Report: Canonical t-SNE Analysis

Ilaria Bonavita

January 9, 2017

Abstract

We introduce here a method to compare two sets of variables which can be considered a variation of canonical correlation analysis in which we replaced the correlation -as criterion to compute similarities between the two sets- by a cost function highly inspired by t-SNE method.

1 Background

1.1 Canonical Correlation Analysis

Let $\mathbf{X} \in \mathbb{R}^{D_x}$ and $\mathbf{Y} \in \mathbb{R}^{D_y}$ be two random vectors (views).
The goal of CCA is to find a pair of L-dimensional projections $\mathbf{W}_1^T\mathbf{X}$, $\mathbf{W}_2^T\mathbf{Y}$ that are maximally correlated but where different dimensions within each view are constrained to be uncorrelated. Assuming that $\mathbf{X}$ and $\mathbf{Y}$ have zero mean, the CCA problem can be written as

$$\begin{aligned} &\max_{\mathbf{W}_1,\mathbf{W}_2} && \mathbb{E}\big[(\mathbf{W}_1^T\mathbf{X})^T(\mathbf{W}_2^T\mathbf{Y})\big] \\ &\text{s.t.} && \mathbb{E}\big[(\mathbf{W}_1^T\mathbf{X})(\mathbf{W}_1^T\mathbf{X})^T\big] = \mathbb{E}\big[(\mathbf{W}_2^T\mathbf{Y})(\mathbf{W}_2^T\mathbf{Y})^T\big] = \mathbf{I}. \end{aligned} \tag{1}$$

where the maximation is over the projection matrices $\mathbf{W}_1 \in \mathbb{R}^{D_x \times L}, \quad \mathbf{W}_2 \in \mathbb{R}^{D_y \times L}$.

1.2 t-Stochastic Neighbour Embedding

1.2.1 Stochastic Neighbour Embedding

Consider a set of n points in a high-dimensional space, $\mathbf{X} \in \mathbb{R}^{D_x}$ that we wish to convert into a set $\mathbf{Y} \in \mathbb{R}^{D_y}$ in a low-dimensional space.

SNE starts by converting the high-dimensional Euclidian distances between datapoints into conditional probabilities that represent similarities. The similarity of datapoint x_j to datapoint x_i is the conditional probability, $p_{j|i}$, that x_i would pick x_j as its neighbour if neighbours were picked in proportion to their probability density under a Gaussian centered at x_i with variance σ_i^2:

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

For the low-dimensional similarities between y_i and y_j a similar conditional probability $q_{j|i}$ can be defined:

$$q_{ij} = \frac{exp(-||y_i - y_j||^2)}{\sum_{k \neq i} exp(-||y_i - y_k||^2)} \tag{3}$$

where we set the variance of the Gaussian to $\frac{1}{\sqrt{2}}$.

If the map points y_i and y_j correctly model the similarity between the high-dimensional datapoints x_i and x_j, the conditional probabilities $p_{j|i}$ and $q_{j|i}$ will be equal. Hence, SNE aims to find a low-dimensional data representation that minimises the mismatch between $p_{j|i}$ and $q_{j|i}$ and it does so minimising the sum of the *Kullback-Leibler* divergences over all datapoints, i.e.:

$$\min_{\mathbf{Y}} \quad C(\mathbf{X}, \mathbf{Y}) = \sum_i KL(P_i||Q_i) = \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}. \tag{4}$$

1.2.2 t-SNE

It has been found that SNE presents two main problems: (1) the cost function is difficult to optimise, (2) a phenomenon known as the "crowding problem".

t-SNE aims to alleviate these problems is two ways: (1) it uses a symetrised version of the SNE cost function with simpler gradients, (2) it uses a Student-t distribution rather than a Gaussian to compute the similarity between two points in the low-dimensional space. To symmetrise the cost function, instead of minimising the sum of the Kullback-Leibler divergences between the conditional probabilities $p_{j|i}$ and $q_{j|i}$, we can minimise a single Kullback-Leibler divergence between a joint probability distribution, P, in the high-dimensional space and a joint probability distribution, Q, in the low-dimensional space:

$$\min_{\mathbf{Y}} \quad C(\mathbf{X}, \mathbf{Y}) = KL(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{5}$$

which has the property that $p_{ij} = p_{ji}$ and $q_{ij} = q_{ji}$.
The joint probability distribution p_{ij} can be defined as:

$$p_{ij} = \frac{p_{i|j} + p_{j|i}}{2n} \tag{6}$$

In the low-dimensional map, we can use a probability distribution that has much heavier tails than a Gaussian to convert distances into probabilities. This allows a moderate distance in the high-dimensional space to be faithfully modelled by a much larger distance in the map and eliminates the unwanted tendency of mapping two moderately dissimilar datapoints as too close.
The join probabilities q_{ij} is then defined as:

$$q_{ij} = \frac{(1 + ||(y_i - y_j)||^2)^{-1}}{\sum_{k \neq l}(1 + ||(y_k - y_l)||^2)^{-1}}. \tag{7}$$

1.2.3 Perplexity

The variance σ_i of the Gaussian centred over each high-dimensional datapoint x_i is a parameter that needs to be optimised. Any particular value of σ_i induces a probability distribution, P_i, over all of the other datapoints. This distribution has an entropy which increases as σ_i increases. A binary search is performed to find the value of σ_i that produces a P_i with a fixed perplexity specified by the user. The perplexity is defined as:

$$Perp \;\; p(P_i) = 2^{H(P_i)}, \tag{8}$$

where $H(P_i)$ is the Shannon entropy of P_i:

$$H(P_i) = -\sum_j p_{j|i} log_2 p_{j|i}. \tag{9}$$

2 Canonical t-SNE Analysis (CtsneA [TODO find a more appealing name])

2.1 Problem formulation

We now want to answer a question: is it possible to change CCA cost function in such a way that the two projection matrices found optimise a criterion different from correlation? Taking inspiration from t-SNE and its ability to preserve the local structure of the high-dimensional data in the low-dimensional space we replaced CCA cost function with a function that attempts to find two projection matrices which minimises a sum of the KL-divergences between the distribution of the two sets of datapoints in the low-dimensional space. More precisely, given $\mathbf{X} \in \mathbb{R}^{D_x}$, $\mathbf{Y} \in \mathbb{R}^{D_y}$ two random vectors (views), the problem can be formulated in this way:

$$\min_{\mathbf{W}_1, \mathbf{W}_2} \quad C(\mathbf{W}_1\mathbf{X}, \mathbf{W}_2\mathbf{Y}) = KL(\tilde{P}||\tilde{Q}) = \sum_i \sum_j \tilde{p}_{ij} \log \frac{\tilde{p}_{ij}}{\tilde{q}_{ij}} \tag{10}$$

where:

$$\tilde{p}_{j|i} = \frac{exp(-||\mathbf{W}_1(x_i - x_j)||^2/2\sigma_i^2)}{\sum_{k\neq i} exp(-||\mathbf{W}_1(x_i - x_k)||^2/2\sigma_i^2)} \tag{11}$$

$$\tilde{q}_{ij} = \frac{(1 + ||\mathbf{W}_2(y_i - y_j)||^2)^{-1}}{\sum_{k\neq l}(1 + ||\mathbf{W}_2(y_k - y_l)||^2)^{-1}}. \tag{12}$$

Although this problem might be similar to the t-SNE one, we point out here the main aspects which made the two methods conceptually and practically very different:

(1) While t-SNE tries to find a map $\mathbf{Y}$ of $\mathbf{X}$ such that the KL-divergence between the distribution of the map points and the original points is minimal, in CtsneA we already have the two sets $\mathbf{X}$ and $\mathbf{Y}$ and we want to find a space such that the distributions of the two sets of points, in the projected space, are similar, i.e. the KL-divergence between the two projected sets of point is minimised.

(2) As a consequence, minimisation in CtsneA is done with respect to $\mathbf{W}_1$ and $\mathbf{W}_2$. while in t-SNE is done with respect to $\mathbf{Y}$.

(3) In t-SNE the choice of the Student-t distribution for $\mathbf{Y}$ is motivated by the need to overcome the "crowding problem"; in CtsneA there is no obvious reason for modelling differently the distributions of $\mathbf{X}$ and $\mathbf{Y}$. Using a symmetrised KL-divergence instead of the classical one seems a more suitable criterion for our method.

Given the considerations made in point (3) we changed the cost function **??** to:

$$C_{JS}(\mathbf{W}_1\mathbf{X}, \mathbf{W}_2\mathbf{Y}) = \frac{1}{2}KL(\tilde{P}||\tilde{Q}) + \frac{1}{2}KL(\tilde{Q}||\tilde{P}) \tag{13}$$

This is only one of the possible ways to symmetrise the KL-divergence and is know as Jensen-Shannon divergence [TODO motivate].

2.2 Gradients

The minimisation of the cost function is performed using a gradient descent method. We report here the computations.
For notation simplicity, let us define the variables $d_{ji} := x_i - x_j$ and $u_{ji} := u_i - u_j$.
Consider first the cost function **??**. The gradient of C with respect to W_1 is given by:

$$\frac{\partial C}{\partial \mathbf{W}_1} = \frac{\partial KL(\tilde{P}||\tilde{Q})}{\partial \mathbf{W}_1} = \sum_i \sum_j \frac{\partial \tilde{p}_{ij}}{\partial \mathbf{W}_1}\Big(log\frac{\tilde{p}_{ij}}{\tilde{q}_{ij}} + 1\Big) \tag{14}$$

where

$$\frac{\partial \tilde{p}_{ij}}{\partial \mathbf{W}_1} = \frac{1}{2n}\Big(\frac{\partial \tilde{p}_{i|j}}{\partial \mathbf{W}_1} + \frac{\partial \tilde{p}_{j|i}}{\partial \mathbf{W}_1}\Big) \tag{15}$$

with

$$\frac{\partial \tilde{p}_{j|i}}{\partial \mathbf{W}_1} = \frac{1}{\sigma_i^2}\Big(- \mathbf{W}_1 d_{ji} d_{ji}^T\Big)\tilde{p}_{j|i} - \tilde{p}_{j|i}\Big(\sum_{k\neq i} -\frac{1}{\sigma_i^2}\mathbf{W}_1 d_{ki} d_{ki}^T \tilde{p}_{k|i}\Big) \tag{16}$$

and, similarly

$$\frac{\partial \tilde{p}_{i|j}}{\partial \mathbf{W}_1} = \frac{1}{\sigma_i^2}\Big(- \mathbf{W}_1 d_{ij} d_{ij}^T\Big)\tilde{p}_{i|j} - \tilde{p}_{i|j}\Big(\sum_{k\neq j} -\frac{1}{\sigma_j^2}\mathbf{W}_1 d_{kj} d_{kj}^T \tilde{p}_{k|j}\Big). \tag{17}$$

The gradient of C with respect to W_2 is given by:

$$\frac{\partial C}{\partial \mathbf{W}_2} = \frac{\partial KL(\tilde{P}||\tilde{Q})}{\partial \mathbf{W}_2} = \sum_i \sum_j -\frac{\tilde{p}_{ij}}{\tilde{q}_{ij}}\frac{\partial \tilde{q}_{ij}}{\partial \mathbf{W}_2} \tag{18}$$

where

$$\frac{\partial \tilde{q}_{ij}}{\mathbf{W}_2} = -\mathbf{W}_2 \tilde{q}_{ij}\Big(\frac{\mathbf{W}_2 u_{ji} u_{ij}^T}{1 + ||\mathbf{W}_2 u_{ij}||^2} + \sum_{k\neq l} -\frac{\mathbf{W}_2 u_{kl} u_{kl}^T}{1 + ||\mathbf{W}_2 u_{kl}||^2}\tilde{q}_{kl}\Big) \tag{19}$$

If we consider the cost function **??** we have that:

$$\frac{\partial C_{JS}}{\partial \mathbf{W}_1} = \sum_i \sum_j \frac{1}{2}\Big[\frac{\partial \tilde{p}_{ij}}{\partial \mathbf{W}_1}\Big(log\frac{\tilde{p}_{ij}}{\tilde{q}_{ij}} + 1 - \frac{\tilde{q}_{ij}}{\tilde{p}_{ij}}\Big)\Big] \tag{20}$$

similarly

$$\frac{\partial C_{JS}}{\partial \mathbf{W}_2} = \sum_i \sum_j \frac{1}{2}\Big[\frac{\partial \tilde{q}_{ij}}{\partial \mathbf{W}_2}\Big(log\frac{\tilde{q}_{ij}}{\tilde{p}_{ij}} + 1 - \frac{\tilde{p}_{ij}}{\tilde{q}_{ij}}\Big)\Big] \tag{21}$$

[NOTE: Is it the right way to symmetrise the cost function? In this way we are still modelling the distribution of **X** with a Gaussian and of **Y** with Student-t...]