Data: high dimensional data $\mathbf{X} = \{\mathbf{X}_1, \ldots, \mathbf{X}_N\} \subseteq \mathbb{R}^D$, Target Perplexity P, Number of iterations T

Result: low dimensional data $\mathbf{Y} = \{\mathbf{Y}_1, \ldots, \mathbf{Y}_N\} \subseteq \mathbb{R}^d$ such that $d \ll D$ (usually d = 2 or 3)

compute squared eucledian distances $d_{ij} = \|\mathbf{X}_i - \mathbf{X}_j\|^2$;

for $i = 0..N$ **do**

  initialize σ_i;

  repeat

    $\forall j = 0..N \quad p_{j|i} = \frac{\exp(-d_{ij}/2\sigma_i^2)}{\sum_{k \neq i} \exp(-d_{ik}/2\sigma_i^2)}$;

    $\mathrm{per}_i = 2^{\sum_j p_{j|i} \log_2 p_{j|i}}$;

  until $per_i = P$;

end

symmetrize $p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}$;

sample inital $\mathbf{Y}$ from gaussian distribution;

for $t = 0..T$ **do**

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

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

  update $\mathbf{Y}^{(t)} = \mathbf{Y}^{(t-1)} + \eta \frac{\partial C}{\partial mathbfY} + \alpha(t)(\mathbf{Y}^{(t-1)} - \mathbf{Y}^{(t-2)})$;

end

Algorithm 1: t-SNE Algorithm