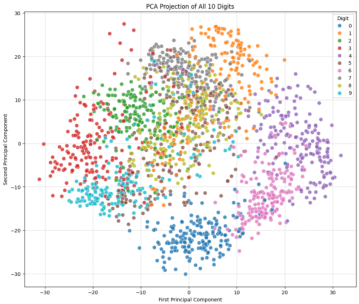
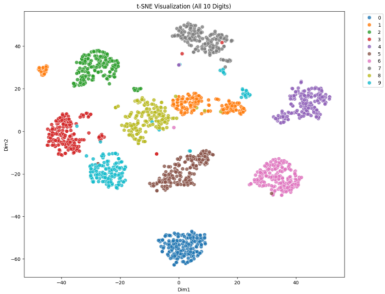
Nonlinear Dimensionality Reduction

For the SKLearn images dataset, compare these two dimensionality reduction/visualization approaches:





Idea: PCA is linear; *t*-SNE is *non*linear . . . how to implement it?

The Entropy of a Random Variable

Suppose X is a random variable with values and probabilities given below.

Number of Cars	Probability
0	0.03
1	0.13
2	0.70
3	0.10
4	0.04

What is the expected value $E(X)$?

Def: The **entropy** of X is $H(X) = -\sum_{x \in X} p(x) \log p(x)$.

- $\log(x)$ meaning ...?
- Example?

Properties of Entropy

- $H(X) \geq 0$
- $H(X) = 0$ if and only if there is x_0 such that $X = x_0$
- $H(X) = \log(n)$ if and only if X is *uniformly distributed* among $\{x_1, x_2, \ldots, x_n\}$
- $0 \leq H(X) \leq \log(n)$

$$
\begin{aligned}
H(X) &= \sum_{x \in X} p(x) \log \frac{1}{p(x)} & (1) \\
&= log|X| - \sum_{x \in X} p(x) \log \frac{p(x)}{\frac{1}{|X|}} & (2)
\end{aligned}
$$

Idea: Entropy gives a measure of how much information/suprise is in a random variable: (No surprise $\implies$ $H(X) = 0$, etc.)

Relative Entropy

Suppose P and Q are two probability distributions on a sample space. Then the **relative entropy** of P relative to Q is given by

$$D_{KL}(P\|Q) = \sum_{x\in X} P(x)\log\frac{P(x)}{Q(x)}$$

This is also called the **Kullback-Leibler (or KL) Divergence**.

Properties:

- $D_{KL}(P\|Q) \neq D_{KL}(Q\|P)$.
- If $P(x) = Q(x)$ for all $x \in X$, then $D_{KL}(P\|Q) = 0$.
- $D_{KL}(P\|Q) \geq 0$. (See next page for proof.)

The KL Divergence

Pf: Use Jensen's inequality at the key step. All the rest is algebra.

$$
\begin{aligned}
& D_{KL}(p_X, p_Y) \\
&= -\sum_{x \in R_X} p_X(x) \ln\left(\frac{p_Y(x)}{p_X(x)}\right) \\
&= \mathrm{E}\left[-\ln\left(\frac{p_Y(X)}{p_X(X)}\right)\right] \\
&> -\ln\left(\mathrm{E}\left[\frac{p_Y(X)}{p_X(X)}\right]\right) \\
&= -\ln\left(\sum_{x \in R_X} p_X(x) \frac{p_Y(x)}{p_X(x)}\right) \\
&= -\ln\left(\sum_{x \in R_X} p_Y(x)\right) \\
&\geq -\ln(1) = 0
\end{aligned}
$$

Key point: What can you say about $\log \frac{P(x)}{Q(x)}$ when $P(x)$ and $Q(x)$ are close . . . ?

Keyer point: We can use the KL divergence as a measure of how "close" two probability distributions are–that is, as a loss function.

t-Stochastic Neighborhood Embeddings (t-SNE)

Suppose you have $\mathbf{x}_1, \ldots, \mathbf{x}_N$ high dimensional objects that you want to project down to (as yet unspecified) points $\mathbf{y}_1, \ldots, \mathbf{y}_N$ in 2 or 3-dimensional space ...

For each index i in $\{1, \ldots, n\}$, define:

$$p_{j|i} = \frac{\exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|\mathbf{x}_i - \mathbf{x}_k\|^2/2\sigma_i^2)} \quad \text{and} \quad p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}$$

These are probability distributions (why?), as are these (why?):

$$q_{ij} = \frac{(1 + \|\mathbf{y}_i - \mathbf{y}_j\|^2)^{-1}}{\sum_k \sum_{l \neq k} (1 + \|\mathbf{y}_k - \mathbf{y}_l\|^2)^{-1}}$$

Idea: Choose $\mathbf{y}_i$ to reflect "similarity" with the $\mathbf{x}_i$. In other words, choose $\mathbf{y}_i$ to minimize the KL Divergence ...

The t-SNE Cost Function

Idea: Use the KL Divergence to create the loss/cost function from each of the probability distributions p_i and q_i:

$$C = \sum_i D_{KL}(p_i \| q_i)$$

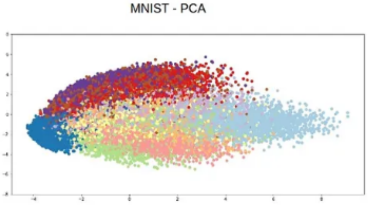
Goal: We want to find $\mathbf{y}_i$ to minimize C. This can be done using gradient descent . . .

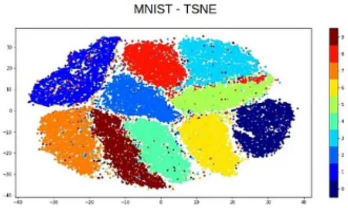
The partial derivatives are given by:

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

t-SNE and MNIST

An example of different visualizations for the MNIST data set:





See the SKLearn TSNE class.