Improving Class Averaging Using CWF

Tejal Bhamre

August 19, 2016

1 Definitions

$Y_1 = H_1X_1 + N_1$
$Y_2 = H_2X_2 + N_2$
$X_1, X_2 \sim N(\mu, \Sigma)$
$N_1, N_2 \sim N(0, \sigma^2 I_n)$

Y_1, Y_2 are independent Gaussian random vectors.

$E(Y_1) = H_1\mu$
$Cov(Y_1) = H_1\Sigma {H_1}^T + \sigma^2 I_n = K_1$
$E(Y_2) = H_2\mu$
$Cov(Y_2) = H_2\Sigma {H_2}^T + \sigma^2 I_n = K_2$

The pdf's are thus

$$f_{X_1}(x_1) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} \exp\{-\frac{1}{2}(x_1-\mu)^T\Sigma^{-1}(x_1-\mu)\} \tag{1}$$

$$f_{X_2}(x_2) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} \exp\{-\frac{1}{2}(x_2-\mu)^T\Sigma^{-1}(x_2-\mu)\} \tag{2}$$

$$f_N(y_1 - H_1x_1) = \frac{1}{(2\pi)^{\frac{n}{2}}\sigma^n} \exp\{-\frac{1}{2}(y_1-H_1x_1)^T\frac{1}{\sigma^2}(y_1-H_1x_1)\} \tag{3}$$

$$f_N(y_2 - H_2x_2) = \frac{1}{(2\pi)^{\frac{n}{2}}\sigma^n} \exp\{-\frac{1}{2}(y_2-H_2x_2)^T\frac{1}{\sigma^2}(y_2-H_2x_2)\} \tag{4}$$

$$f_{Y_1}(y_1) = \frac{1}{(2\pi)^{\frac{n}{2}}|K_1|^{\frac{1}{2}}} \exp\{-\frac{1}{2}(y_1-H_1\mu)^T{K_1}^{-1}(y_1-H_1\mu)\} \tag{5}$$

$$f_{Y_2}(y_2) = \frac{1}{(2\pi)^{\frac{n}{2}}|K_2|^{\frac{1}{2}}} \exp\{-\frac{1}{2}(y_2-H_2\mu)^T{K_2}^{-1}(y_2-H_2\mu)\} \tag{6}$$

2 Calculation

$$\begin{bmatrix} X_1 \\ Y_1 \end{bmatrix} = \begin{bmatrix} I & 0 \\ H_1 & I \end{bmatrix} \times \begin{bmatrix} X_1 \\ N_1 \end{bmatrix} \tag{7}$$

$$\sim N \left[\begin{bmatrix} \mu \\ H_1\mu \end{bmatrix}, \begin{bmatrix} \Sigma & \Sigma H_1^T \\ H_1\Sigma & H_1\Sigma H_1^T + \sigma^2 I \end{bmatrix} \right] \tag{8}$$

Using conditional distributions (Wiki MND)

$$f_{X_1|Y_1}(x_1|y_1) \sim N(\alpha, L) \tag{9}$$

$$f_{X_2|Y_2}(x_2|y_2) \sim N(\beta, M) \tag{10}$$

where

$\alpha = \mu + \Sigma H_1^T (H_1 \Sigma H_1^T + \sigma^2 I)^{-1}(y_1 - H_1\mu)$

$L = \Sigma - \Sigma H_1^T (H_1 \Sigma H_1^T + \sigma^2 I)^{-1} H_1 \Sigma$

$\beta = \mu + \Sigma H_2^T (H_2 \Sigma H_2^T + \sigma^2 I)^{-1}(y_2 - H_2\mu)$

$M = \Sigma - \Sigma H_2^T (H_2 \Sigma H_2^T + \sigma^2 I)^{-1} H_2 \Sigma$

So

$$P(x_1 - x_2|y_1, y_2) \sim N(\alpha - \beta, L + M) \tag{11}$$

Let $X_1 - X_2 = X_3$. Then

$$P(||x_3||_\infty < \epsilon|y_1, y_2) = \int_{-\epsilon}^{\epsilon} \frac{1}{(2\pi)^{\frac{n}{2}} |L + M|^{\frac{1}{2}}} \exp\{-\frac{1}{2}(x_3 - (\alpha - \beta)^T)(L + M)^{-1}(x_3 - (\alpha - \beta)\} dx_3 \tag{12}$$

For small ϵ this is

$$= \frac{(2\epsilon)^n}{(2\pi)^{\frac{n}{2}} |L + M|^{\frac{1}{2}}} \exp\{-\frac{1}{2}(\alpha - \beta)^T (L + M)^{-1}(\alpha - \beta)\} \tag{13}$$

So we can define our metric after taking log and dropping out the constant term

$$-\frac{1}{2}\log(|L + M|) - \frac{1}{2}(\alpha - \beta)^T (L + M)^{-1}(\alpha - \beta) \tag{14}$$

3 Implementation

Note that α and β defined above are denoised images obtained from CWF after deconvolution. In practice, we use α, β, L, M projected onto the subspace spanned by the principal components (of the respective angular frequency block).

We obtain an initial list of nearest neighbors for eacg image using the Initial Alignment procedure in ASPIRE (steerable PCA + bispectum). This list is then refined using the metric defined above.

We test this method on simulated projection images and observe an improvement in the nearest neighbors detected.