T-61.5030 Advanced course in neural computing

Solutions for exercise 5

1. We shall prove that PCA miminizes the difference

$$E_{\mathbf{x},\mathbf{y}}[d(\mathbf{x},\mathbf{y})^2 - d(\mathbf{x}',\mathbf{y}')^2] = E_{\mathbf{x},\mathbf{y}}[d(\mathbf{x},\mathbf{y})^2] - E_{\mathbf{x},\mathbf{y}}[d(\mathbf{x}',\mathbf{y}')^2]$$

where $\mathbf{x}$ and $\mathbf{y}$ are n-dimensional data points, $\mathbf{x}'$ and $\mathbf{y}'$ are their orthonormal projections to a lower-dimensional space, and the minimum is taken over all orthonormal projections. The equality follows from the linearity of the expectation operator. The first term on the right side does not depend on the projection, so minimizing the difference corresponds to the maximizing the second term.

Assume that $E_{\mathbf{x}}[\mathbf{x}] = 0 \Rightarrow E_{\mathbf{x}}[\mathbf{W}\mathbf{x}] = 0 \quad \forall \mathbf{W}$, where $\mathbf{W}$ is some linear projection. Note: if $\mathbf{x} = \mathbf{x}_0 + \mu$ where $E_{\mathbf{x}_0}[\mathbf{x}_0] = 0$, then $d(\mathbf{x},\mathbf{y}) = d(\mathbf{x}_0 + \mu, \mathbf{y}_0 + \mu) = d(\mathbf{x}_0, \mathbf{y}_0)$, so the assumption does not cause loss of generality. We may rewrite the expected squared distance of the projections as

$$E_{\mathbf{x},\mathbf{y}}[\|\mathbf{x}'\|^2 + \|\mathbf{y}'\|^2 - 2\mathbf{x}'^T\mathbf{y}'] = 2E_{\mathbf{x}}[\|\mathbf{x}'\|^2]$$

where the inner product term disappears because $\mathbf{x}$ and $\mathbf{y}$ are zero-mean, and the quadratic terms have equal expectations because $\mathbf{x}$ and $\mathbf{y}$ come from the same distribution.

The projection basis is an orthonormal set of m vectors, $m < n$. Denote this basis $[\mathbf{u}_1, \ldots \mathbf{u}_m]$. The projection of point $\mathbf{x}$ can be written as

$$\mathbf{x}' = \sum_{k=1}^{m} (\mathbf{u}_k^T \mathbf{x})\mathbf{u}_k$$

and its squared length is

$$\|\mathbf{x}'\|^2 = \sum_{k=1}^{m} (\mathbf{u}_k^T \mathbf{x})^2 \ .$$

The expectation of the squared length, which we must maximize, is

$$E\left[\sum_{k=1}^{m} (\mathbf{u}_k^T \mathbf{x})^2\right] = \sum_{k=1}^{m} E[\mathbf{u}_k^T \mathbf{x}\mathbf{x}^T \mathbf{u}_k] = \sum_{k=1}^{m} \mathbf{u}_k^T E[\mathbf{x}\mathbf{x}^T]\mathbf{u}_k = \sum_{k=1}^{m} \mathbf{u}_k^T \mathbf{R}\mathbf{u}_k \qquad (1)$$

where $\mathbf{R}$ is the covariance matrix of $\mathbf{x}$.

As restrictions we have $\mathbf{u}_k^T\mathbf{u}_k = 1$, and $\mathbf{u}_k^T\mathbf{u}_l = 0$ for $l \neq k$. We will take them into account with Lagrange multipliers: λ_{kk} for the first kind of restrictions and λ_{kl} for the second. Since λ_{kl} are arbitrary multipliers, for later convenience we will write $-\lambda_{kl}$ instead.

At the minimum, the $\mathbf{u}_k$-gradient of the above expression, plus the $\mathbf{u}_k$-gradients of the restriction functions times the corresponding Lagrange multipliers, is zero for all $\mathbf{u}_k$. That is, we have

$$2\mathbf{R}\mathbf{u}_k - 2\lambda_{kk}\mathbf{u}_k + 2\sum_{l=m+1, l\neq k}^{n} (-\lambda_{kl})\mathbf{u}_l = 2\left(\mathbf{R}\mathbf{u}_k - \sum_{l=m+1}^{n} \lambda_{kl}\mathbf{u}_l\right) = 0 \ , \quad 1 \leq k \leq m \ .$$

Collecting all $\mathbf{u}_k$, $k = 1, \ldots, m$, into a $n \times m$ matrix $\mathbf{U}$, and the multipliers λ_{kl}, $k, l = 1, \ldots, m$, into a $m \times m$ matrix $\mathbf{\Lambda}$, we have

$$\mathbf{RU} - \mathbf{U\Lambda} = \mathbf{0} \tag{2}$$

where $\mathbf{0}$ is a $n \times m$ matrix of zeroes.

Suppose that the off-diagonal elements of $\mathbf{\Lambda}$ are zero: $\lambda_{kl} = 0$ for $l \neq k$. Then it is easy to see that if $\mathbf{u}_k$ are some eigenvectors of $\mathbf{R}$ and λ_{kk} are the corresponding eigenvalues, we have $\mathbf{Ru}_k = \lambda_{kk}\mathbf{u}_k$ and equation (2) is satisfied. But then the expected squared length of the projection, equation (1), becomes

$$\sum_{k=1}^{m} \lambda_{kk}(\mathbf{u}_k^T \mathbf{u}_k) = \sum_{k=1}^{m} \lambda_{kk} \ .$$

The choice of eigenvectors that yields the largest value is when the λ_{kk}, $1 \leq k \leq m$, are the m largest eiqenvalues. But this is exactly the result PCA gives. Therefore PCA also minimizes our initial cost function.[1]

Note: any orthogonal rotation of the basis vectors $\mathbf{U}$ that preserves the projection space also satisfies equation (2). To show this, write $\mathbf{U}' = \mathbf{UW}$ where $\mathbf{W}$ is an (orthogonal) rotation matrix. Since $\mathbf{W}$ is orthogonal, $\mathbf{WW}^T = \mathbf{W}^T\mathbf{W} = \mathbf{I}$, $\mathbf{U} = \mathbf{U}'\mathbf{W}^T$, and equation (2) becomes

$$\mathbf{RU}'\mathbf{W}^T - \mathbf{U}'\mathbf{W}^T\mathbf{\Lambda} = \mathbf{0} \Rightarrow \mathbf{RU}'\mathbf{W}^T\mathbf{W} - \mathbf{U}'\mathbf{W}^T\mathbf{\Lambda W} = \mathbf{RU}' - \mathbf{U}'\mathbf{\Lambda}' = \mathbf{0}$$

where we denoted $\mathbf{\Lambda}' = \mathbf{W}^T\mathbf{\Lambda W}$.

2.

$$\lambda_1 = 1 + \sigma^2 \tag{3}$$

$$\mathbf{q}_1 = \mathbf{s} \tag{4}$$

$$\mathbf{X}(n) = \mathbf{s} + \mathbf{V}(n) \tag{5}$$

$$\begin{aligned}\Rightarrow \mathbf{R} = E[\mathbf{XX}^T] &= E[(\mathbf{s} + \mathbf{V})(\mathbf{s}^T + \mathbf{V}^T)] = E[\mathbf{ss}^T + \mathbf{sV}^T + \mathbf{Vs}^T + \mathbf{VV}^T] \\ &= \mathbf{ss}^T + \mathbf{s}E[\mathbf{V}^T] + E[\mathbf{V}]\mathbf{s}^T + E[\mathbf{VV}^T] = \mathbf{ss}^T + \sigma^2\mathbf{I} \end{aligned} \tag{6}$$

In the last equality, $\mathbf{V(n)}$ is zero-mean, so the terms with the mean vanish, and since $\mathbf{V(n)}$ is a white-noise component, its covariance matrix is diagonal with equal variances σ^2. Hence we get

$$\mathbf{Rq}_1 = (\mathbf{ss}^T + \sigma^2\mathbf{I})\mathbf{q}_1 = \mathbf{ss}^T\mathbf{s} + \sigma^2\mathbf{s} = (\|\mathbf{s}\|^2 + \sigma^2)\mathbf{s} = (1 + \sigma^2)\mathbf{s} = \lambda_1\mathbf{q}_1 \tag{7}$$

where we have used the definition of $\mathbf{q}_1$ in the first equality, the fact that $\mathbf{s}$ is a unit vector in the fourth, and the definitions of λ_1 and $\mathbf{q}_1$ again in the last.

[1] Technically, this argument only shows that the PCA solution makes the gradient zero, not that it maximizes the expected squared length. For a more complete discussion, see P. Baldi and K. Hornik, Neural Networks and Principal Component Analysis: Learning from Examples Without Local Minima, *Neural Networks*, Vol. 2, pages 53-58, 1989.

3. From Haykin, Eq. 8.46, we have the update formula

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \eta y(n)[\mathbf{x}(n) - y(n)\mathbf{w}(n)] \ . \tag{8}$$

Since we assume $E[\mathbf{x}(n)] = 0$, then also $E[y(n)] = E[\mathbf{w}(n)^T\mathbf{x}(n)] = 0$, and hence the variance of the output at iteration n is

$$\begin{aligned}\sigma_y^2(n) = E[y(n)^2] = E[\mathbf{w}(n)^T\mathbf{x}(n)\mathbf{x}(n)^T\mathbf{w}(n)]^T \\ = \mathbf{w}(n)^T E[\mathbf{x}(n)\mathbf{x}(n)^T]\mathbf{w}(n) = \mathbf{w}(n)^T\mathbf{R}\mathbf{w}(n)\end{aligned} \tag{9}$$

where $\mathbf{R} = E[\mathbf{x}(n)\mathbf{x}(n)^T]$ is the correlation matrix of the inputs. Note that $\mathbf{R}$ does not depend on n since the $\mathbf{x}(n)$ are drawn from the same distribution for all n.

From Haykin, Eq. 8.54 we have the convergence result (Ljung, 1977; Kushner and Clark, 1978):

$$\lim_{n\to\infty} \mathbf{w}(n) = \mathbf{q}_1, \ \text{infinitely often with probability 1} \tag{10}$$

where $\mathbf{q}_1$ is the eigenvector associated with the largest eigenvalue λ_1 of $\mathbf{R}$.

Taking the limit on both sides of (9) we have (infinitely often with probability 1)

$$\lim_{n\to\infty} \sigma_y^2(n) = (\lim_{n\to\infty} \mathbf{w}(n)^T)\mathbf{R}(\lim_{n\to\infty} \mathbf{w}(n)) = \mathbf{q}_1^T\mathbf{R}\mathbf{q}_1 \ . \tag{11}$$

By the definition of $\mathbf{q}_1$ we have $\mathbf{R}\mathbf{q}_1 = \lambda_1\mathbf{q}_1$ and $||\mathbf{q}_1|| = 1$. The above equation therefore becomes

$$\lim_{n\to\infty} \sigma_y^2(n) = \mathbf{q}_1^T\lambda_1\mathbf{q}_1 = \lambda_1||\mathbf{q}_1||^2 = \lambda_1 \tag{12}$$

which shows that as n approaches infinity, the variance of the filter output approaches (infinitely often with probability 1) the largest eigenvalue of the covariance matrix of the inputs.

4. From Haykin, Eq. (8.144) we have

$$\tilde{\mathbf{q}} = \sum_{j=1}^{N} \alpha_j \phi(\mathbf{x}_j) \ . \tag{13}$$

Multiplying $\tilde{\mathbf{q}}$ by its transpose, we form the inner product

$$\tilde{\mathbf{q}}^T\tilde{\mathbf{q}} = \sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i\alpha_j\phi^T(\mathbf{x}_i)\phi(\mathbf{x}_j) = \sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i K_{ij}\alpha_j = \boldsymbol{\alpha}^T\mathbf{K}\boldsymbol{\alpha} \ , \tag{14}$$

where $\mathbf{K}$ is the inner-product kernel matrix.

From Haykin, Eq. (8.151) we also have

$$\mathbf{K}\boldsymbol{\alpha} = \lambda\boldsymbol{\alpha} \ . \tag{15}$$

Premultiplying both sides of this equation by $\boldsymbol{\alpha}^T$, we have

$$\boldsymbol{\alpha}^T\mathbf{K}\boldsymbol{\alpha} = \lambda\boldsymbol{\alpha}^T\boldsymbol{\alpha} \ . \tag{16}$$

Combining equations (14) and (16), we have for $k = 1, \ldots, p$

$$\tilde{\mathbf{q}}_k^T \tilde{\mathbf{q}}_k = \lambda \boldsymbol{\alpha}_k^T \boldsymbol{\alpha}_k \ . \tag{17}$$

Hence, for the condition $\tilde{\mathbf{q}}_k^T \tilde{\mathbf{q}}_k = 1$ for $k = 1, \ldots, p$ to be satisfied, we require that

$$\boldsymbol{\alpha}_k^T \boldsymbol{\alpha}_k = \frac{1}{\lambda_k} \tag{18}$$

for $k = 1, \ldots, p$, where λ_p is the smallest nonzero eigenvalue of the kernel matrix $\mathbf{K}$.