Machine Learning HW7 t-SNE

What I have done

Modify the code a little bit and make it back to symmetric

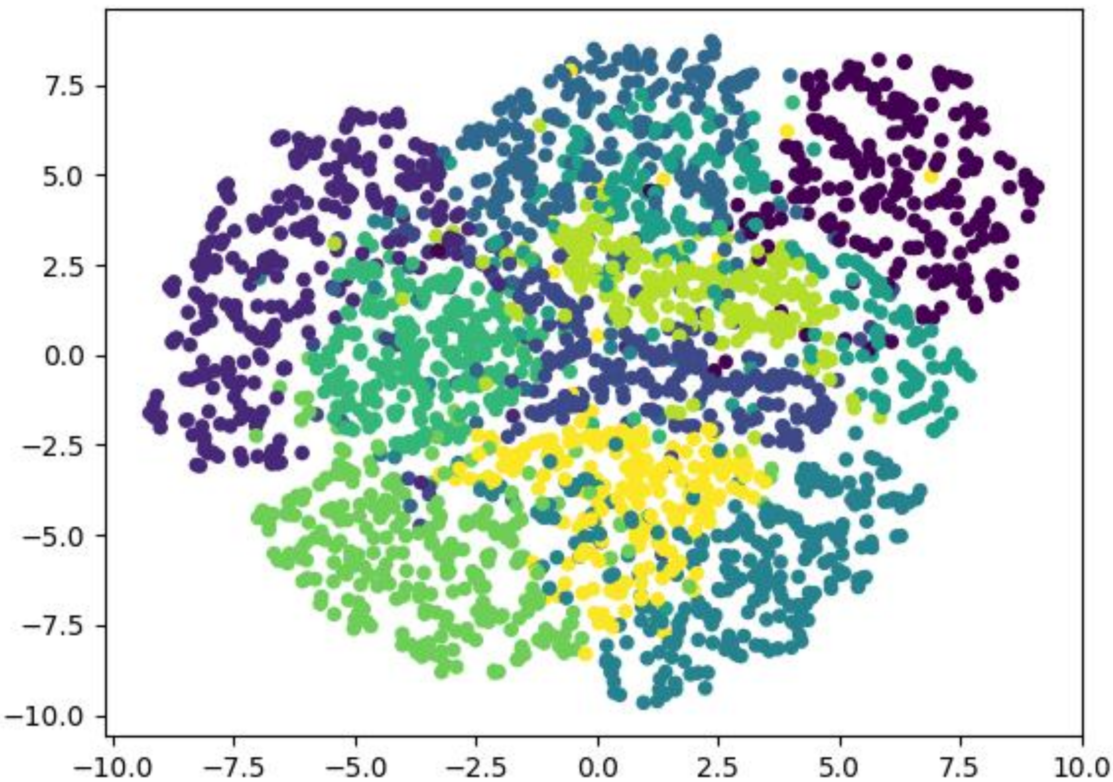
SNE

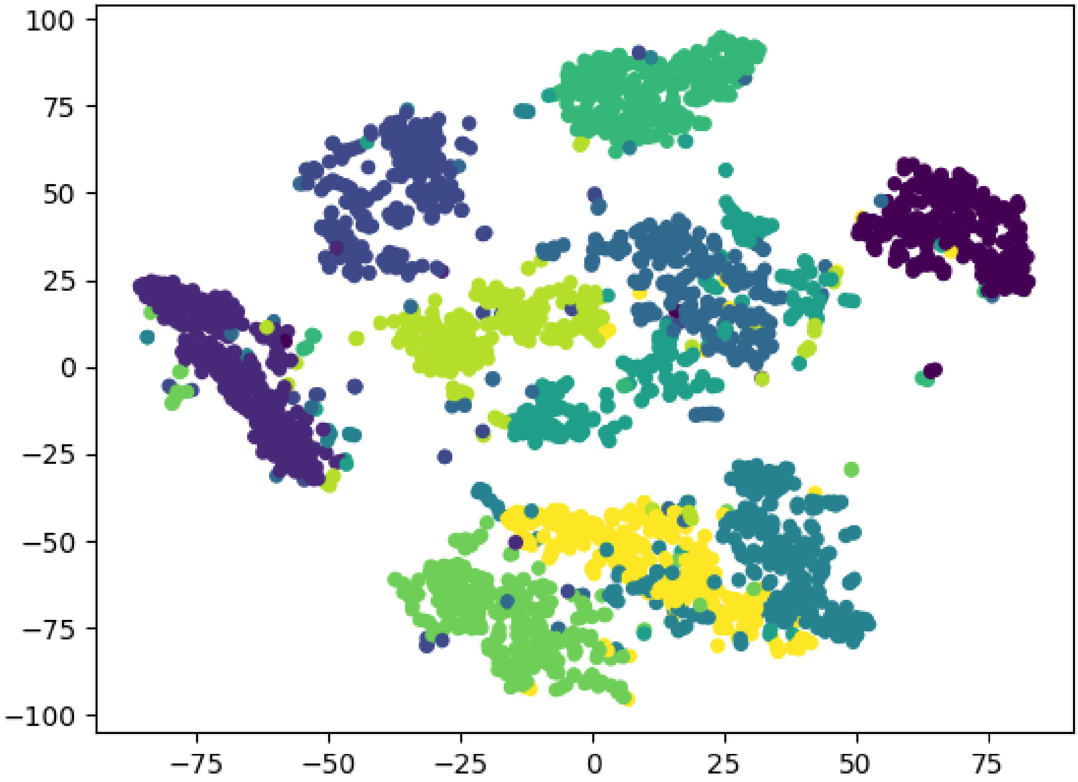
similarity in low dimension Q
derivative of y
symmetric sne
Compute pairwise affinities sum_Y = np.sum(np.square(Y), 1) Q = -2. * np.dot(Y, Y.T) Q = np.add(np.add(Q, sum_Y).T, sum_Y) Q = np.exp(-Q) Q[range(n), range(n)] = 0 Q = Q / np.sum(Q) Q = np.maximum(Q, 1e-12) # Compute gradient PQ = P - Q for i in range(n): dY[i, :] = 2. * np.sum(np.tile(PQ[:, i], (no_dims, 1)).T * (Y[i, :] - Y), 0)
t-sne
Compute pairwise affinities sum_Y = np.sum(np.square(Y), 1) num = -2. * np.dot(Y, Y.T) num = 1. / (1. + np.add(np.add(num, sum_Y).T, sum_Y)) num[range(n), range(n)] = 0. Q = num / np.sum(num) Q = np.maximum(Q, 1e-12) # Compute gradient PQ = P - Q for i in range(n): dY[i, :] = 4. * np.sum(np.tile(PQ[:, i] * num[:, i], (no_dims, 1)).T * (Y[i, :] - Y), 0)

What I have visualized

Compare between symmetric sne and t-sne

Using data from https://lvdmaaten.github.io/tsne/code/tsne_python.zip
You can tell there is crowding problem when using symmetric sne.
symmetric sne
t-sne


7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0



Distribution of pairwise similarities

Here I trimmed zero in pairwise similarty matrix and use normalized histogram to visualize.

P_ = np.trim_zeros(np.sort(P.flatten())) for i in range(len(P_)): if P_[i] <= 1e-12: P_[i] = 0 P_ = np.trim_zeros(P_)

binwidth = 1 bins = np.arange(min(np.log(P_)), max(np.log(P_)) + binwidth, binwidth)

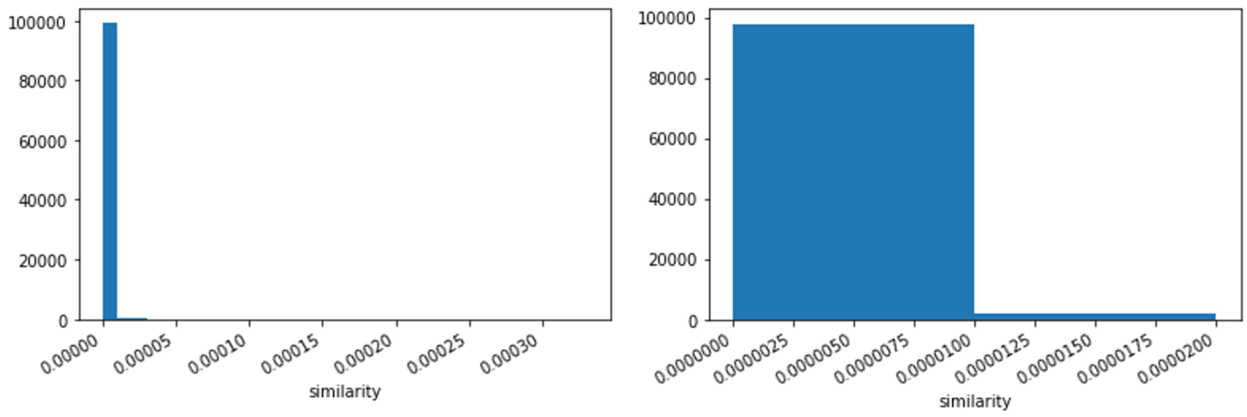
plt.figure() plt.hist(np.log(P_), bins=bins, normed=1) plt.xlabel('log similarity')

plt.show()

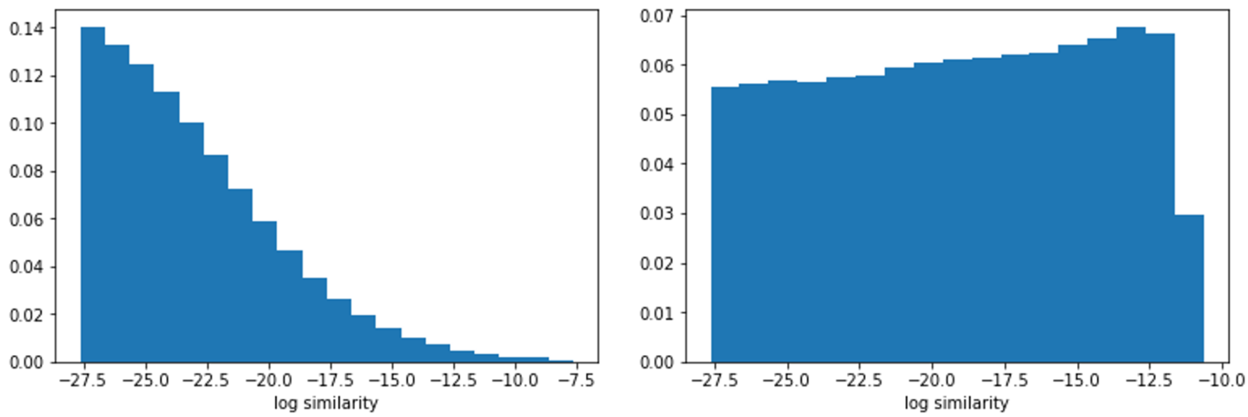
symmetric sne

t-sne

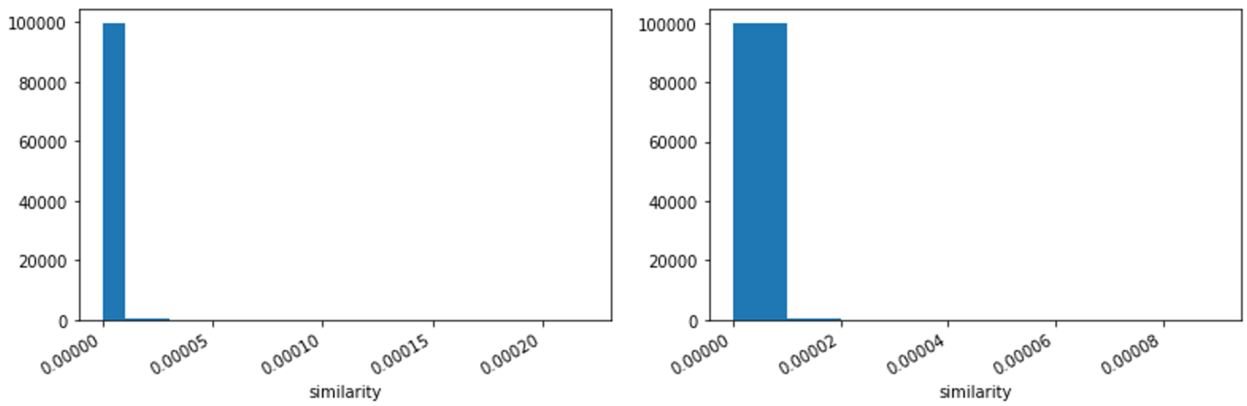
Plot similarity directly.



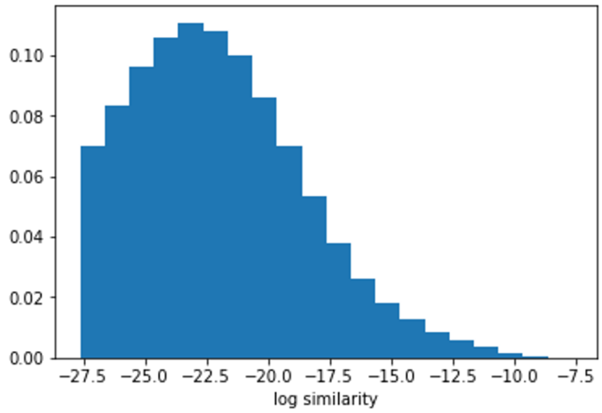
Plot log transformed similarity.

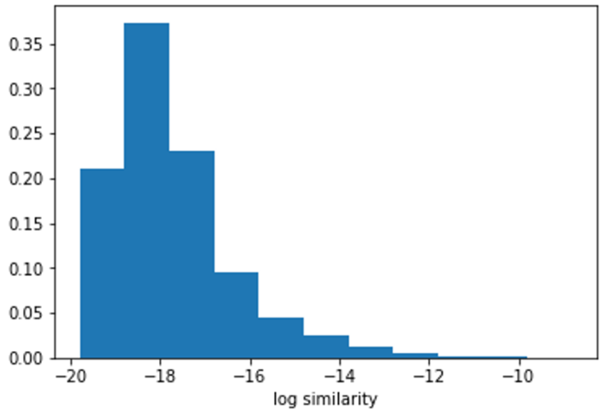


Plot similarity directly.



Plot log transformed similarity.





Different settings of perplexity

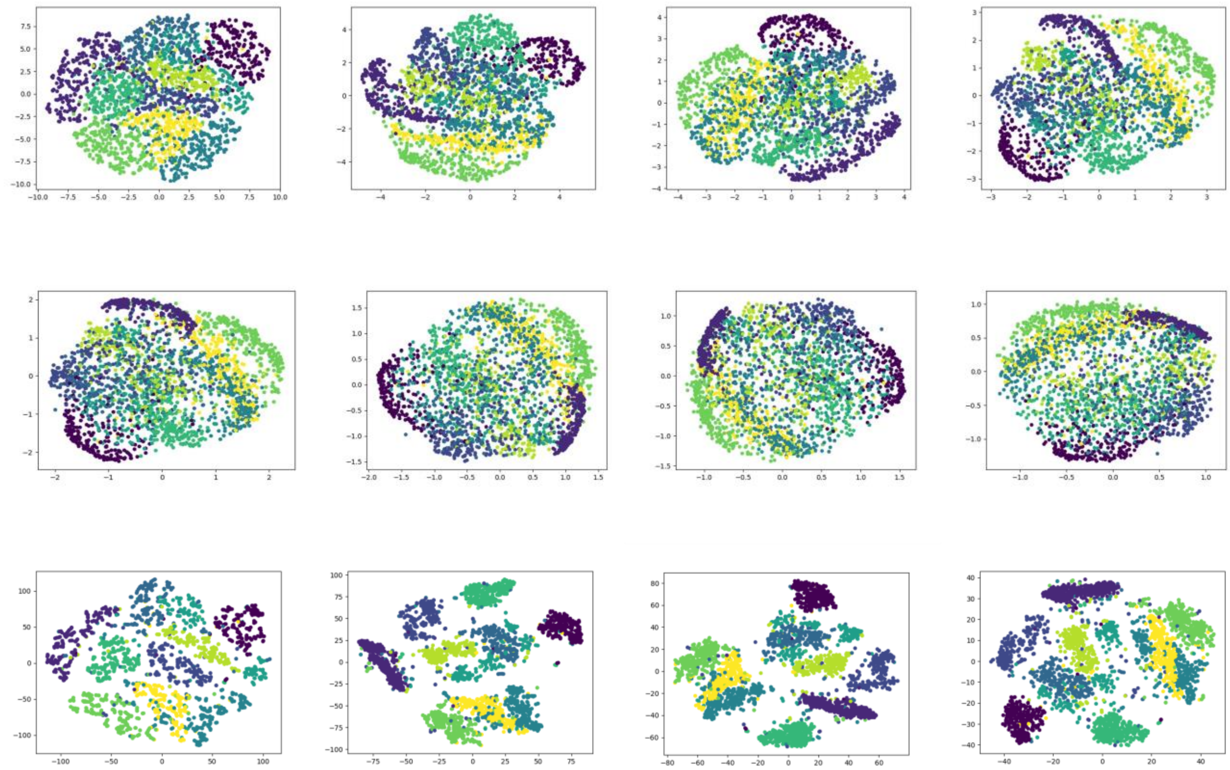
perplexity setting: 5, 30, 50, 100, 250, 500, 750, 1000

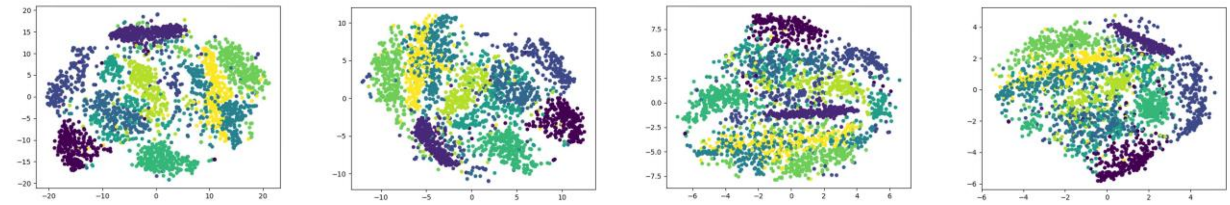
With small perplexity, there are sub-clusters (top-left image, perplexity: 5)

With large perplexity, clusters become more divergent and overlap to neighbor cluster.

symmetric sne

t-sne





What you have learned

sne, symmetric sne, t-sne 的原理、差别

perplexity 參數造成的影響

使用不同機率模型來擬和高維及低維空間資料分布

perplexity 可以解釋為一個點附近的有效鄰近點的個數