Cortical representation learning regulated by acetylcholine

Junichiro Hirayama†, Junichiro Yoshimoto‡† and Shin Ishii†‡
{junich-h, juniti-y, ishii}@is.aist-nara.ac.jp
† Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara 630-0192, Japan
‡ CREST, Japan Science and Technology Agency

January 29, 2004

Abstract

A brain needs to detect an environmental change and to quickly learn internal representations necessary in a new environment. This article presents a theoretical model of cortical representational learning that can adapt to dynamic environments, incorporating the results by previous studies on a functional role of acetylcholine (ACh). We adopt the probabilistic principal component analysis (PPCA) as a functional model of cortical representational learning, and present an on-line learning method for PPCA according to Bayesian inference. Our approach is examined in two types of simulations with synthesized and realistic datasets, in which our model is able to re-learn new representation bases after environmental changes. Our model implies the possibility that a higher-level recognition regulates the cortical ACh release in the lower-level, and that the ACh level alters the learning dynamics of a local representation unit in order to continuously acquire appropriate representations in a dynamic environment.

1 Introduction

External environments surrounding an animal can be regarded as static for a short time, but dynamic over a long time period. To adapt to such environments, a brain needs to detect an environmental change and to quickly learn internal representations necessary in a new environment. This article presents a theoretical model of cortical representational learning that can adapt to dynamic environments, incorporating results by previous studies on the functional role of acetylcholine (ACh), a neuromodulatory chemical [10, 7, 5].

Theoretically, learning of internal representation in the cortex can be modeled by means of probabilistic generative models [4, 14, 11, 16]. In this article, we adopt the probabilistic principal component analysis (PPCA) [18] as a simple model of a cortical representational system and highlight the functional role of ACh within the learning of this model. We intend to discuss how our model adapts to environmental changes but not the validity of our probabilistic model itself.

2 Model

2.1 Detecting environmental changes

To allow a PPCA to detect an environmental change, we introduce an additional structure to the probabilistic generative model of PPCA. The generative model for an n-dimensional observed variable $\boldsymbol{x}_t \in \Re^n$ is given by

$$\boldsymbol{x}_t = \boldsymbol{\Theta}\tilde{\boldsymbol{y}}_t + \boldsymbol{\xi}_t + z_t\boldsymbol{\zeta}_t, \tag{1}$$

where $\boldsymbol{\xi}_t \sim \mathcal{N}_n(\boldsymbol{\xi}_t \mid \boldsymbol{0}, \sigma_x^2\boldsymbol{I}_n), \boldsymbol{\zeta}_t \sim \mathcal{N}_n(\boldsymbol{\zeta}_t \mid \boldsymbol{0}, \sigma_\zeta^2\boldsymbol{I}_n)$, and t denotes the discrete time. The first two terms provide the original PPCA generative model with the following notations: $\boldsymbol{\Theta} \equiv (\boldsymbol{W}, \boldsymbol{\mu}) \in \Re^{n\times(m+1)}$ and $\tilde{\boldsymbol{y}}_t \equiv (\boldsymbol{y}_t', 1)' \in \Re^{(m+1)}$, where a prime ($'$) denotes a transpose. $\boldsymbol{y}_t \equiv (y_{t,1}, \cdots, y_{t,m})' \in \Re^m$ $(m \le n)$ is a latent variable corresponding to a principal component score, which is generated independently at each time step from a standard Gaussian distribution. $\boldsymbol{\xi}_t \in \Re^n$ is a white noise and $\mathcal{N}_p(\cdot \mid \cdot, \cdot)$ denotes a p-dimensional Gaussian density function[1]. $\boldsymbol{I}_n$ is an $n \times n$ identity matrix, and σ_x^2 $(\sigma_x^2 > 0)$ is an observation noise variance which is assumed to be a known constant for simplicity. $\boldsymbol{W} \equiv (\boldsymbol{w}_1, \cdots, \boldsymbol{w}_m) \in \Re^{n\times m}$ is the principal component loading matrix, where each column $\boldsymbol{w}_j \in \Re^n (j = 1, \cdots, m)$ is a principal

[1] $\mathcal{N}_p(\boldsymbol{x} \mid \boldsymbol{m}, \boldsymbol{\Sigma}) \equiv (2\pi)^{-p/2}|\boldsymbol{\Sigma}|^{-1}e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{m})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{m})}$, where $\boldsymbol{x} \in \Re^p$ is a random vector. $\boldsymbol{m} \in \Re^p$ and $\boldsymbol{\Sigma} \in \Re^{p\times p}$ are a mean vector and a covariance matrix, respectively.

component vector. $\boldsymbol{\mu} \in \Re^n$ is the expected observation. The third term is an additional noise, where σ_ζ^2 is a constant noise variance and is known. $z_t \in \{0,1\}$ is an indicator variable regarded as a latent variable. The joint probability distribution for a triplet $(\boldsymbol{x}_t, \boldsymbol{y}_t, z_t)$ is given by

$$p(\boldsymbol{x}_t, \boldsymbol{y}_t, z_t = 0 \mid \boldsymbol{\Theta}) = (1-r)\mathcal{N}_m(\boldsymbol{y}_t \mid \boldsymbol{0}, \boldsymbol{I}_m)\mathcal{N}_n\left(\boldsymbol{x}_t \mid \boldsymbol{\Theta}\tilde{\boldsymbol{y}}_t, \sigma_x^2 \boldsymbol{I}_n\right), \quad (2a)$$

$$p(\boldsymbol{x}_t, \boldsymbol{y}_t, z_t = 1 \mid \boldsymbol{\Theta}) = r\mathcal{N}_m(\boldsymbol{y}_t \mid \boldsymbol{0}, \boldsymbol{I}_m)\mathcal{N}_n\left(\boldsymbol{x}_t \mid \boldsymbol{\Theta}\tilde{\boldsymbol{y}}_t, \sigma_\epsilon^2 \boldsymbol{I}_n\right), \quad (2b)$$

where $\sigma_\epsilon^2 = \sigma_x^2 + \sigma_\zeta^2$. The prior probability for the indicator variable, $P(z_t = 1) = 1 - P(z_t = 0) = r$, is assumed to be known such to represent the *a priori* knowledge of the occurrence probability of environmental changes.

The model (1) can be regarded as a simple version of the mixture of PPCA [19] consisting of two PPCA components, (2a) and (2b), which have the same parameter $\boldsymbol{\Theta}$ except for different Gaussian noises. This model can detect an environmental change according to the following mechanism. Consider a situation where the model parameter $\boldsymbol{\Theta}$ has been estimated from the previous observations $\boldsymbol{x}_{t-1}, \boldsymbol{x}_{t-2}, \ldots$ and then a new observation $\boldsymbol{x}_t$ is given. If $\boldsymbol{x}_t$ is generated in the current environment, it can be described sufficiently by the component (2a) with the regular noise variance σ_x^2, thus the posterior probability of $z_t = 1$ becomes small. If $\boldsymbol{x}_t$ is generated in a novel environment, it can be regarded as an outlier in the component (2a). In this case, the observation can be described better by the component (2b) with a larger noise variance σ_ϵ^2, thus the posterior probability of $z_t = 1$ becomes large. Accordingly, the posterior probability of $z_t = 1$ can be viewed as a confidence of environmental change between time steps $t-1$ and t.

2.2 On-line Variational Bayes learning

We used the on-line variational Bayes (VB) method [17] to infer the model parameter $\boldsymbol{\Theta}$. Let $(X_{1:t}, Y_{1:t}, Z_{1:t}) \equiv \{(\boldsymbol{x}_\tau, \boldsymbol{y}_\tau, \boldsymbol{z}_\tau) \mid \tau = 1, \ldots, t\}$ be a sequence of observations and corresponding latent variables. The objective of the Bayesian inference is to obtain a posterior distribution of unknown variables, $p(Y_{1:t}, Z_{1:t}, \boldsymbol{\Theta} \mid X_{1:t})$, when given observation variables $X_{1:t}$. For this purpose, an on-line variational free energy with a time-dependent forgetting factor $\lambda(s) \in [0,1]$ $(s = 1, \ldots, t)$ is defined by

$$F^\lambda[q](t) = T^\lambda(t)L^\lambda(t) - H(t) \quad (3a)$$

$$L^\lambda(t) = \eta(t) \sum_{\tau=1}^{t} \left(\prod_{s=\tau+1}^{t} \lambda(s) \right) \times E\left[\log \frac{p(\boldsymbol{x}_\tau, \boldsymbol{y}_\tau, z_\tau \mid \boldsymbol{\Theta})}{q_\tau(\boldsymbol{y}_\tau, z_\tau | \boldsymbol{x}_\tau)}\right] \quad (3b)$$

$$H(t) = E\left[\log \frac{q_\theta(\boldsymbol{\Theta} \mid X_{1:t})}{p(\boldsymbol{\Theta})}\right], \quad (3c)$$

where a trial distribution $q(Y_{1:t}, Z_{1:t}, \boldsymbol{\Theta} \mid X_{1:t})$ is introduced to approximate the true posterior distribution $p(Y_{1:t}, Z_{1:t}\boldsymbol{\Theta} \mid X_{1:t})$, and is assumed to be factorized as $q_\theta(\boldsymbol{\Theta} \mid X_{1:t}) \prod_{\tau=1}^{t} q_\tau(\boldsymbol{y}_\tau, z_\tau \mid \boldsymbol{x}_\tau)$. $E[\cdot]$ denotes the expectation over the trial distribution q. Furthermore, $p(\boldsymbol{\Theta})$ is the prior distribution of $\boldsymbol{\Theta}$. $T^\lambda(t) \equiv \sum_{\tau=1}^{t} \left(\prod_{s=\tau+1}^{t} \lambda(s)\right)$ is an effective data number and $\eta(t) \equiv 1/T^\lambda(t)$ is the normalization term. The on-line VB method for the model (1) is derived as a sequential maximization process of the variational free energy (3). When a datum $\boldsymbol{x}_t$ is observed at time t, F^λ is maximized with respect to q_t in the on-line VB-Estep while q_τ $(\tau = 1, \ldots, t-1)$ and q_θ are fixed. In the next step, called the on-line VB-Mstep, F^λ is maximized with respect to q_θ while q_τ $(\tau = 1, \ldots, t)$ is fixed. These two steps are executed every time a new datum is observed.

The on-line VB method needs only to maintain the expected sufficient statistics, instead of storing all observed data; this scheme is more natural for learning by animals than the batch one. The expected sufficient statistics are defined by

$$\langle f(\boldsymbol{x}, \boldsymbol{y}, z)\rangle(t) = \eta(t) \sum_{\tau=1}^{t} \left(\prod_{s=\tau+1}^{t} \lambda(s) \right) E[f(\boldsymbol{x}_\tau, \boldsymbol{y}_\tau, z_\tau)], \quad (4)$$

where $f(\boldsymbol{x}, \boldsymbol{y}, z)$ is given by a quadratic function of $\boldsymbol{x}$, $\boldsymbol{y}$ and z_t. This calculation can be done incrementally as

$$\langle f(\boldsymbol{x}, \boldsymbol{y}, z)\rangle(t) = (1 - \eta(t))\langle f(\boldsymbol{x}, \boldsymbol{y}, z)\rangle(t-1) + \eta(t)E[f(\boldsymbol{x}_t, \boldsymbol{y}_t, z_t)], \quad (5)$$

where the normalization term $\eta(t)$ acts as the learning rate to control the speed of updating $\langle f(\boldsymbol{x}, \boldsymbol{y}, z)\rangle(t)$. $\eta(t)$ can also be calculated incrementally, because its reciprocal $T^\lambda(t)$ is given by the following step-wise equation:

$$T^\lambda(t) = 1 + \lambda(t)T^\lambda(t-1). \quad (6)$$

It is shown that the on-line VB method achieves a stochastic approximation of the Bayesian inference if scheduling like $\lambda(s) \stackrel{s\to\infty}{\longrightarrow} 1$ is used [17].

2.3 Regulating learning dynamics

The learning rate $\eta(t)$ regulates the updating speed of expected sufficient statistics as shown in Eq. (5), so that it balances the adaptability and stability of the on-line learning process. Because $\eta(t)$ is dependent on the sequence of $\lambda(t)$ as shown in Eq. (6), in which $T^{\lambda}(t)$ is the reciprocal to $\eta(t)$, the scheduling of the forgetting factor $\lambda(t)$ is essential for modulating the dynamics of on-line learning. We introduces here the following scheduling scheme:

$$\lambda(t) = (1-\alpha)\,\lambda(t-1) + \alpha(1 - q_t(z_t = 1)), \quad (7)$$

where $q_t(z_t = 1)$ denotes the posterior probability of $z_t = 1$, which informs of the occurrence of an environmental change. α $(0 < \alpha < 1)$ is a smoothing constant to reduce an excessive sensitivity to outliers that may appear even in a static environment. It is expected that when an environment changes, a temporal increase of $q_t(z_t = 1)$ results in the decrease of $\lambda(t)$, and facilitates the learning by placing more weight on the recent data than the previous data.

Moreover, we also introduce a refractory period (RP) into the scheduling scheme, in order that $\lambda(t)$ recovers after dropping to almost zero in response to an environmental change; if $\lambda(t)$ is below a threshold ϕ at time $t = t_0$, $q_t(z_t = 1)$ in Eq. (7) is explicitly replaced by 0 during $t = t_0 + 1, \ldots, t_0 + \nu$, where ϕ and ν are known constants. The RP is especially required for high-dimensional cases as shown in the simulation in Section 3.2

This on-line VB learning achieves stochastic approximation of the Bayesian inference if the environment continues to be static, since $q_t(z_t = 1) = 0$ for any t implies $\lambda(t) \stackrel{t\to\infty}{\longrightarrow} 1$ from any $\lambda(0)$.

3 Simulations

Our learning model was evaluated using two types of computer simulations, which employed synthesized and real datasets.

3.1 Synthesized data

The basic features of our approach were examined by using simple two-dimensional synthesized data. A two-dimensional vector $\boldsymbol{x}_t$ was generated according to Eq. (1) with $z_t = 0$ at each time step t, where the actual parameter $\boldsymbol{\Theta} = (\boldsymbol{W}, \boldsymbol{\mu})$ was usually fixed but occasionally changed as follows: $\boldsymbol{\Theta} = \begin{pmatrix} 5 & 10 \\ -1 & 10 \end{pmatrix}$ for $t = 1, \ldots, 200$, $\begin{pmatrix} 1 & -10 \\ 5 & 10 \end{pmatrix}$ for $t = 201, \ldots, 400$, and $\begin{pmatrix} -3 & -10 \\ 3 & -10 \end{pmatrix}$ for $t = 401, \ldots, 600$. Although the on-line variational free energy (3a) may be useful for the on-line determination of the principal component dimensionality m, we assumed here a situation in which m of actual data was known as $m = 1$. The known constants were set as follows: $\sigma_x^{-2} = 10^{-2}, \sigma_\zeta^{-2} = 10^{-6}, r = 0.001, \alpha = 0.05$, and $\gamma = 0.001$. In this simulation, the RP was not used.

Figure 1 shows learning processes under the following three conditions: 1) the forgetting factor was fixed at $\lambda(t) = 1$ for any t; 2) fixed at $\lambda(t) = 0.8$ for any t; and, 3) $\lambda(t)$ was scheduled by Eq. (7). Only the direction of the estimated principal component vector, represented as the angle from the x_1-axis, is shown in this figure. The estimator of the learning model is given as the expectation[2] of the model parameter over the trial posterior distribution. When the forgetting factor $\lambda_m(s) (s = 1, \ldots, T)$ was set at constant of 1, the estimator closely approached the true value in a static environment, but it could not follow environmental changes. When the forgetting factor was set at a smaller constant of 0.8, the estimator could alter its value in response to environmental changes, but a high variance remained. Because of this variance, the estimator could not be improved in a static environment even when the time elapsed. When the forgetting factor was regulated by our method, on the other hand, the inference exhibited high performance. Namely, the estimator could alter its value just after the environment change, while it was improved in a static environment as the number of observed data increased. The figure also shows the modulation of $\lambda(t)$ and $\eta(t)$, which makes the learning flexible as described above.

3.2 Realistic data: face images

Our approach was evaluated by using a dataset of realistic face images, assuming that a representational system is used for our recognition of images. Basis vectors (principal components) extracted from a set of face images using PCA were called "eigenfaces" [20]. The dataset used here consists of 100 gray-scale photographs of frontal faces and 100 of half-profile faces, registered in Yale Face Database B [8]. The subjects in this dataset were six males and one female in various lighting conditions. We standardized the images such that all the images contained 49×41 pixels, and the centers of eyes for frontal views and the centers of faces for half-profile views took the same coordinate. The pixel values were normalized

[2] In our case, the posterior distribution is Gaussian, thus the parameter expectation is identical to its mean.

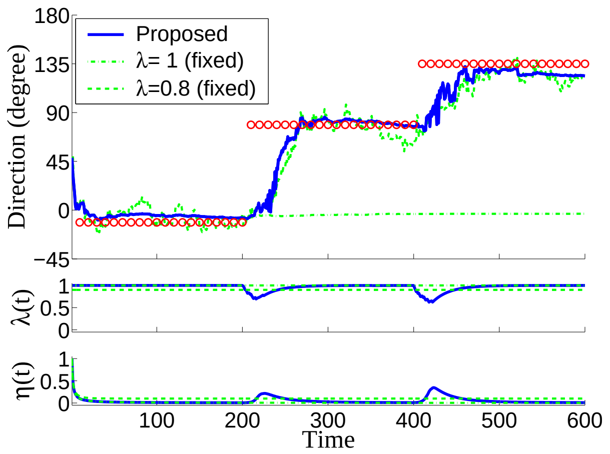

Figure 1: The direction of the estimated principal component vector and modulatory variables. The horizontal axis denotes the time step t. The top panel shows the estimator in the following three cases: 1) $\lambda(t) = 1.0$; 2) $\lambda(t) = 0.8$; and 3) $\lambda(t)$ is controlled by our proposed scheduling scheme. Only the direction of the principal component vector, the angle from the x_1-axis, is shown. A mark '$\circ$' represents a real value in each time step. The middle and bottom panels show the time series of $\lambda(t)$ and $\eta(t)$, respectively.

to be within $[0, 1]$, thus each image was represented as a 2,009-dimensional vector of normalized pixel values. Figure 2 shows the first five eigenfaces extracted from the frontal and half-profile face images, using the standard PCA.

Figure 2: The first five eigenfaces (first to fifth from left to right) extracted by the standard PCA from the frontal (upper row) and half-profile (lower row) face images.

The learning process was divided into two phases. In the first phase, 100 observations were randomly selected from the frontal faces and sequentially provided to the learning model. This phase is called the "frontal condition." In the next phase, called the "half-profile condition," 100 observations were selected from half-profile faces. The principal component dimensionality m was fixed at 14, and the known constants were set as: $\sigma_x^{-2} = 60, \sigma_\zeta^{-2} = 5, r = 0.001, \alpha = 0.02$, and $\gamma = 0.001$. The scheduling of the forgetting factor described in Section 2.2 was used with or without the RP, where $\phi = 0.05$ and $\nu = 30$.

Figure 3 shows the obtained first eigenface with the largest norm during the on-line learning process with the RP. In the latter half of the frontal condition, at $t = 60$ and 90, the eigenface successfully captured the features of frontal faces. The eigenface was then modified quickly into that of half-profile faces at $t = 130, 150$ and 180, through a transient phase like at $t = 110$. The figure also shows the learning processes with or without the RP, which are evaluated in comparison with the result by usual PCA. The eigenface obtained by our on-line learning without the RP did not approach that by the usual PCA; in contrast, that with the RP behaved well as the time elapsed within both the frontal and half-profile conditions. The time courses of $\eta(t)$ and $\lambda(t)$ are also shown in this figure. In the case with the RP, $\eta(t)$ and $\lambda(t)$ shift in time to properly adapt to the condition change.

4 Discussion

Physiological studies have reported that a high ACh level facilitates the plasticity of receptive field and reorganization of representational maps [2, 12, 6, 3, 15]. Moreover, it has also reported that cortical ACh levels tend to increase when facing novel stimuli or environments and to gradually decrease through habituation processes [1, 13, 9]. These works provide evidences that a cortical ACh level increases in response to an environmental change and regulate the learning of cortical representational system.

The simulations in Section 3 showed that our online VB learning could alter its dynamics in order to re-learn new representation bases necessary in novel environments. The dynamics change is based essentially on modification of the learning rate $\eta(t)$ in response to environmental changes. Here, we present an idea that the cortical ACh level corresponds functionally to the learning rate in the learning of cortical representational systems. This idea is based mainly on the following computational perspective on the functional role of ACh, which has been presented by Hasselmo [10] standing on physiological facts: 1) a high ACh level within a local circuit leads to a predominant influence of external stimuli, which induces learning of new memories; and 2) a low ACh level,

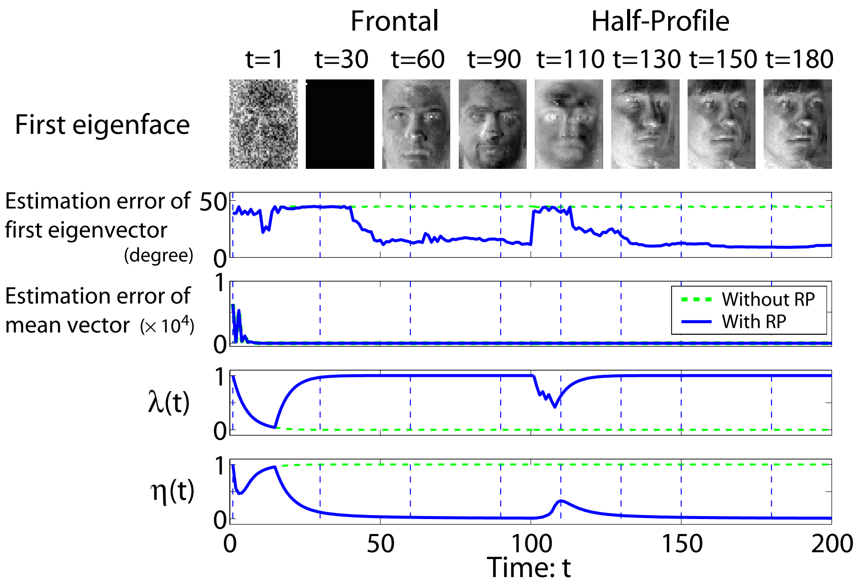

Figure 3: Top row: the first eigenface obtained during the on-line learning process. The other four rows, from the second to the bottom: the angle between the first basis vector (i.e., with the largest norm) obtained in our learning and that by the standard PCA, the distance between the estimated mean vector and the true mean, $\lambda(t)$ and $\eta(t)$, where each dash or solid line denotes the case without the refractory period (RP) or with the RP, respectively. A set of dashed vertical lines in each panel denotes the time steps at which the first eigenfaces on the top row are displayed.

in contrast, leads to a predominant influence of local intrinsic activities, which corresponds to recalling of previously-learned information. This function of ACh is analogous to the function of $\eta(t)$ in our on-line learning, because it regulates the updating of expected sufficient statistics which can be regarded as memories of past observations and inferences.

Our model is related to the previous studies by Yu and Dayan [21, 22], in both of which they used a probabilistic generative model as a model of cortical processing and incorporate a function of ACh into the model. They have advocated that cortical ACh level conveys an "uncertainty" information about the inference or learning, while we hypothesized an ACh level as the learning rate in cortical representational learning. Their three-layered hidden Markov model [21] assumed that an ACh signal conveyed an top-down information to the intermediate hidden layer and influenced its inference. Their subsequent model [22], a factor analysis model in which the mean vector of a Gaussian hidden variable shifted in time, assumed that an ACh level reported an uncertainty about the posterior of the mean vector of hidden variable and influenced its learning. Our model employs a similar idea to [21], because the ACh level in our model is regulated by a kind of top-down information, i.e., the posterior probability of the index variable, $q_t(z_t)$. However, our hypothesized function of ACh in learning is different from that in [22]: the ACh level in [22] influenced only the learning of the mean vector as mentioned above, and the learning of basis vectors, which was essentially important for obtaining appropriate internal representations, was not explicitly addressed; in contrast, the ACh level in our model influences the learning of basis vectors by controlling the updating of expected sufficient statistics. In addition, the hypothesized function of ACh in our model shows a consistency with the Hasselmo's hypothesis more clearly than that in [22], because the learning rate in our model controls the updating of memories more directly than the uncertainty in [22].

5 Conclusion

We proposed an on-line learning scheme for a modified mixture of PPCA, which could alter its dynamics in order to re-learn new representation bases necessary in novel environments. Based on existing both experimental and computational evidences, we presented an idea that an ACh level in the cortex performs as the learning rate which regulates the adaptability and stability in our on-line leaning, standing on a statistical approach to understanding brain functions.

A Algorithm

In this implementation, we assume a natural conjugate prior for $p(\boldsymbol{\Theta})$, given by

$$p(\boldsymbol{\Theta}) = \prod_{h=1}^{n} \mathcal{N}_{m+1}\left(\boldsymbol{\theta}_h \mid \boldsymbol{e}_h, \gamma^{-1}\boldsymbol{I}_{m+1}\right). \quad (8)$$

Here, $\boldsymbol{\theta}_h = (w_{h1}, w_{h2}, \ldots, w_{hm}, \mu_h)' \in \Re^{(m+1)}$ is a transposed row vector of $\boldsymbol{\Theta}$, where w_{ij} is the (i,j)-element of matrix $\boldsymbol{W}$ and μ_j is the j-th element of vector $\boldsymbol{\mu}$. $\boldsymbol{e}_h \equiv (\delta_{h,1}, \cdots, \delta_{h,m}, 0)' \in \Re^{(m+1)}$, $\delta_{i,j}$ is the Kronecker's delta, and γ ($\gamma > 0$) is a constant inverse variance. The mean of each principal component vector $\boldsymbol{w}_j$ ($j = 1, \cdots, m$) over the prior distribution (8) becomes orthogonal with the others, and its norm equals 1. Since the principal component vectors are estimated as orthonormal bases when there are no observed data, this prior distribution is suitable for PCA.

The algorithm of our on-line VB learning for the modified MPPCA is summarized as follows.

1. **Initialization phase**:

   Initialize the trial posterior of parameter, $q_\theta(\boldsymbol{\Theta})$, such as to be equal to the prior distribution:

   $$q_\theta(\boldsymbol{\Theta}) = \prod_{h=1}^{n} \mathcal{N}_{m+1}\left(\boldsymbol{\theta}_h \mid \hat{\boldsymbol{\theta}}_h, \hat{\gamma}^{-1}\hat{\boldsymbol{V}}\right), \quad (9)$$

   where $\hat{\boldsymbol{\theta}}_h = \boldsymbol{e}_h, \hat{\gamma} = \gamma$ and $\hat{\boldsymbol{V}} = \boldsymbol{I}_{m+1}$. Initialize $\lambda(0)$ simply at $\lambda(0) = 1$.

2. **Inference phase**:

   After observing a datum $\boldsymbol{x}_t$ at time step t, the following procedure is executed.

   On-line VB-Estep: $F^\lambda(t)$ is maximized with respect to $q_t(\boldsymbol{y}_t, z_t \mid \boldsymbol{x}_t)$. The solution does not depend on $\lambda(s)$ ($s = 1, \cdots, t$) and is given by

   $$\begin{aligned} &q_t(\boldsymbol{y}_t, z_t = i|\boldsymbol{x}_t) \\ &\quad = C \exp\Big[E\left[\log p(\boldsymbol{x}_t, \boldsymbol{y}_t, z_t = i|\boldsymbol{\Theta})\right]\Big], \end{aligned} \quad (10)$$

   for $i = 0, 1$. C is a normalization term. Based on this posterior distribution, the forgetting factor $\lambda(t)$ is given by equation (7). The effective data number $T^\lambda(t)$ and the learning rate $\eta(t)$ are updated by using the forgetting factor $\lambda(t)$:

   $$T^\lambda(t) = 1 + \lambda(t)T^\lambda(t), \quad \eta(t) = 1/T^\lambda(t). \quad (11)$$

   On-line VB-Mstep: $F^\lambda(t)$ is maximized with respect to $q_\theta(\boldsymbol{\Theta}|X_{1:t})$. The solution is given by

   $$q_\theta(\boldsymbol{\Theta}|X_{1:t}) = \prod_{h=1}^{n} \mathcal{N}_{m+1}\left(\boldsymbol{\theta}_h \mid \hat{\boldsymbol{\theta}}_h, \hat{\gamma}^{-1}\hat{\boldsymbol{V}}\right), \quad (12)$$

   where

   $$\begin{aligned} \hat{\gamma} =& \sigma_x^{-2} T^\lambda(t)\langle(1-z)\rangle(t) \\ &+ \sigma_\epsilon^{-2} T^\lambda(t)\langle z\rangle(t) + \gamma \quad (13) \\ \hat{\boldsymbol{V}} =& \frac{1}{\hat{\gamma}}\left(\sigma_x^{-2} T^\lambda(t)\langle(1-z)\tilde{\boldsymbol{y}}\tilde{\boldsymbol{y}}'\rangle(t)\right. \\ &\left. + \sigma_\epsilon^{-2} T^\lambda(t)\langle z\tilde{\boldsymbol{y}}\tilde{\boldsymbol{y}}'\rangle(t) + \gamma\boldsymbol{I}_{m+1}\right) \quad (14) \\ \begin{pmatrix} \hat{\boldsymbol{\theta}}_1' \\ \vdots \\ \hat{\boldsymbol{\theta}}_n' \end{pmatrix} =& \frac{1}{\hat{\gamma}}\left(\sigma_x^{-2} T^\lambda(t)\langle(1-z)\boldsymbol{x}\tilde{\boldsymbol{y}}'\rangle(t)\right. \\ &\left. + \sigma_\epsilon^{-2} T^\lambda(t)\langle z\boldsymbol{x}\tilde{\boldsymbol{y}}'\rangle(t)\right)\hat{\boldsymbol{V}}^{-1}. \quad (15) \end{aligned}$$

3. **Obtaining the mean model parameter**:

   The expectation of parameter $\boldsymbol{\Theta}$ over the trial posterior distribution, $\boldsymbol{\Theta}^*$, has been obtained by equation (15), as $\boldsymbol{\Theta}^* = \left(\hat{\boldsymbol{\theta}}_1, \cdots, \hat{\boldsymbol{\theta}}_n\right)'$.

References

[1] E. Acquas, C. Wilson, and H. C. Fibiger. Conditioned and unconditioned stimuli increase frontal cortical and hippocampal acetylcholine release: effects of novelty, habituation, and fear. *The journal of Neuroscience*, 16(9):3089–3096, 1996.

[2] J. S. Bakin and N. M. Weinberger. Induction of a physiological memory in the cerebral cortex by stimulation of the nucleus basalis. *Proc. Natl. Acad. Sci. USA*, 93:11219–11224, 1996.

[3] D.T. Blake, N.N. Byl, and M.M. Merzenich. Representation of the hand in the cerebral cortex. *Behavioural Brain Research*, 135:179–184, 2002.

[4] P. Dayan, G. E. Hinton, R. Neal, and R. S. Zemel. The Helmholtz Machine. *Neural Computation*, 7:1022–1037, 1995.

[5] K. Doya. Metalearning and neuromodulation. *Neural Network*, 15(4–6):495–506, 2002.

[6] J.-M. Edeline. Learning-induced physiological plasticity in the thalamo-cortical sensory systems: a critical evaluation of receptive field plasticity, map changes and their potential mechanisms. *Progress in Neurobiology*, 57:165–224, 1999.

[7] J.-M. Fellous and C. Linster. Computational models of neuromodulation. *Neural Computation*, 10:771–805, 1998.

[8] A.S. Georghiades, P.N. Belhumeur, and D.J. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 23(6):643–660, 2001.

[9] M. G. Giovannini, A. Rakovska, R. S. Benton, M. Pazzagli, L. Bianchi, and G. Pepeu. Effects of novelty and habituation on acetylcholine, GABA, and glutamate release from the frontal cortex and hippocampus of freely moving rats. *Neuroscience*, 106(1):43–53, 2001.

[10] M. E. Hasselmo. Neuromodulation and cortical function: modeling the physiological basis of behavior. *Behavioural Brain Research*, 67:1–27, 1995.

[11] G. E. Hinton and Z. Ghahramani. Generative models for discovering sparse distributed representations. *Philosophical Transactions Royal Society B*, 352:1177–1190, 1997.

[12] M. P. Kilgard and M. M. Merzenich. Cortical map reorganization enabled by nucleus basalis activity. *Science*, 279:1714–1718, 1998.

[13] M. I. Miranda, L. Ramírez-Lugo, and F. Bermúdez-Rattoni. Cortical cholinergic activity is related to the novelty of the stimulus. *Brain Research*, 882:230–235, 2000.

[14] B. Olshausen and D. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.

[15] S. Penschuck, C. H. Chen-Bee, N. Parakash, and R. D. Frostig. In vivo modulation of a cortical functional sensory representation shortly after topical cholinergic agent application. *J Comp Neurol*, 452(1):38–50, 2002.

[16] R. P. N. Rao and D. H. Ballard. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2(1):79–87, 1999.

[17] M. Sato. Online model selection based on the varational Bayes. *Neural Computaion*, 13:1649–1681, 2001.

[18] M. Tipping and C. Bishop. Probabilistic principal component analysis. Technical report, Neural Computing Research Group, Aston University, 1997.

[19] M. E. Tipping and C. M. Bishop. Mixtures of probabilistic principal component analyzers. *Neural Computation*, 11:443–482, 1999.

[20] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[21] A. Yu and P. Dayan. Acetylcholine in cortical inference. *Neural Networks*, 15(4–6):719–730, 2002.

[22] A. Yu and P. Dayan. Expected and unexpected uncertainty: ACh and NE in the neocortex. In *Advances in Neural Information Processing Systems 15*, Cambridge, MA, 2002. MIT Press.