Machine Learning	Prof. Dr. Mario Fritz
Sheet 7	***Tutorial 2022.07.07***

Exercise 1: Approximation interpretation of PCA

Suppose we want to find an orthogonal set of M linear basis vectors $\boldsymbol{u}_j \in \mathbb{R}^D$, and the corresponding scores $\boldsymbol{z}_i \in \mathbb{R}^M$, such that we minimize the average reconstruction error (Equation 1)

$$J = \frac{1}{N} \sum_{i=1}^{N} \|\boldsymbol{x}_i - \hat{\boldsymbol{x}}_i\|^2 = \frac{1}{N} \sum_{i=1}^{N} \|\boldsymbol{x}_i - \boldsymbol{U}\boldsymbol{z}_i\|^2 \tag{1}$$

where $\boldsymbol{U}$ is an orthonormal matrix with $\boldsymbol{u}_j$ as its j-th column. Show that the optimal solution is obtained by setting $\boldsymbol{U}^* = \boldsymbol{V}_M$, where $\boldsymbol{V}_M$ contains the M eigenvectors with largest eigenvalues of the empirical covariance matrix $\boldsymbol{\Sigma} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i \boldsymbol{x}_i^\top$. (We assume the $\boldsymbol{x}_i$ have zero mean, for notational simplicity.) Furthermore, the optimal low-dimensional encoding of the data is given by $\boldsymbol{z}_i = \boldsymbol{U}^\top \boldsymbol{x}_i$, which is an orthogonal projection of the data onto the column space spanned by the eigenvectors. (Hint: start with the case of $M = 1, 2$ and then proof by induction; Use Lagrange multipliers to include the orthonormality constraints $\boldsymbol{u}_i^\top \boldsymbol{u}_j = \delta_{ij}$.)

Solution: We use $\boldsymbol{u}_j \in \mathbb{R}^D$ to denote the j-th principal direction, $\boldsymbol{x}_i \in \mathbb{R}^D$ to denote the i-th high-dimensional observation, $\boldsymbol{z}_i \in \mathbb{R}^L$ to denote the i-th low-dimensional representation (i.e., the projection), and $\tilde{\boldsymbol{z}}_j \in \mathbb{R}^N$ to denote the $[z_{1j}, ..., z_{Nj}]$, which is the j-th component of all the low-dimensional vectors.

Let us start by estimating the best 1d solution, $\boldsymbol{u}_1 \in \mathbb{R}^D$, and the corresponding projected points $\tilde{\boldsymbol{z}}_1 \in \mathbb{R}^N$. The reconstruction error is given by:

$$\begin{aligned}
J(\boldsymbol{u}_1, \boldsymbol{z}_1) &= \frac{1}{N} \sum_{i=1}^{N} \|\boldsymbol{x}_i - z_{i1}\boldsymbol{u}_1\|^2 = \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{x}_i - z_{i1}\boldsymbol{u}_1)^\top (\boldsymbol{x}_i - z_{i1}\boldsymbol{u}_1) \\
&= \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{x}_i^\top \boldsymbol{x}_i - 2z_{i1}\boldsymbol{u}_1^\top \boldsymbol{x}_i + z_{i1}^2 \boldsymbol{u}_1^\top \boldsymbol{u}_1) \\
&= \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{x}_i^\top \boldsymbol{x}_i - 2z_{i1}\boldsymbol{u}_1^\top \boldsymbol{x}_i + z_{i1}^2)
\end{aligned}$$

where we have used the orthonormality $\boldsymbol{u}_1^\top \boldsymbol{u}_1 = 1$.

Taking derivatives w.r.t. z_{i1} and equating to zero gives:

$$\frac{\partial}{\partial z_{i1}} J(\boldsymbol{u}_1, \boldsymbol{z}_1) = \frac{1}{N}[-2\boldsymbol{u}_1^\top \boldsymbol{x}_i + 2z_{i1}] \stackrel{!}{=} 0 \quad \Rightarrow \quad z_{i1} = \boldsymbol{u}_1^\top \boldsymbol{x}_i$$

i.e., the optimal reconstruction weights are obtained by orthogonally projecting the data onto $\boldsymbol{u}_1$.

Plugging the optimal $\boldsymbol{z}_1$ into the expression of J, we obtain:

$$J(\boldsymbol{u}_1) = \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{x}_i^\top \boldsymbol{x}_i - z_{i1}^2) = \text{const} - \frac{1}{N} \sum_{i=1}^{N} z_{i1}^2$$

Also, we see that the variance of the projected data is given by:

$$\mathrm{Var}[\tilde{\boldsymbol{z}}_1] = \mathbb{E}\left[\tilde{\boldsymbol{z}}_1^2\right] - (\mathbb{E}\left[\tilde{\boldsymbol{z}}_1\right])^2 = \frac{1}{N}\sum_{i=1}^{N} z_{i1}^2 - 0 = \frac{1}{N}\sum_{i=1}^{N} z_{i1}^2$$

since $\mathbb{E}\left[z_{i1}\right] = \mathbb{E}\left[\boldsymbol{x}_i^\top \boldsymbol{u}_1\right] = \mathbb{E}\left[\boldsymbol{x}_i\right]^\top \boldsymbol{u}_1 = 0$

Therefore, minimizing the reconstruction error $J(\boldsymbol{u}_1)$ is equivalent to maximizing the variance of the projected data, i.e.,

$$\underset{\boldsymbol{u}_1}{\mathrm{argmin}}\, J(\boldsymbol{u}_1) = \mathrm{argmax}_{\boldsymbol{u}_1} \mathrm{Var}[\tilde{\boldsymbol{z}}_1]$$

The variance of the projected data can also be written as:

$$\frac{1}{N}\sum_{i=1}^{N} z_{i1}^2 = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{u}_1^\top \boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{u}_1 = \boldsymbol{u}_1^\top \hat{\boldsymbol{\Sigma}} \boldsymbol{u}_1$$

which is exactly the objective in conventional PCA, i.e.,

$$\boldsymbol{u}_1 = \mathrm{argmax}_{\boldsymbol{u}_1} \boldsymbol{u}_1^\top \hat{\boldsymbol{\Sigma}} \boldsymbol{u}_1 \quad \text{s.t.} \quad \boldsymbol{u}_1^\top \boldsymbol{u}_1 = 1$$

Solving this by using Lagrange multipliers, we see that $\boldsymbol{u}_1$ is the eigenvector of the covariance matrix $\hat{\boldsymbol{\Sigma}}$ with the largest associated eigenvalue.

Assume that it holds that $\forall j \leq M-1$ that $z_{ij} = \boldsymbol{u}_j^\top \boldsymbol{x}_i$, and $\boldsymbol{u}_j$ is the eigenvector of the covariance matrix $\hat{\boldsymbol{\Sigma}}$ with the j-th largest associated eigenvalue (And the orthogonormality holds $\boldsymbol{u}_i \boldsymbol{u}_j = \delta_{ij}$ for $i, j \leq M-1$).

Now prove the case for $j = M$.

$$J = \frac{1}{N}\sum_{i=1}^{N} \|\boldsymbol{x}_i - z_{i1}\boldsymbol{u}_1 - z_{i1}\boldsymbol{u}_2 - \ldots - z_{iM}\boldsymbol{u}_M\|^2 \tag{2}$$

Optimizing w.r.t. z_{iM} (setting the derivative of J w.r.t. z_{iM} equals to zero) gives:

$$\frac{\partial J}{\partial z_{iM}} = \frac{1}{N}[-2\boldsymbol{u}_M^\top \boldsymbol{x}_i + 2z_{iM}] \stackrel{!}{=} 0 \quad \Rightarrow \quad z_{iM} = \boldsymbol{u}_M^\top \boldsymbol{x}_i$$

Substituting the solutions for all $z_{i1}, \ldots., z_{iM}$ and $\boldsymbol{u}_1, \ldots., \boldsymbol{u}_{M-1}$, we have:

$$\begin{aligned}
J(\boldsymbol{u}_M) &= \frac{1}{N}\sum_{i=1}^{N}[\boldsymbol{x}_i^\top \boldsymbol{x}_i - \boldsymbol{u}_1^\top \boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{u}_1 - \ldots - \boldsymbol{u}_{M-1}^\top \boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{u}_{M-1} - \boldsymbol{u}_M^\top \boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{u}_M] \\
&= \frac{1}{N}\sum_{i=1}^{N}(\mathrm{const} - \boldsymbol{u}_M^\top \boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{u}_M) \\
&= \mathrm{const} - \boldsymbol{u}_M^\top \hat{\boldsymbol{\Sigma}} \boldsymbol{u}_M
\end{aligned}$$

Incorporating the orthogonormality constraints $\boldsymbol{u}_i \boldsymbol{u}_j = \delta_{ij}$ via Langrange multipliers:

$$\tilde{J}(\boldsymbol{u}_M) = -\boldsymbol{u}_M^\top \hat{\boldsymbol{\Sigma}} \boldsymbol{u}_M + \lambda_M(\boldsymbol{u}_M^\top \boldsymbol{u}_M - 1) + \sum_{j=1}^{M-1} \lambda_{jM}(\boldsymbol{u}_M^\top \boldsymbol{u}_j - 0)$$

The stationary points occur when

$$0 = 2\hat{\boldsymbol{\Sigma}} \boldsymbol{u}_M - 2\lambda_M \boldsymbol{u}_M + \sum_{j=1}^{M-1} \lambda_{jM} \boldsymbol{u}_j.$$

Left multiplying with $\boldsymbol{u}_j^\top$, and using the orthogonality constraints, we see that $\lambda_{jM} = 0$ for $j = 1, ..., M-1$.

We therefore obtain

$$\hat{\boldsymbol{\Sigma}}\boldsymbol{u}_M = \lambda_M \boldsymbol{u}_M$$

and so $\boldsymbol{u}_M$ must be an eigenvector of $\hat{\boldsymbol{\Sigma}}$ with eigenvalue λ_M. So the reconstruction error $\mathrm{const} - \boldsymbol{u}_M^\top \hat{\boldsymbol{\Sigma}} \boldsymbol{u}_M$ is minimized by choosing $\boldsymbol{u}_M$ to be the eigenvector having the largest eigenvalue amongst those not previously selected.

Exercise 2: PCA and Kernel PCA

Show that the conventional linear PCA algorithm is recovered as a special case of kernel PCA if we choose the linear kernel function given by $k(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{x}^\top \boldsymbol{x}'$.

Solution: W.l.o.g. assuming that the data is centered. For kernel PCA, the eigenvectors $\boldsymbol{a}$ of the kernel matrix K (associated with largest eigenvalues) are computed:

$$\boldsymbol{K}\boldsymbol{a} = \lambda \boldsymbol{a}$$

For linear kernel function $k(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{x}^\top \boldsymbol{x}'$, the kernel matrix is equivalent to $\boldsymbol{K} = \boldsymbol{X}\boldsymbol{X}^\top$.

Hence,

$$\begin{aligned}
& \boldsymbol{K}\boldsymbol{a} = \lambda \boldsymbol{a} \\
\iff \quad & \boldsymbol{X}\boldsymbol{X}^\top \boldsymbol{a} = \lambda \boldsymbol{a} \\
\iff \quad & \boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{X}^\top \boldsymbol{a} = \lambda \boldsymbol{X}^\top \boldsymbol{a} \\
\iff \quad & \boldsymbol{C}'\boldsymbol{X}^\top \boldsymbol{a} = \lambda \boldsymbol{X}^\top \boldsymbol{a} \\
\iff \quad & \boldsymbol{C}'\boldsymbol{u} = \lambda \boldsymbol{u}
\end{aligned}$$

where $\boldsymbol{C}' = \boldsymbol{C}$ corresponds to the (scaled) covariance matrix, $\boldsymbol{u} = \boldsymbol{X}^\top \boldsymbol{a}$ corresponds to the eigenvector of $\boldsymbol{C}'$.

The projection of data matrix in conventional PCA recovers that of linear kernel PCA:

$$\boldsymbol{X}\boldsymbol{U} = \boldsymbol{X}(\boldsymbol{X}^\top \boldsymbol{A}) = (\boldsymbol{X}\boldsymbol{X}^\top)\boldsymbol{A} = \boldsymbol{K}\boldsymbol{A}$$

Exercise 3: Probabilistic PCA

Probabilistic PCA is a simple example of the linear-Gaussian framework. First, a latent variable $\boldsymbol{z}$ is introduced (which corresponds to the principal-component subspace). Next, we define a Gaussian prior distribution $p(\boldsymbol{z})$ (Equation 3) over the latent variable, together with a Gaussian conditional distribution $p(\boldsymbol{x}|\boldsymbol{z})$ (Equation 4) for the observed variable $\boldsymbol{x}$ conditioned on the value of the latent variable. Specifically, the mean of $\boldsymbol{x}$ is a general linear function of $\boldsymbol{z}$ governed by the $D \times M$ matrix $\boldsymbol{W}$ and the D-dimensional vector $\boldsymbol{\mu}$.

$$p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}|0, \boldsymbol{I}) \tag{3}$$

$$p(\boldsymbol{x}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{W}\boldsymbol{z} + \boldsymbol{\mu}, \sigma^2 \boldsymbol{I}) \tag{4}$$

i) Compute the marginal distribution $p(\boldsymbol{x})$ and posterior distribution $p(\boldsymbol{z}|\boldsymbol{x})$.

ii) Given a data set $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^n$ of observed data points, write down the corresponding log likelihood function.

iii) Verify that the maximum likelihood solution for the parameter $\boldsymbol{\mu}$ is given by $\boldsymbol{\mu}_{\mathrm{ML}} = \bar{\boldsymbol{x}}$, where $\bar{\boldsymbol{x}}$ is the mean of the data vectors.

iv) The maximum likelihood solution for the parameter $\boldsymbol{W}$ has a form of

$$\boldsymbol{W}_{\mathrm{ML}} = \boldsymbol{U}_M(\boldsymbol{L}_M - \sigma^2\boldsymbol{I})^{1/2}\boldsymbol{R} \tag{5}$$

where $\boldsymbol{U}_M$ is a $D \times M$ matrix matrix whose columns are given by the M eigenvectors of the data covariance matrix $\boldsymbol{\Sigma}$ with the largest eigenvalues, the $M \times M$ diagonal matrix $\boldsymbol{L}_M$ has elements given by the corresponding eigenvalues λ_i, and $\boldsymbol{R}$ is an arbitrary $M \times M$ orthogonal matrix.

Show that in the limit $\sigma^2 \to 0$, the posterior mean $\mathbb{E}\left[\boldsymbol{z}|\boldsymbol{x}\right]$ for the probabilistic PCA model becomes an orthogonal projection onto the principal subspace, as in conventional PCA.

Solution:

i) Both the marginal and posterior will also be Gaussian as a result of the linear Gaussian model:

$$x = \boldsymbol{W}\boldsymbol{z} + \boldsymbol{\mu} + \boldsymbol{\varepsilon}$$

And we have

$$\begin{aligned}
\mathbb{E}\left[\boldsymbol{x}\right] &= \mathbb{E}\left[\boldsymbol{W}\boldsymbol{z} + \boldsymbol{\mu} + \boldsymbol{\varepsilon}\right] = \boldsymbol{\mu} \\
\mathrm{Cov}[\boldsymbol{x}] &= \mathbb{E}\left[(\boldsymbol{W}\boldsymbol{z} + \boldsymbol{\varepsilon})(\boldsymbol{W}\boldsymbol{z} + \boldsymbol{\varepsilon})^\top\right] \\
&= \mathbb{E}\left[\boldsymbol{W}\boldsymbol{z}\boldsymbol{z}^\top\boldsymbol{W}^\top\right] + \mathbb{E}\left[\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^\top\right] = \boldsymbol{W}\boldsymbol{W}^\top + \sigma^2\boldsymbol{I}
\end{aligned}$$

Hence,

$$p(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{C}) \qquad \text{where} \quad \boldsymbol{C} = \boldsymbol{W}\boldsymbol{W}^\top + \sigma^2\boldsymbol{I} \tag{6}$$

Making use of matrix inversion identity (Equation (C.7) in [1])

$$(\boldsymbol{A} + \boldsymbol{B}\boldsymbol{D}^{-1}\boldsymbol{C})^{-1} = \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{B}(\boldsymbol{D} + \boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{B})^{-1}\boldsymbol{C}\boldsymbol{A}^{-1}$$

to compute the $\boldsymbol{C}^{-1}$, we have

$$\boldsymbol{C}^{-1} = \sigma^{-1}\boldsymbol{I} - \sigma^{-2}\boldsymbol{W}\boldsymbol{M}^{-1}\boldsymbol{W}^\top$$

where $\boldsymbol{M} = \boldsymbol{W}^\top\boldsymbol{W} + \sigma^2\boldsymbol{I}$ is an $M \times M$ matrix.

Then, making use of Equation (2.116) from [1] (See Figure 1), we have:

$$p(\boldsymbol{z}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{M}^{-1}\boldsymbol{W}^\top(\boldsymbol{x} - \boldsymbol{\mu}), \sigma^{-2}\boldsymbol{M}^{-1}) \tag{7}$$

ii) The log likelihood function is given by:

$$\begin{aligned}
\log p(\boldsymbol{X}|\boldsymbol{\mu}, \boldsymbol{W}, \sigma^2) &= \sum_{n=1}^{N} \log p(\boldsymbol{x}_n|\boldsymbol{\mu}, \boldsymbol{W}, \sigma^2) \\
&= -\frac{ND}{2}\log(2\pi) - \frac{N}{2}\log|\boldsymbol{C}| - \frac{1}{2}\sum_{n=1}^{N}(\boldsymbol{x}_n - \boldsymbol{\mu})^\top\boldsymbol{C}^{-1}(\boldsymbol{x}_n - \boldsymbol{\mu}) \qquad (8)
\end{aligned}$$

iii) The log likelihood is a quadratic function of $\boldsymbol{\mu}$, thus taking the derivative and letting it equal to zero leads to the unique maximum solution $\boldsymbol{\mu}_{\mathrm{ML}} = \bar{\boldsymbol{x}} = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{x}_n$.

iv) Taking the limit $\sigma^2 \to 0$, we have $\boldsymbol{M} = \boldsymbol{W}^\top \boldsymbol{W}$

From the posterior in Equation 7, we have that:

$$\mathbb{E}\left[\boldsymbol{z}|\boldsymbol{x}\right] = \boldsymbol{M}^{-1}\boldsymbol{W}^\top(\boldsymbol{x}-\boldsymbol{\mu}) = \boldsymbol{M}^{-1}\boldsymbol{W}_{\mathrm{ML}}^\top(\boldsymbol{x}-\bar{\boldsymbol{x}}) = (\boldsymbol{W}_{\mathrm{ML}}^\top \boldsymbol{W}_{\mathrm{ML}})^{-1}\boldsymbol{W}_{\mathrm{ML}}^\top(\boldsymbol{x}-\bar{\boldsymbol{x}})$$

Substituting for $\boldsymbol{W}_{\mathrm{ML}}$ using Equation 4, in which we take $\boldsymbol{R} = \boldsymbol{I}$ for compatibility with conventional PCA. Using the orthogonality property $\boldsymbol{U}_M^\top \boldsymbol{U}_M = \boldsymbol{I}$ and setting $\sigma^2 = 0$, we have

$$\begin{aligned}\boldsymbol{W}_{\mathrm{ML}} &= \boldsymbol{U}_M \boldsymbol{L}_M^{1/2} \\ \mathbb{E}\left[\boldsymbol{z}|\boldsymbol{x}\right] &= \boldsymbol{L}_M^{-1/2}\boldsymbol{U}_M^\top(\boldsymbol{x}-\bar{\boldsymbol{x}})\end{aligned}$$

Note that this corresponds to the whitening operation:

1.	centering :	$\boldsymbol{x}-\bar{\boldsymbol{x}}$
2.	projection :	$\boldsymbol{U}_M^\top(\boldsymbol{x}-\bar{\boldsymbol{x}})$
2.	rescaling :	$\boldsymbol{L}_M^{-1/2}\boldsymbol{U}_M^\top(\boldsymbol{x}-\bar{\boldsymbol{x}})$

Exercise 4: Neural Network: Properties of activation functions

i) Show that the derivative of the *sigmoid* (i.e., $\sigma(z) = \frac{1}{1+e^{-z}}$) and the *tanh* (i.e., $\sigma(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$) activation function can be expressed in terms of the function value itself.

ii) Show that the derivative of the binary cross-entropy error function (Equation 10) with respect to the activation a_k for an output unit having a logistic sigmoid activation function satisfies Equation 9.

iii) Show that the derivative of the multiclass cross-entropy error function (Equation 11) with respect to the activation a_k for output units having a softmax activation function satisfies Equation 9.

$$\frac{\partial E}{\partial a_k} = y_k - t_k \tag{9}$$

Solution:

i) The derivative of *sigmoid*:

$$\begin{aligned}\sigma'(z) &= \frac{\mathrm{d}}{\mathrm{d}z}\left(\frac{1}{1+e^{-z}}\right) \\ &= -\left(\frac{1}{1+e^{-z}}\right)^2 \cdot \frac{\mathrm{d}}{\mathrm{d}z}\left(1+e^{-z}\right) \\ &= -\left(\frac{1}{1+e^{-z}}\right)^2 \left(-e^{-z}\right) \\ &= \frac{1}{1+e^{-z}} \cdot \frac{e^{-z}}{1+e^{-z}} \\ &= \sigma(z)(1-\sigma(z))\end{aligned}$$

The derivative of *tahn*:

$$
\begin{aligned}
\mathit{tahn}'(z) &= \frac{\mathrm{d}}{\mathrm{d}z}\left(\frac{e^z - e^{-z}}{e^z + e^{-z}}\right)\\
&= \frac{(e^z - e^{-z})'(e^z + e^{-z}) - (e^z - e^{-z})(e^z + e^{-z})'}{(e^z + e^{-z})^2}\\
&= \frac{(e^z + e^{-z})(e^z + e^{-z}) - (e^z - e^{-z})(e^z - e^{-z})}{(e^z + e^{-z})^2}\\
&= \frac{(e^z + e^{-z})^2 - (e^z - e^{-z})^2}{(e^z + e^{-z})^2}\\
&= 1 - \frac{(e^z - e^{-z})^2}{(e^z + e^{-z})^2}\\
&= 1 - \mathit{tahn}^2(z)
\end{aligned}
$$

Both derivatives can be expressed by the function value itself.

ii) We know that $y_k = \sigma(a_k)$ where σ is the logistic sigmoid function. As shown above, we have the derivative $\sigma' = \sigma(1-\sigma)$. Thus, differentiating Equation 10 w.r.t. the activation a_k corresponding to a particular data point k, we obtain

$$
\begin{aligned}
\frac{\partial E(\boldsymbol{w})}{\partial a_k} &= \frac{\partial}{\partial a_k}\left[-\left(t_k \cdot \log y_k + (1-t_k)\cdot \log\left(1-y_k\right)\right)\right]\\
&= -\left(t_k \frac{1}{y_k} y_k(1-y_k) + (1-t_k)\frac{1}{1-y_k}(-y_k(1-y_k))\right)\\
&= -t_k(1-y_k) + (1-t_k)y_k\\
&= -t_k + y_k
\end{aligned}
$$

iii) Similar to ii), we first denote $y_{kn} = y_k(\boldsymbol{x}_n, \boldsymbol{w})$ the k-th entry of the output vector on data point n. We know that $y_{kn} = \frac{\exp\left(a_{kn}\right)}{\sum_{j=1}^{K} exp(a_{jn})}$. and the derivative of softmax activation function is:

$$
\frac{\partial y_{kn}}{\partial a_{jn}} = y_{kn}(\delta_{kj} - y_{jn})
$$

Therefore,

$$
\begin{aligned}
\frac{\partial E(\boldsymbol{w})}{\partial a_{jn}} &= -\sum_{k=1}^{K} t_{kn}\frac{1}{y_{kn}}\left(y_{kn}\left(\delta_{kj} - y_{jn}\right)\right)\\
&= -\sum_{k=1}^{K} t_{kn}(\delta_{kj} - y_{jn})\\
&= -\sum_{k=1}^{K} t_{kn}\delta_{kj} + \sum_{k=1}^{K} t_{kn}y_{jn}\\
&= -t_{jn} + y_{jn}
\end{aligned}
$$

where we have used the fact that $\sum_{k=1}^{K} t_{kn} = 1$.

Exercise 5: Neural Network: Probabilistic Interpretation of Classification Models

i) Consider a binary classification problem in which the target values are $t \in \{0,1\}$, with a network output $y(\boldsymbol{x}, \boldsymbol{w})$ that represents $p(t=1|\boldsymbol{x})$, and suppose that there is a probability ε

that the class label on a training data point has been incorrectly set. Assuming independent and identically distributed data, write down the error function corresponding to the negative log likelihood. Verify that when $\varepsilon = 0$, the error function is reduced to the usual cross-entropy error function:

$$E(\boldsymbol{w}) = -\sum_{n=1}^{N} \{t_n \cdot \log y_n + (1 - t_n) \cdot \log(1 - y_n) \tag{10}$$

Note that this error function (that consider mislabelling) makes the model robust to incorrectly labelled data, in contrast to the usual cross-entropy error function.

ii) Show that maximizing likelihood for a multiclass neural network model in which the network outputs have the interpretation $y_k(\boldsymbol{x}, \boldsymbol{w}) = p(t_k = 1|\boldsymbol{x})$ is equivalent to the minimization of the cross-entropy error function:

$$E(\boldsymbol{w}) = -\sum_{n=1}^{N} \sum_{k=1}^{K} \{t_{kn} \cdot \log\, y_k(\boldsymbol{x}_n, \boldsymbol{w})\} \tag{11}$$

Solution:

i) First, we use t to denote the observed target label, and t_r to denote the real label. Then, we have that:

$$p(t = 1|\boldsymbol{w}, \boldsymbol{x}) = (1 - \varepsilon) \cdot p(t_r = 1|\boldsymbol{w}, \boldsymbol{x}) + \varepsilon \cdot p(t_r = 0|\boldsymbol{w}, \boldsymbol{x})$$
$$p(t = 0|\boldsymbol{w}, \boldsymbol{x}) = (1 - \varepsilon) \cdot p(t_r = 0|\boldsymbol{w}, \boldsymbol{x}) + \varepsilon \cdot p(t_r = 1|\boldsymbol{w}, \boldsymbol{x})$$

Note that the network is aimed to predict the real label t_r instead of the noisy one t, i.e., we model $p(t_r = 1|\boldsymbol{w}, \boldsymbol{x}) = y(\boldsymbol{w}, \boldsymbol{x})$.

Hence,

$$p(t = 1|\boldsymbol{w}, \boldsymbol{x}) = (1 - \varepsilon) \cdot y(\boldsymbol{w}, \boldsymbol{x}) + \varepsilon \cdot (1 - y(\boldsymbol{w}, \boldsymbol{x}))$$
$$p(t = 0|\boldsymbol{w}, \boldsymbol{x}) = (1 - \varepsilon) \cdot (1 - y(\boldsymbol{w}, \boldsymbol{x})) + \varepsilon \cdot y(\boldsymbol{w}, \boldsymbol{x})$$

Combining the two cases ($t = 0/1$), we have:

$$p(t|\boldsymbol{w}) = (1 - \varepsilon) \cdot y^t (1 - y)^{1-t} + \varepsilon (1 - y)^t y^{1-t}$$

Given a data set with N points, the error function corresponding to the negative log likelihood function is then:

$$E(\boldsymbol{w}) = -\sum_{n=1}^{N} \log \left((1 - \varepsilon) \cdot y_n^{t_n} \, (1 - y_n)^{1-t_n} + \varepsilon (1 - y_n)^{t_n} y_n^{1-t_n} \right) \tag{12}$$

When $\varepsilon = 0$, it is obvious that the equation above will reduce to Equation 10.

ii) For the given interpretation of $y_k(\boldsymbol{x}, \boldsymbol{w})$, the probability that the observed sample has target vector $\boldsymbol{t}$ is given by:

$$p(\boldsymbol{t}|\boldsymbol{w}) = \prod_{k=1}^{K} y_k^{t_k}$$

Then, for a data set of N points, the log likelihood function will be:

$$l(\boldsymbol{w}) = \sum_{n=1}^{N} \log p(\boldsymbol{t}_n|\boldsymbol{w}) = \sum_{n=1}^{N} \log \left(\prod_{k=1}^{K} y_k^{t_k} \right) = \sum_{n=1}^{N} \sum_{k=1}^{K} t_{kn} \log y_{kn}$$

Maximizing the log likelihood function w.r.t. $\boldsymbol{w}$ is exactly minimizing the cross-entropy in Equation 11.

Appendix

Marginal and Conditional Gaussians

Given a marginal Gaussian distribution for $\mathbf{x}$ and a conditional Gaussian distribution for $\mathbf{y}$ given $\mathbf{x}$ in the form

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Lambda}^{-1}) \quad (2.113)$$

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}|\mathbf{A}\mathbf{x} + \mathbf{b}, \mathbf{L}^{-1}) \quad (2.114)$$

the marginal distribution of $\mathbf{y}$ and the conditional distribution of $\mathbf{x}$ given $\mathbf{y}$ are given by

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y}|\mathbf{A}\boldsymbol{\mu} + \mathbf{b}, \mathbf{L}^{-1} + \mathbf{A}\boldsymbol{\Lambda}^{-1}\mathbf{A}^{\mathrm{T}}) \quad (2.115)$$

$$p(\mathbf{x}|\mathbf{y}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\Sigma}\{\mathbf{A}^{\mathrm{T}}\mathbf{L}(\mathbf{y} - \mathbf{b}) + \boldsymbol{\Lambda}\boldsymbol{\mu}\}, \boldsymbol{\Sigma}) \quad (2.116)$$

where

$$\boldsymbol{\Sigma} = (\boldsymbol{\Lambda} + \mathbf{A}^{\mathrm{T}}\mathbf{L}\mathbf{A})^{-1}. \quad (2.117)$$

Figure 1: Commonly used results for linear Gaussian models.

References

[1] C. M. Bishop. *Pattern recognition and machine learning*. springer, 2006.

[2] J. Friedman, T. Hastie, R. Tibshirani, et al. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.

[3] K. P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.