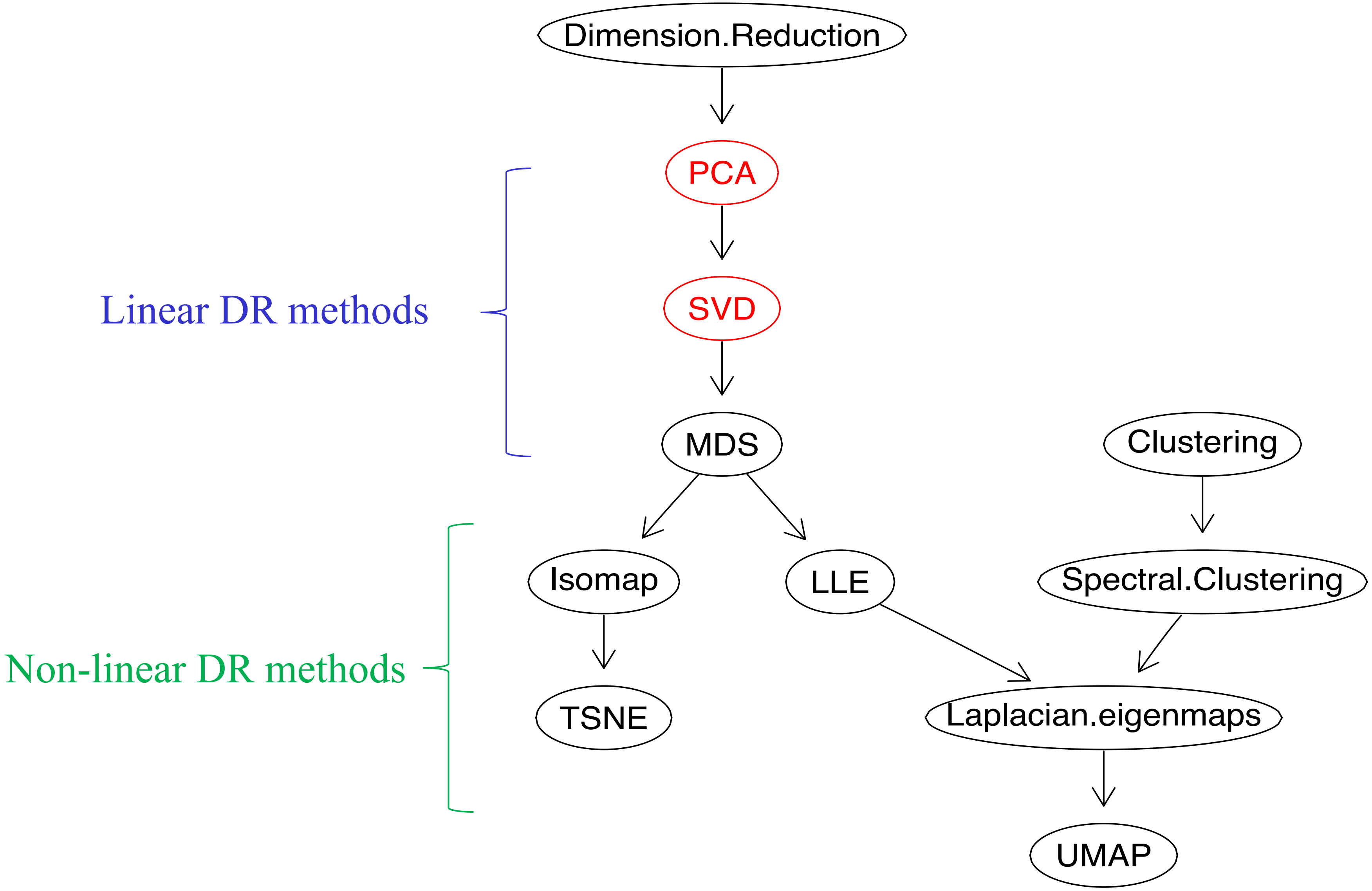
Dimension Reduction Methods:
From PCA to TSNE and UMAP

Maxwell Lee

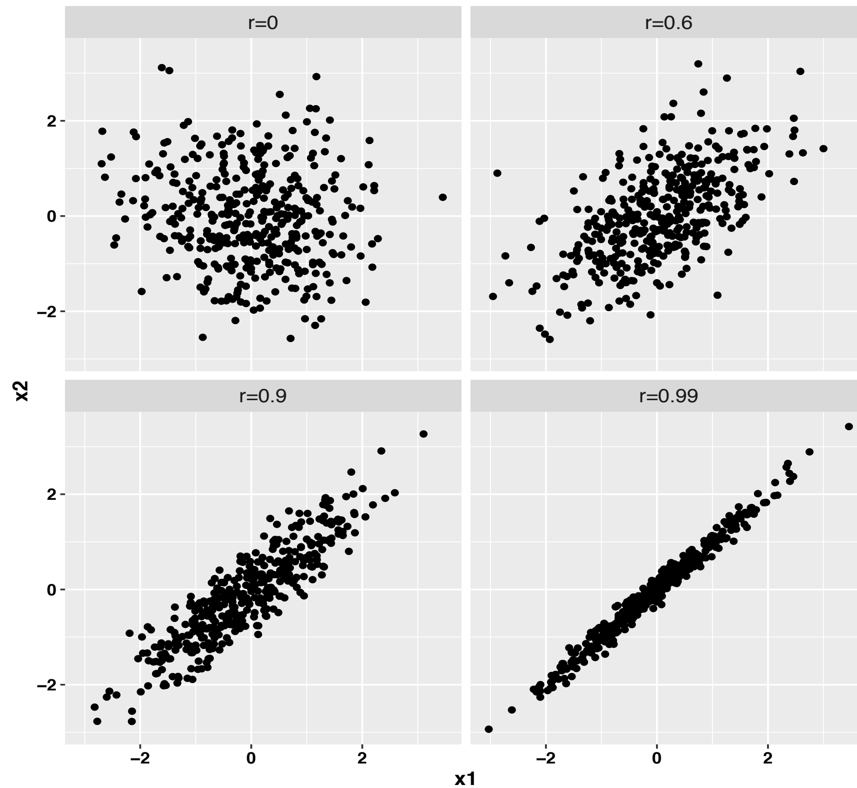
High-dimension Data Analysis Group
Laboratory of Cancer Biology and Genetics
Center for Cancer Research
National Cancer Institute

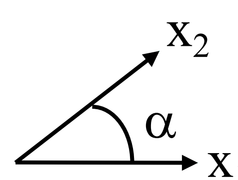
April 30, 2020

Road Map for Dimension Reduction Methods



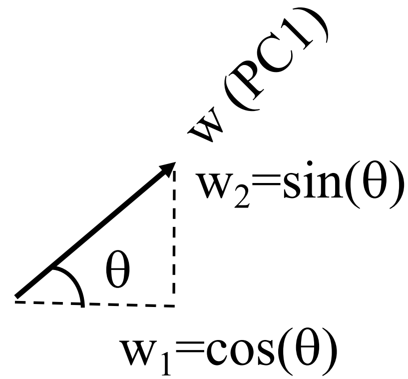
Three Dimensional Object and Its Two Dimensional Image
Ellipse as Shadow of Sphere

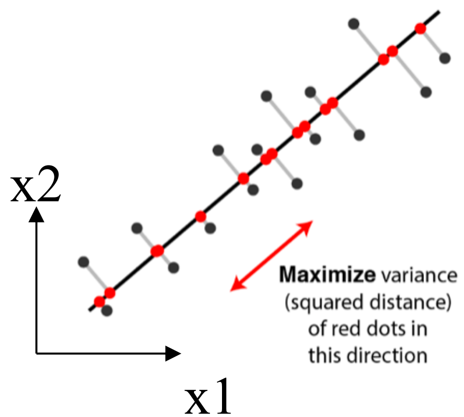




$$r = \cos(\alpha)$$

Algorithm of PCA:
How Does PCA Find the Direction of Its Principal Component PC1?





X: data matrix

X^T: transposition

S: X^TX

w: direction vector

z: principal component

λ: eigen value

$$z = Xw$$

$$\mathrm{var}(z) = w^TSw$$

Choose w to maximize w^TSw

subject to $w^Tw = 1$

$$Sw = \lambda w$$

w is the eigen vector and λ is eigen value

Singular Value Decomposition (SVD)

p column vectors
n row vectors

$$\begin{bmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ . & & & . \\ . & & & . \\ x_{n1} & x_{n2} & \dots & x_{np} \end{bmatrix} = X$$

$$Z = XW$$
$$Z_s = XW\Lambda^{-1/2}$$
$$X = Z_s\Lambda^{1/2}W^T$$

$$X = U\Sigma V^T$$

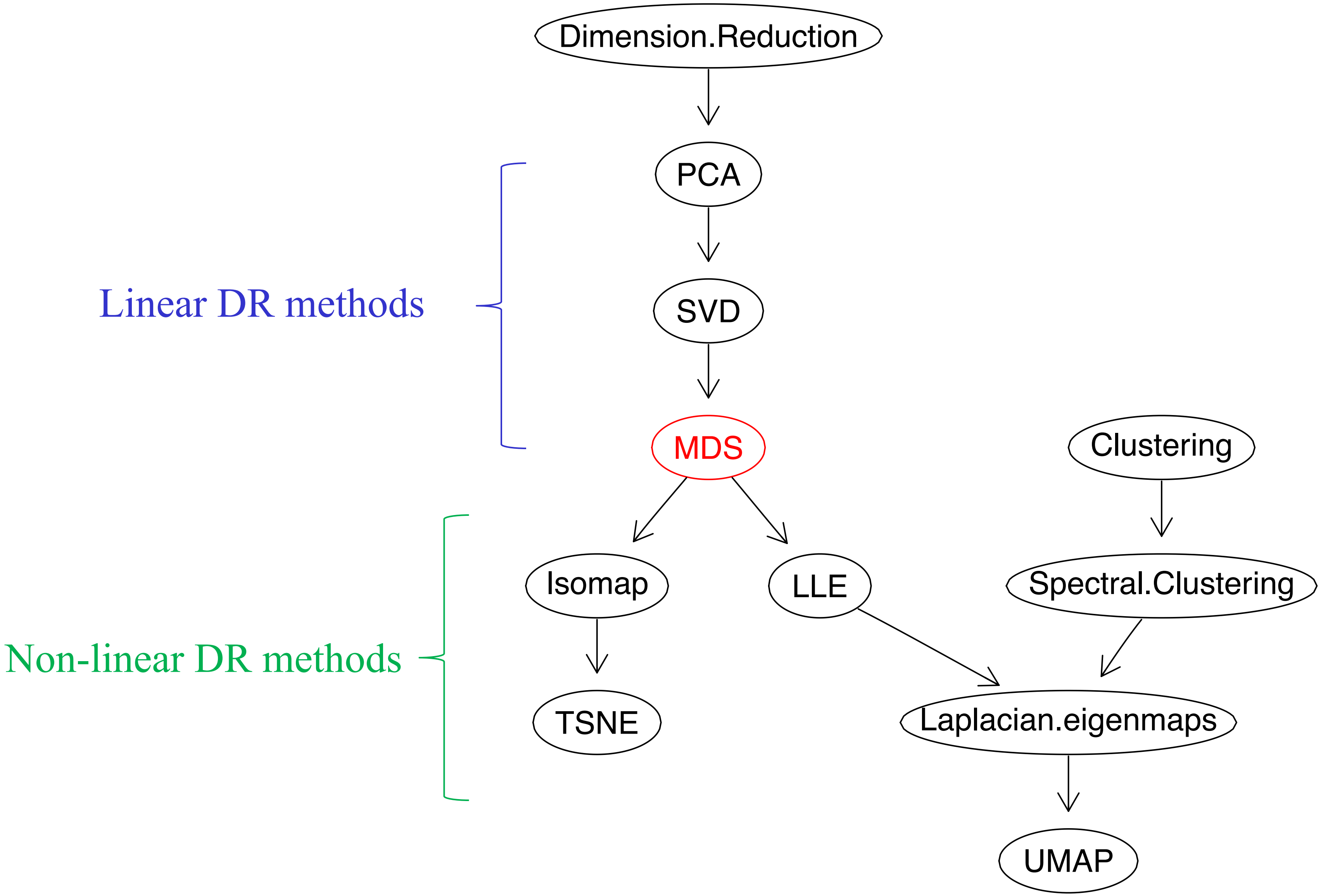
X: data matrix
Z: principal component
W: Eigen vector
Λ: Eigen value
V: right singular vector
U: left singular vector
Σ: singular value
X^TX: covariance matrix
XX^T: Gram matrix or kernel matrix

$$X^TX = V\Sigma^2V^T$$
$$XX^T = U\Sigma^2U^T$$

$$Z = U\Lambda^{1/2}$$

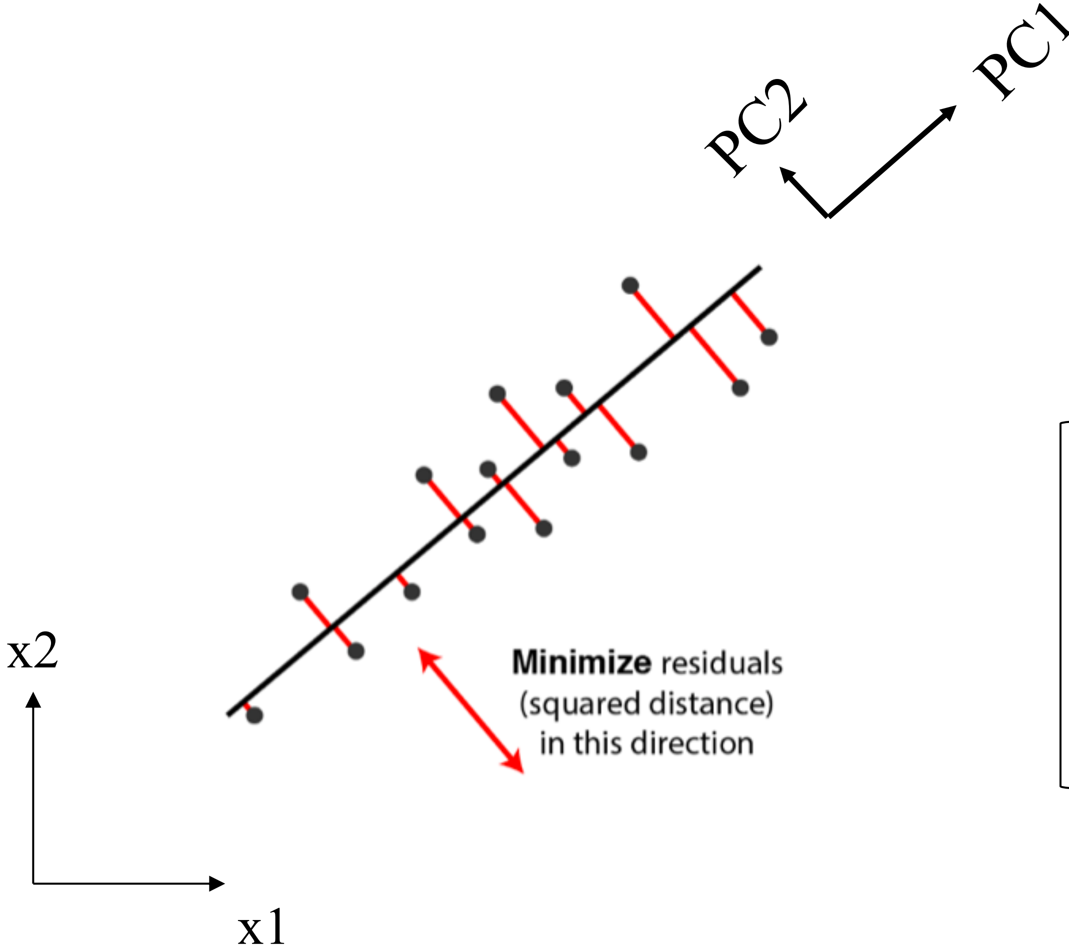
Eugenio Beltrami and Camille Jordan, 1873-1874

Road Map for Dimension Reduction Methods



Globe Versus Google Map

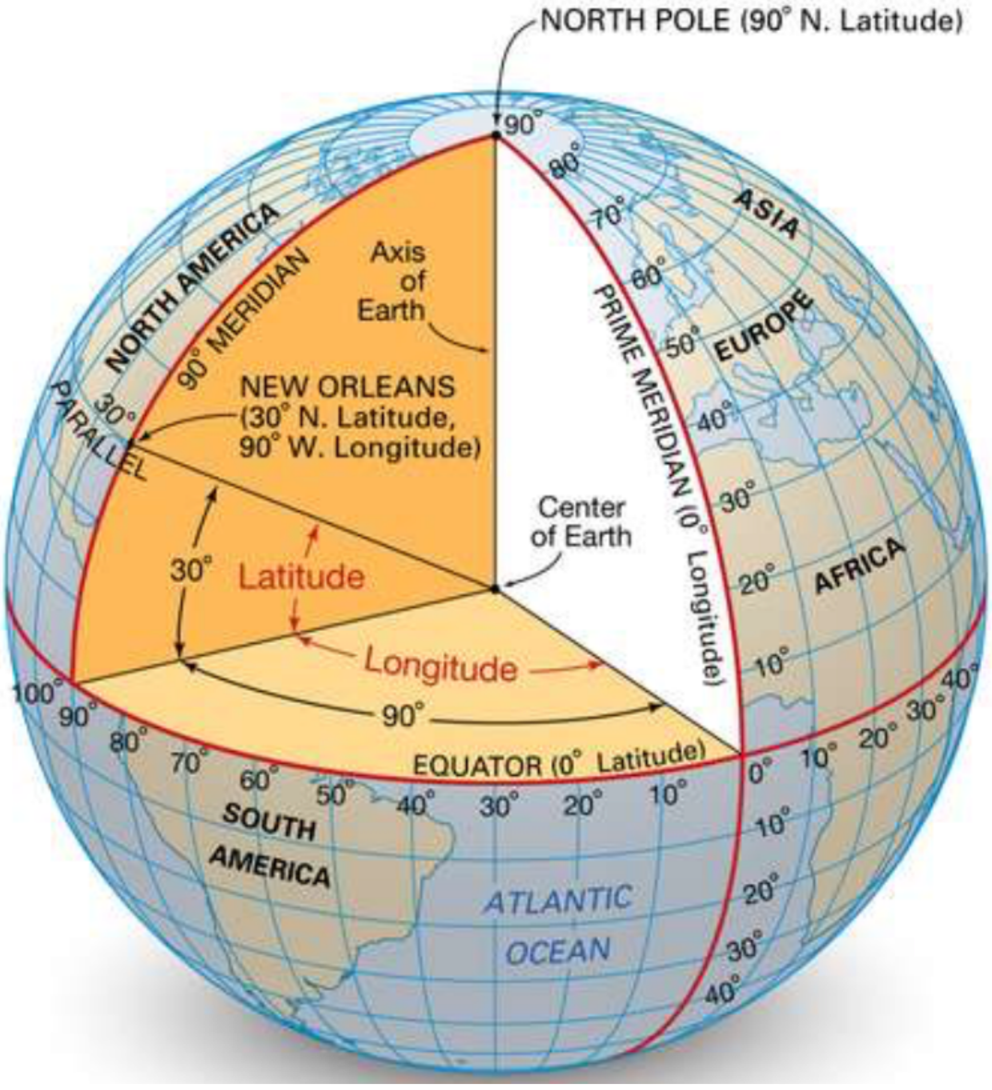
Principal Component Analysis (PCA)



$$\begin{bmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ . & & & . \\ . & & & . \\ x_{n1} & x_{n2} & \dots & x_{np} \end{bmatrix}$$

Karl Pearson 1901

3D Map Data



NORTH POLE (90° N. Latitude)
NORTH AMERICA
90° MERIDIAN
Axis of Earth
ASIA
EUROPE
PRIME MERIDIAN (0° Longitude)
NEW ORLEANS (30° N. Latitude, 90° W. Longitude)
30° PARALLEL
Center of Earth
30°
Latitude
Longitude
90°
AFRICA
EQUATOR (0° Latitude)
SOUTH AMERICA
ATLANTIC OCEAN

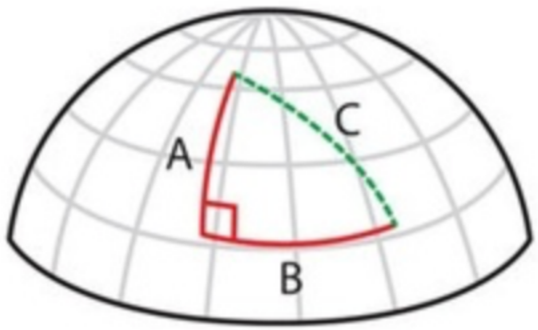
Eight European Cities Pairwise Distance Matrix

	Athens	Berlin	Dublin	London	Madrid	Paris	Rome	Warsaw
Athens	0	1119	1777	1486	1475	1303	646	1013
Berlin	1119	0	817	577	1159	545	736	327
Dublin	1777	817	0	291	906	489	1182	1135
London	1486	577	291	0	783	213	897	904
Madrid	1475	1159	906	783	0	652	856	1483
Paris	1303	545	489	213	652	0	694	859
Rome	646	736	1182	897	856	694	0	839
Warsaw	1013	327	1135	904	1483	859	839	0

Triangle on a Curved Surface vs on a Plane

Non-Euclidean

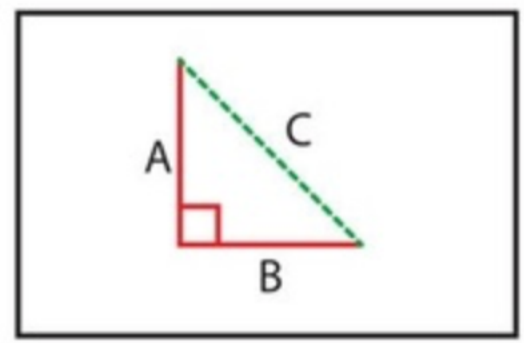
$A^2 + B^2 > C^2$



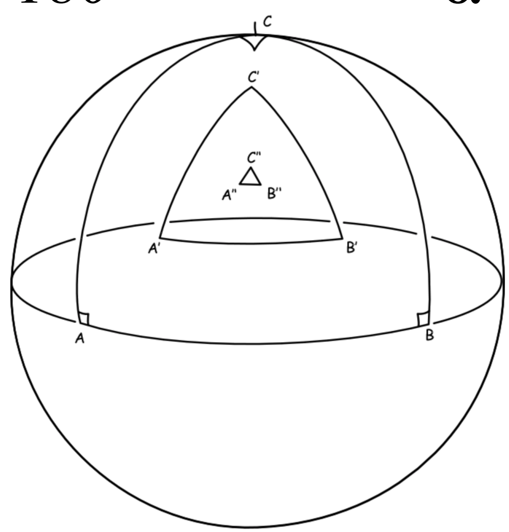
$\alpha + \beta + \gamma > 180^0$

Euclidean

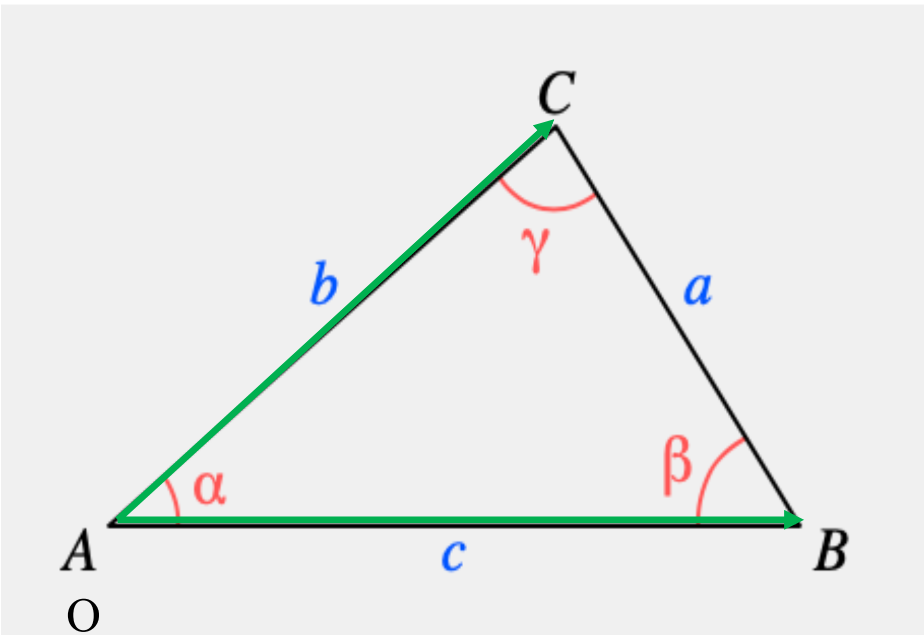
$A^2 + B^2 = C^2$



$\alpha + \beta + \gamma = 180^0$



The Dot Product of Two Vectors is the Difference Between the Squared Distances (Law of Cosines)



$$a^2 = b^2 + c^2 - 2bc\cos(\alpha)$$

$$bc\cos(\alpha) = -½(a^2 - b^2 + c^2)$$

$$\mathbf{b \cdot c} = -1/2(a^2 - b^2 - c^2)$$

Warren Torgerson in 1958

Eigen Decomposition of Gram Matrix (Similarity Matrix)

$\mathbf{b \cdot c} = -1/2(a^2 - b^2 - c^2)$

$$G = \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1n} \\ g_{21} & g_{22} & \cdots & g_{2n} \\ . & & & . \\ . & & & . \\ g_{n1} & g_{n2} & \cdots & g_{nn} \end{bmatrix}$$

g_{ij} is dot product between element i and j which captures similarity or relatedness

$$G = XX^T$$

$$G = U\Lambda U^T$$

$$Z = U\Lambda^{1/2}$$

$$Z = XV$$

$$Z_s = XV\Lambda^{-1/2}$$

$$U\Lambda^{1/2} = Z$$

X: X_{np}

G: Gram matrix or kernel matrix

U: Eigen vector

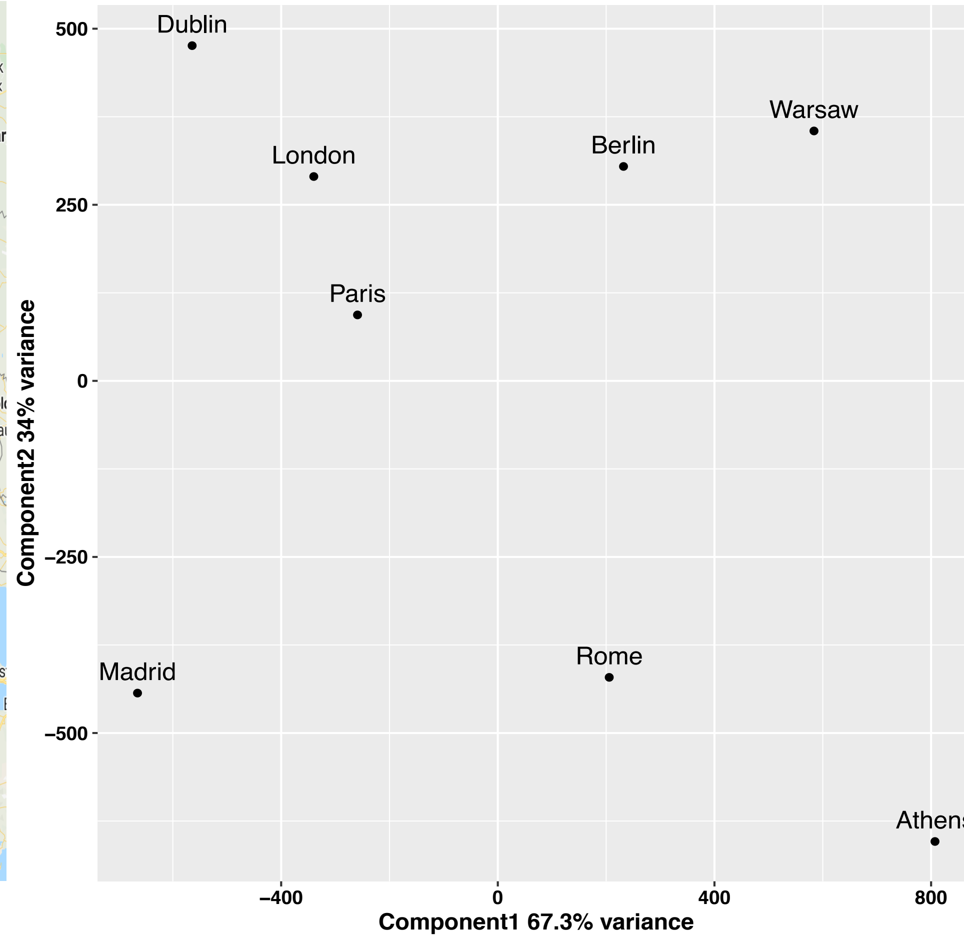
Λ: Eigen value

Z: principal component

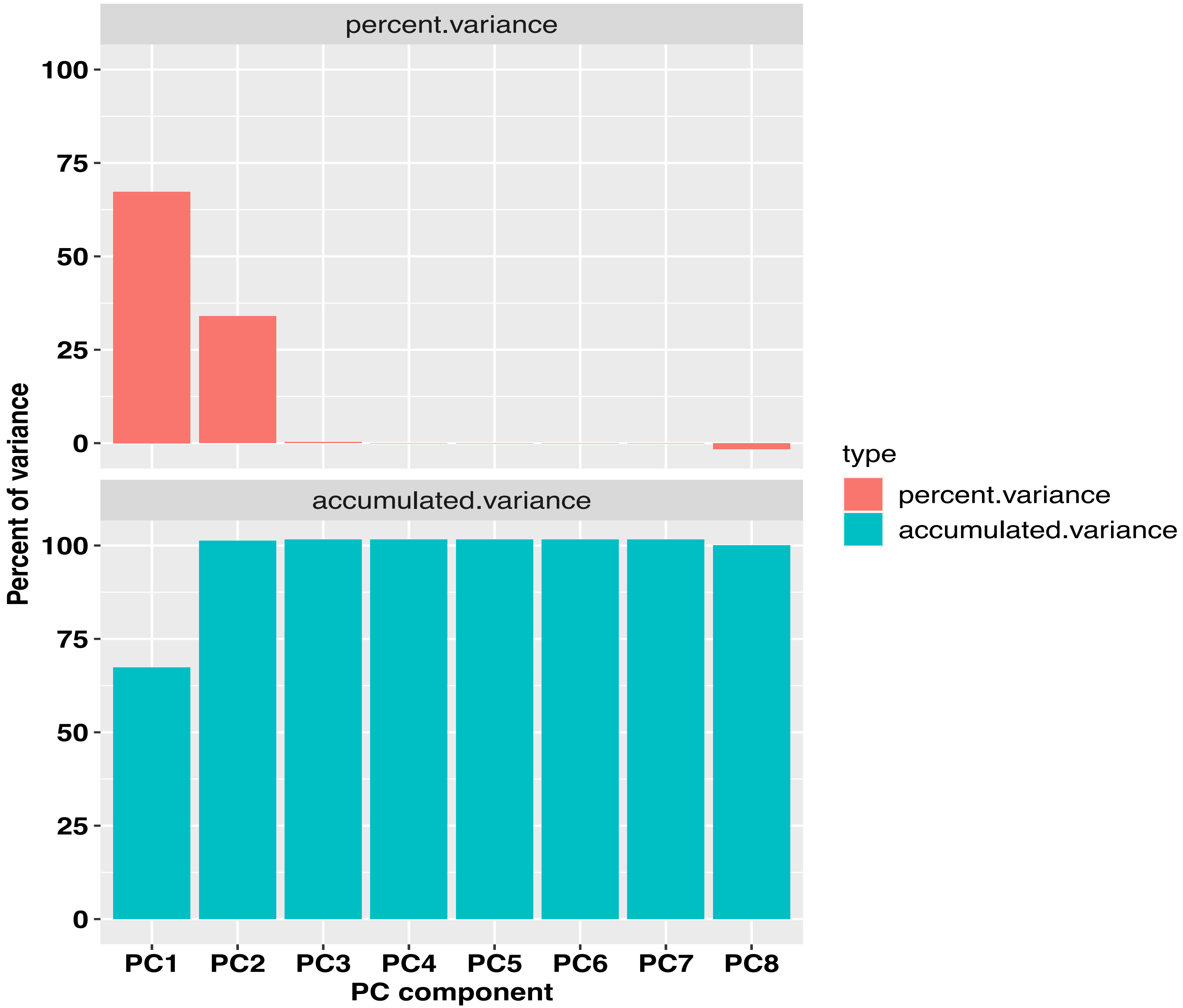
Z_s: standardized Z

$Z_s = U$

Google Map vs MDS Projection



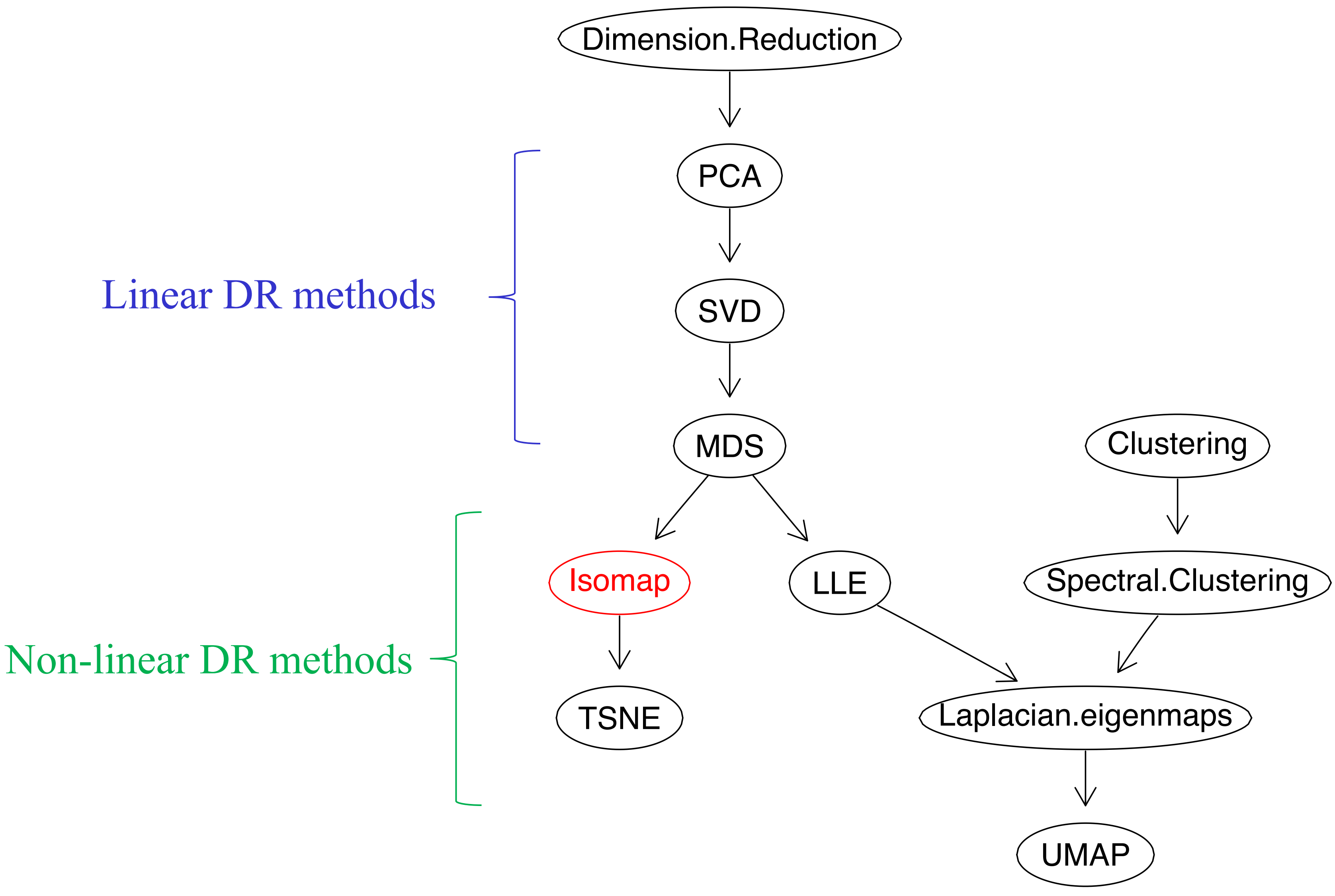
Variance of MDS Components



PCA vs MDS

	PCA	MDS
Data matrix	X_{np}	D_{nn}
Dot product	$S_{pp} = X^T X$	$G_{nn} = XX^T$
transformation	NA	$G = -1/2(HDH)$
Eigen decomposition	$X^T X = V \Lambda V^T$	$XX^T = U \Lambda U^T$
SVD	$Z = XV$	$Z = U\Sigma$

Road Map for Dimension Reduction Methods



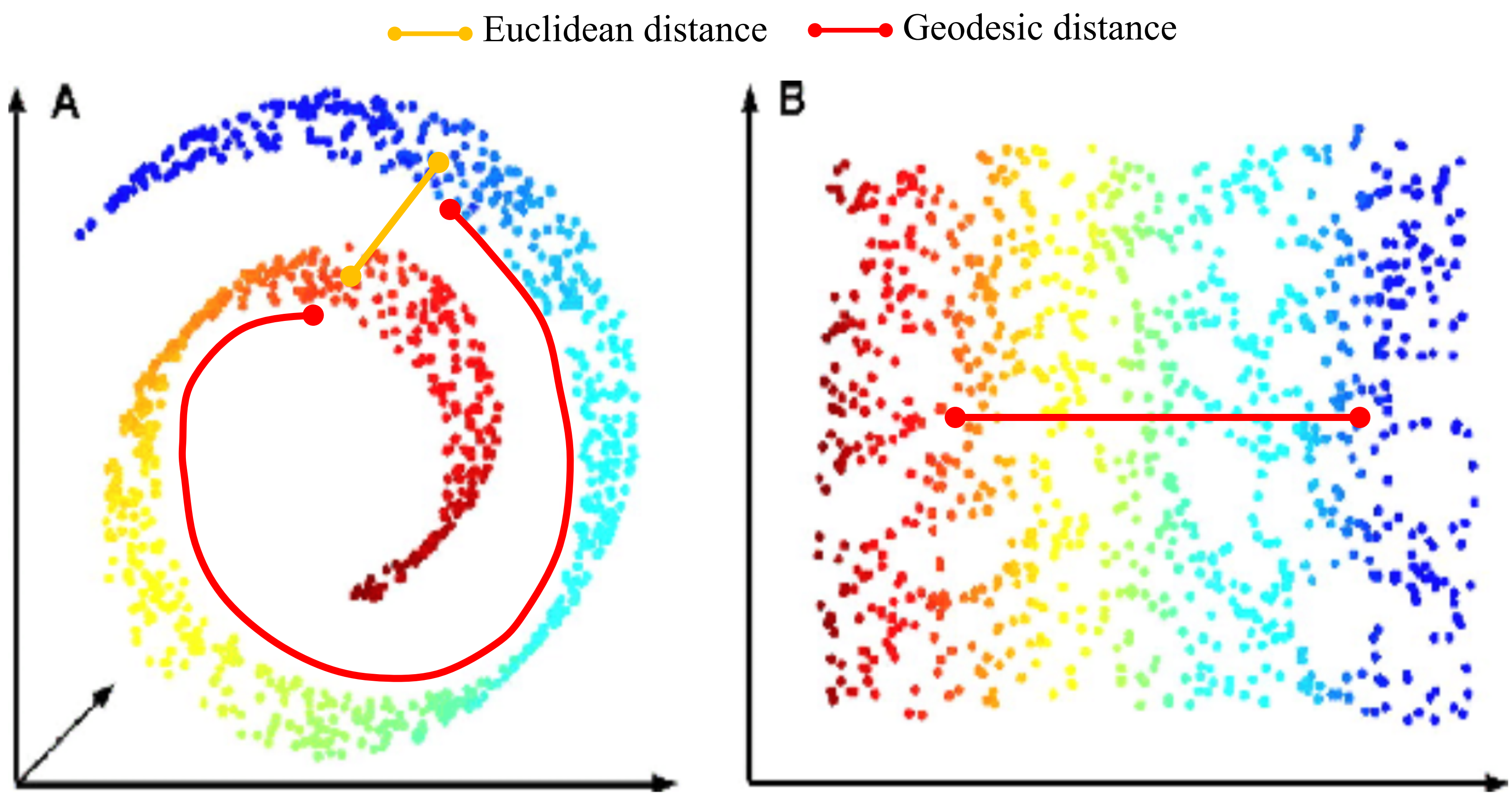
Euclidean Distance Versus Geodesic Distance

Euclidean distance

Geodesic distance

Nonlinear Dimension Reduction of Swiss Roll Dataset

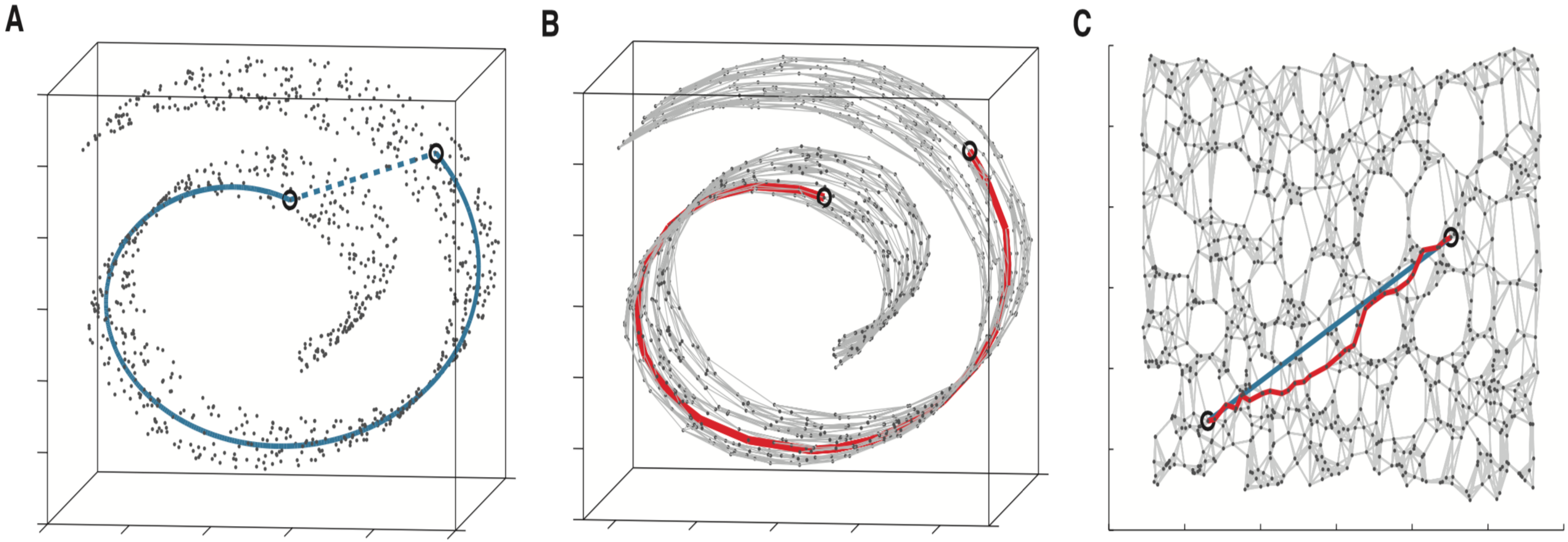
Swiss roll manifold in 3D —unfolding→ 2D sheet



3-dimension 2-dimension

Isomap

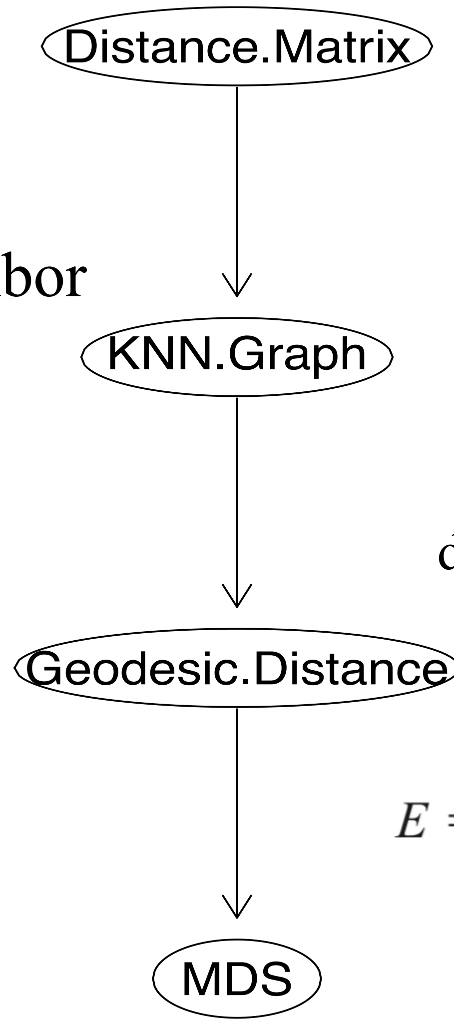
Swiss roll manifold in 3D KNN graph in 3D KNN graph in 2D

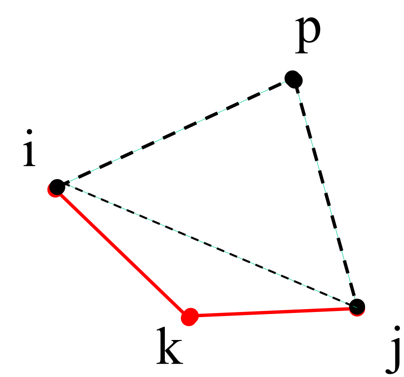


Joshua Tenenbaum et al Science in 2000

Algorithm of Isomap

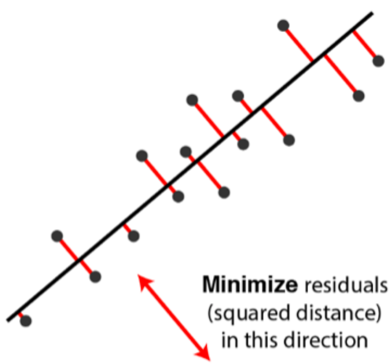
KNN: k nearest neighbor





$d_{ij} = d_{ik} + d_{kj}$

d_{ij} through k is the shortest path.



$$E = \|\tau(D_G) - \tau(D_Y)\|_{L^2}$$

$\tau(D) = -1/2(HDH)$

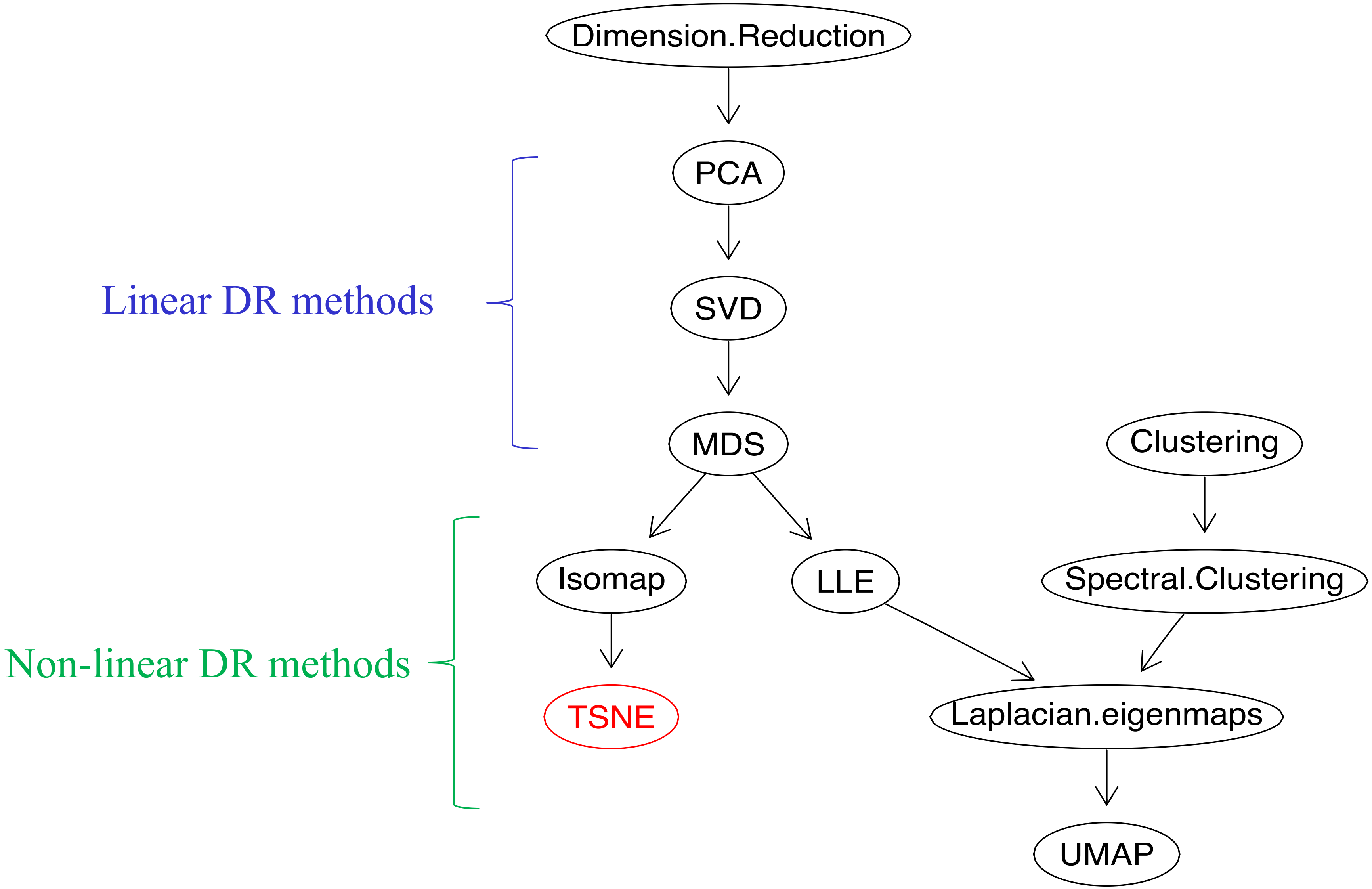
$H = I - 1/n(ee^T)$

Joshua Tenenbaum et al Science 2000

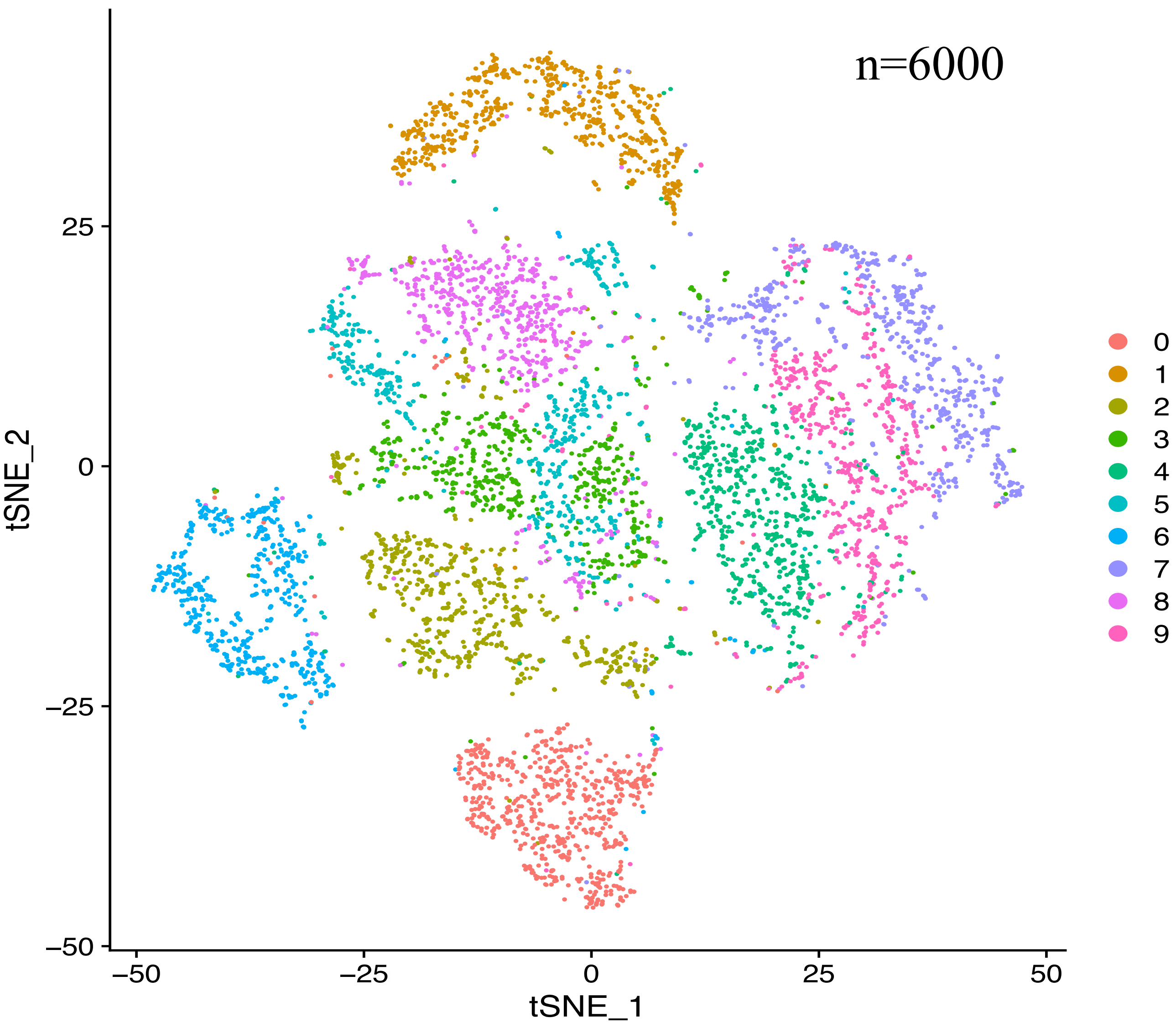
Summary of ISOMAP

- ISOMAP is an extension of MDS. We use geodesic distance instead of Eucledian distance for curve or curved surface.
- For KNN graph, K should be chosen to make graph connected but not too large to retain local linearity.
- However, there are issues about stability and performance.

Road Map for Dimension Reduction Methods

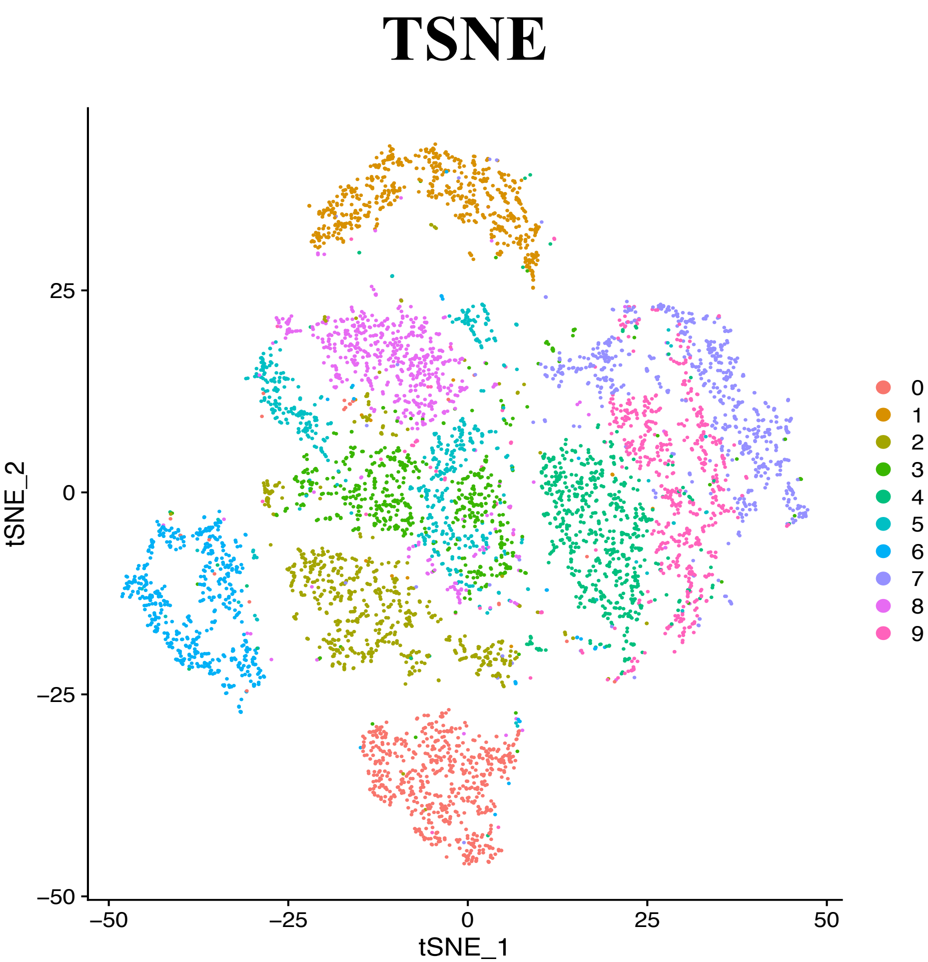


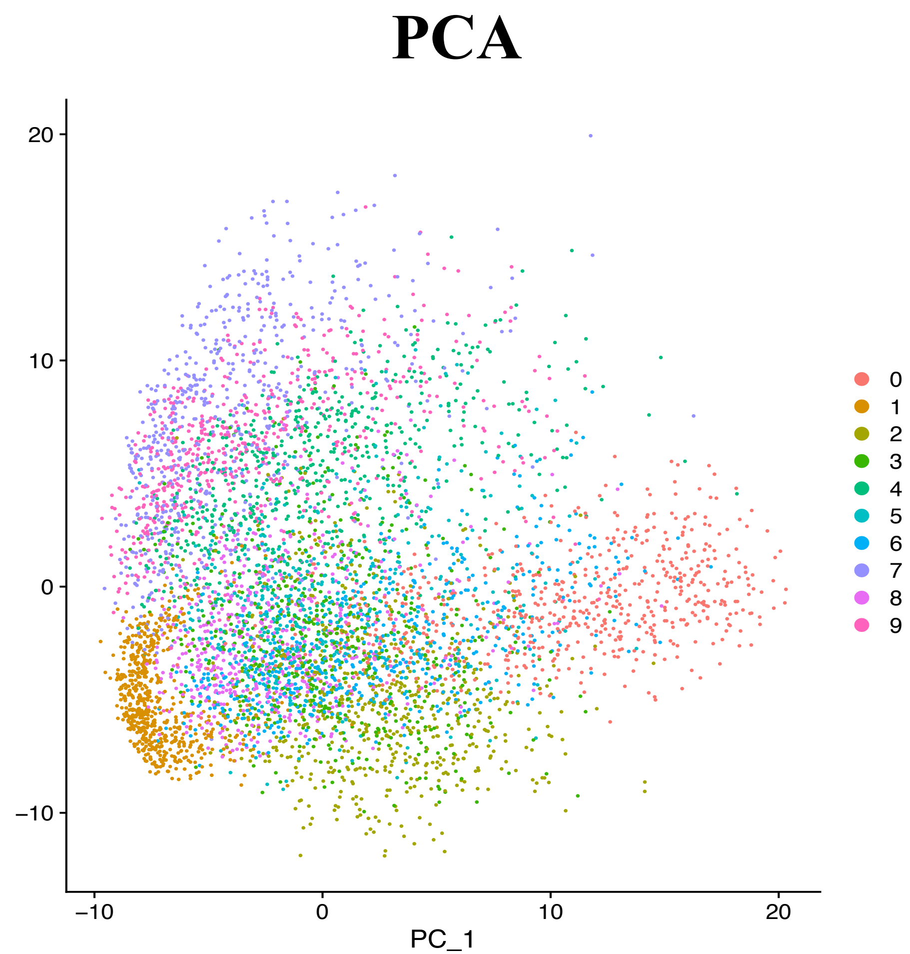
T-distributed Stochastic Neighbor Embedding (TSNE)



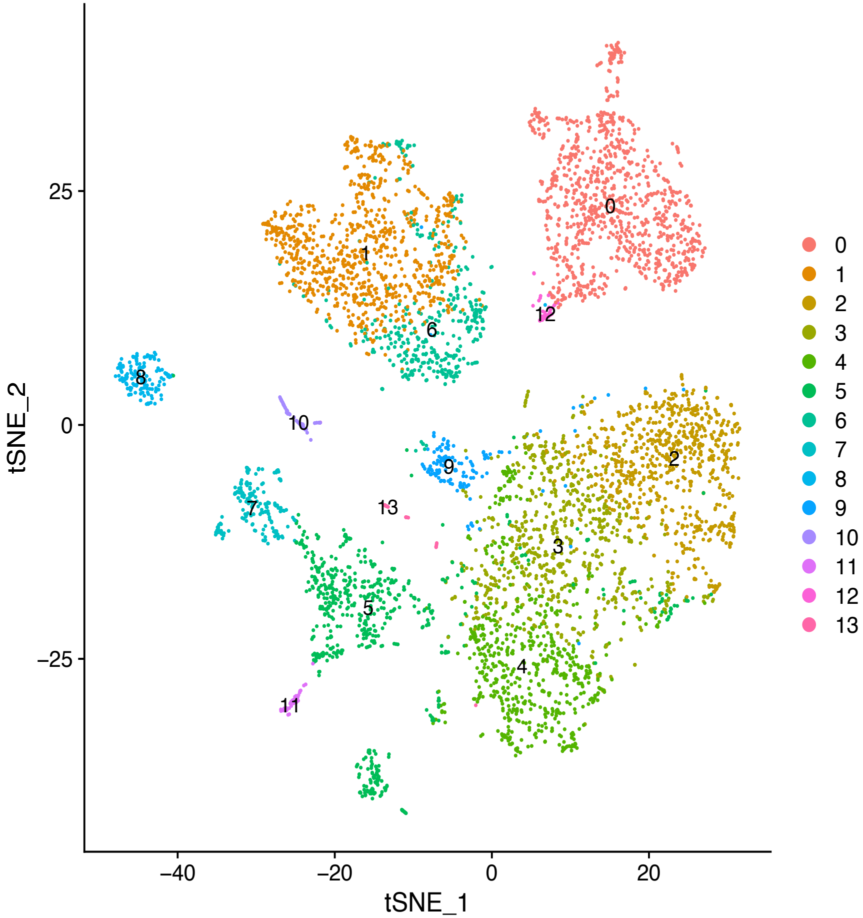
Laurens van der Maaten and Geoffrey Hinton, JMLR 2008

TSNE Versus PCA of the Same MNIST Dataset

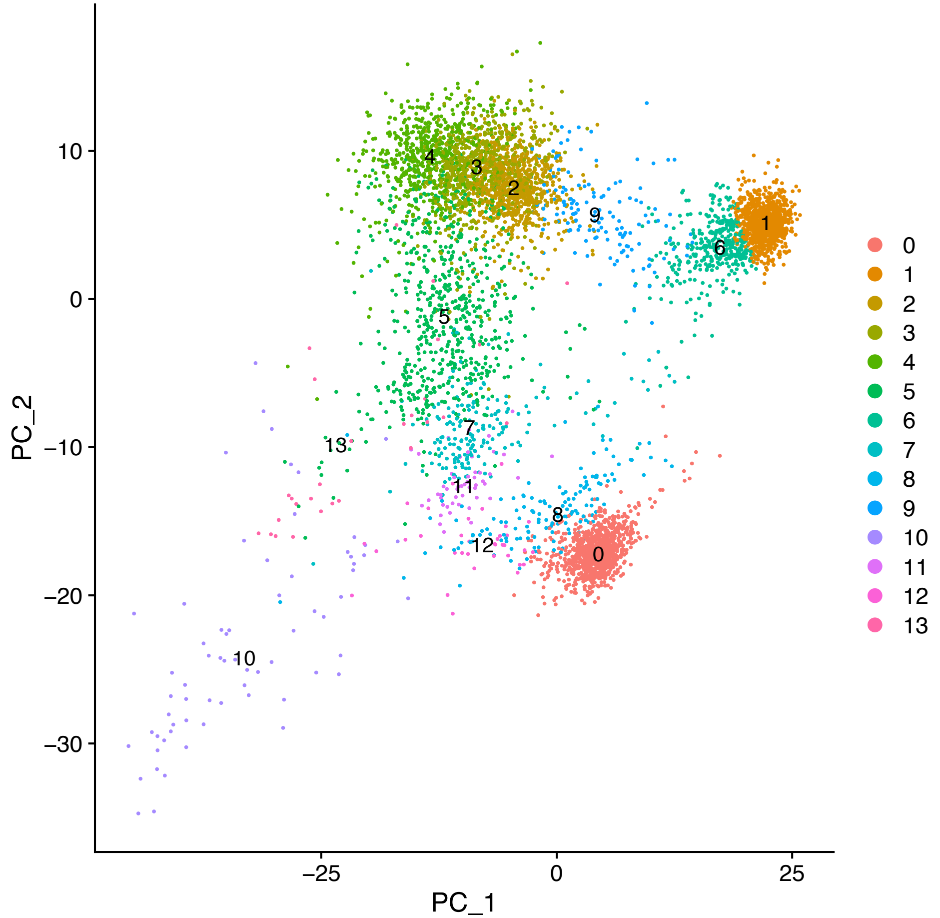




TSNE and PCA of a Single Cell RNAseq Data

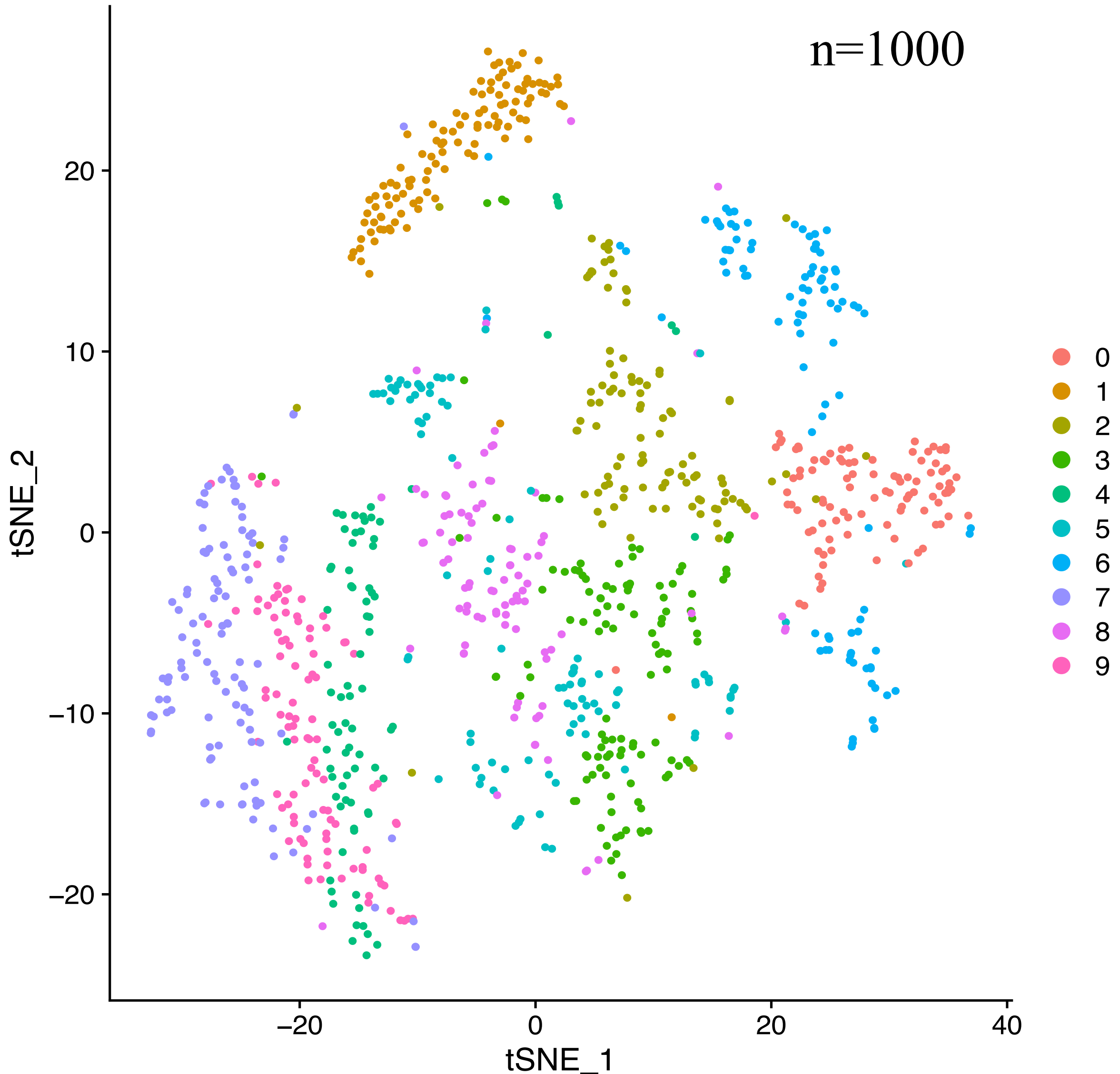


Cells in cluster are more spread out.

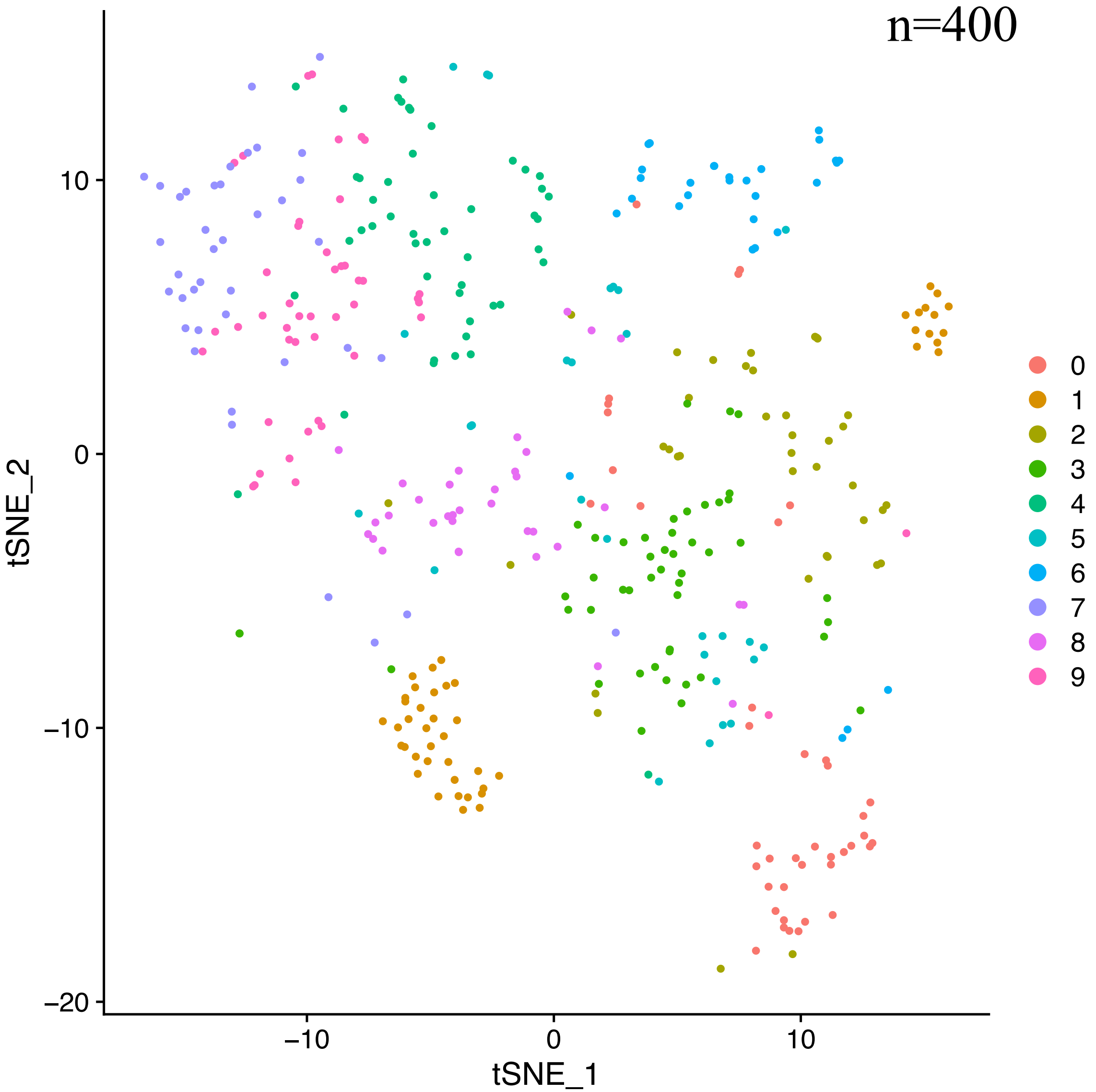


Clusters are driven by outliers.

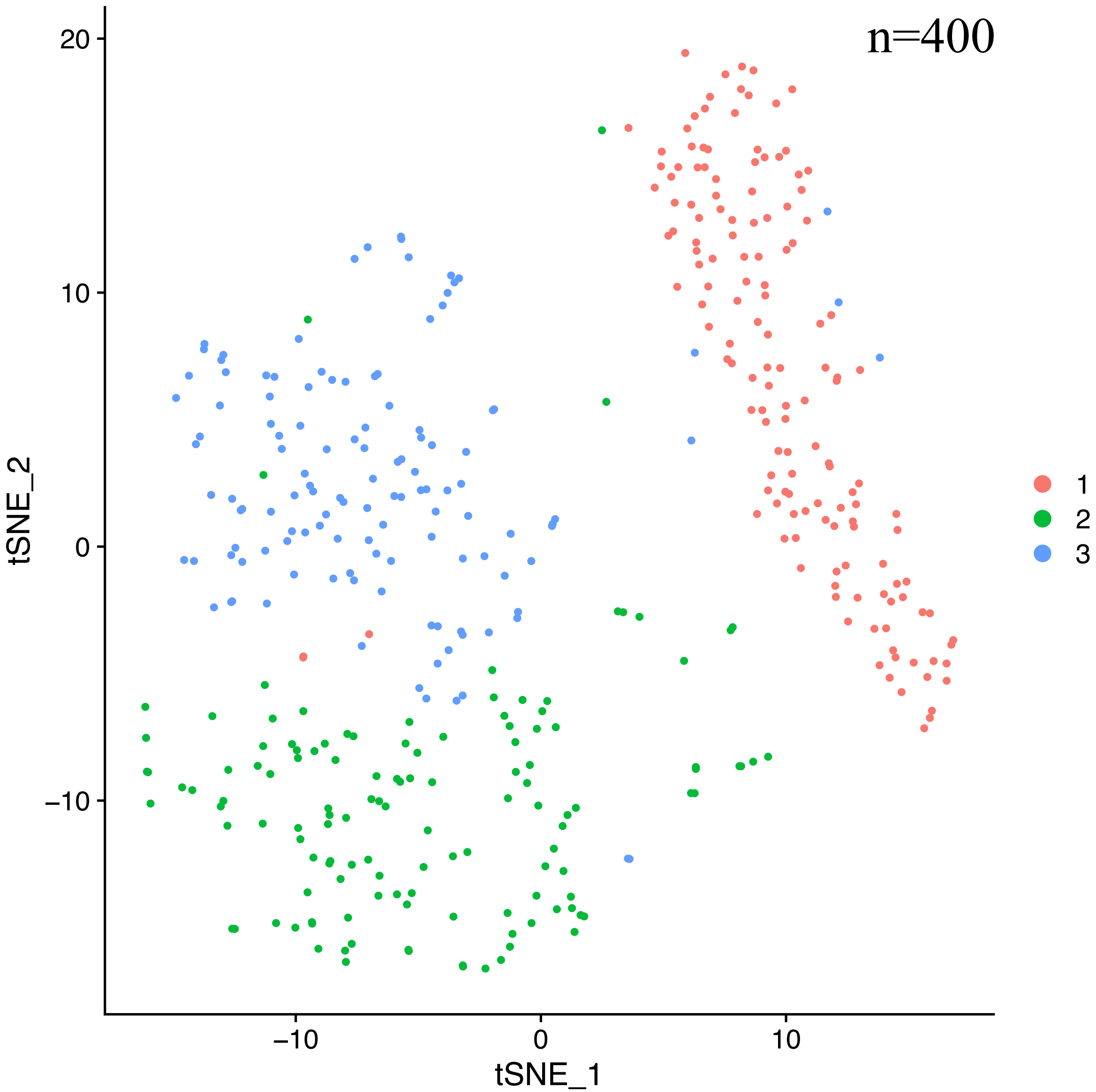
Effect of Sample Size on TSNE Plot



Effect of Sample Size on TSNE Plot



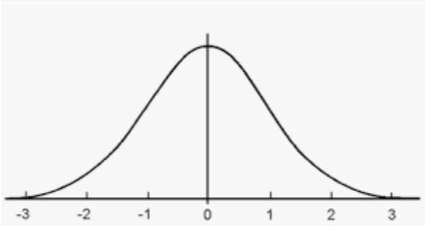
Effect of Cluster Numbers on TSNE Plot



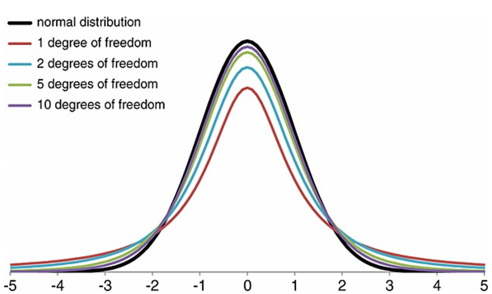
T-distributed Stochastic Neighbor Embedding (TSNE)

Gaussian kernel

$$y = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$



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



Distance.Matrix

↓

HD.Gaussian

↓

LD.t.Distribution

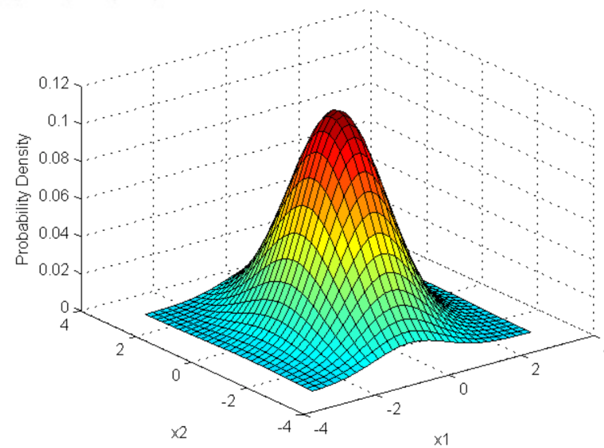
↓

KL.divergence

Radial basis function (RBF) kernel

$$Z_s = (x - \mu)^T \Sigma^{-1} (x - \mu)$$

$$p(x; \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right)$$



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

Laurens van der Maaten and Geoffrey Hinton, JMLR 2008

Road Map for Dimension Reduction Methods