Question 1

According to Wikipedia Similarity is the inverse of distance metrics:

$$q_{ij} = \frac{(1 + \|z_i - \mu_j\|^2/\alpha)^{-\frac{\alpha+1}{2}}}{\sum_{j'}(1 + \|z_i - \mu_{j'}\|^2/\alpha)^{-\frac{\alpha+1}{2}}},$$

Nominator is proportion to inverse square of Euclidian distance. The denominator is constant for particular point. So it can be considered as similarity measure.

Question 2

The authors of the papers want to find new distribution P

Which holds following properties:

1) strengthen predictions
2) put more emphasis on data points assigned with high confidence,
3) normalize loss contribution of each centroid to prevent large clusters from distorting the hidden feature space.

By raising qi to the second power and then normalizing by frequency per cluster we make low values lower and high values greater. So all 3 properties are holds.

Question 3

We have following formula for gradient of points:

$$\frac{\partial L}{\partial z_i} = \frac{\alpha+1}{\alpha}\sum_j (1 + \frac{\|z_i - \mu_j\|^2}{\alpha})^{-1} \qquad (4)$$

$I\ assign\ new\ points\ z_i = z_i - \alpha * \frac{\delta L}{\delta z_i}$

And fit these value as target values to NN

Since NN uses $MSE = \sum(\check{y} - y)^2\ where\ \check{y} = predicted\ \ and\ y - actual\ when\ it\ calculates\ gradient\ L = \sum -2(\check{y} - y)\ \ Since\ y_i\ = z_i - \alpha * \frac{\delta L}{\delta z_{i_i}}\ \ and\ neural\ network\ predicts\ z_i\ then\ L = -2\left(z - z + a * \frac{\delta L}{\delta z_i}\right) =$

$-2 * a * \frac{\delta L}{\delta z_i}\ so\ we\ go\ in\ antigradient\ direction$

At each step alphas are proportional to inverse of square root of number iteration in order to make model converges.

Question 4

First we train layer-wise encoders and decoders. In layer wise training we construct following neural network:

Out1 = Dropout(Input)

Out2 = g1(W1*Out1+b1)

Out3 = Dropout(Out2)

Out3 = g2(W2*Out3+b2)

And we minimize (Out3-Input) ^2. All function g1, g2 are RELU except last encoder layer and last decoder layer, they are sigmoids. Dropout at each layer 0.2 After we need construct neural networks from pretrained encoders and decoders... We sequentially connect encoders and then connect decoders in mirror manner without dropout layers. So neurons on each layer: 784-500-500-2000-10-2000-500-500-784. Then we train NN trying to minimize (Out-Input) ^2 function

Question 5

Let’s take a look at Student t-distribution

$$q_{ij} = \frac{(1 + \|z_i - \mu_j\|^2/\alpha)^{-\frac{\alpha+1}{2}}}{\sum_{j'}(1 + \|z_i - \mu_{j'}\|^2/\alpha)^{-\frac{\alpha+1}{2}}},$$

Denominator can be computed in advance for each z and u O(kn) Compute particular q takes O(1). Compute each q take O(nk)

$$p_{ij} = \frac{q_{ij}^2/f_j}{\sum_{j'} q_{ij'}^2/f_{j'}},$$

Each f_j can be computed in O(n). Denominator can be computed in advance for each i and j in O(nk)

Particular p can be computed in O(1). To compute each p O(nk)

$$\begin{aligned}\frac{\partial L}{\partial z_i} \quad &= \quad \frac{\alpha+1}{\alpha}\sum_j (1 + \frac{\|z_i - \mu_j\|^2}{\alpha})^{-1} \\ &\times (p_{ij} - q_{ij})(z_i - \mu_j),\end{aligned}$$

Gradient by z can be computed in O(nk)

$$\begin{aligned}\frac{\partial L}{\partial \mu_j} \quad &= \quad -\frac{\alpha+1}{\alpha}\sum_i (1+\frac{\|z_i-\mu_j\|^2}{\alpha})^{-1} \\ &\qquad \times (p_{ij}-q_{ij})(z_i-\mu_j).\end{aligned}$$

Gradient by u can be computed in O(nk)

Moving centroid takes O(k)

Training NN takes O(h*g) where h-number of epochs g-size of NN. They are constants so O (1)

So toal complexity O(nk)