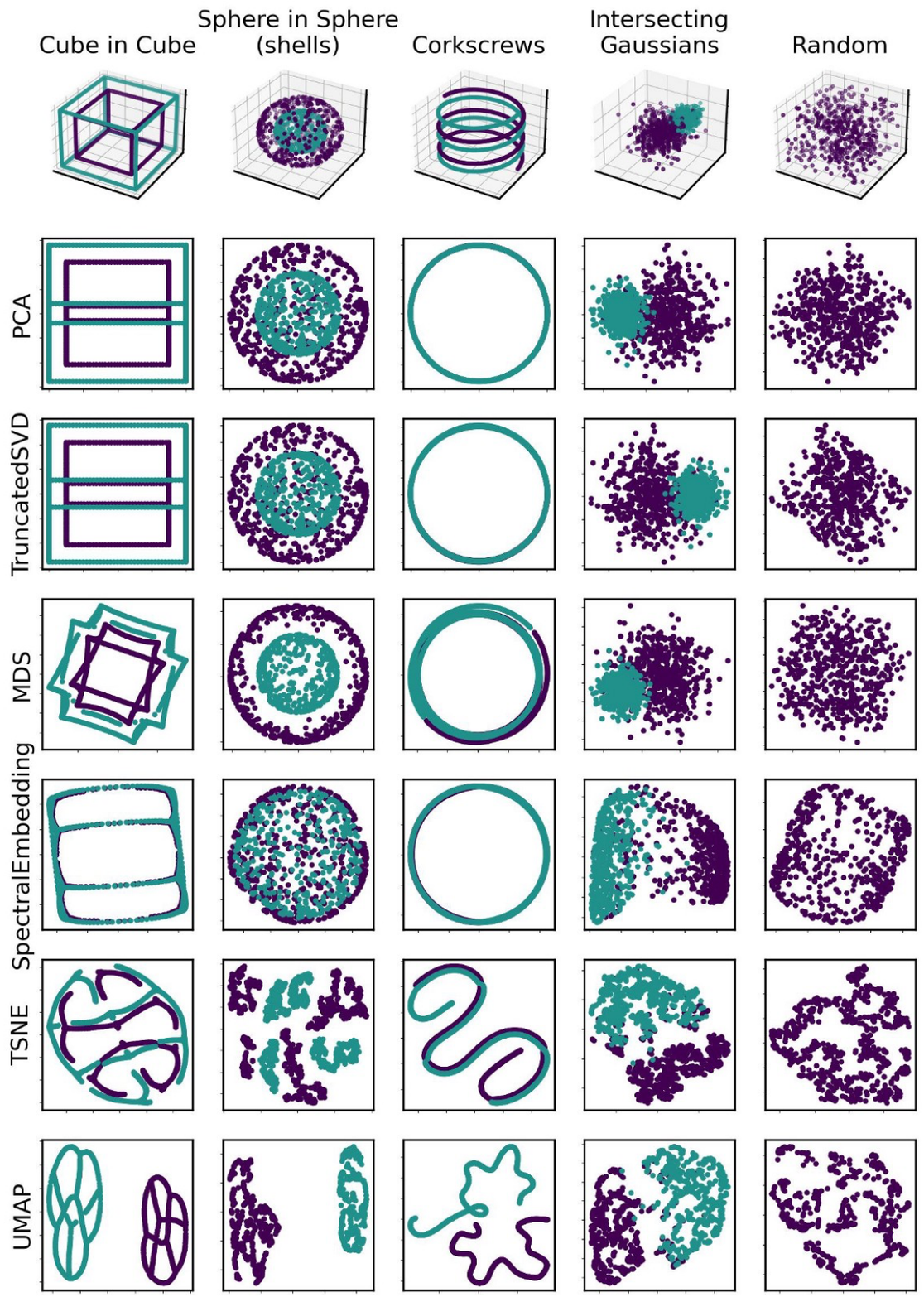
NONLINEAR DATA



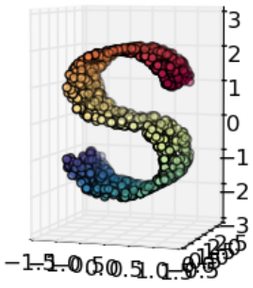
MANIFOLDS

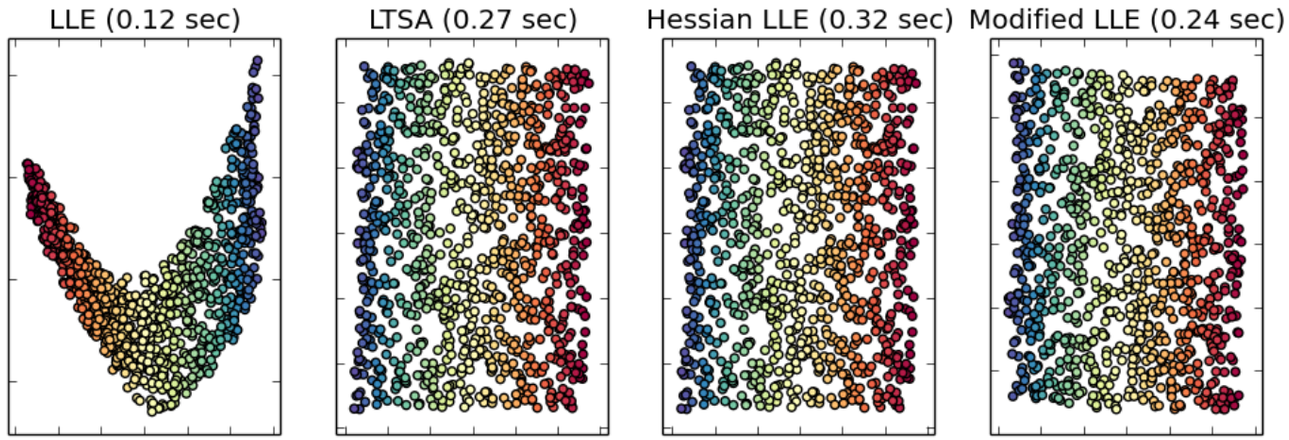
MANIFOLDS

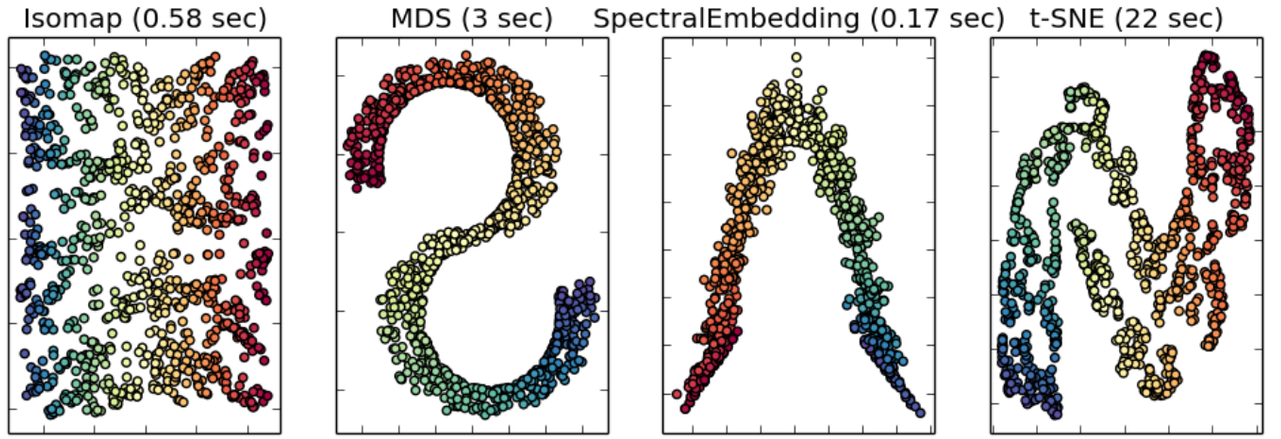
- Manifolds are another approach to dimensionality reduction
- The general principle is to
  - 1)Define a notion of distance between elements in the original space
  - 2)Define a notion of distance between elements in a reduced, target space
  - 3)Minimize the difference between distances in original and target space
- In many cases, the process is nonlinear, i.e., we choose distances such as
  - We care more about preserving close proximity than exact distance for nodes that are "far" from each other


Cube in Cube
Sphere in Sphere (shells)
Corkscrews
Intersecting Gaussians
Random
PCA
TruncatedSVD
MDS
SpectralEmbedding
TSNE
UMAP

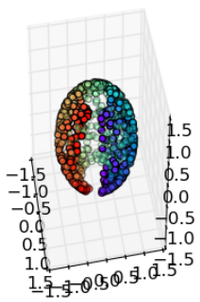
Manifold Learning with 1000 points, 10 neighbors

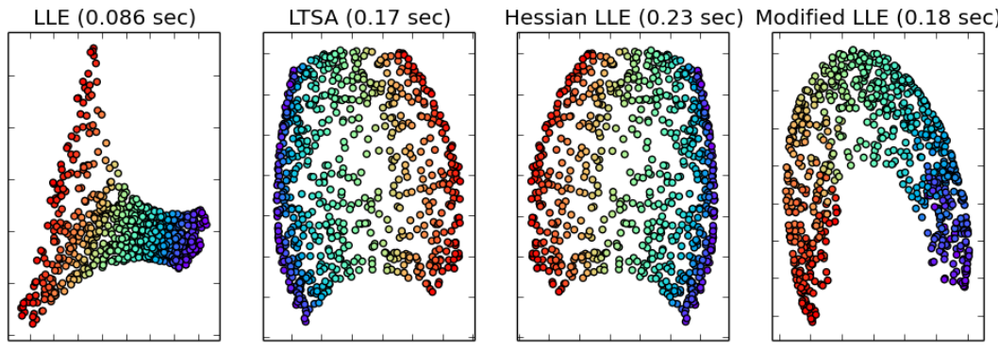


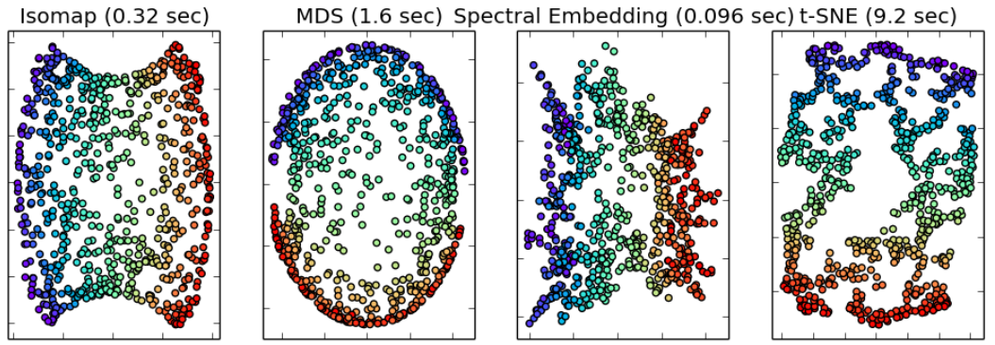


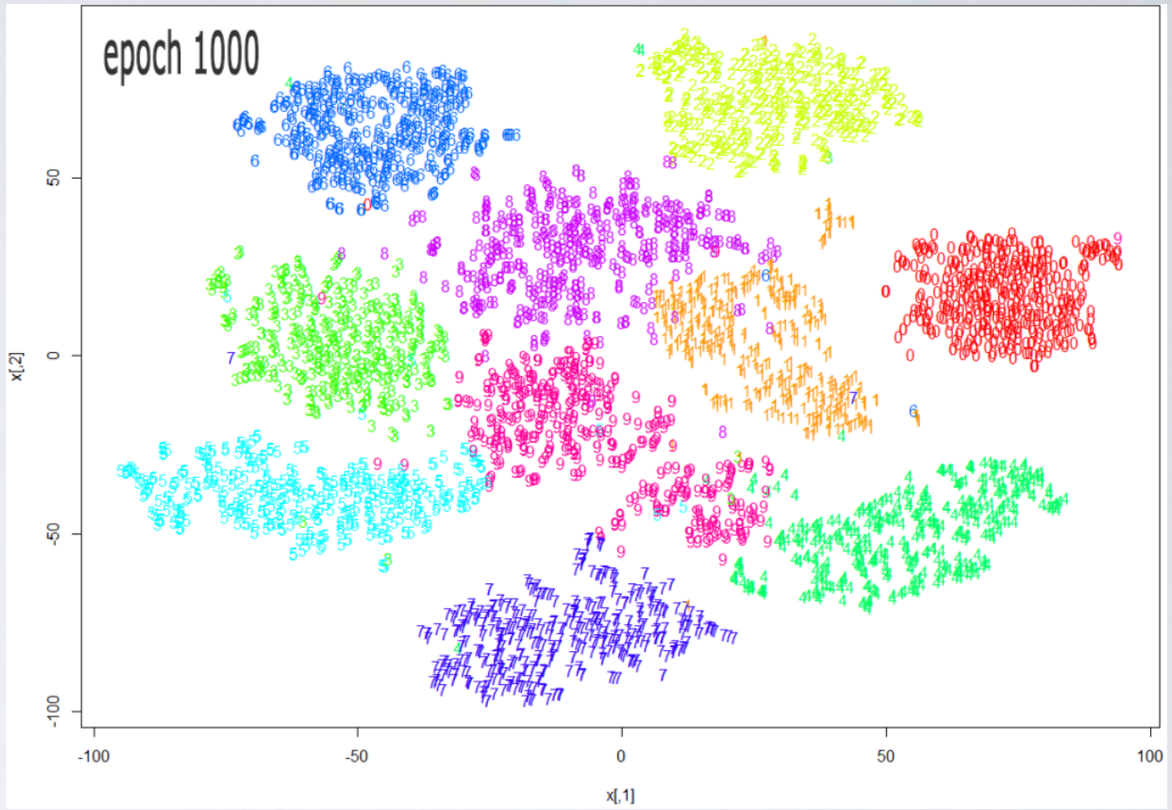


Manifold Learning with 1000 points, 10 neighbors

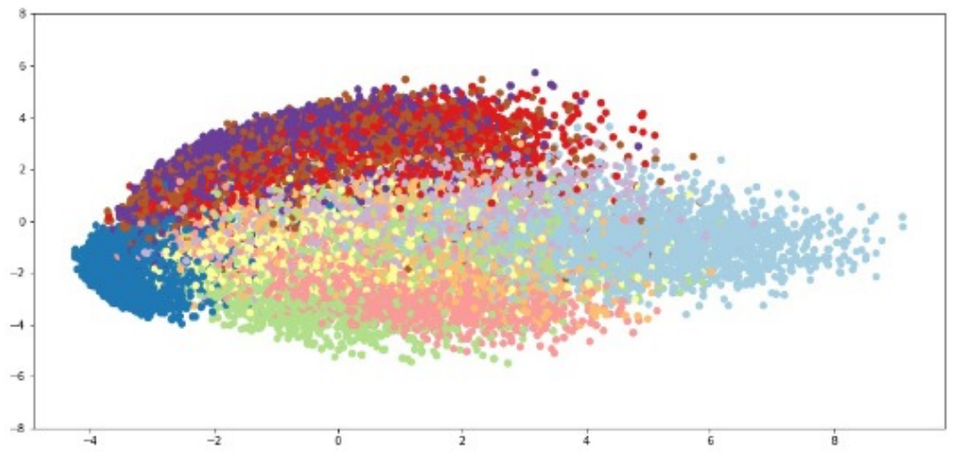




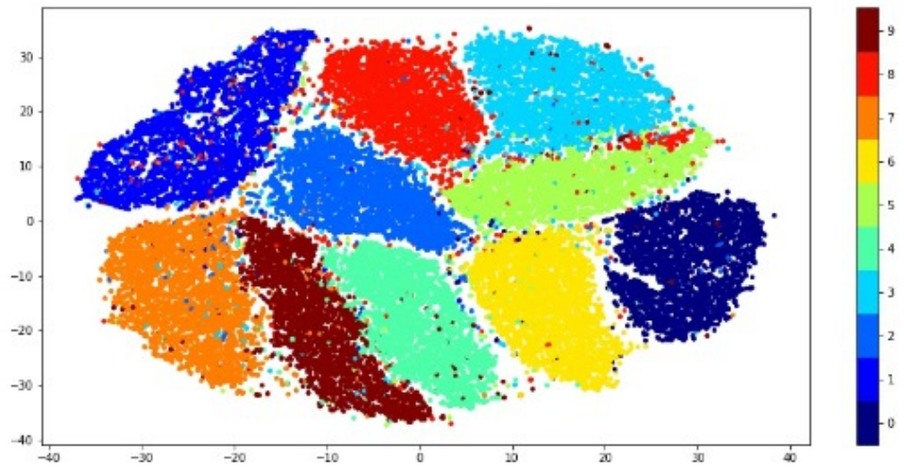




MNIST - PCA



MNIST - TSNE



MDS

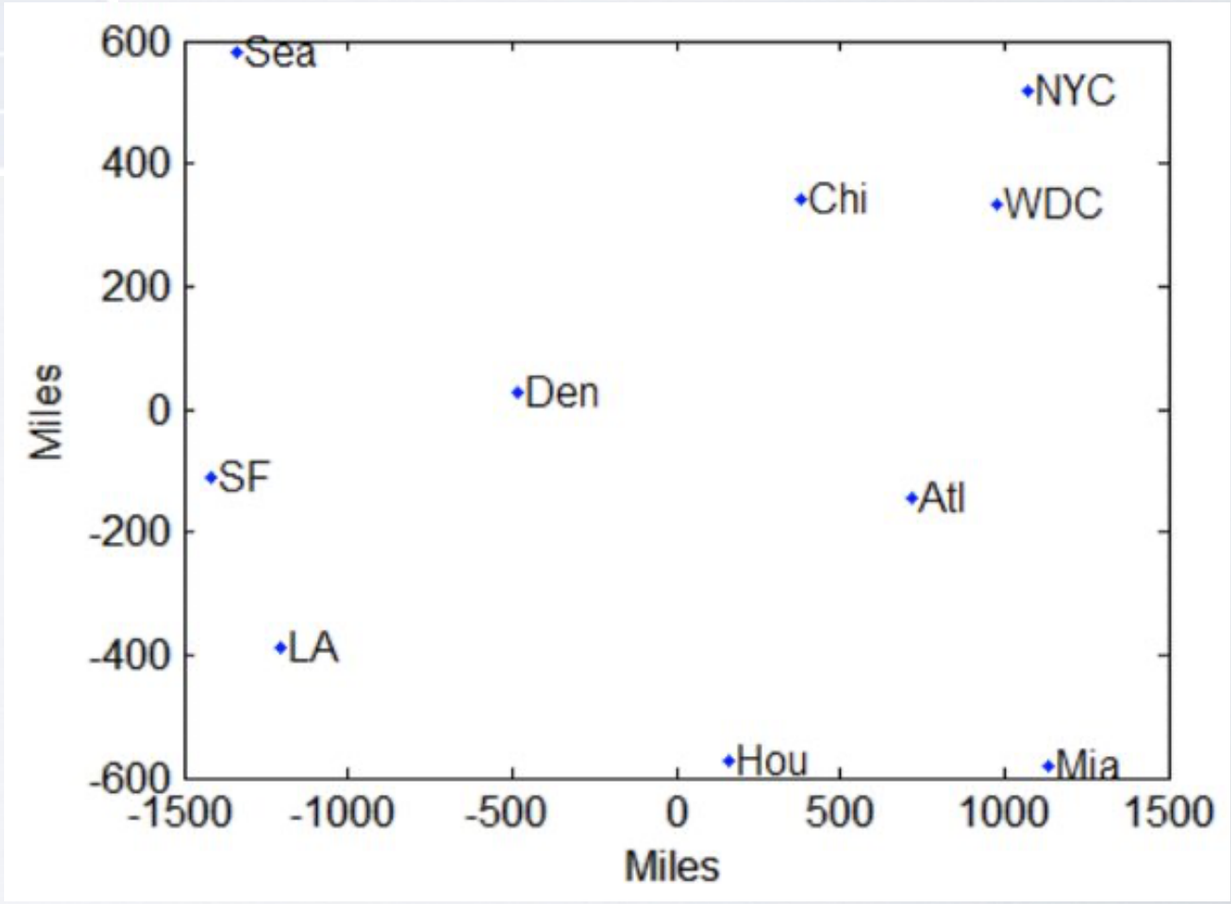
- MDS: Multi-dimensional Scaling:
  - Simply minimize distance between original space and target space
    - e.g., d-dimensional forced to 2-dimensional

- How to do it?
  - 1)Compute all pairwise distances between Objects=>similarity matrix
    - n x f matrix => n x n matrix
  - 2)Compute PCA on this similarity matrix
    - PCA preserves columns information => preserve distance on a similarity matrix

- Problems:
  - Very costly (nb features=nb elements), n^2
  - Try to preserve all distances, therefore extremely constrained

MDS

	Atl	Chi	Den	Hou	LA	Mia	NYC	SF	Sea	WDC
Atl	0	587	1212	701	1936	604	748	2139	2182	543
Chi	587	0	920	940	1745	1188	713	1858	1737	597
Den	1212	920	0	879	831	1726	1631	949	1021	1494
Hou	701	940	879	0	1374	968	1420	1645	1891	1220
LA	1936	1745	831	1374	0	2339	2451	347	959	2300
Mia	604	1188	1726	968	2339	0	1092	2594	2734	923
NYC	748	713	1631	1420	2451	1092	0	2571	2408	205
SF	2139	1858	949	1645	347	2594	2571	0	678	
Sea	2182	1737	1021	1891	959	2734	2408	678	0	
WDC	543	597	1494	1220	2300	923	205	2442	2329	

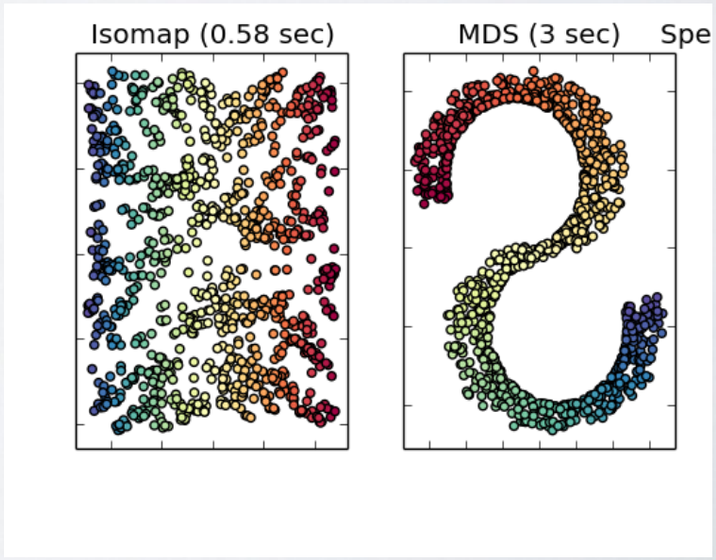


ISOMAP

- Variation of MDS
  - 1)We define a graph such as two elements are connected if they are at distance<threshold. (Alternative: fixed number of neighbors)
    - Put a weight on edges=euclidean distance
  - 2)Compute a similarity matrix, such as distance= weighted shortest path distance
  - 3)Apply MDS on it

- Computing shortest paths on a graph is fast
  - Floyd–Warshall algorithm

- Much less constraints



T-SNE

T-SNE

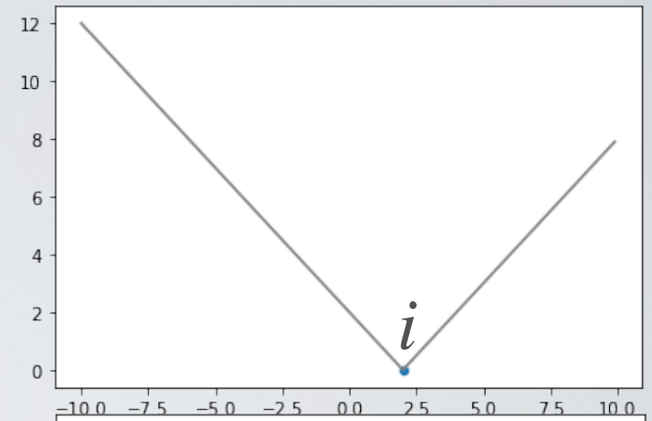
- t-SNE : t-distributed stochastic neighbor embedding
- Non-linear dimensionality reduction
- Currently the most popular method for visualizing data in low dimensions

T-SNE

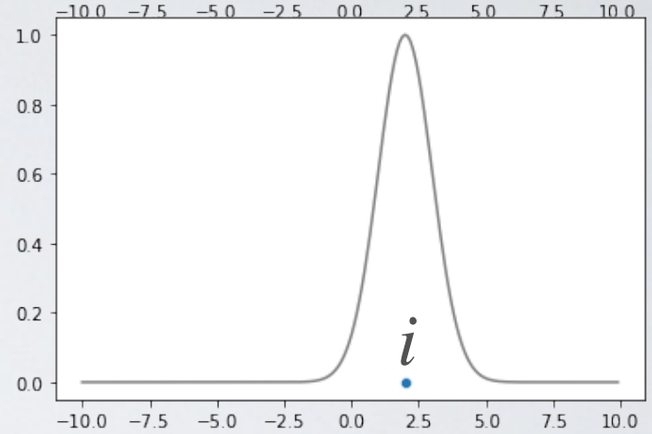
- General principle:
  - Define a notion of similarity $p_{j|i}$ in the high dimensional space P
    - Based on normal distribution
  - Define a notion of similarity $q_{j|i}$ in the low dimensional space Q
    - Based on student-t distribution, tends to "exaggerate" differences
  - For each point of initial coordinates x_i, find a new coordinate y_i in the lower dimensional space, such as to minimize the difference between P and Q
    - $\forall_{i,j} p_{j|i} \approx q_{j|i}$

SNE

Euclidean



Normal

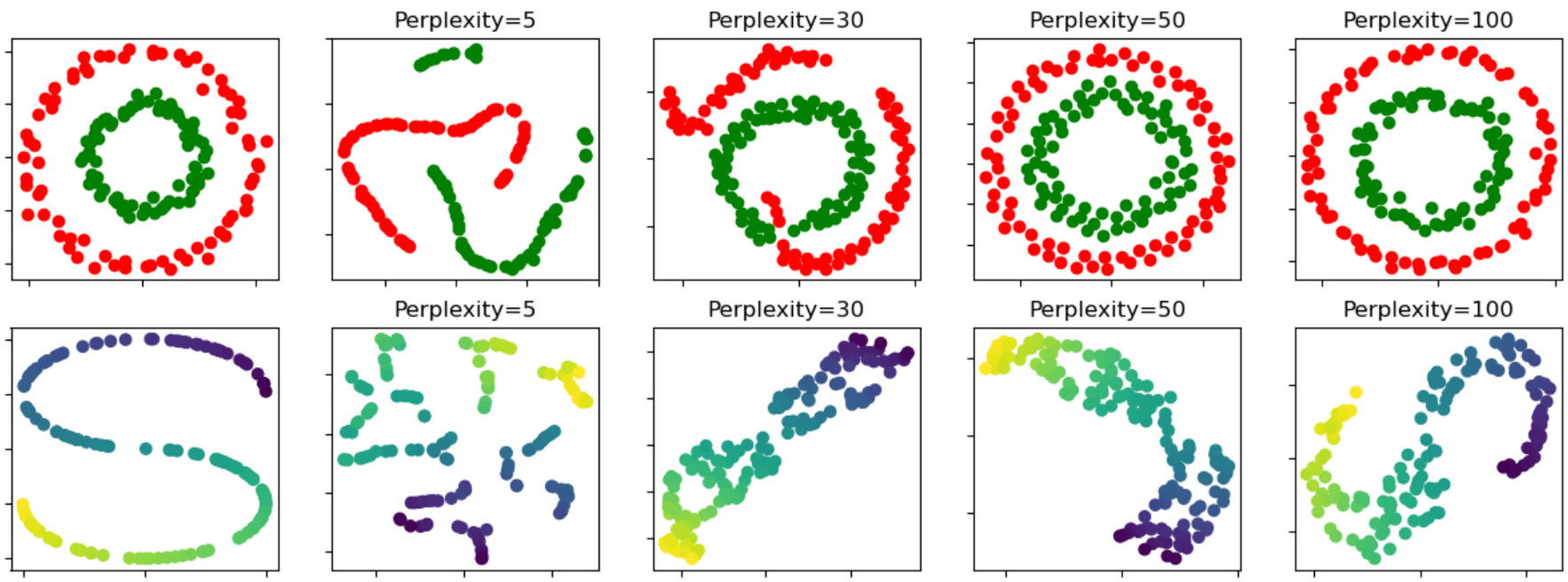


- Distance in the original space $\boldsymbol{P}$
  - To compute how far j is from i, consider a normal distribution centered in j with variance σ
  - Mathematically: the raw distance is given as $s^P_{j|i} = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}}$
  - $p_{j|i} = \frac{s^P_{j|i}}{\sum_{k \neq i} s^P_{j|k}}$
    - Normalizes the similarity by sum of similarity to all other points.
    - With proper σ, local definition of similarity

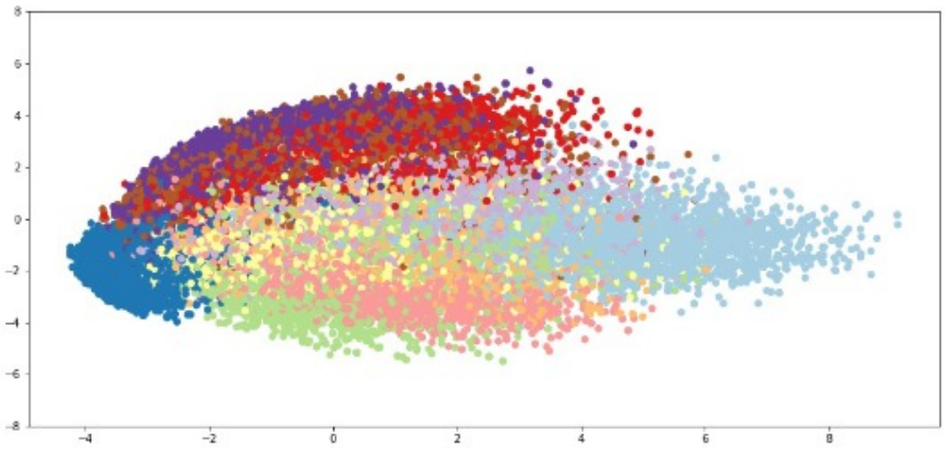
T-SNE: PERPLEXITY

- There is a perplexity parameter σ: it controls how much each point cares more about close neighbors compared with farther neighbors
  - Low σ: Preserve mostly local distances
  - High σ: Give more importance to long-range distances
    - More expensive, more similar to a PCA

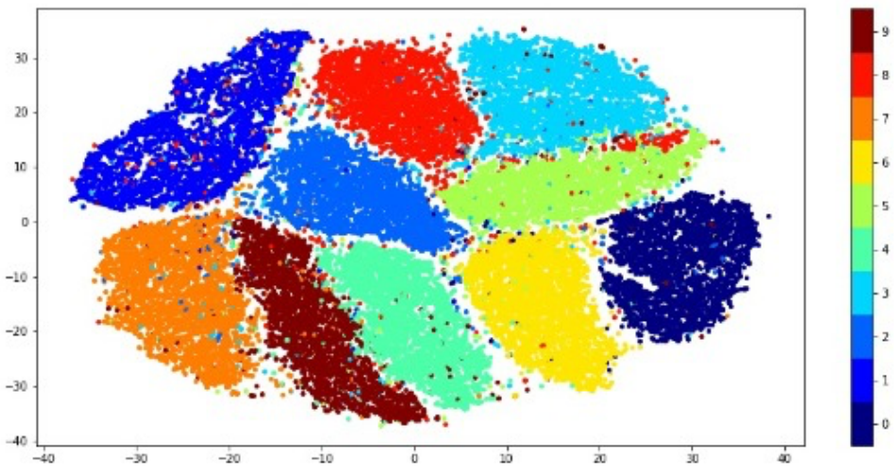
INFLUENCE OF PERPLEXITY



MNIST - PCA



MNIST - TSNE



LOW DIMENSIONAL EMBEDDINGS

EMBEDDINGS

- A recent usage of low dimensional embeddings is to encode complex objects as vectors
  - Words as Vector => Word2Vec
  - Nodes (of graph) as Vectors => Node2Vec
  - Documents as Vectors => Doc2Vec
  - ....

WORD EMBEDDING

WORD EMBEDDING

- Words can be understood as a (very) high dimensional space
  - Using One Hot encoding: vocabulary of 1000 words=>1000 columns

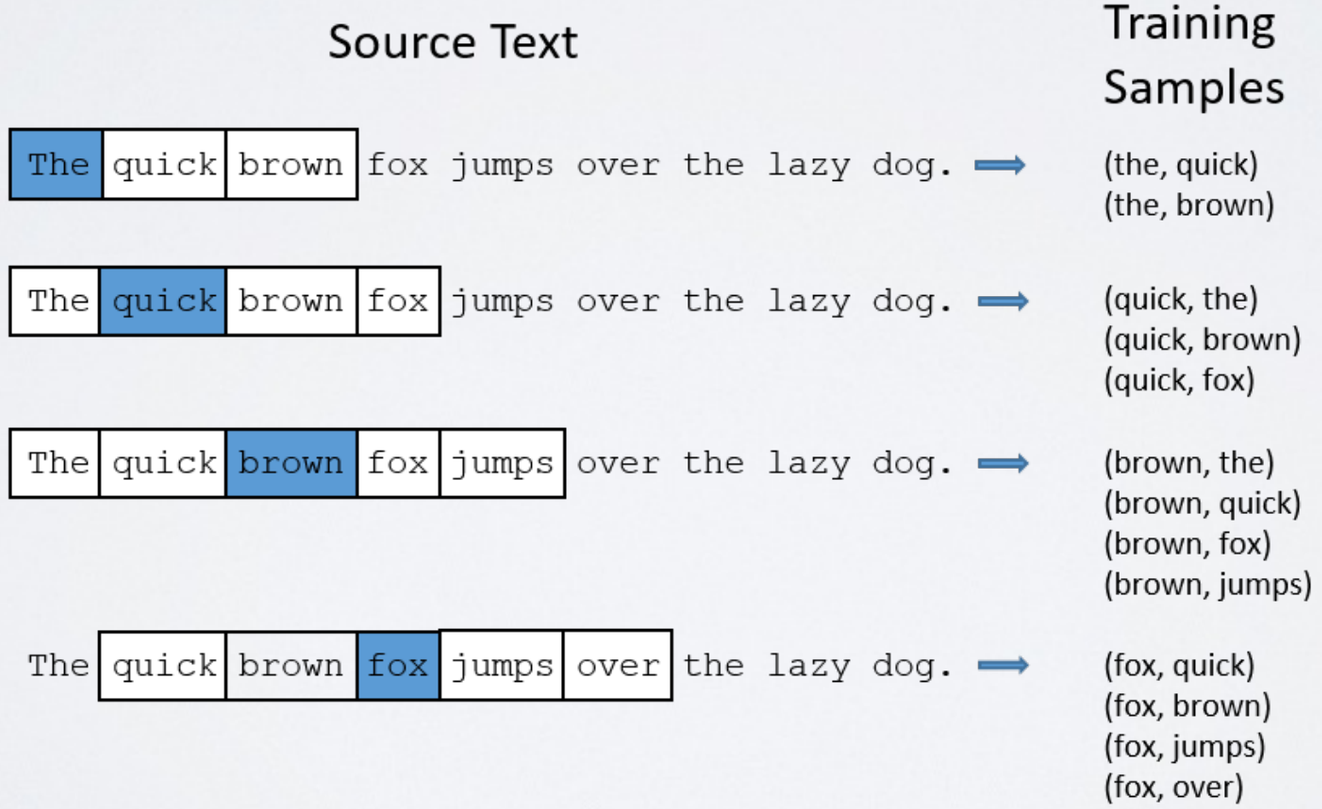
- Could we assign a vector in "low dimension", encoding the "semantic" of a word?
  - Two words with similar meanings should be close

SKIPGRAM

Word embedding

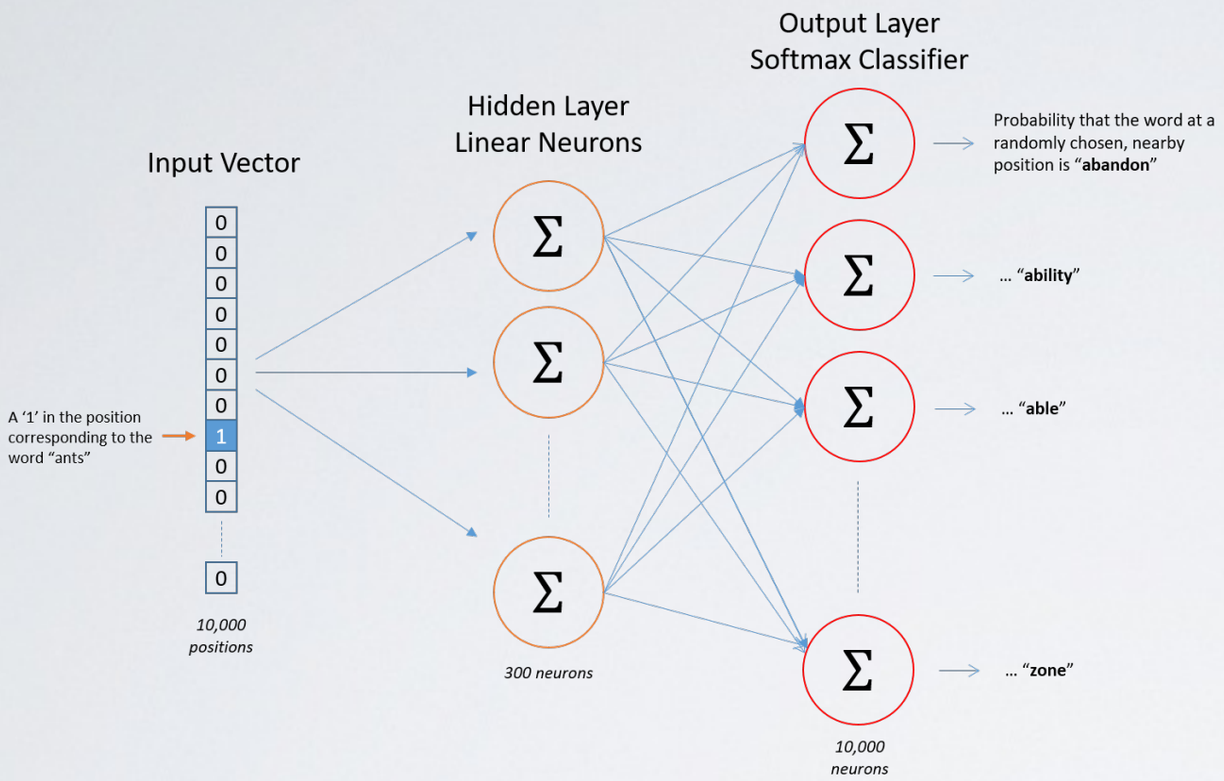
Corpus => Word = vectors

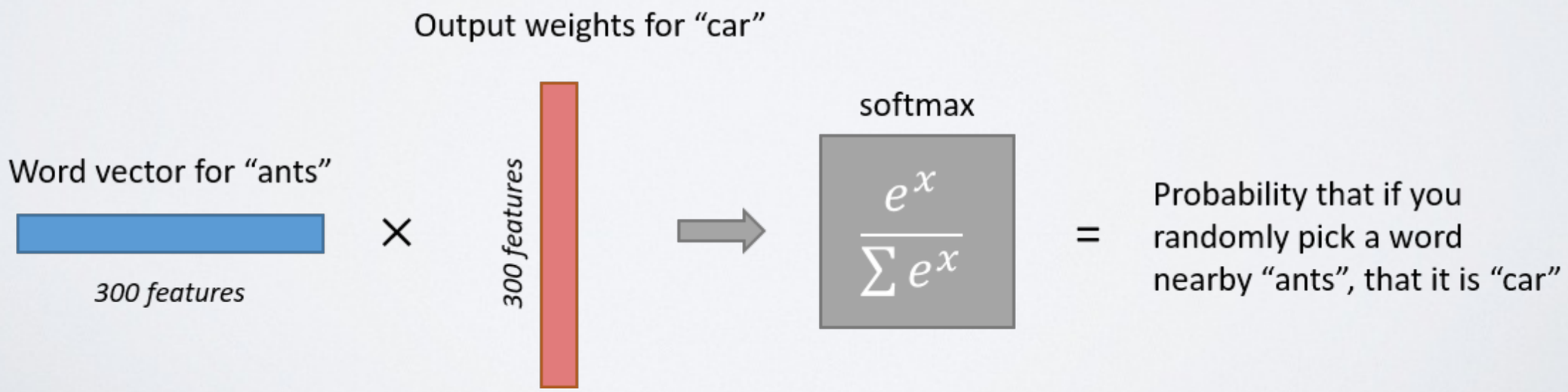
Similar embedding= similar **context**



[http://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model/]

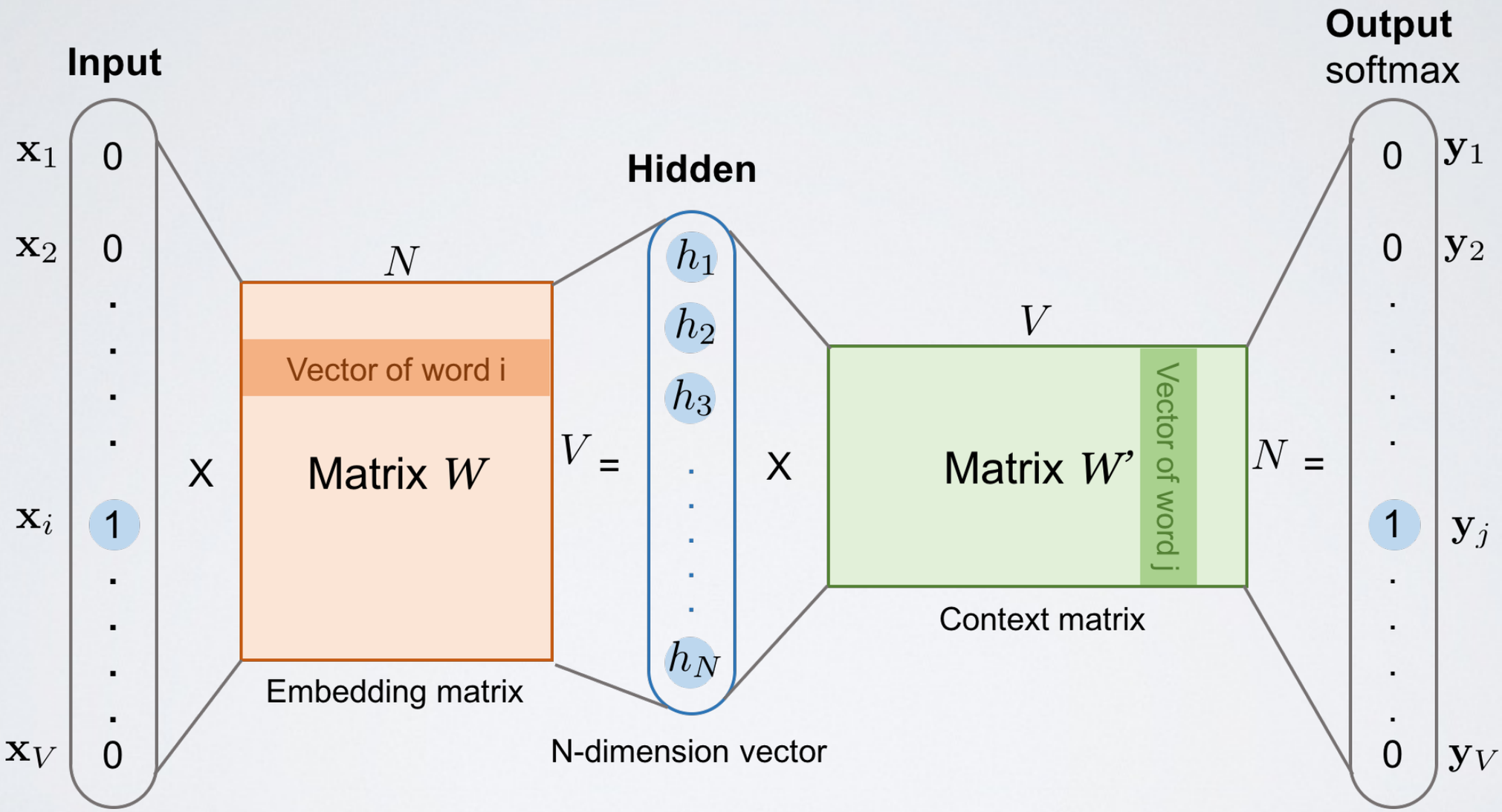
SKIPGRAM





https://towardsdatascience.com/word2vec-skip-gram-model-part-1-intuition-78614e4d6e0b

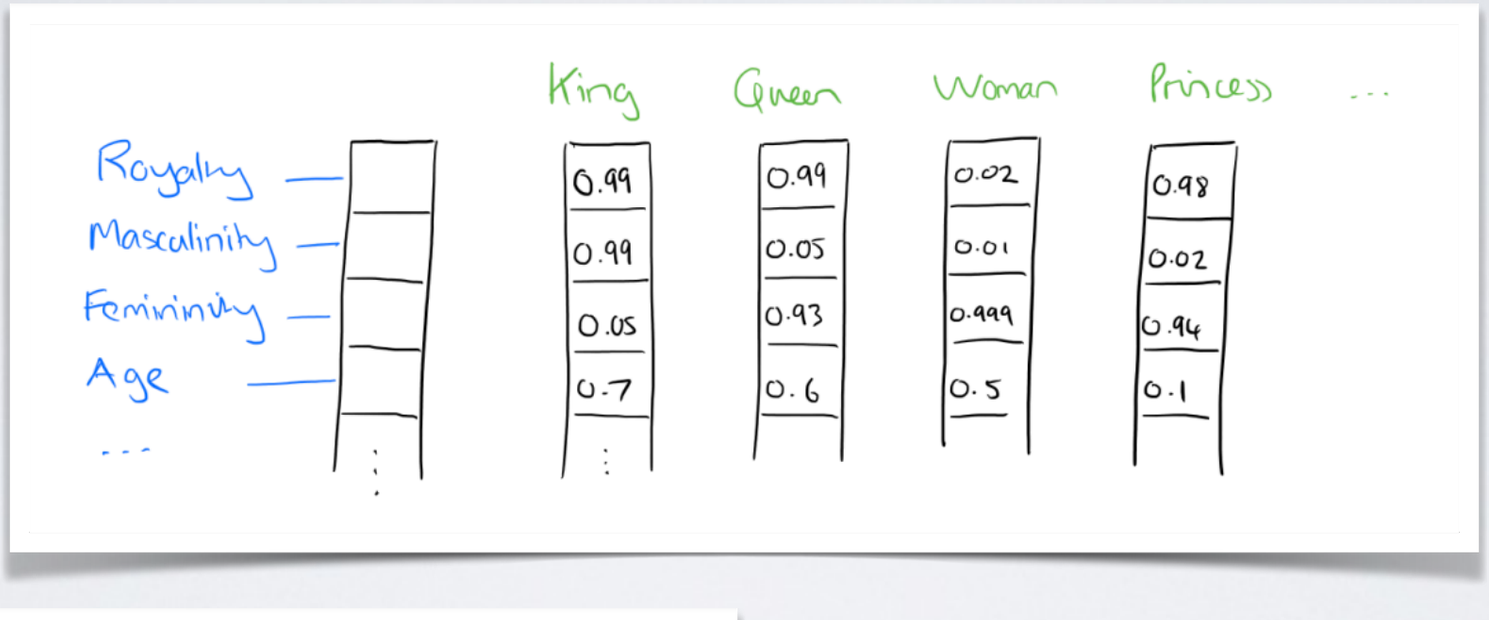
SKIPGRAM

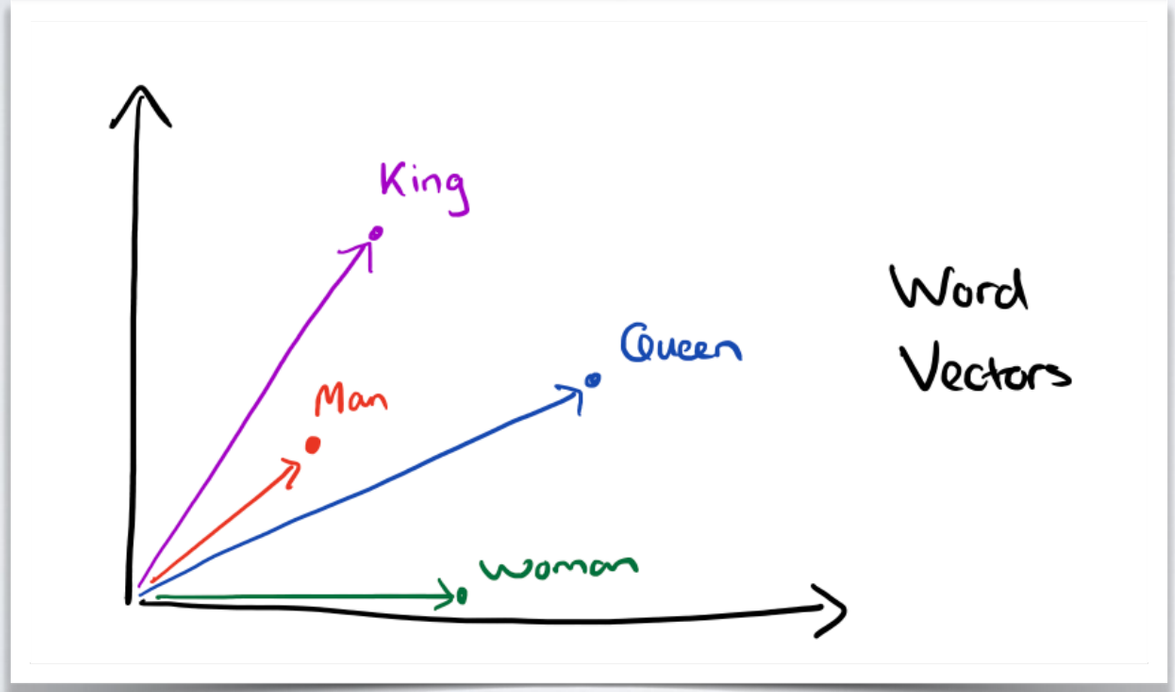


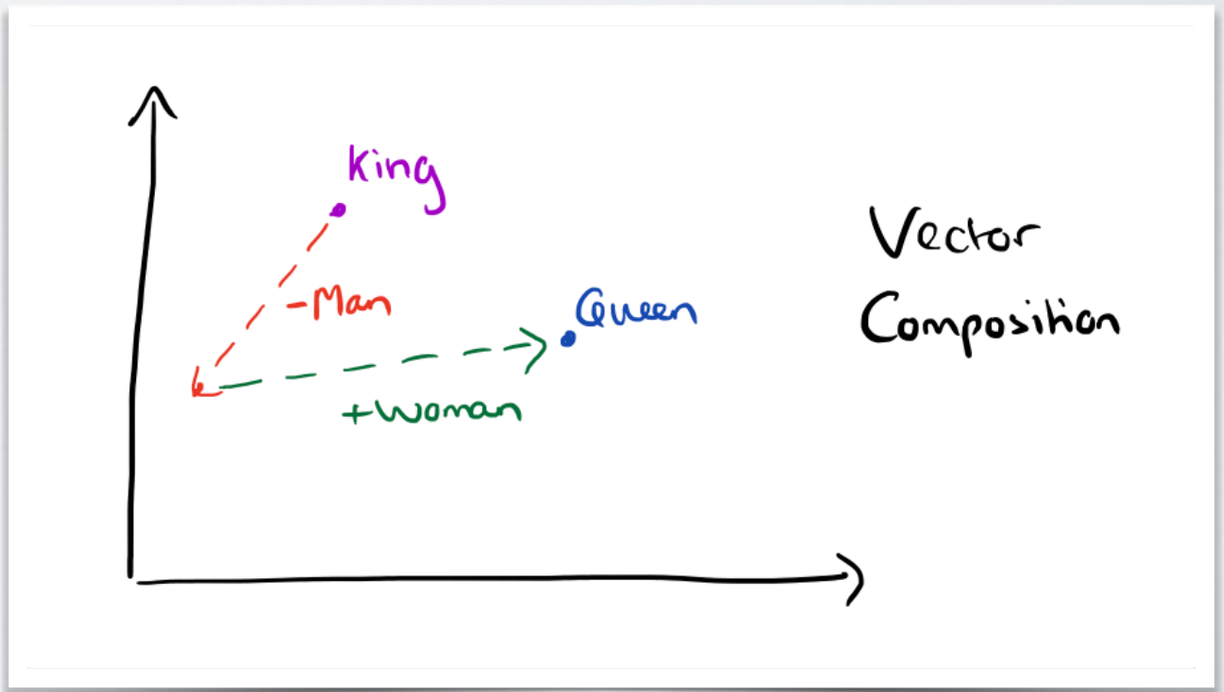
N=embedding size. **V**=vocabulary size

https://towardsdatascience.com/word2vec-skip-gram-model-part-1-intuition-78614e4d6e0b

SKIPGRAM







[https://blog.acolyer.org/2016/04/21/the-amazing-power-of-word-vectors/]

SKIPGRAM

Table 8: *Examples of the word pair relationships, using the best word vectors from Table 4 (Skip-gram model trained on 783M words with 300 dimensionality).*

Relationship	Example 1	Example 2	Example 3
France - Paris	Italy: Rome	Japan: Tokyo	Florida: Tallahassee
big - bigger	small: larger	cold: colder	quick: quicker
Miami - Florida	Baltimore: Maryland	Dallas: Texas	Kona: Hawaii
Einstein - scientist	Messi: midfielder	Mozart: violinist	Picasso: painter
Sarkozy - France	Berlusconi: Italy	Merkel: Germany	Koizumi: Japan
copper - Cu	zinc: Zn	gold: Au	uranium: plutonium
Berlusconi - Silvio	Sarkozy: Nicolas	Putin: Medvedev	Obama: Barack
Microsoft - Windows	Google: Android	IBM: Linux	Apple: iPhone
Microsoft - Ballmer	Google: Yahoo	IBM: McNealy	Apple: Jobs
Japan - sushi	Germany: bratwurst	France: tapas	USA: pizza

[https://blog.acolyer.org/2016/04/21/the-amazing-power-of-word-vectors/]

PRE-TRAINED

- You can easily train word2vec on your own dataset, but it needs to be large enough
  - https://radimrehurek.com/gensim/models/word2vec.html
- You can use pre-trained embeddings, trained on enormous corpus (Twitter, Wikipedia…)
  - e.g., Glove: https://nlp.stanford.edu/projects/glove/

USAGE

- Single words=> Use directly vectors

- Short texts=> Weighted average vectors (more weights to more important words, e.g., rare words: TF-IDF…)

- Long texts=> More tricky. Need other approaches (Doc2vec, RNN)

USAGE

- Parameters:
  - Embedding dimensions *d*
  - Context size

GRAPH EMBEDDING

GENERIC "SKIPGRAM"

- Algorithm that takes an input:
  - The element to embed
  - A list of "context" elements
- Provide as output:
  - An embedding with interesting properties
    - Works well for machine learning
    - Similar elements are close in the embedding
    - Somewhat preserves the overall structure

DEEPWALK

- Skipgram for graphs:
  - 1)Generate "sentences" using random walks
  - 2)Apply Skipgram

- Parameters:
  - Same as Skipgram
    - Embedding dimensions *d*
    - Context size
  - Parameters for "sentence" generation: length of random walks, number of walks starting from each node, etc.

Perozzi, B., Al-Rfou, R., & Skiena, S. (2014, August). Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 701-710). ACM.

NODE2VEC

- Use biased random walk to tune the context to capture *what we want*
  - "Breadth first" like RW => local neighborhood (edge probability ?)
  - "Depth-first" like RW => global structure ? (Communities ?)
  - 2 parameters to tune:
    - **p**: bias towards revisiting the previous node
    - **q**: bias towards exploring undiscovered parts of the network

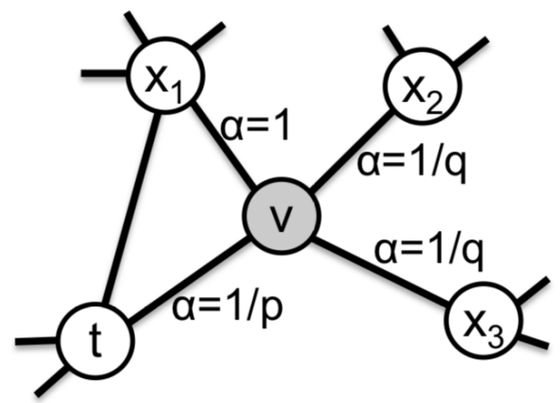


Figure 2: Illustration of the random walk procedure in *node2vec*. The walk just transitioned from t to v and is now evaluating its next step out of node v. Edge labels indicate search biases α.

Grover, A., & Leskovec, J. (2016, August). node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 855-864). ACM.

EMBEDDING ROLES

STRUC2VEC/ROLE2VEC

- In node2vec/Deepwalk, the context collected by RW contains the **labels** of encountered nodes

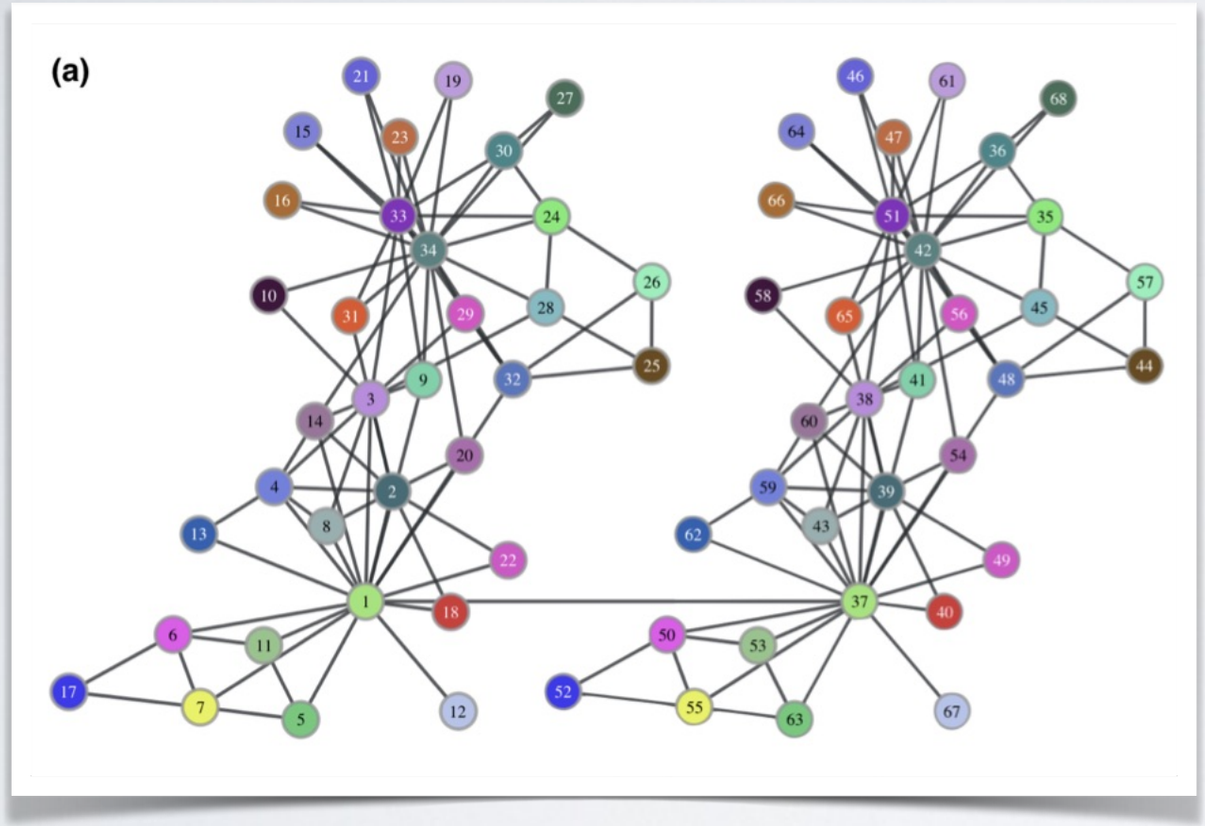
- Instead, we could memorize the **properties** of the nodes: attributes if available, or computed attributes (degrees, CC, …)

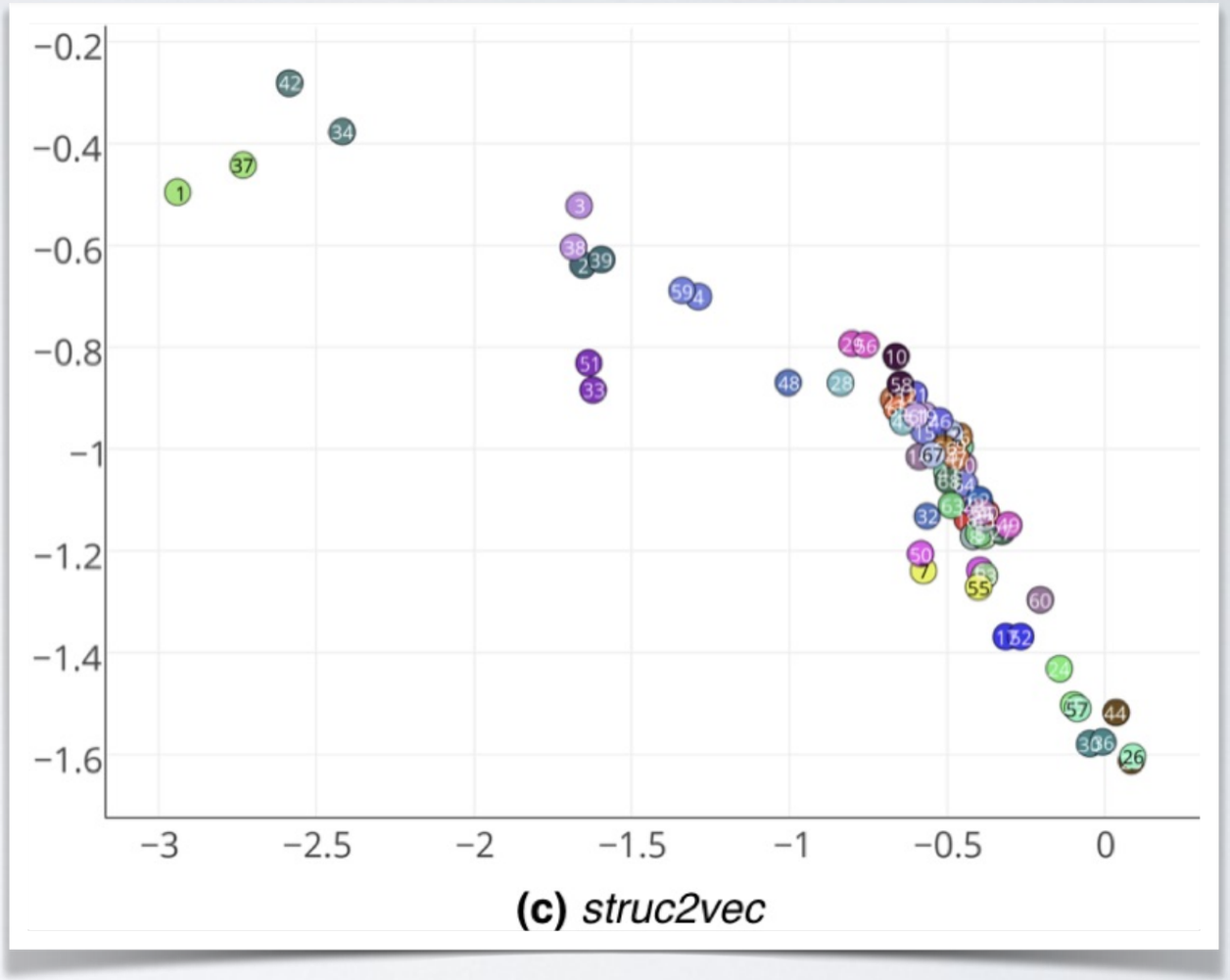
- =>Nodes with a same context will be nodes in a same "position" in the graph

- =>Capture the role of nodes instead of proximity

Ribeiro, L. F., Saverese, P. H., & Figueiredo, D. R. (2017, August). struc2vec: Learning node representations from structural identity. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 385-394). ACM.

STRUCT2VEC : DOUBLE ZKC





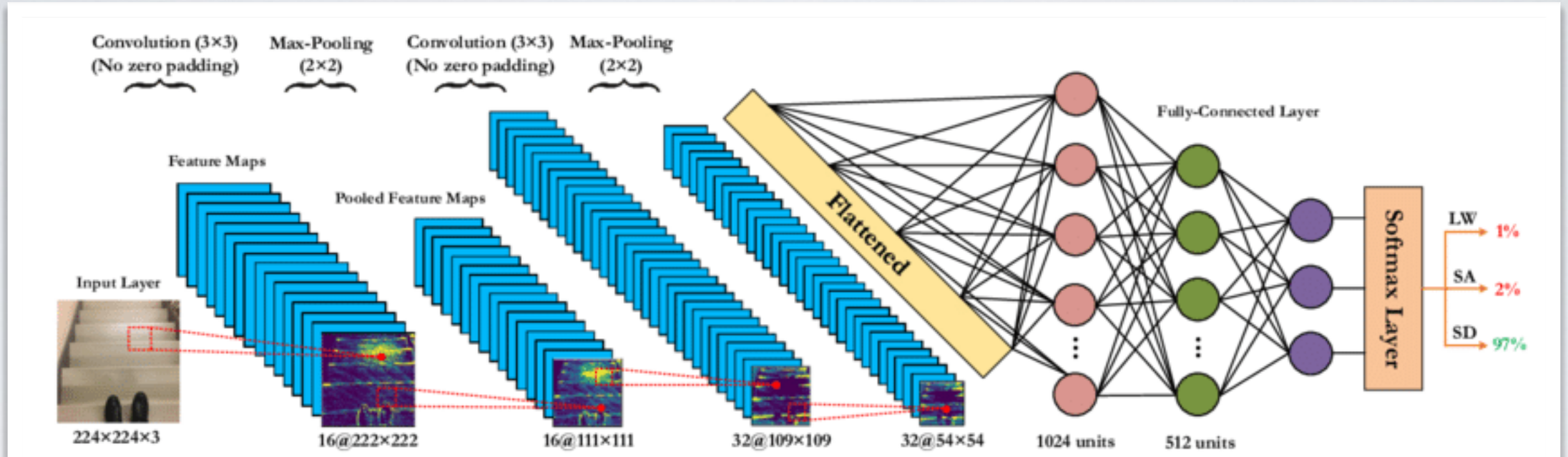
Ribeiro, L. F., Saverese, P. H., & Figueiredo, D. R. (2017, August). struc2vec: Learning node representations from structural identity. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 385-394). ACM.

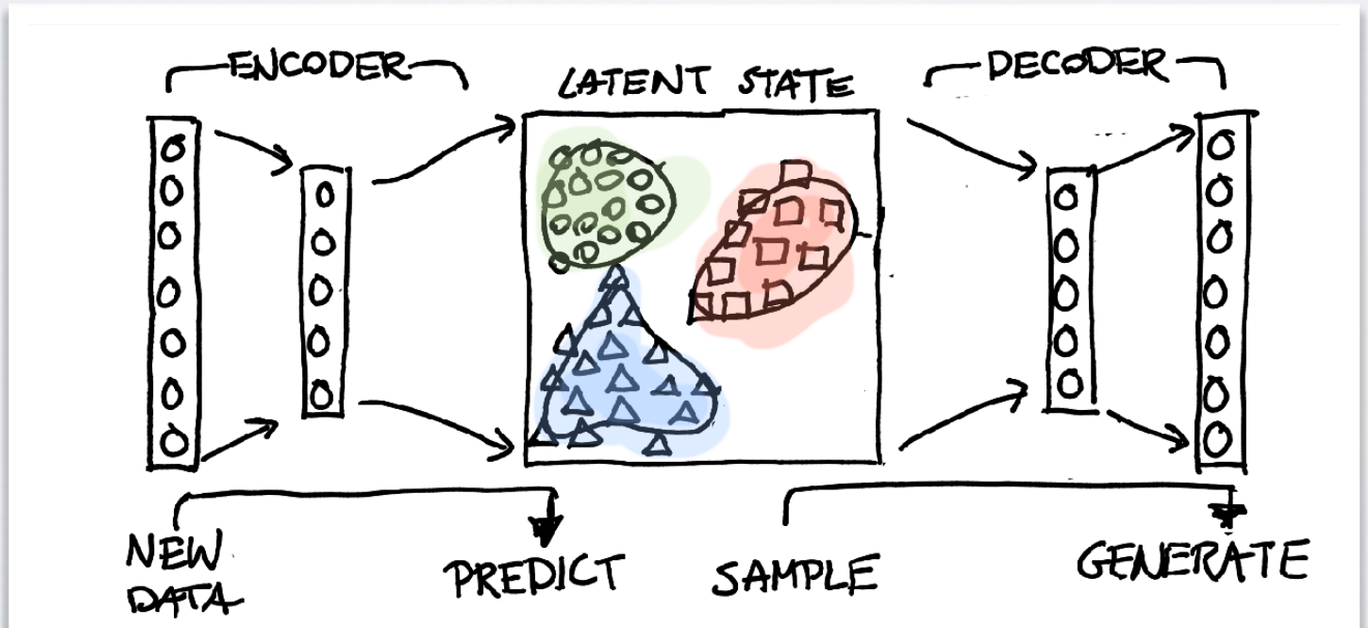
DEEP LEARNING AND EMBEDDINGS

SHALLOW TO DEEP

- Deep neural networks are also commonly used to produce complex data embedding
  - Skipgram/Word2Vec/Node2Vec are just particular cases of a general principle
- After each layer of a DNN, items are represented as vectors
  - Usually, at some steps, those layers are low-dimensional
  - Often, the last step or the middle step
  - These can be used as embedding for other tasks

SHALLOW TO DEEP





APPLICATIONS

- Image modification: modify some values of the embedding of an object (image, music, graph…) to reconstruct a slightly different version of it

- Clustering
  - Train a DNN on image classification task, then use clustering on the embeddings to discover similar images

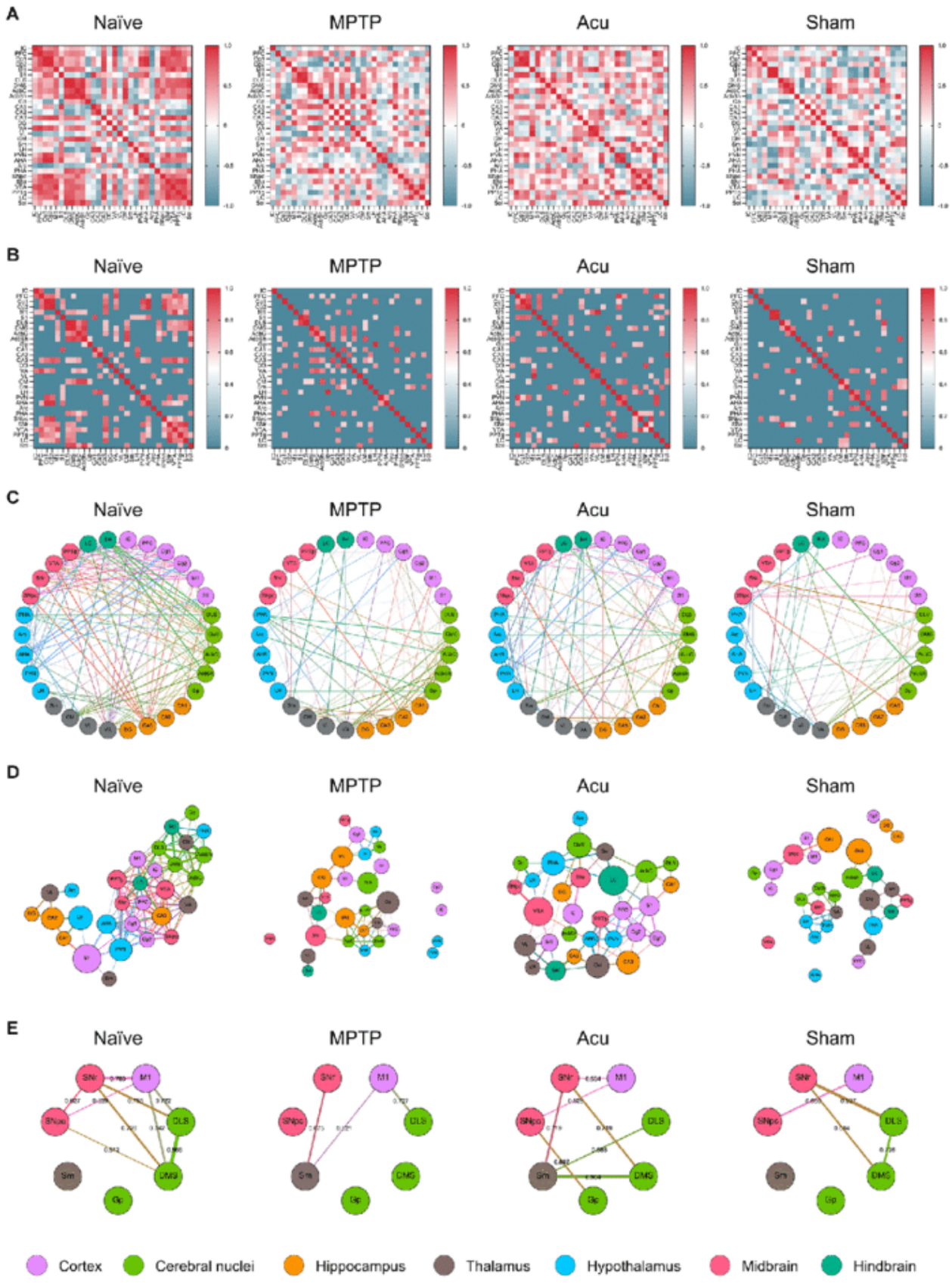
- Visualization
  - Using Tsne on an embedding, we can have a global view of the organization of our data
    - Music, photos, graphs, books…

OBJECTS/VECTORS TO GRAPHS

GRAPH<->VECTORS

- Graph Embedding: Graph->Vectors

- What about Vectors->Graphs
  - Simple approach: Correlation matrix
  - =>Represent the relations between features in a dataset
    - 1)Compute the correlation between all variables(spearman/Pearson)
    - 2)Keep only correlations above a threshold
    - 3)Correlation values can be represented as weights

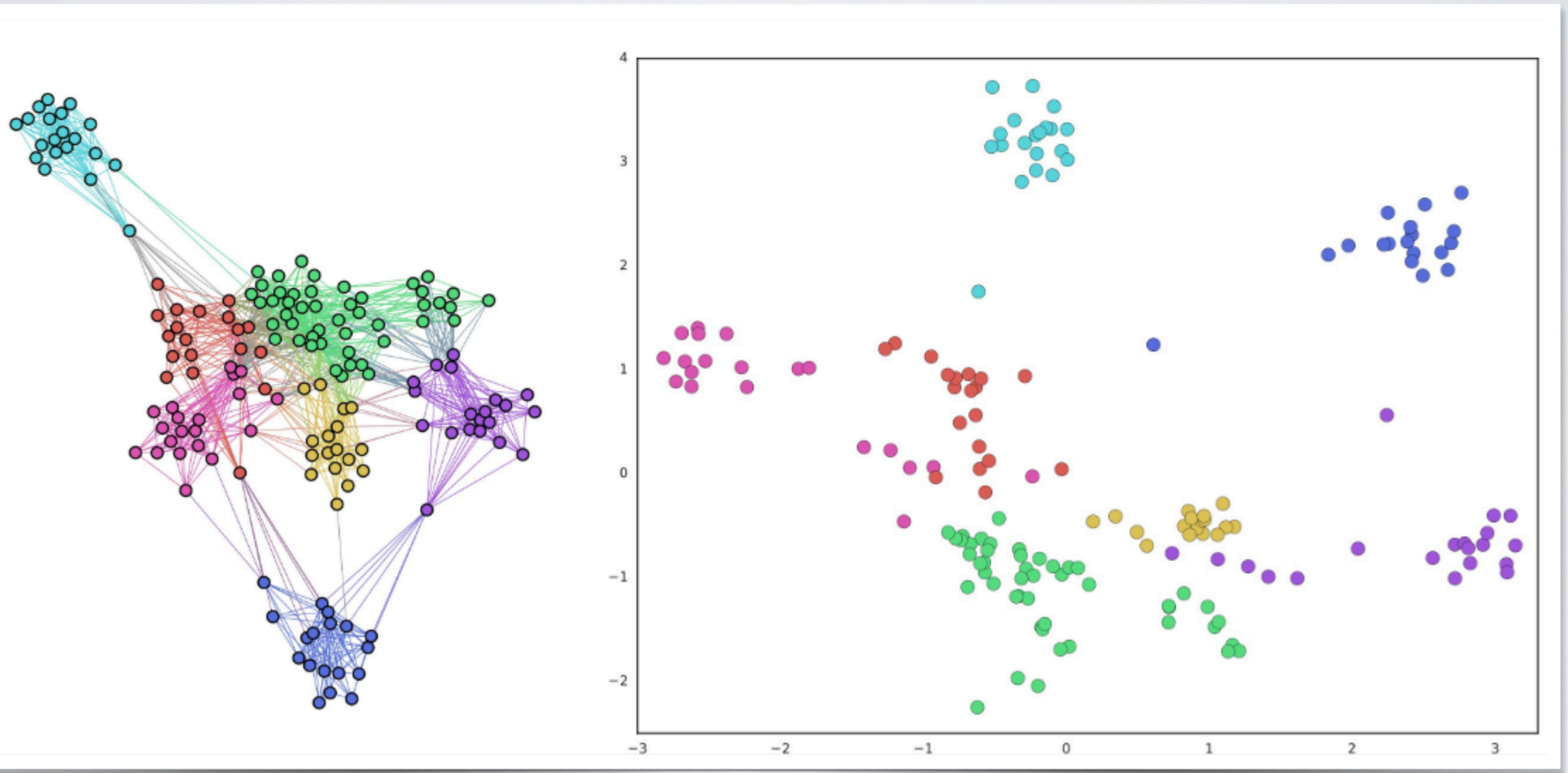

A
Naïve
MPTP
Acu
Sham
B
Naïve
MPTP
Acu
Sham
C
Naïve
MPTP
Acu
Sham
D
Naïve
MPTP
Acu
Sham
E
Naïve
MPTP
Acu
Sham
Cortex
Cerebral nuclei
Hippocampus
Thalamus
Hypothalamus
Midbrain
Hindbrain

ITEM-ITEM GRAPH

- We can use graphs as an alternative to dimensionality reduction for visualization
  - PCA / tSNE: project items in 2D, close items are similar
    - Some impossibilities, e.g., palm (part of the hand, tree)
  - Networks can also be viewed in 2D and preserve the similarity information

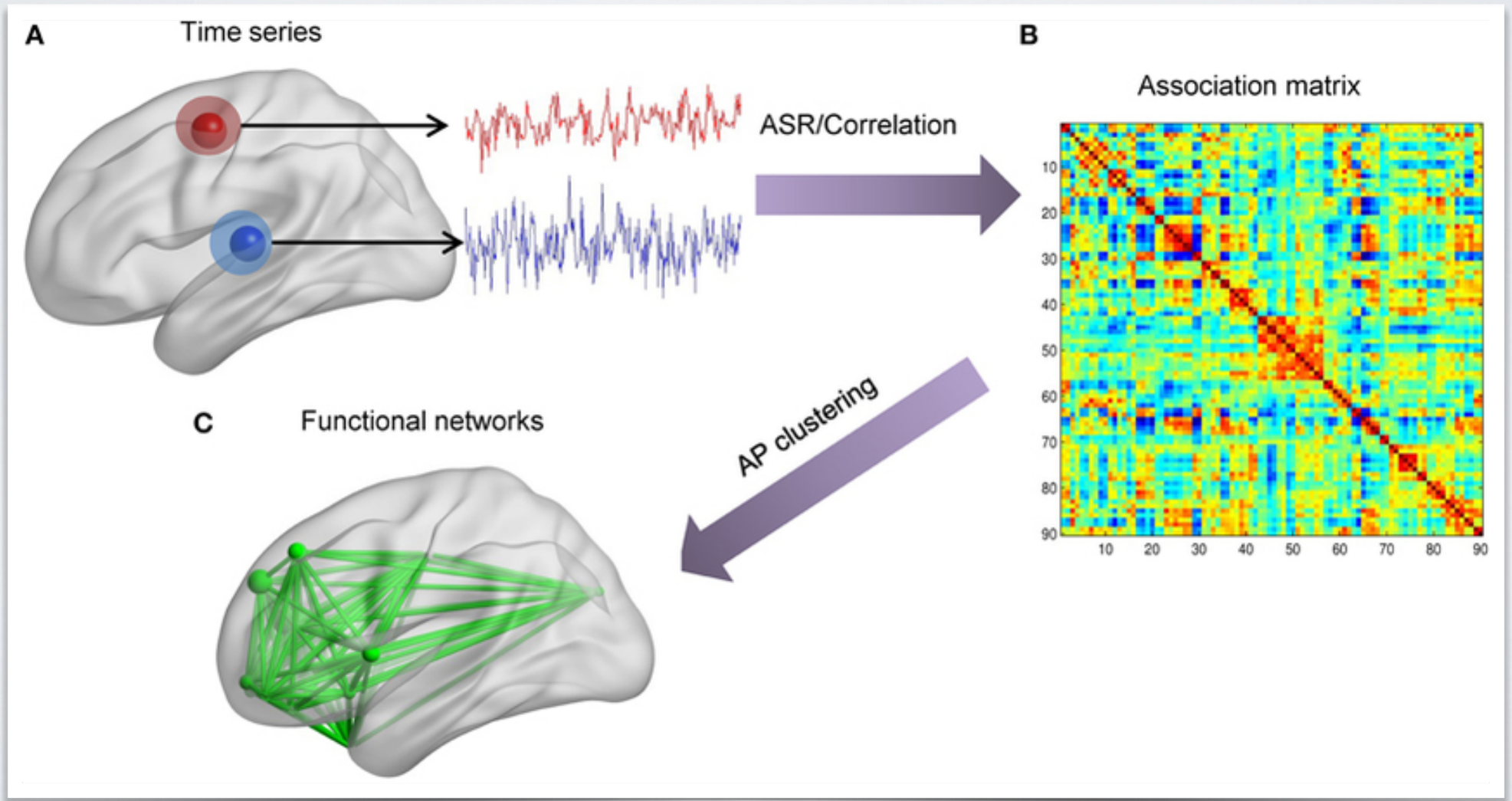
- Approach:
  - 1)Compute a distance between elements
    - Euclidean
    - Cosine (in recommendation settings for instance)
  - 2)Keep as edge values above a threshold

ITEM-ITEM GRAPH



ITEM-ITEM GRAPH

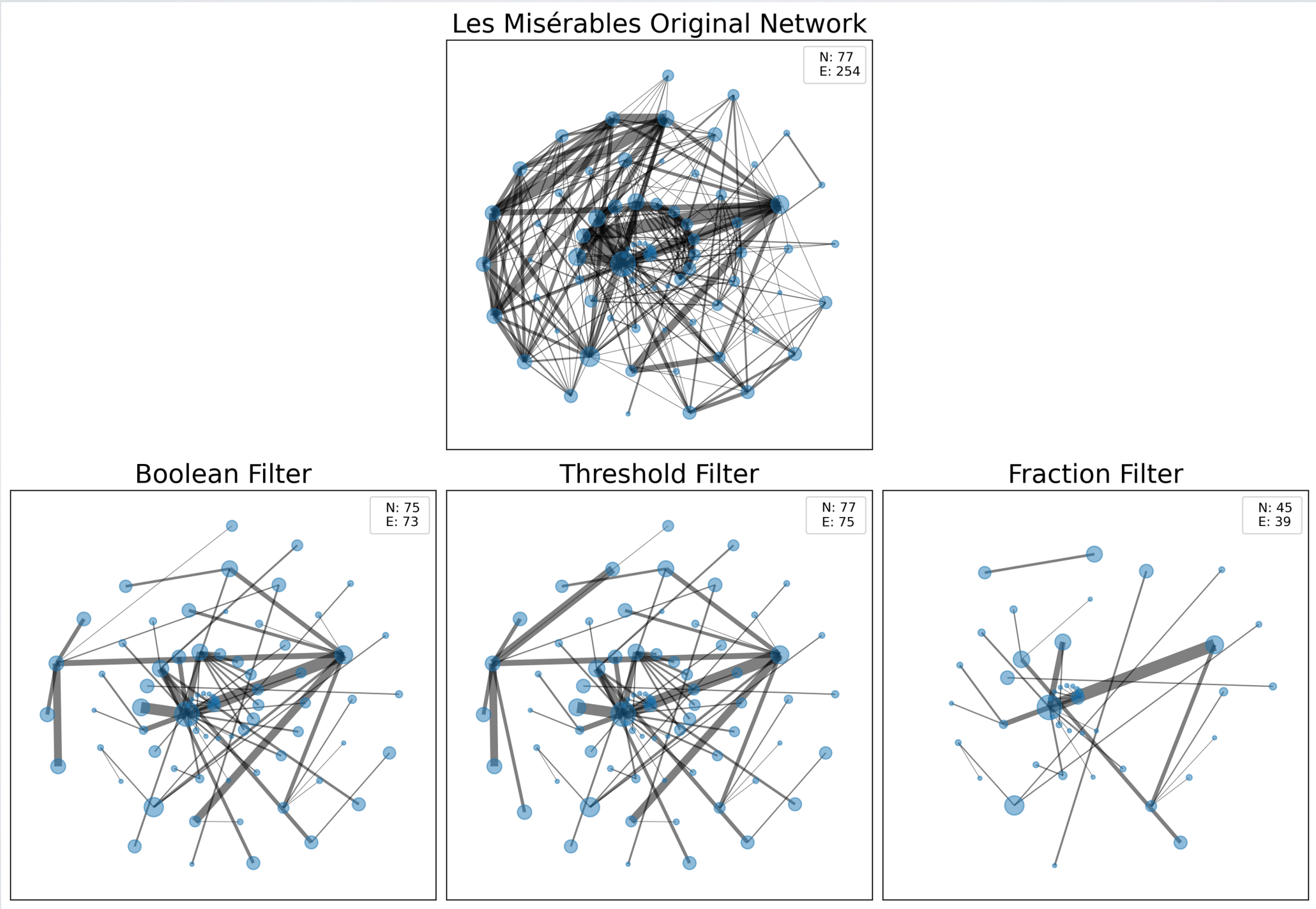
- Typical application case: Brain signal analysis
  - Distance is computed as signal correlation on fMRI, i.e., regional brain activity



BACKBONE EXTRACTION

- In many cases, the network created might be too dense to be analyzed properly
  - Too low threshold: everything is connected
  - Too high: disconnected graph
- A solution is to use Backbone extraction
  - Methods that retain only the most important edges, based on different principles
  - e.g., https://gitlab.liris.cnrs.fr/coregraphie/netbone

BACKBONE EXTRACTION



PROJECT

OBJECTIF

- Rendre un dossier court (6 Pages maximum+ Figures)
  - De qualité professionnelle
  - => Article de data-journalisme
  - => Article scientifique décrivant une étude empirique
  - => Rapport pour un client, un employeur
- Faire parler les données
- Utilisation des outils vus en cours, mais d'autres outils sont autorisés
  - Interdiction de se concentrer sur une tâche supervisé (Pas l'objectif de ce cours)

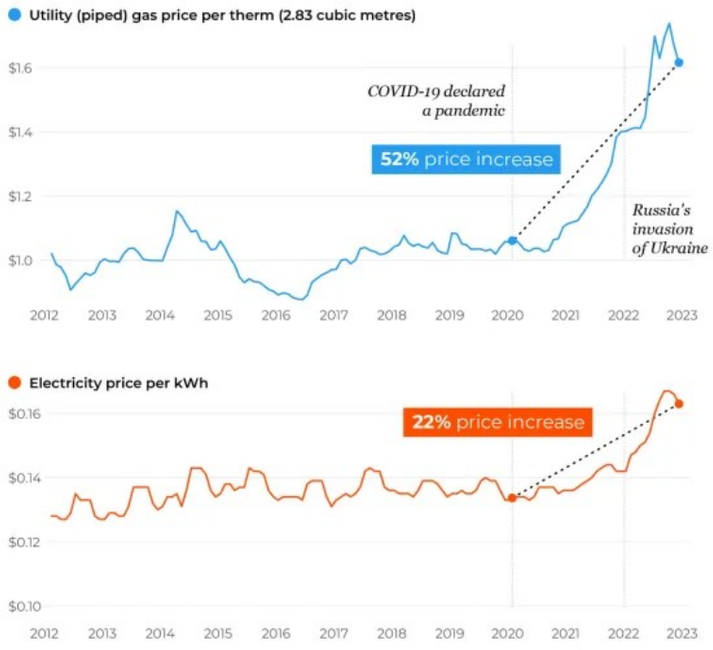
OBJECTIF

- Quelques exemples

ENERGY

Increasing energy prices

During the past three years, average utility company prices for electricity and gas in US cities have increased by almost 40%.



The average US home monthly consumption

NOV 2019 ENERGY BILL	
Gas	
75 Therms	$79.40
Electricity	
886 kWh	$117.80
Total	$197.20

1.4X

NOV 2022 ENERGY BILL	
Gas	
75 Therms	$121.20
Electricity	
886 kWh	$144.40
Total	$265.80

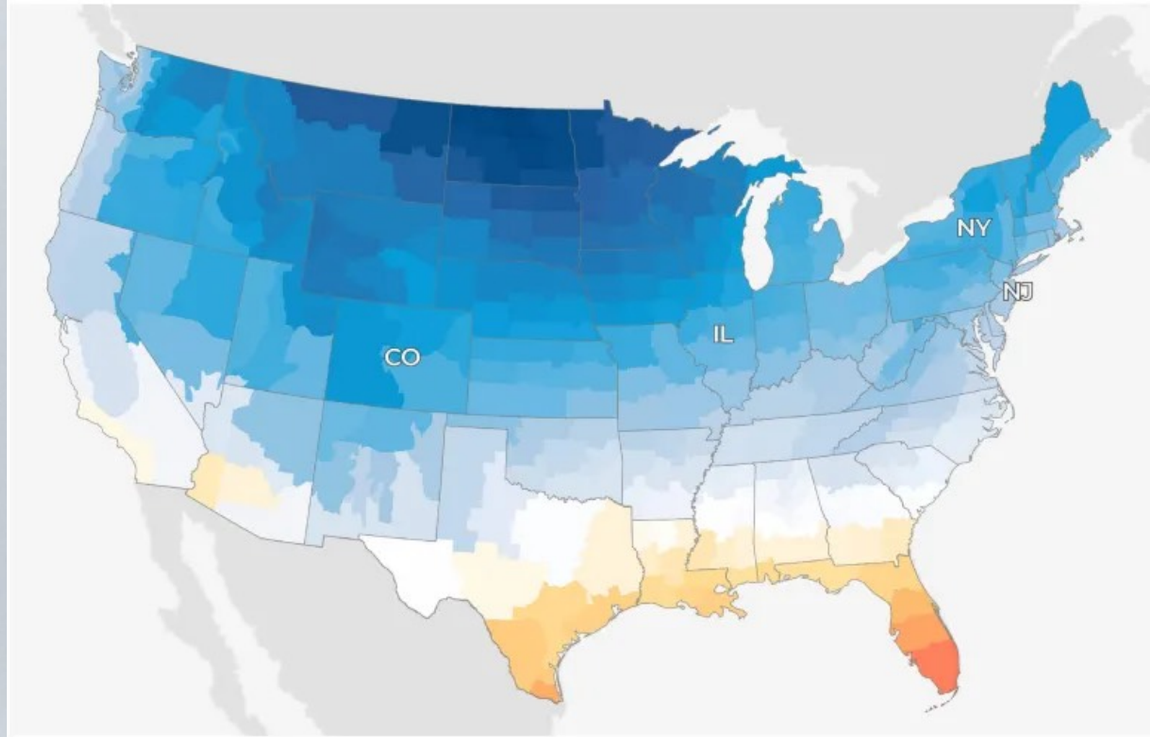
Source: United States Bureau of Labor Statistics, United States Energy Information Administration | January 27, 2023

@AJLabs ALJAZEERA

UNITED STATES

Average winter temperatures

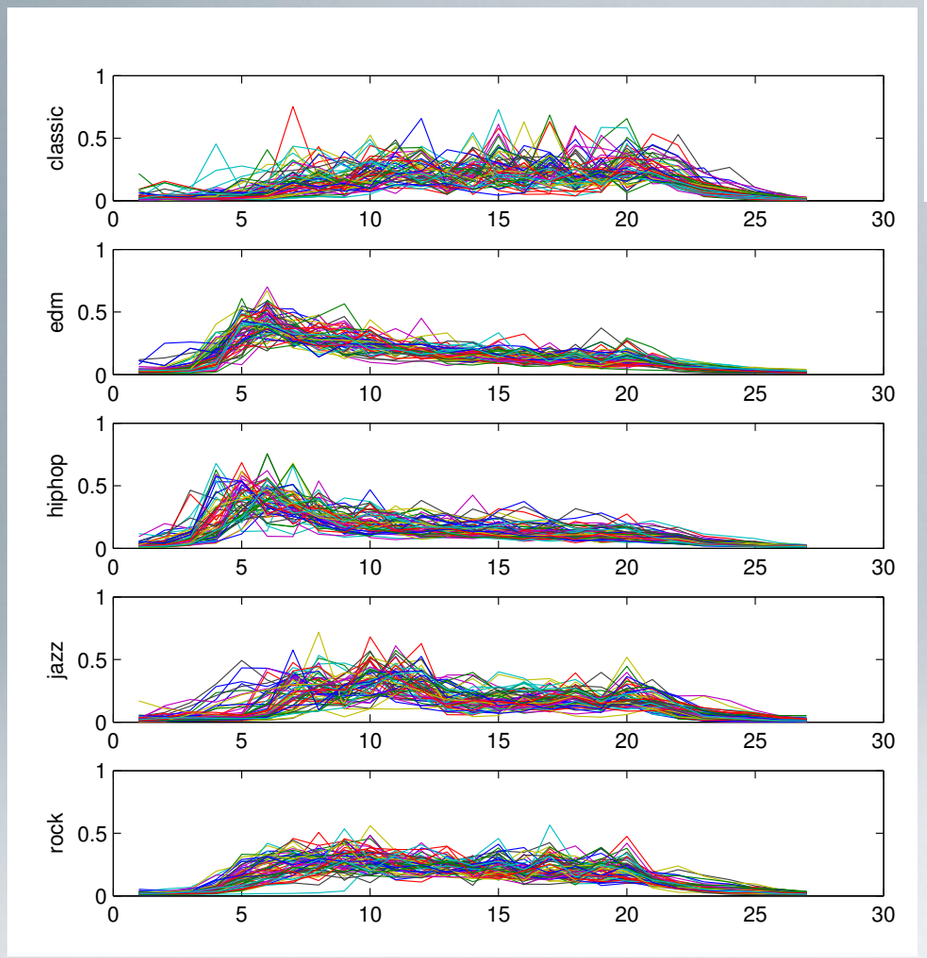
The average temperature across the US in December was 1C (33F), with many parts of the Midwest dropping below -17C (0F).

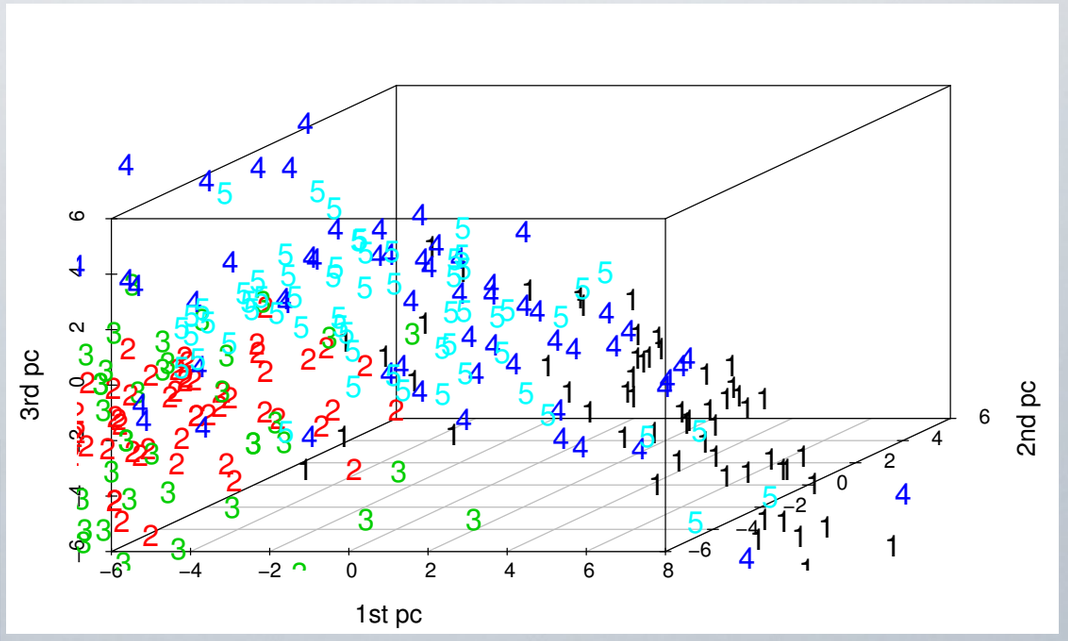


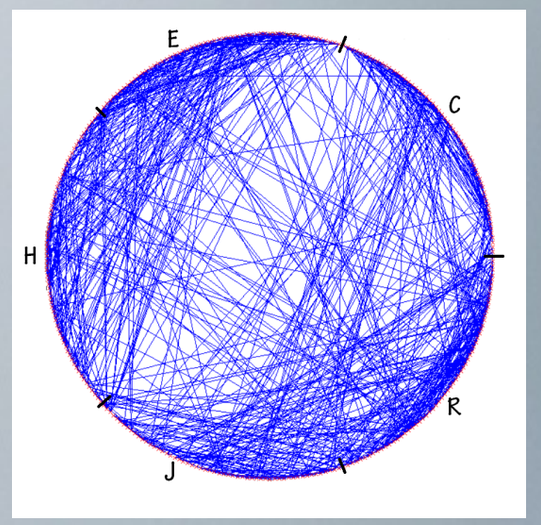
Source: Climate.gov, National Centers for Environmental Information | January 27, 2023

@AJLabs ALJAZEERA

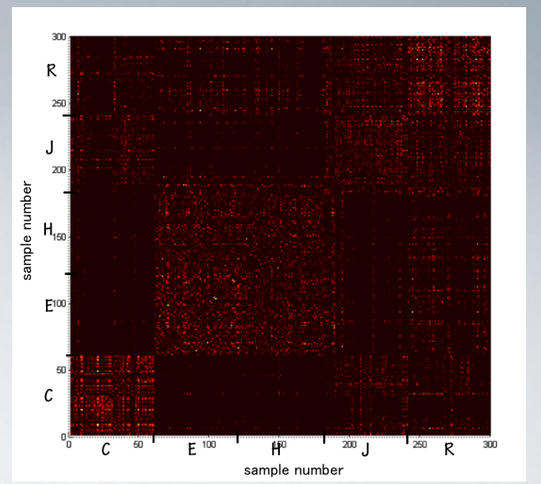
https://www.aljazeera.com/features/longform/2023/1/27/staying-warm-this-winter-how-cold-affects-those-most-vulnerable



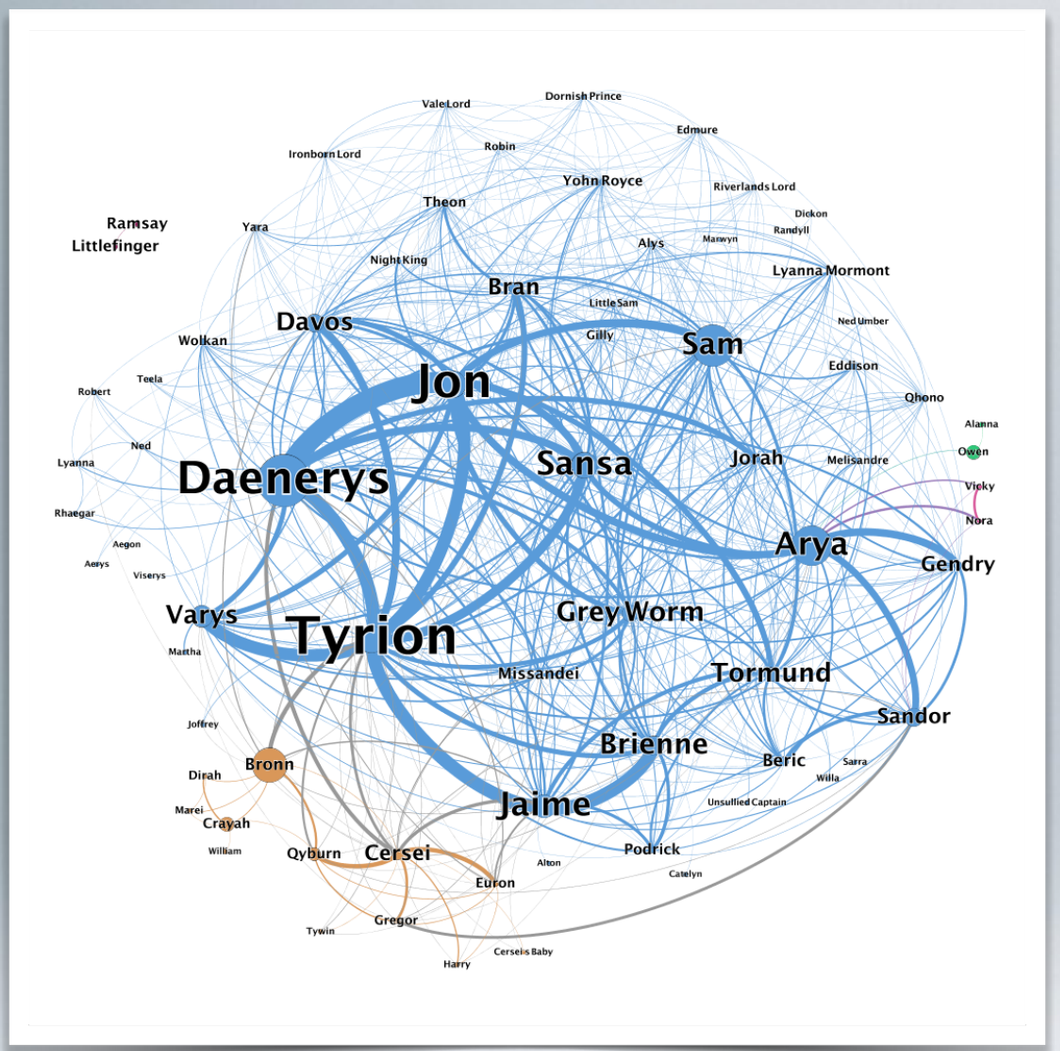


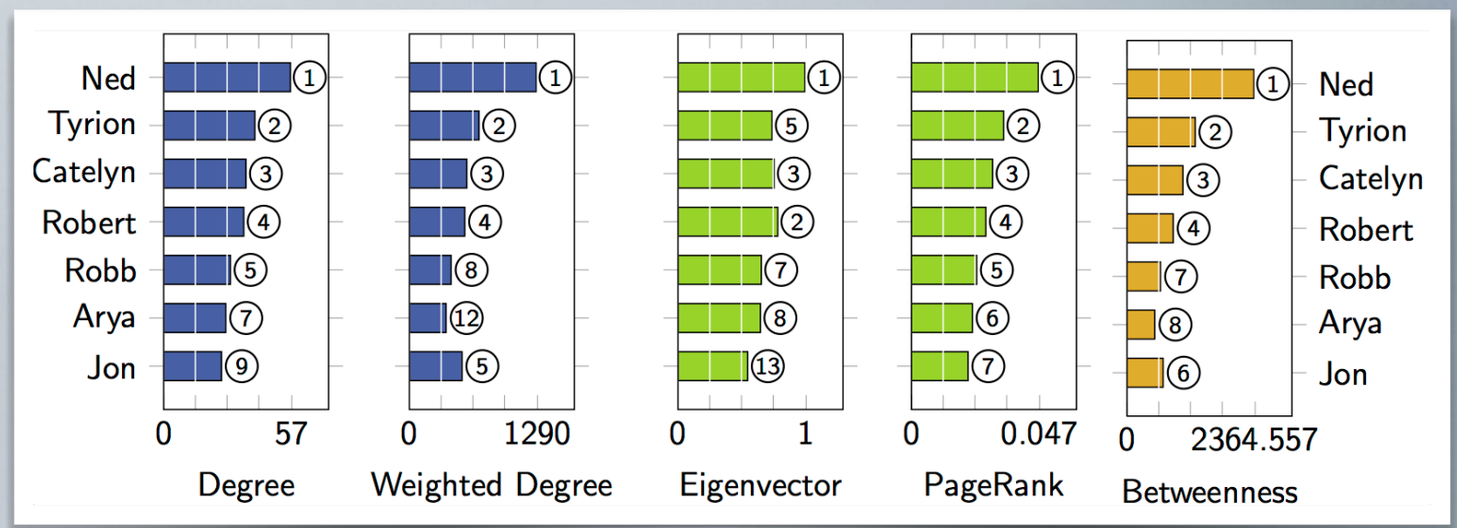


'igure 5. Graph showing the proximity between songs with 7 eighbors. Each red dot represents a song, each blue edge repre onnection between two neighbors. Abbreviations: C is classical, E i I is hip-hop, J is jazz, R is rock

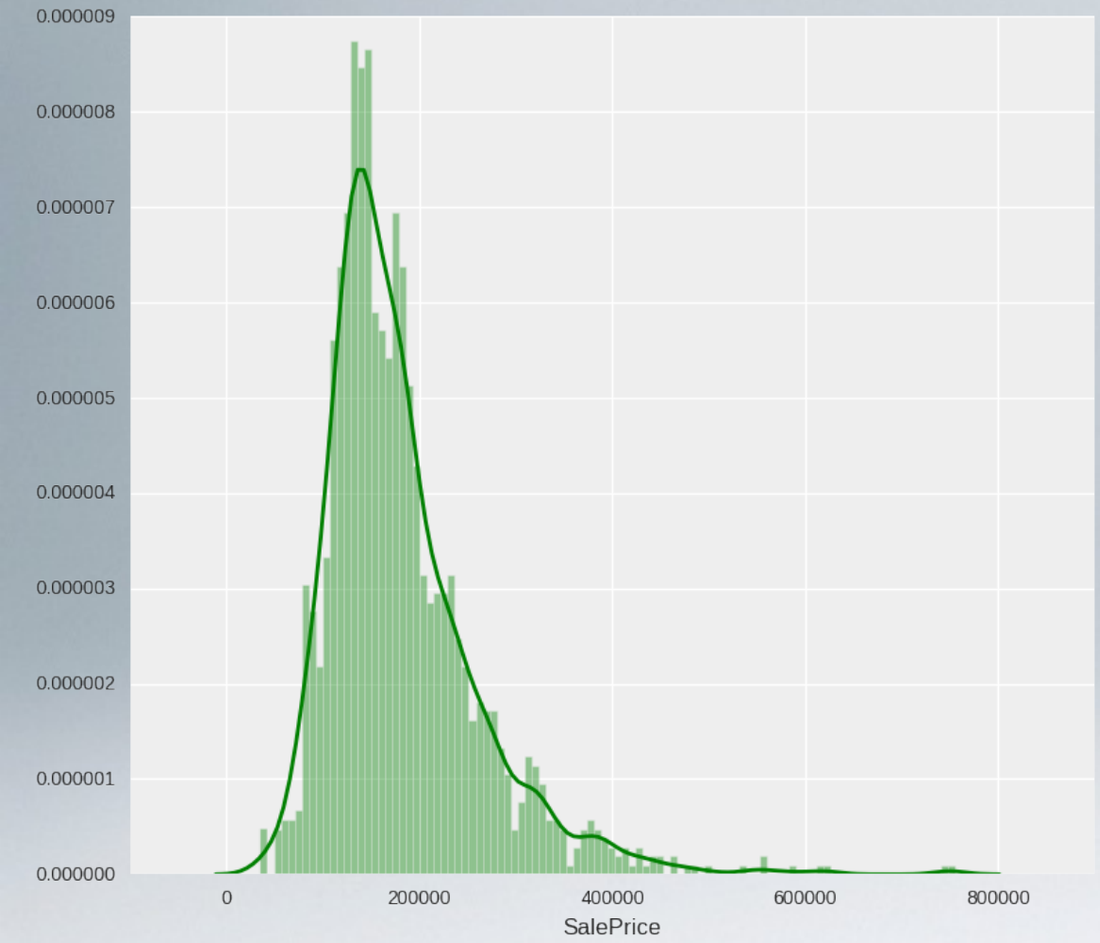


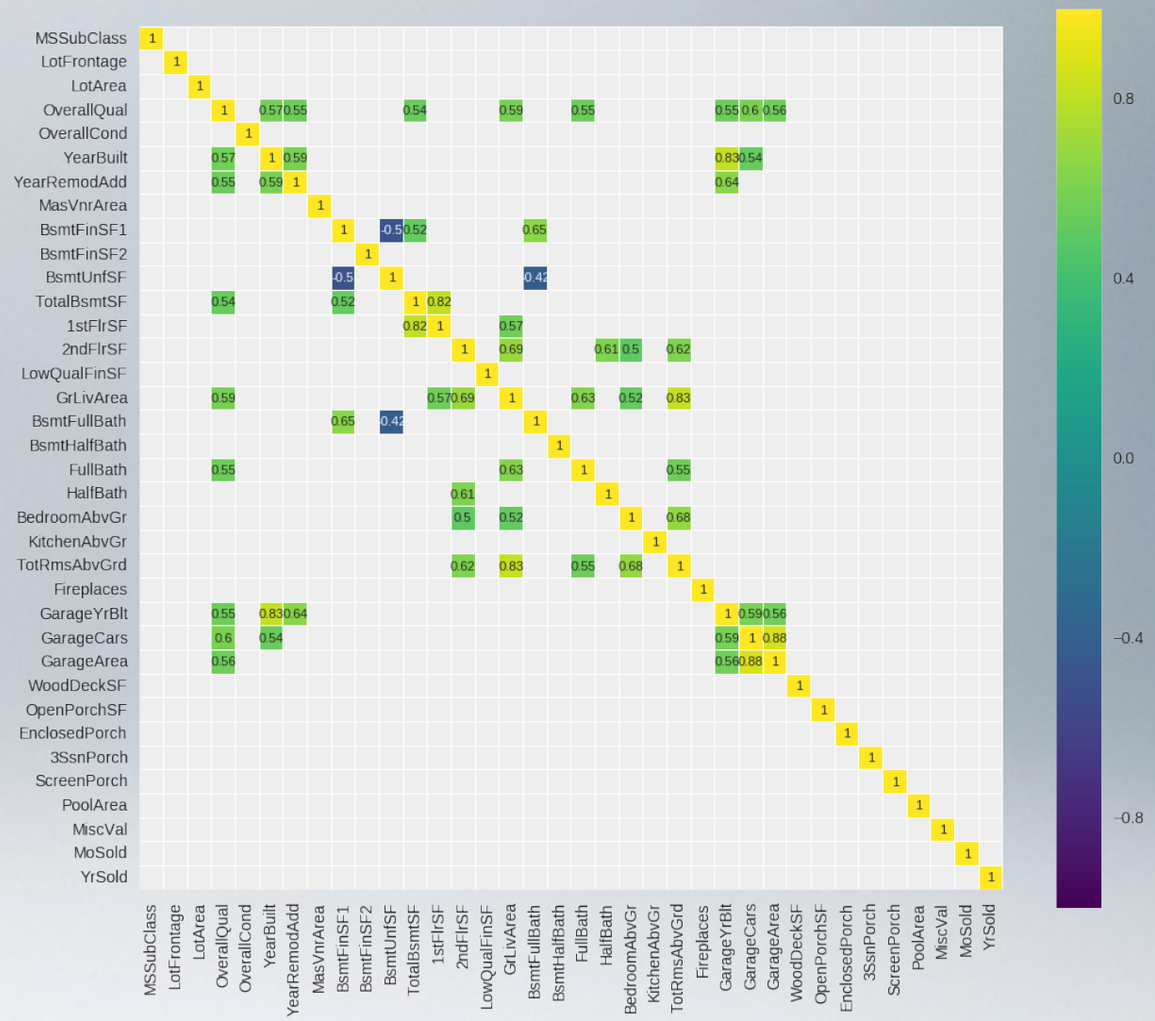
https://cs229.stanford.edu/proj2015/129_report.pdf





https://networkofthrones.wordpress.com/

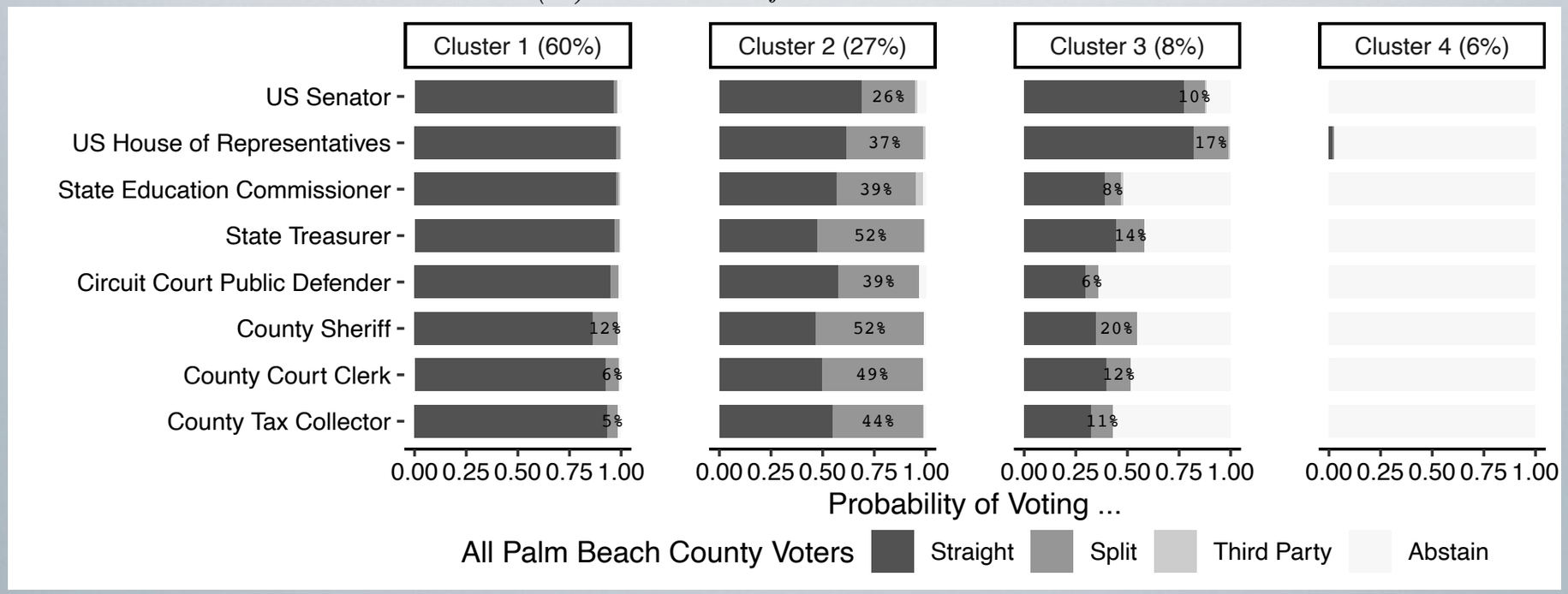




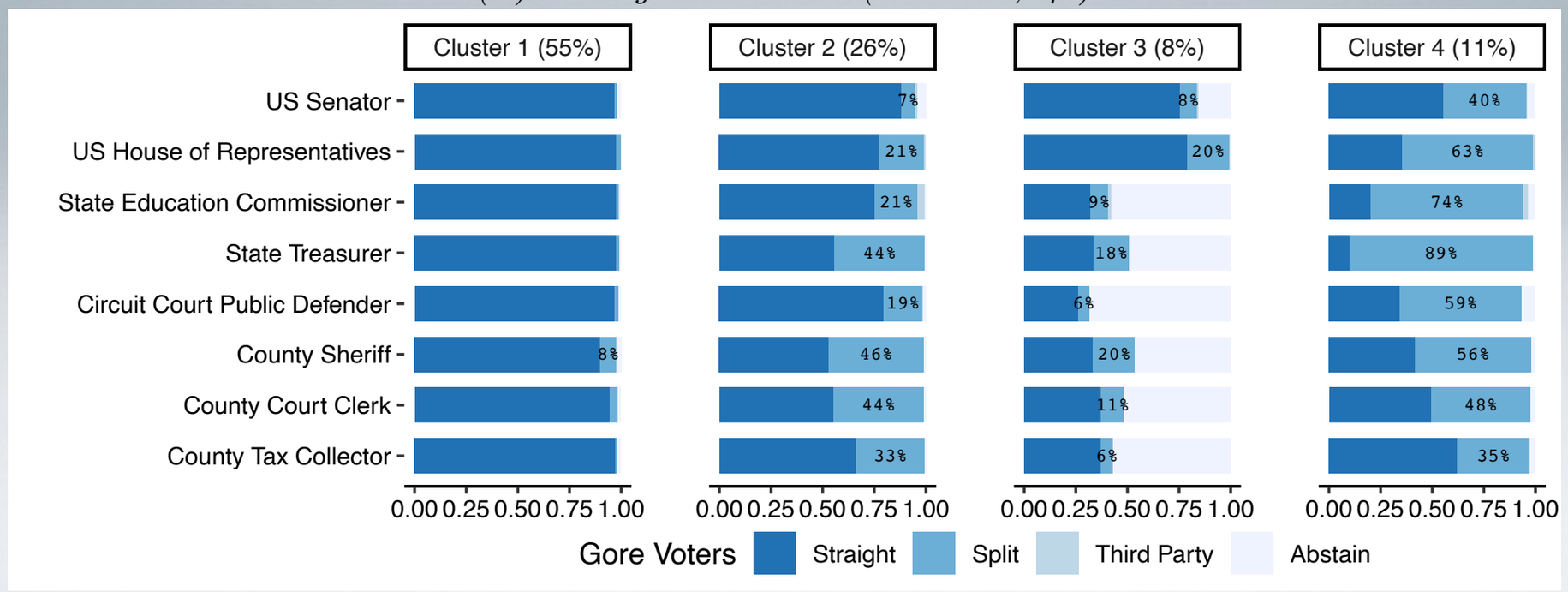
https://www.kaggle.com/code/ekami66/detailed-exploratory-data-analysis-with-python

Figure 1: Clusters of Voting Profiles in Palm Beach County Florida, 2000

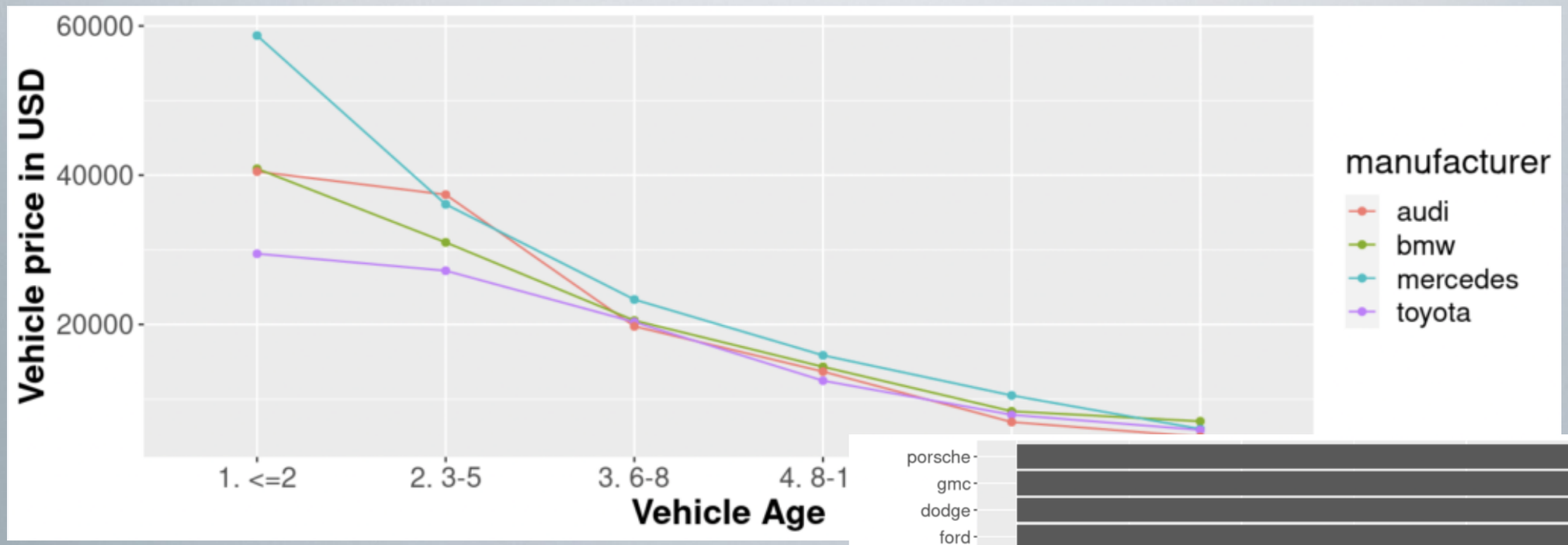
(A) All Voters for Gore or Bush

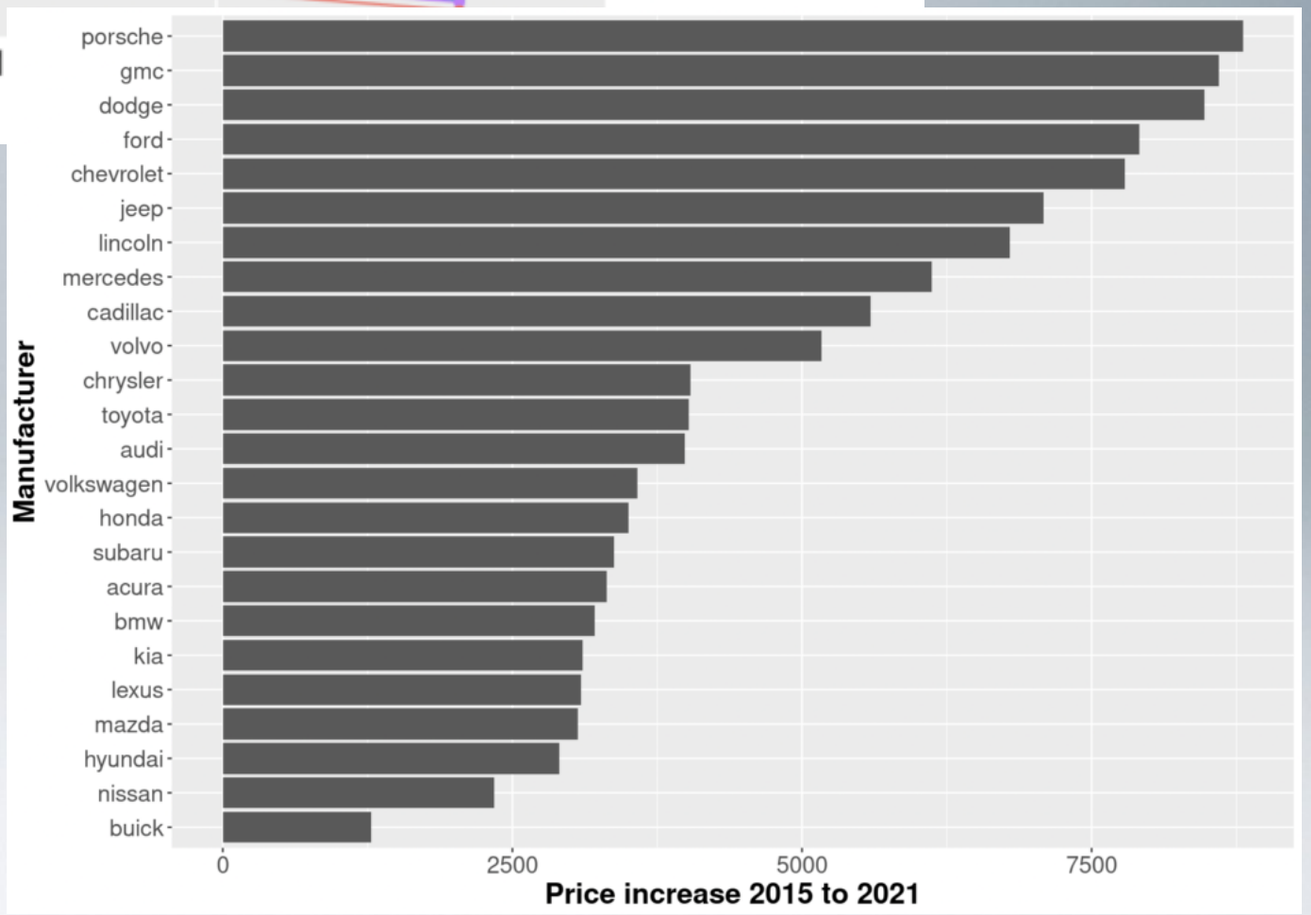


(B) Among Gore Voters (N = 210,640)



https://osf.io/v3rhz/download





https://nycdatascience.com/blog/student-works/data-analysis-on-car-pricing-and-its-factors/