Dimension reduction: PCA, LDA, tSNE

Diane Lingrand

UNIVERSITÉ CÔTE D'AZUR
Master Data Science M1

2020 - 2021

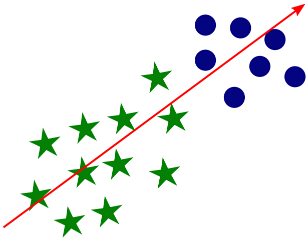
Outline

PCA (1901 Karl Pearson, 1936 H. Hotelling)

- Unsupervised
- Analysis of variance-covariance matrix
- Reducing the dimension of data
- Visualisation of data of the reduced dimension is 2 or 3
- Interpretation : dependance between variables
- PCA : often as pre-processing

Geometrical interpretation

- original variables : $X_1, X_2, .., X_p$
- principal components : $C_1, C_2, ...C_k, ..., C_q$ with $q \leq p$
- $C_k = \sum_j a_{jk} X_j$ with :
  - C_k and C_j not correlated
  - maximum variance and
  - decreasing importance



Variance decomposition

$$
\begin{aligned}
\sigma^2 &= \frac{1}{2n(n-1)} \sum_{i=1}^{n} \sum_{j=1}^{n} (\mathrm{x_i} - \mathrm{x_j})^{\mathsf{T}} (\mathrm{x_i} - \mathrm{x_j}) \\
&= \frac{1}{2n(n-1)} \sum_{i=1}^{n} \sum_{j=1}^{n} ((\mathrm{x_i} - \boldsymbol{\mu}) - (\mathrm{x_j} - \boldsymbol{\mu}))^{\mathsf{T}} ((\mathrm{x_i} - \boldsymbol{\mu}) - (\mathrm{x_j} - \boldsymbol{\mu})) \\
&= \frac{1}{2n(n-1)} \sum_{i=1}^{n} \sum_{j=1}^{n} ((\mathrm{x_i} - \boldsymbol{\mu})^{\mathsf{T}} (\mathrm{x_i} - \boldsymbol{\mu}) + (\mathrm{x_j} - \boldsymbol{\mu})^{\mathsf{T}} (\mathrm{x_j} - \boldsymbol{\mu}) \\
&\qquad - 2(\mathrm{x_i} - \boldsymbol{\mu})^{\mathsf{T}} (\mathrm{x_j} - \boldsymbol{\mu})) \\
&= \frac{1}{n-1} \sum_{i=1} (\mathrm{x_i} - \boldsymbol{\mu})^{\mathsf{T}} (\mathrm{x_i} - \boldsymbol{\mu})
\end{aligned}
$$

Projection on a line

- projection on a line directed by v : vv^T with constraint $v^T v = 1$
- variance of projected data :

$$
\begin{aligned}
\sigma_v^2 &= \frac{1}{n-1} \sum_{i=1} (vv^T(x_i - \boldsymbol{\mu}))^T (vv^T(x_i - \boldsymbol{\mu})) \\
&= \frac{1}{n-1} \sum_{i=1} (x_i - \boldsymbol{\mu})^T v \underbrace{v^T v}_{1} v^T (x_i - \boldsymbol{\mu}) \\
&= \frac{1}{n-1} \sum_{i=1} (x_i - \boldsymbol{\mu})^T vv^T (x_i - \boldsymbol{\mu}) \\
&= \frac{1}{n-1} \sum_{i=1} ((x_i - \boldsymbol{\mu})^T v)(v^T (x_i - \boldsymbol{\mu})) \\
&= \frac{1}{n-1} \sum_{i=1} (v^T (x_i - \boldsymbol{\mu}))((x_i - \boldsymbol{\mu})^T v) \\
&= \frac{1}{n-1} v^T \left[\sum_{i=1} (x_i - \boldsymbol{\mu})(x_i - \boldsymbol{\mu})^T \right] v \\
&= v^T \Sigma v
\end{aligned}
$$

- Σ : variance covariance matrix, positive definite (real eigenvalues)

Maximisation of projected variance

- max of $\sigma_v^2 = \mathrm{v}^\mathsf{T}\Sigma\mathrm{v}$
- constraint : $\mathrm{v}^T\mathrm{v} = 1$
- Lagrangian : $\mathcal{L} = \mathrm{v}^\mathsf{T}\Sigma\mathrm{v} + \lambda(1 - \mathrm{v}^T\mathrm{v})$
- max $\Rightarrow \frac{\partial\mathcal{L}}{\partial\mathrm{v}} = 0 \Rightarrow \Sigma\mathrm{v} = \lambda\mathrm{v}$
    - eigenvalues : λ
    - eigenvectors : v
    - variance : $\sigma_v^2 = \mathrm{v}^\mathsf{T}\Sigma\mathrm{v} = \mathrm{v}^T\lambda\mathrm{v} = \lambda$
    - highest variance : highest lambda value

Principal component

- eigenvalues, ordered - eigenvectors
- $tr(\Sigma) = \sigma^2 = \sum_{i=1}^{n} \lambda_i$
- each eigenvalue participates to the global variance

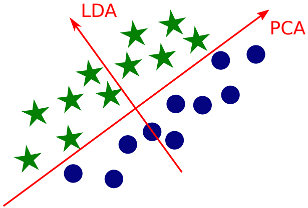
Examples

- PCA on Iris dataset :
  - from dimension 4 to dimension 3 for visualisation (`https://scikit-learn.org/stable/auto_examples/decomposition/plot_pca_iris.html`)
  - from dimension 4 to dimension 2 (`https://scikit-learn.org/stable/auto_examples/decomposition/plot_pca_vs_lda.html`)
  - explained variance ratio (first two components) : [0.92461872 0.05306648]
- PCA on MNIST : What will be the smallest dimension after PCA such that 95% of the variance is explained ?
  - answer : 153

Outline

LDA (1936, Sir Ronald Fisher ; 1948, R. C. Rao)

- context :
  - supervised
  - classification, q classes, n data, d dimension
- idea : find the factors (linear combination of components) that :
  - maximizes variance between classes
  - minimizes variance inside classes
- dimensionality reduction



Decomposition of variance

$$
\begin{aligned}
\sigma^2 &= \frac{1}{2n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\mathrm{x_i}-\mathrm{x_j})^{\mathsf{T}}(\mathrm{x_i}-\mathrm{x_j}) \\
&= \frac{1}{n}\sum_{i=1}^{n}(\mathrm{x_i}-\boldsymbol{\mu})^{\mathsf{T}}(\mathrm{x_i}-\boldsymbol{\mu}) \\
&= \frac{1}{n}\sum_{k=1}^{q}\sum_{i\in C_k}(\mathrm{x_i}-\boldsymbol{\mu})^{\mathsf{T}}(\mathrm{x_i}-\boldsymbol{\mu}) \\
&= \frac{1}{n}\sum_{k=1}^{q} SS(k)
\end{aligned}
$$

Decomposition of variance within/between class

$$
\begin{aligned}
SS(k) &= \sum_{i \in C_k} (\mathrm{x_i} - \boldsymbol{\mu})^{\mathsf{T}} (\mathrm{x_i} - \boldsymbol{\mu}) \\
&= \sum_{i \in C_k} (\mathrm{x_i} - \boldsymbol{\mu}(k) + \boldsymbol{\mu}(k) - \boldsymbol{\mu})^T (\mathrm{x_i} - \boldsymbol{\mu}(k) + \boldsymbol{\mu}(k) - \boldsymbol{\mu}) \\
&= \sum_{i \in C_k} \big(\|\mathrm{x_i} - \boldsymbol{\mu}(k)\|^2 + \|\boldsymbol{\mu}(k) - \boldsymbol{\mu}\|^2 \\
&\qquad\qquad + 2(\mathrm{x_i} - \boldsymbol{\mu}(k))^T (\boldsymbol{\mu}(k) - \boldsymbol{\mu}) \big) \\
&= \sum_{i \in C_k} \big(\|\mathrm{x_i} - \boldsymbol{\mu}(k)\|^2 + \|\boldsymbol{\mu}(k) - \boldsymbol{\mu}\|^2 \big) \\
&= \underbrace{\sum_{i \in C_k} \|\mathrm{x_i} - \boldsymbol{\mu}(k)\|^2}_{\textit{within}} + \underbrace{n(k) \|\boldsymbol{\mu}(k) - \boldsymbol{\mu}\|^2}_{\textit{between}}
\end{aligned}
$$

Total variance

$$
\begin{aligned}
\sigma^2 &= \frac{1}{n}\sum_{k=1}^{q}\left[\sum_{i\in C_k}\|\mathrm{x_i}-\boldsymbol{\mu}(k)\|^2 + n(k)\|\boldsymbol{\mu}(k)-\boldsymbol{\mu}\|^2\right] \\
&= \frac{1}{n}\sum_{k=1}^{q} n(k)\left[\frac{1}{n(k)}\sum_{i\in C_k}\|\mathrm{x_i}-\boldsymbol{\mu}(k)\|^2 + \|\boldsymbol{\mu}(k)-\boldsymbol{\mu}\|^2\right] \\
&= \frac{1}{n}\sum_{k=1}^{q} n(k)\left[\sigma_w^2(k)+\sigma_b^2(k)\right] \\
&= \sigma_w^2+\sigma_b^2
\end{aligned}
$$

Within class variance of projection on v

$$
\begin{aligned}
\sigma^2_{vw} &= \frac{1}{n}\sum_{k=1}^{q}\sum_{i\in C_k}(\mathrm{v}\mathrm{v}^{\mathrm{T}}\mathrm{x_i} - \mathrm{v}\mathrm{v}^{\mathrm{T}}\boldsymbol{\mu}(k))^T(\mathrm{v}\mathrm{v}^{\mathrm{T}}\mathrm{x_i} - \mathrm{v}\mathrm{v}^{\mathrm{T}}\boldsymbol{\mu}(k)) \\
&= \frac{1}{n}\sum_{k=1}^{q}\sum_{i\in C_k}(\mathrm{x_i} - \boldsymbol{\mu}(k))^T\mathrm{v}\mathrm{v}^{\mathrm{T}}\mathrm{v}\mathrm{v}^{\mathrm{T}}(\mathrm{x_i} - \boldsymbol{\mu}(k)) \\
&= \frac{1}{n}\sum_{k=1}^{q}\sum_{i\in C_k}\mathrm{v}^{\mathrm{T}}(\mathrm{x_i} - \boldsymbol{\mu}(k))(\mathrm{x_i} - \boldsymbol{\mu}(k))^T\mathrm{v} \\
&= \mathrm{v}^{\mathrm{T}}\left[\frac{1}{n}\sum_{k=1}^{q}\sum_{i\in C_k}(\mathrm{x_i} - \boldsymbol{\mu}(k))(\mathrm{x_i} - \boldsymbol{\mu}(k))^T\right]\mathrm{v} \\
&= \mathrm{v}^{\mathrm{T}}\left[\frac{1}{n}\sum_{k=1}^{q}n(k)\left(\frac{1}{n(k)}\sum_{i\in C_k}(\mathrm{x_i} - \boldsymbol{\mu}(k))(\mathrm{x_i} - \boldsymbol{\mu}(k))^T\right)\right]\mathrm{v} \\
&= \mathrm{v}^{\mathrm{T}}\mathrm{W}\mathrm{v}
\end{aligned}
$$

W represents the weighted mean of within class variance.

Between class variance of projections on v

$$
\begin{aligned}
\sigma^2_{vb} &= \frac{1}{n}\sum_{k=1}^{q}(\mathrm{v}\mathrm{v}^{\mathsf{T}}\boldsymbol{\mu}(k) - \mathrm{v}\mathrm{v}^{\mathsf{T}}\boldsymbol{\mu})^T(\mathrm{v}\mathrm{v}^{\mathsf{T}}\boldsymbol{\mu}(k) - \mathrm{v}\mathrm{v}^{\mathsf{T}}\boldsymbol{\mu}) \\
&= \mathrm{v}^T\left[\sum_{k=1}^{q}\frac{(\boldsymbol{\mu}(k) - \boldsymbol{\mu})^T(\boldsymbol{\mu}(k) - \boldsymbol{\mu})}{n}\right]\mathrm{v} \\
&= \mathrm{v}^{\mathsf{T}}\mathrm{B}\mathrm{v}
\end{aligned}
$$

B represents the variance of the barycenter of each class.

Total variance of projections on v

$$
\begin{aligned}
\sigma_v^2 &= \frac{1}{n}\sum_{i=1}^{n}(\mathrm{v}\mathrm{v}^\mathrm{T}\mathrm{x_i} - \mathrm{v}\mathrm{v}^\mathrm{T}\boldsymbol{\mu})^T(\mathrm{v}\mathrm{v}^\mathrm{T}\mathrm{x_i} - \mathrm{v}\mathrm{v}^\mathrm{T}\boldsymbol{\mu}) \\
&= \mathrm{v}^T\left[\sum_{i=1}^{n}\frac{(\mathrm{x_i}-\boldsymbol{\mu})^\mathrm{T}(\mathrm{x_i}-\boldsymbol{\mu})}{n}\right]\mathrm{v} \\
&= \mathrm{v}^\mathrm{T}\Sigma\mathrm{v}
\end{aligned}
$$

As previously seen :

$$\sigma_v^2 = \sigma_{vw}^2 + \sigma_{vb}^2 \Rightarrow 1 = \frac{\sigma_{vw}^2}{\sigma_v^2} + \frac{\sigma_{vb}^2}{\sigma_v^2}$$

and thus :

$$0 < \frac{\sigma_{vb}^2}{\sigma_v^2} = \frac{\mathrm{v}^\mathrm{T}\mathrm{B}\mathrm{v}}{\mathrm{v}^\mathrm{T}\Sigma\mathrm{v}} < 1$$

Maximisation

- We want to maximize $\frac{\sigma_{vb}^2}{\sigma_v^2} = \frac{\mathrm{v}^T\mathrm{B}\mathrm{v}}{\mathrm{v}^T\Sigma\mathrm{v}}$
- Derivation with respect to v :

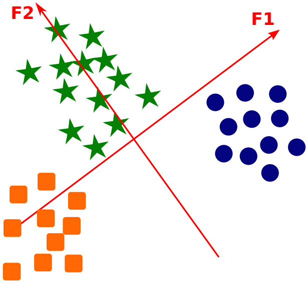
$$\frac{\partial\frac{\mathrm{v}^T\mathrm{B}\mathrm{v}}{\mathrm{v}^T\Sigma\mathrm{v}}}{\partial\mathrm{v}} = 0 \Rightarrow (\mathrm{v}^T\Sigma\mathrm{v})\mathrm{B}\mathrm{v} = (\mathrm{v}^T\mathrm{B}\mathrm{v})\Sigma\mathrm{v} \Rightarrow \mathrm{B}\mathrm{v} = \left(\frac{\mathrm{v}^T\mathrm{B}\mathrm{v}}{\mathrm{v}^T\Sigma\mathrm{v}}\right)\Sigma\mathrm{v}$$

- Let $\lambda = \frac{\mathrm{v}^T\mathrm{B}\mathrm{v}}{\mathrm{v}^T\Sigma\mathrm{v}}$. Thus : $\Sigma^{-1}\mathrm{B}\mathrm{v} = \lambda\mathrm{v}$ Eigenvalues !
- Ordered eigenvalues (decreasing) are related to the eigenvectors defining a new space for the classification. In this new space, a new data is labelled according to the closest class barycenter (Euclidean distance)
- In original space : a new data is labelled according to the closest class barycenter using the Mahalanobis distance :

$$d(\mathrm{x}, \boldsymbol{\mu}(k)) = (\mathrm{x} - \boldsymbol{\mu}(k))^T \mathrm{W}^{-1} (\mathrm{x} - \boldsymbol{\mu}(k))$$

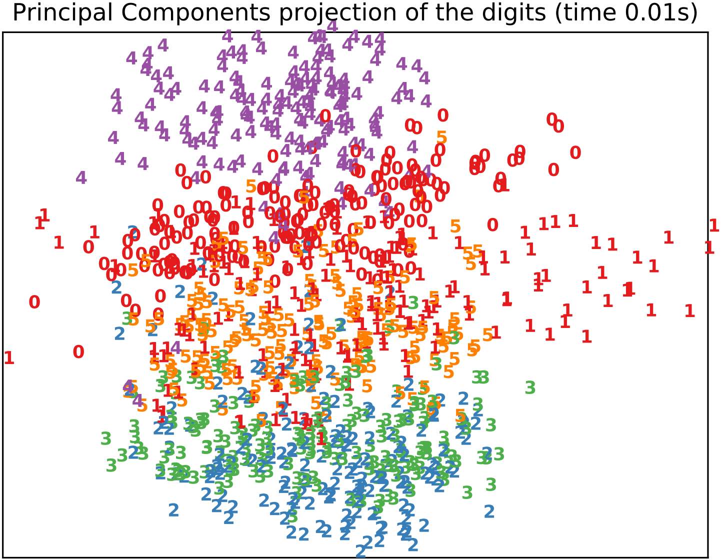
Algorithm

- center the data : replace each x_i by $x_i - \boldsymbol{\mu}$
- compute the variance-covariance matrix Σ
- compute the between class variance matrix B
- diagonalise $\Sigma^{-1}B$ and order eigenvalues



Properties of LDA

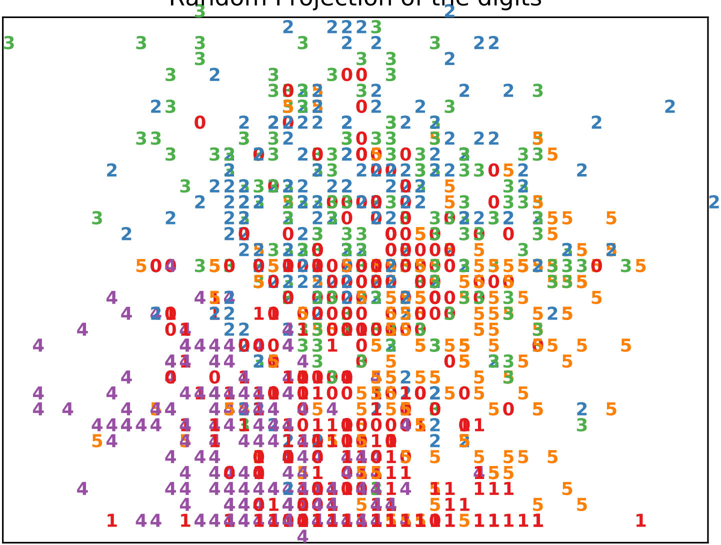
- at most (q-1) non zero eigenvalues
- the amount of between class variance is decreasing with eigenvalues
- LDA = PCA of class barycenters weighted by the size of classes, with a Σ^{-1} metric
- drawbacks :
  - if "shape" of classes are not similar (different dispersion) : the metric W^{-1} is computed on the whole data
  - a class is represented by its barycenter : what if the barycenter is not representative ?


Principal Components projection of the digits (time 0.01s)

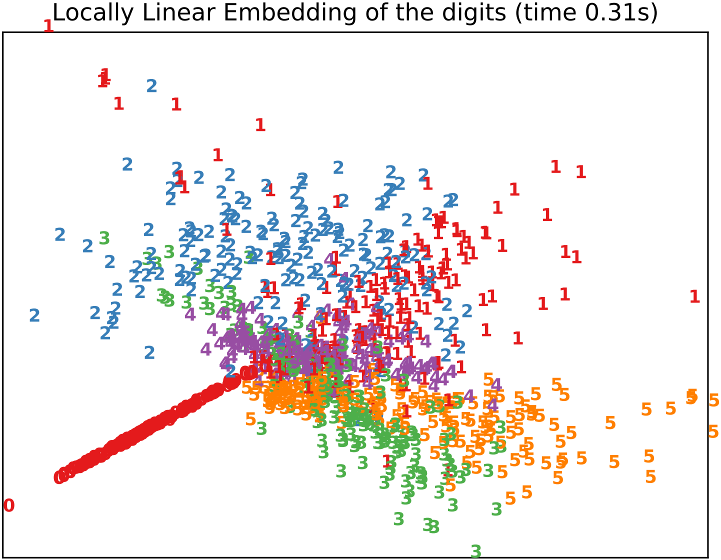
Outline

Random projection

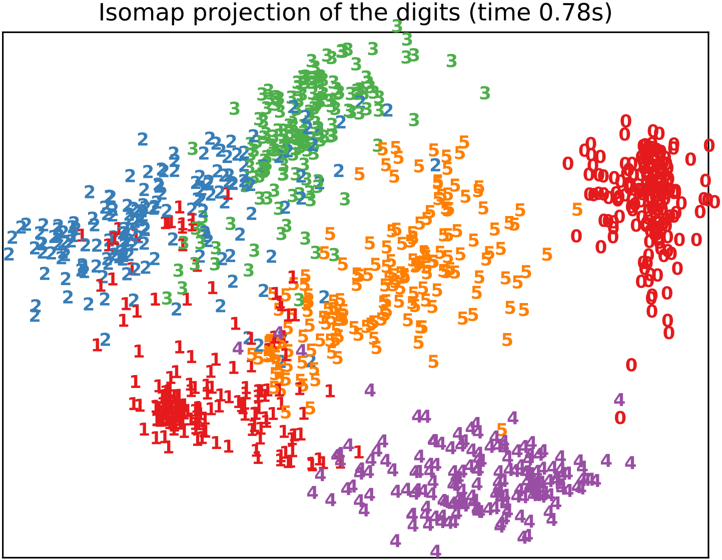
Random Projection of the digits



LLE (Locally Linear Embedding)



IsoMap



Idea of t-SNE

- Build map in which distances between points reflect similarities in the data
  - typical map dimension : 2 or 3
  - preserving local structures
  - previous approaches : IsoMap, LLE (Locally Linear Embedding)
  - t-SNE : try to avoid all points collapsing
- Non linear dimension reduction
  - converts affinities of data points to probabilities represented by Gaussian joint probabilities
  - affinities in the embedded space are represented by Student's t-distributions (heavy tailed)
  - minimisation of Kullback-Leibler divergence of the two distributions (gradient descent) : gives the coordinates in the embedded space
- Exact algorithm of tSNE is computationally expensive (huge compared to PCA)
- Stochastic algorithm : multiple restarts with different seeds can yield different results

Similarities between points in the original space

- point of reference : p_i
  - fit a gaussian locally to this point and examine neighbors p_j
  - compute similarities for points belonging to the neighborhood
  - measure the density of all these points and normalize

$$p_{ij} = \frac{\exp -\|x_i - x_j\|^2 / 2\sigma^2}{\sum_k \sum_{l \neq k} \exp -\|x_k - x_l\|^2 / 2\sigma^2}$$

  - high probability for a pair of points (i, j) if they are similar
- in practice
  - normalization is done only on pairs of points involving p_i

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

  - with a bandwidth σ_i : fixed perplexity (a fixed number of points fall in mode of this Gaussian). This allows to adapt to different region with different densities
  - symmetry

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

Similarities between points in the embedded space

- point of reference : p_i
  - fit a distribution : Student t-test with one dof
  - similarity between two points p_i and p_j in the low dim. space

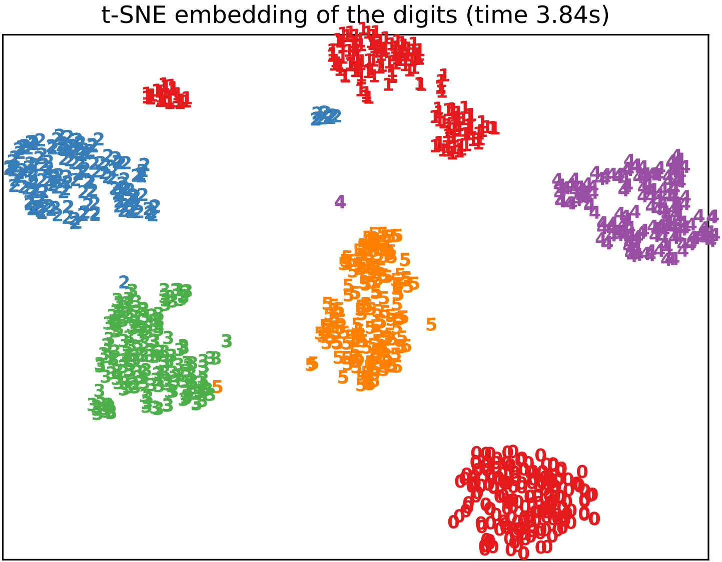
$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_k \sum_{l \neq k} (1 + \|y_k - y_l\|^2)^{-1}}$$

- goal of t-SNE : p_{ij} and q_{ij} as identical as possible
  - so that the structure of the map is similar to the structure of the data
- Kullback-Leibler :

$$KL(P||Q) = \sum_i \sum_{j \neq i} p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

  - initially : random points
  - move points in order to minimize KL
  - KL preserves local structures
    - if large p_{ij} value : KL forces q_{ij} to be large also (otherwise, high penalty)
    - if small p_{ij} value : small penalty

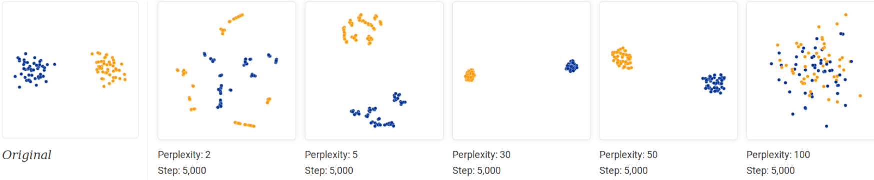
Example using the 64 D digit dataset



Short intro to t-SNE

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

- Perplexity (usually between 5 and 50) from `https://distill.pub/2016/misread-tsne/`



- Early exaggeration factor : optimization in two steps :
  - exaggeration phase : joint probabilities in the original space are artificially multiplied by a factor
  - final optimization
- Learning rate ϵ : not too small, not too large.
- Maximum number of iterations : 5000 ?
- angle (not used in the exact method)

KL Gradient interpretation

$$\sum_{j \neq i} (p_{ij} - q_{ij})(1 + \|y_i - y_j\|^2)^{-1}(y_i - y_j)$$

- N-body simulation
  - spring
  - exertion / compression

Barnes-Hut t-SNE

- approximation of t-SNE, more scalable.
  - many of the pairwise interactions between points are similar
- Another parameter : angle :
  - tradeoff between performance and accuracy
  - usual range : from 0.2 to 0.8
    - larger angles imply that we can approximate larger regions by a single point, leading to better speed but less accurate results.
- Limitations :
  - target dimension less than 3. Mostly 2.
  - only for dense dataset (for sparse dataset use exact t-SNE)