Computational Systems Biology
Deep Learning in the Life Sciences

6.802 6.874 20.390 20.490 HST.506

David Gifford
Lecture 6
February 25, 2019

# The Zen of PCA, t-SNE, and Autoencoders

Massachusetts Institute of Technology

http://mit6874.github.io

1

## What's on tap today!

- Embedding data in a lower dimensional space
- Linear reduction of dimensionality
  - Principle Component Analysis
- Non-linear embedding
  - t-distributed Stochastic Network Embedding (t-SNE)
  - Autoencoders

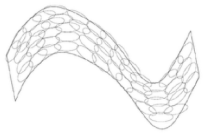
## Dimensionality reduction has multiple applications

- Uses:
  - Data Visualization
  - Data Reduction
  - Data Classification
  - Trend Analysis
  - Factor Analysis
  - Noise Reduction
- Examples:
  - How many unique "sub-sets" are in the sample?
  - How are they similar / different?
  - What are the underlying factors that influence the samples?
  - Which time / temporal trends are (anti)correlated?
  - Which measurements are needed to differentiate?
  - How to best present what is "interesting"?
  - Which "sub-set" does this new sample rightfully belong?

A manifold is a topological space that locally resembles Euclidean space near each point

A manifold embedding is a structure preserving mapping of a high dimensional space into a manifold

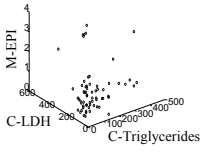
Manifold learning learns a lower dimensional space that enables a manifold embedding



## 1. Principal Component Analysis

## Example data

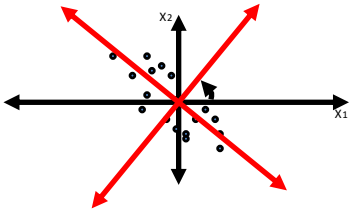
- Example: 53 Blood and urine measurements (wet chemistry) from 65 people (33 alcoholics, 32 non-alcoholics)
- Trivariate plot




| | H-WBC | H-RBC | H-Hgb | H-Hct | H-MCV | H-MCH | H-MCHC |
|---|---|---|---|---|---|---|---|
| A1 | 8.0000 | 4.8200 | 14.1000 | 41.0000 | 85.0000 | 29.0000 | 34.0000 |
| A2 | 7.3000 | 5.0200 | 14.7000 | 43.0000 | 86.0000 | 29.0000 | 34.0000 |
| A3 | 4.3000 | 4.4800 | 14.1000 | 41.0000 | 91.0000 | 32.0000 | 35.0000 |
| A4 | 7.5000 | 4.4700 | 14.9000 | 45.0000 | 101.0000 | 33.0000 | 33.0000 |
| A5 | 7.3000 | 5.5200 | 15.4000 | 46.0000 | 84.0000 | 28.0000 | 33.0000 |
| A6 | 6.9000 | 4.8600 | 16.0000 | 47.0000 | 97.0000 | 33.0000 | 34.0000 |
| A7 | 7.8000 | 4.6800 | 14.7000 | 43.0000 | 92.0000 | 31.0000 | 34.0000 |
| A8 | 8.6000 | 4.8200 | 15.8000 | 42.0000 | 88.0000 | 33.0000 | 37.0000 |
| A9 | 5.1000 | 4.7100 | 14.0000 | 43.0000 | 92.0000 | 30.0000 | 32.0000 |

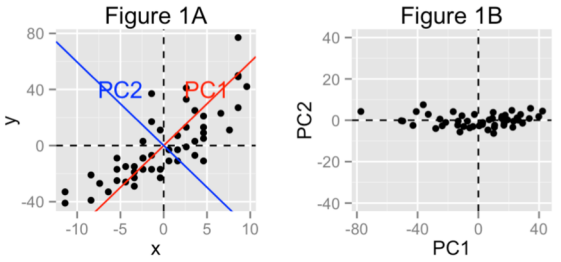
## Principal Component = axis of greatest variability

Suppose we have a population measured on p random variables $X_1,\ldots,X_p$. Note that these random variables represent the p-axes of the Cartesian coordinate system in which the population resides. Our goal is to develop a new set of p axes (linear combinations of the original p axes) in the directions of greatest variability:



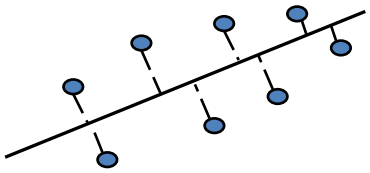

This is accomplished by rotating the axes.

## Data projected onto PC1




## Selecting Principal Components

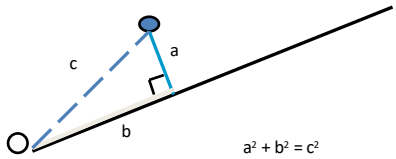
- Given m points in a n dimensional space, for large n, how does one project on to a 1 dimensional space?
- Formally, minimize sum of squares of distances to the line.



- Why sum of squares? Because it allows fast minimization, assuming the line passes through 0

## We maximize the sum of square of projections to minimize the sum of squares distances

- Minimizing sum of squares of distances to the line is the same as maximizing the sum of squares of the projections on that line, thanks to Pythagoras.




$a^2 + b^2 = c^2$

$a^2 = c^2 - b^2$

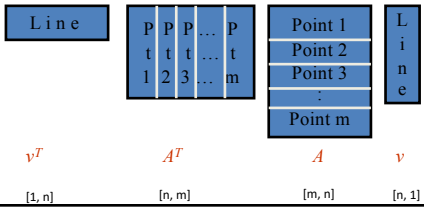
## Sum of squares of projections for all m points onto vector x in matrix form

$\max(v^T A^T A v)$, subject to $v^T v = 1$

$\Sigma = A^T A$

$v^T \Sigma v = \lambda^2$

$\Sigma v = \lambda^2 v$




## Principle Component Analysis (PCA)

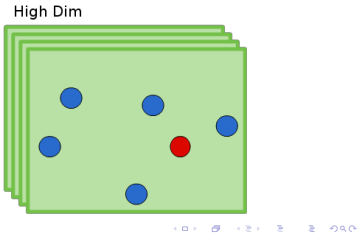
- How do we find the eigenvectors $v_i$?
- We use singular value decomposition to decompose $\Sigma$ into an orthogonal rotation matrix $U$ and a diagonal scaling matrix $S$:

$$\Sigma = USU^T \quad (22)$$
$$\Sigma U = (USU^T)U \quad (23)$$
$$= US \quad (24)$$

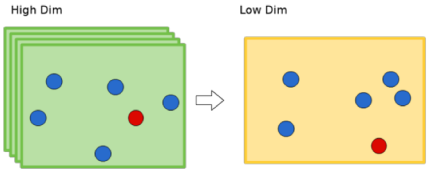
- The columns of $U$ are the $v_i$, and $S$ is the diagonal matrix of eigenvalues $\lambda_i^2$

2. tSNE non-linear embedding

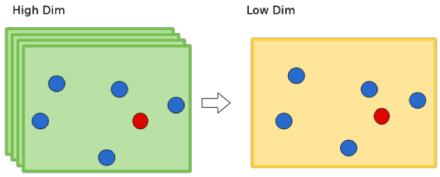
Distance Preservation  Neighbor Preservation



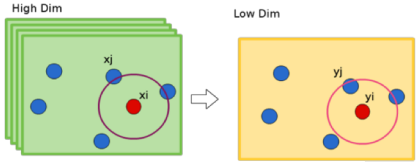

## Neighborhood not preserved




## Neighborhood preserved




## Measure pairwise distances in high dimensional space




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

Set the bandwidth $\sigma_i$ such that the conditional has a fixed perplexity (effective number of neighbors) $Perp(P_i) = 2^{H(P_i)}$, typical value is about 5 to 50

## We want to choose an embedding that minimizes divergence between low and high dimension similarities

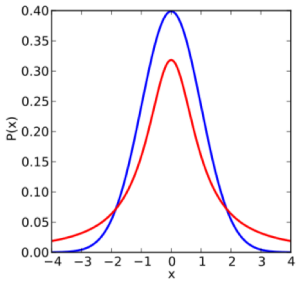
- Similarity of datapoints in High Dimension

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

- Similarity of datapoints in Low Dimension

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

## Low dimensional embedding using a Student t-distribution to avoid overcrowding




Red – Student t-distribution (1 degree of freedom)
Blue - Gaussian

## We can use gradient methods to find an embedding

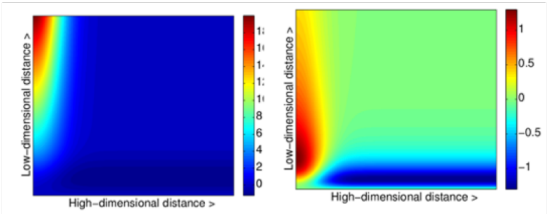
- Cost function

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

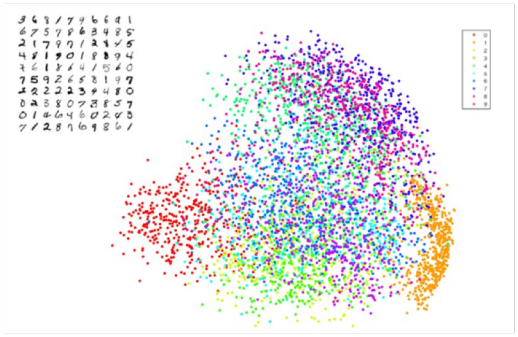
  - Large $p_{ij}$ modeled by small $q_{ij}$: Large penalty
  - Small $p_{ij}$ modeled by large $q_{ij}$: Small penalty
  - t-SNE mainly preserves local similarity structure of the data
- Gradient

$$\frac{\partial C}{\partial y_i} = 4 \sum_{j \neq i} (p_{ij} - q_{ij})(1 + ||y_i - y_j||^2)^{-1}(y_i - y_j)$$

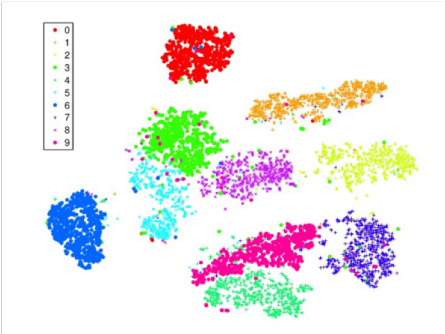
## Interpretation of SNE (left) and t-SNE (right) gradients



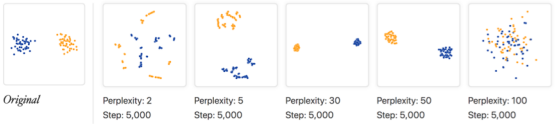

## PCA of MNIST digits



## t-SNE of MNIST digits



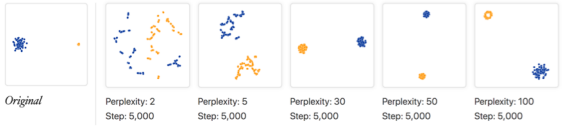

## Perplexity matters




https://distill.pub/2016/misread-tsne/

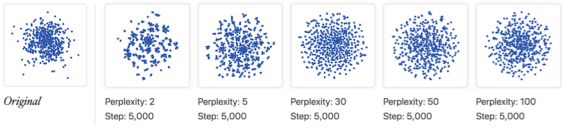
## Number of steps matter


## Cluster sizes are not meaningful

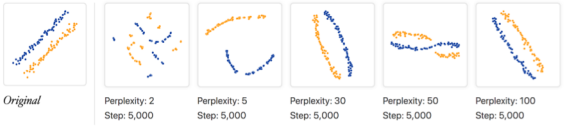



## Distance is not always preserved

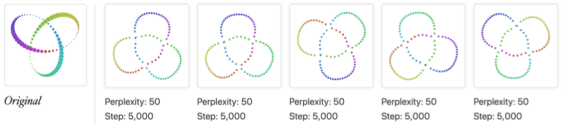

## False clusters may appear



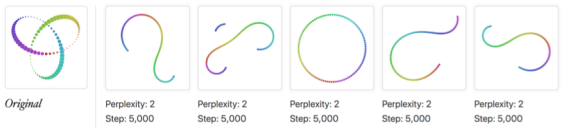

## Relationships are not always preserved



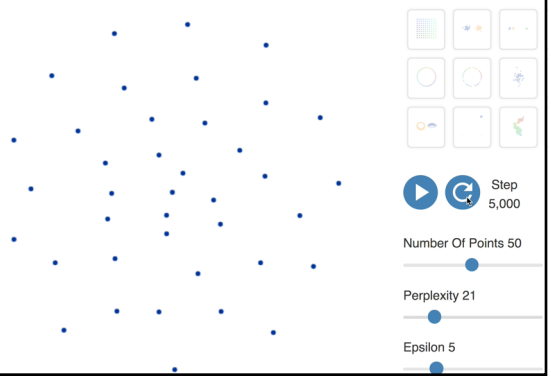

## Different runs may produce similar results…



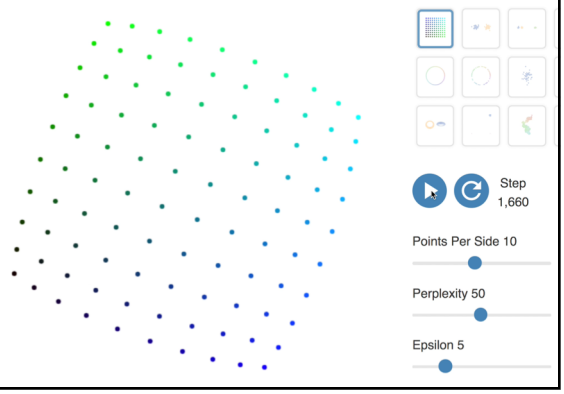
## Or not…



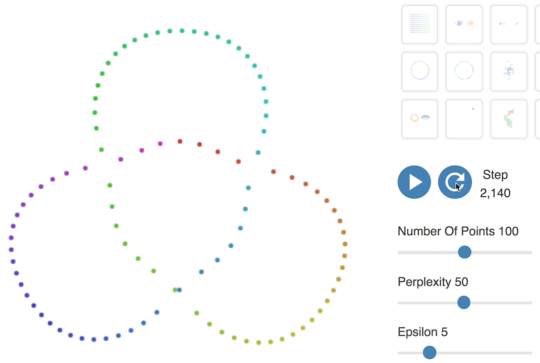

## t-SNE of equidistant points




## t-SNE of square grid

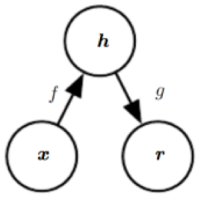



## t-SNE of 3D Knot



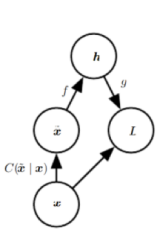

3. Autoencoders embed data into a latent space

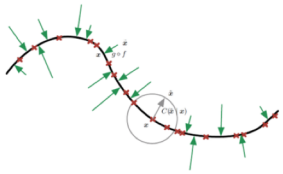
## Autoencoders learn a latent representation of data




$$L(\boldsymbol{x}, g(f(\boldsymbol{x})))$$

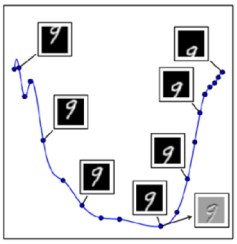
## Denoising autoencoders recover signal corrupted by noise

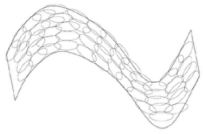


$$L(\boldsymbol{x}, g(f(\tilde{\boldsymbol{x}}))),$$

## We can learn manifolds with autoencoders



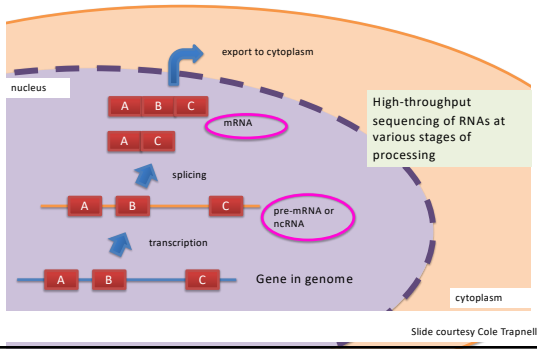


## Interesting on-line demos

http://dpkingma.com/sgvb_mnist_demo/demo_old.html
http://elf-project.sourceforge.net/autoencoder.html
http://vdumoulin.github.io/morphing_faces/online_demo.html

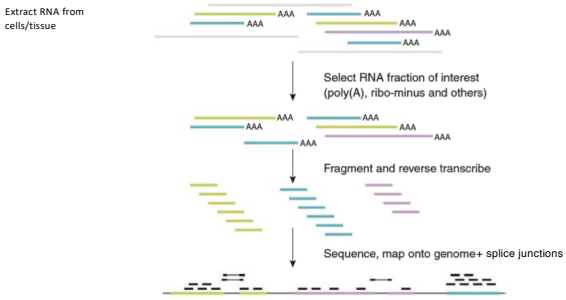
## 4. RNA-seq data has 3,000 – 20,000 gene expression levels per sample

## RNA-Seq characterizes RNA molecules



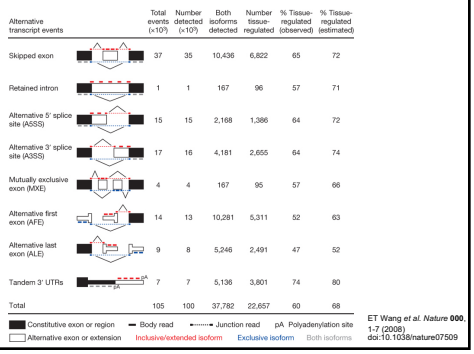

Slide courtesy Cole Trapnell

## RNA-Seq: millions of short reads from fragmented mRNAs



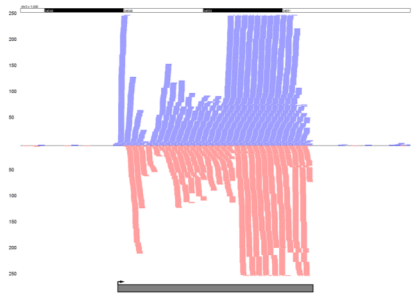

Pepke et. al. *Nature Methods* 2009

Pervasive tissue-specific regulation of alternative mRNA isoforms.



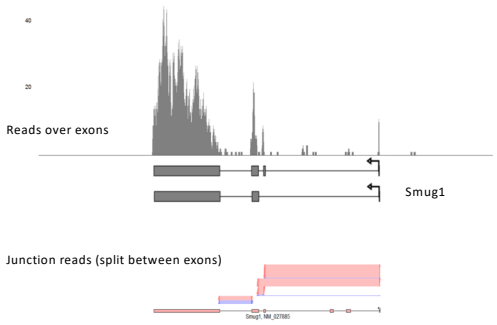

ET Wang *et al. Nature* **000**, 1-7 (2008) doi:10.1038/nature07509

As crude measure of expression is
Reads Per Kilobase of gene per Million reads (RPKM)



Sox2

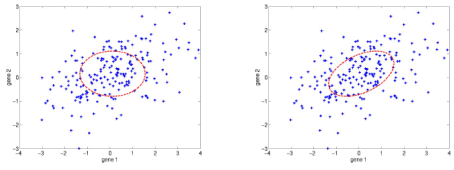
## RNA-seq reads map to exons and across exons




## 5. The significance of differential expression

## Statistical tests: example

- The alternative hypothesis $H_1$ is more expressive in terms of explaining the observed data




null hypothesis  alternative hypothesis

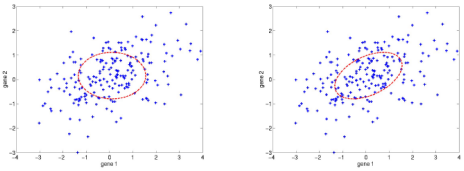
- We need to find a way of testing whether this difference is significant

## Degrees of freedom

- How many degrees of freedom do we have in the two models?

$$H_0: \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \sim N\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}\right)$$

$$H_1: \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \sim N\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}\right)$$



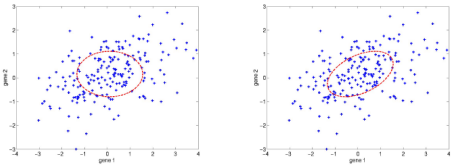

$H_0$  $H_1$

## Degrees of freedom

- How many degrees of freedom do we have in the two models?

$$H_0: \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \sim N\left( \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix} \right)$$

$$H_1: \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \sim N\left( \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \right)$$



$H_0$ $H_1$

- The observed data overwhelmingly supports $H_1$

## Test statistic

- Likelihood ratio statistic

$$T(X^{(1)}, \ldots, X^{(n)}) = 2 \log \frac{P(X^{(1)}, \ldots, X^{(n)} | \hat{H}_1)}{P(X^{(1)}, \ldots, X^{(n)} | \hat{H}_0)} \quad (1)$$

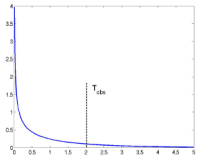
Larger values of $T$ imply that the model corresponding to the null hypothesis $H_0$ is much less able to account for the observed data

- To evaluate the P-value, we also need to know the sampling distribution for the test statistic

In other words, we need to know how the test statistic $T(X^{(1)}, \ldots, X^{(n)})$ varies if the null hypothesis $H_0$ is correct

## Test statistic cont'd

- For the likelihood ratio statistic, the sampling distribution is $\chi^2$ with degrees of freedom equal to the difference in the number of free parameters in the two hypotheses




- Once we know the sampling distribution, we can compute the P-value

$$p = Prob(T(X^{(1)}, \ldots, X^{(n)}) \geq T_{obs} \,|\, H_0) \quad (2)$$

FIN - Thank You