OPEN ACCESS Freely available online

PLoS COMPUTATIONAL BIOLOGY

Maximization of Learning Speed in the Motor Cortex Due to Neuronal Redundancy

Ken Takiyama[1], **Masato Okada**[1,2]*

1 Graduate School of Frontier Sciences, The University of Tokyo, Complex Science and Engineering, Chiba, Japan, **2** RIKEN Brain Science Institute, Wako, Japan

Abstract

Many redundancies play functional roles in motor control and motor learning. For example, kinematic and muscle redundancies contribute to stabilizing posture and impedance control, respectively. Another redundancy is the number of neurons themselves; there are overwhelmingly more neurons than muscles, and many combinations of neural activation can generate identical muscle activity. The functional roles of this neuronal redundancy remains unknown. Analysis of a redundant neural network model makes it possible to investigate these functional roles while varying the number of model neurons and holding constant the number of output units. Our analysis reveals that learning speed reaches its maximum value if and only if the model includes sufficient neuronal redundancy. This analytical result does not depend on whether the distribution of the preferred direction is uniform or a skewed bimodal, both of which have been reported in neurophysiological studies. Neuronal redundancy maximizes learning speed, even if the neural network model includes recurrent connections, a nonlinear activation function, or nonlinear muscle units. Furthermore, our results do not rely on the shape of the generalization function. The results of this study suggest that one of the functional roles of neuronal redundancy is to maximize learning speed.

Citation: Takiyama K, Okada M (2012) Maximization of Learning Speed in the Motor Cortex Due to Neuronal Redundancy. PLoS Comput Biol 8(1): e1002348. doi:10.1371/journal.pcbi.1002348

Editor: Jörn Diedrichsen, University College London, United Kingdom

Received March 16, 2011; **Accepted** November 26, 2011; **Published** January 12, 2012

Funding: This work was partially supported by a Grant-in-Aid for Scientific Research (A) (Grant No. 20240020) and a Grant-in-Aid for Special Purposes (Grant No. 10J04910) from the Ministry of Education, Culture, Sports, Science and Technology of Japan. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: okada@k.u-tokyo.ac.jp

Introduction

In the human brain, numerous neurons encode information about external stimuli, e.g., visual or auditory stimuli, and internal stimuli, e.g., attention or motor planning. Each neuron exhibits different responses to stimuli, but neural encoding, especially in the visual and auditory cortices, can be explained by the maximization of stimulus information [1–3]. This maximization framework can also explain learning that occurs when the same stimuli are repeatedly presented; previous neurophysiological experiments have suggested that perceptual learning causes changes in neural encoding to enhance the Fisher information of a visual stimulus [4]. However, a recent study has suggested that information maximization alone is insufficient to explain neural encoding. Salinas has suggested that "how encoded information is used" needs to be taken into account: neural encoding is influenced by the downstream circuits and output units to which neurons project, and it is ultimately influenced by animal behavior [5]. In the motor cortex, neural encoding is influenced by the characteristics of muscles (output units) because motor cortex neurons send motor commands to muscles through the spinal cord. In adaptation experiments, some motor cortex neurons exhibit rotations in their preferred directions (PDs), and these rotations result in a population vector that is directed toward a planned target [6]. Neural encoding therefore changes to minimize errors between planning and behavior, suggesting that neural encoding is influenced by behavior and properties of output units.

A critical problem exists in the relationship between motor cortex neurons and output units: the neuronal redundancy problem, or overcompleteness, which refers to the fact that the number of motor cortex neurons far exceeds the number of output units. Many different combinations of neural activities can therefore generate identical outputs. Neurophysiological and computational studies have revealed that the motor cortex exhibits neuronal redundancy [7,8]. However, it remains unknown how neuronal redundancy influences neural encoding. In other words, we do not yet understand the functional roles of neuronal redundancy in motor control and learning, though other types of redundancies are known to play various functional roles [9].

One of these types of redundancy is muscle redundancy: many combinations of muscle activities can generate identical movements. The functional roles of this muscle redundancy include impedance control to achieve accurate movements [10], reduction of motor variance by constructing muscle synergies [11], and learning internal models by changing muscle activities [12]. Another redundancy is kinematic redundancy: many combinations of joint angles result in identical hand positions. This redundancy ensures the stability of posture even if one joint is perturbed [13], and it facilitates of motor learning by increasing motor variance in a dimension irrelevant to the desired movements [14]. Redundancies therefore play important functional roles in motor control and learning.

Similar to the muscle and kinematic redundancies, neuronal redundancy likely has functional roles in motor control and learning. However, the functional roles of this redundancy are

Author Summary

There are overwhelmingly more neurons than muscles in the motor system. The functional roles of this neuronal redundancy remains unknown. Our analysis, which uses a redundant neural network model, reveals that learning speed reaches its maximum value if and only if the model includes sufficient neuronal redundancy. This result does not depend on whether the distribution of the preferred direction is uniform or a skewed bimodal, both of which have been reported in neurophysiological studies. We have confirmed that our results are consistent, regardless of whether the model includes recurrent connections, a nonlinear activation function, or nonlinear muscle units. Additionally, our results are the same when using either a broad or a narrow generalization function. These results suggest that one of the functional roles of neuronal redundancy is to maximize learning speed.

unclear. Here, using a redundant neural network, we investigate these functional roles by varying the number of model neurons while holding the number of output units constant. This manipulation allows us to control the degree of neuronal redundancy because, if a neural network includes a large number of neurons and a small number of output units, many different combinations of neural activities can generate identical outputs. It should be noted that we used a redundant neural network model that can explain neurophysiological motor cortex data [7]. The key conclusion arising from our study is that one of the functional roles of neuronal redundancy is the maximization of learning speed.

Initially, a linear model with a fixed decoder was used. Analytical calculations revealed that neuronal redundancy is a necessary and sufficient condition to maximize learning speed. This maximization is invariant whether the distribution of PDs is unimodal [6] or bimodal [15–17]; both distributions have been reported in neurophysiological investigations. Second, numerical simulations confirmed the invariance of our results, even when the neural network included an adaptable decoder, a nonlinear activation function, recurrent connections, or nonlinear muscle units. Third, we show that our results do not depend on learning rules by using weight and node perturbation, both of which are representative stochastic gradient methods [18]. Finally, we demonstrate that our hypothesis does not depend on the shape of the generalization function which shape depends on the task (broad or sharp in force field [19,20] or visuomotor rotation adaptation [21], respectively). Our results strongly support our hypothesis that neuronal redundancy maximizes learning speed.

Results

Neuronal redundancy is defined as the dimensional gap between the number of neurons N and the number of outputs M. It is synonymous with overcompleteness [22]: many combinations of neural activities $\mathbf{A}\in\mathbb{R}^{N\times 1}$ can generate identical outputs $\mathbf{x}\in\mathbb{R}^{M\times 1}$ through a decoder $\mathbf{Z}\in\mathbb{R}^{M\times N}$ ($\mathbf{x}=\mathbf{ZA}$) because there are more neurons than necessary, i.e., $N\gg M$ (Figure 1). It should be noted that neuronal redundancy is defined not by N but by the relationship between N and M. In most parts of this study, the number of constrained tasks T is the same as M and is set to two, i.e., $M=T=2$, so there is neuronal redundancy if $N>2$. Thus, throughout this paper, the extent of neuronal redundancy can be expressed simply using the number of neurons. In this study, we can change only the neuronal redundancy; N can be increased while T is held constant at two, enabling the investigation of the functional roles of neuronal redundancy. In the *Importance of Neuronal Redundancy* section, we distinguish the effects of neuronal redundancy from the effects of neuron number by varying both N and T.

In this study, we discuss the relationship between neuronal redundancy and learning speed by assuming adaptation to either a visuomotor rotation or a force field. These tasks are simulated by using a rotational perturbation $\mathbf{R}=\begin{pmatrix}\cos\phi & -\sin\phi\\ \sin\phi & \cos\phi\end{pmatrix}$ where ϕ is the rotational angle. Due to this perturbation, if an error occurs between target position $\mathbf{t}_{k(t)}=(\cos\theta_{k(t)},\sin\theta_{k(t)})^T$ and output (motor command) $\mathbf{x}$ in the tth trial, neural activities $\mathbf{A}(\theta_{k(t)})$ are modified to minimize the error, where $\theta_{k(t)}$ is the angle of the $k(t)$th target which is radially and equally distributed ($t=1,\ldots,Trial$, $k(t)\in 1,\ldots,K$, $\theta_{k(t)}=2\pi\frac{k(t)}{K}$). To model the learning process in the motor cortex, we used a linear rate model, which

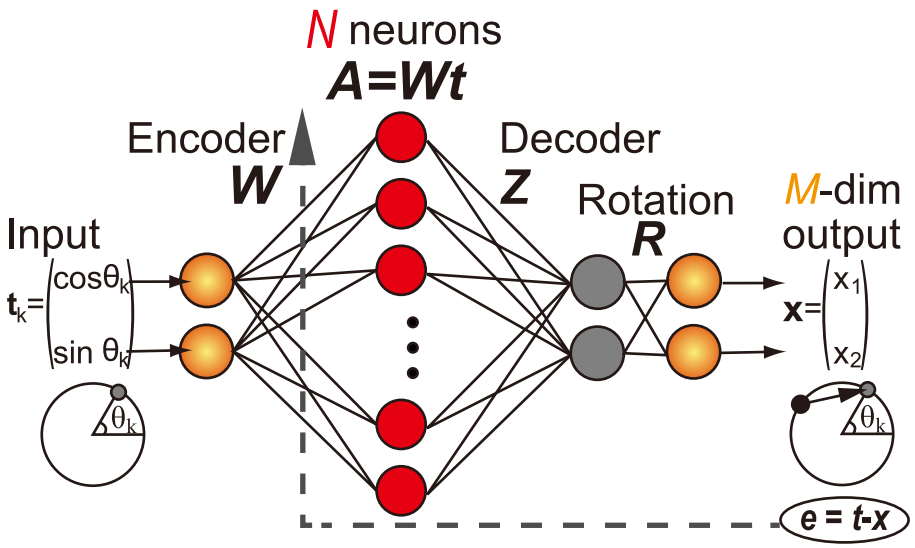

Figure 1. Graphical model of a redundant neural network.
doi:10.1371/journal.pcbi.1002348.g001

can reproduce neurophysiological data [7] and be easily analyzed. In this model, $\mathbf{x}$ is given by a weighted average of $\mathbf{A}$, and each component of $\mathbf{Z}$ is accordingly set to $\mathcal{O}(\frac{1}{N})$, i.e., (i,j)th component of $\mathbf{Z}$ is defined as $Z_{ij}=\frac{1}{N}z_{ij}$, where z_{ij} is a variable that is independent of N. Because of this assumption, the learning rate is set to NB such that the trial-to-trial variation of $\mathbf{x}$ do not depend on N ($\mathcal{O}(1)$), but the optimized learning rate η^* is $\mathcal{O}(N)$ (see Text S1), i.e., $\eta^*=NB^*$, suggesting that we consider the quasi-optimal learning rate in this study. It should be noted that, because the following results do not depend on B, our results hold when the optimal learning rate is used. Furthermore, even when each component of $\mathbf{Z}$ is $\mathcal{O}(1)$, the following results are invariant if we set the learning rate to its optimal value (see Text S1). Our study shows that neuronal redundancy is necessary and sufficient to maximize learning speed.

Neuronal redundancy maximizes learning speed

Fixed homogeneous decoder. In the case of a fixed decoder, $\mathbf{Z}=\frac{1}{N}\begin{pmatrix}\cos\varphi_1 & \dots & \cos\varphi_N\\ \sin\varphi_1 & \dots & \sin\varphi_N\end{pmatrix}$, the ith neuron has uniform force amplitude (FA) ($\frac{1}{N^2}(\cos^2\varphi_i+\sin^2\varphi_i)=\frac{1}{N^2}$) and force direction (FD), φ_i, which is randomly sampled from a uniform distribution. Because of its uniformity, we refer to this decoder as a fixed homogeneous decoder. This model corresponds to the one proposed by Rokni et al. [7].

In this case, the squared error can be calculated recursively as

$$E^{t+1}=\frac{1}{2}(\mathbf{e}^{t+1})^T\mathbf{e}^{t+1}=\frac{1}{2}(\mathbf{e}^t)^T(\mathbf{I}-B\mathbf{\Lambda})^T(\mathbf{I}-B\mathbf{\Lambda})\mathbf{e}^t, \quad (1)$$

where $\mathbf{e}=\mathbf{t}-\mathbf{x}=\mathbf{t}-\mathbf{RZA}$. Here, we assume that a single target is repeatedly presented for simplicity (general case is discussed in the *Methods* section), $\mathbf{I}$ is the identity matrix, $\mathbf{\Lambda}=N\mathbf{RZZ}^T\mathbf{R}^T$, NB is the learning rate, and neural activity $\mathbf{A}$ is updated as

$$\mathbf{A}^{t+1}=\mathbf{A}^t+BN\mathbf{Z}^T\mathbf{R}^T\mathbf{e}^t \quad (2)$$

for the tth trial to minimize the squared error. Multiplication by N in equation (2) is included for the purpose of scaling; it ensures that the amount of trial-to-trial variation in $\mathbf{A}$ does not explicitly depend on N. Equation (1) can thus be simplified as

$$E^{t+1}=\frac{1}{2}(\mathbf{v}^{t+1})^T\mathbf{v}^{t+1}=\frac{1}{2}(\mathbf{v}^t)^T(\mathbf{I}-\lambda)^T(\mathbf{I}-\lambda)\mathbf{v}^t, \quad (3)$$

where the diagonal elements of λ, λ_1 and λ_2, are eigenvalues of $\mathbf{\Lambda}$, $\mathbf{\Lambda}$ is decomposed as $\mathbf{V}^T\lambda\mathbf{V}$ ($\mathbf{V}^T\mathbf{V}=\mathbf{I}$), $\mathbf{v}^t=\mathbf{V}\mathbf{e}^t$, and learning speed is therefore determined based on the eigenvalues of

$$\mathbf{\Lambda}=\frac{1}{N}\begin{pmatrix}\sum_{i=1}^N\cos^2(\varphi_i-\phi) & \sum_{i=1}^N\cos(\varphi_i-\phi)\sin(\varphi_i-\phi)\\ \sum_{i=1}^N\cos(\varphi_i-\phi)\sin(\varphi_i-\phi) & \sum_{i=1}^N\sin^2(\varphi_i-\phi)\end{pmatrix} \quad (4)$$

each component of which is $\mathcal{O}(1)$. The larger λ_i becomes, the faster learning becomes ($i=1,2$). It should be noted that learning speed and λ_i do not explicitly depend on N.

Analytical calculations can yield necessary and sufficient conditions to maximize learning speed (see the *Methods* section). The following self-averaging properties [23] maximize learning speed or maximize the minimum eigenvalue of $\mathbf{\Lambda}$:

$$\frac{1}{N}\sum_i^N\cos^2\varphi_i=\int_0^{2\pi}d\varphi P(\varphi)\cos^2\varphi=\frac{1}{2}, \quad (5)$$

$$\frac{1}{N}\sum_i^N\sin^2\varphi_i=\int_0^{2\pi}d\varphi P(\varphi)\sin^2\varphi=\frac{1}{2}, \quad (6)$$

and

$$\frac{1}{N}\sum_i^N\cos\varphi_i\sin\varphi_i=\int_0^{2\pi}d\varphi P(\varphi)\cos\varphi\sin\varphi=0, \quad (7)$$

where $P(\varphi)$ is the probability distribution in which FDs are randomly sampled. It remains unknown what kind of conditions can satisfy the self-averaging properties. The self-averaging properties are satisfied if and only if the neural network model includes sufficient neuronal redundancy. In other words, learning speed is maximized if and only if $N\to\infty$. If the neural network includes neuronal redundancy, the self-averaging properties exist. Conversely, if the self-averaging properties exist, the neural network model should include sufficient neuronal redundancy because Monte Carlo integration shows a fluctuation of $\mathcal{O}(1/\sqrt{N})$ [24]. Thus, in the case of a fixed homogeneous decoder, neuronal redundancy plays a functional role in maximizing learning speed.

We numerically confirmed the above analytical results. Figures 2A and 2B show the learning speed and learning curves calculated using the results of 1,000 sets of randomly sampled φ values, an identical target sequence ($K=8$), and $\phi=\pi/3$. The more neuronal redundancy grows, the faster learning speed becomes. Figure 2C shows the relationship between learning speed and neuronal redundancy. The horizontal axis denotes the number of neurons, and the vertical axis denotes the increase in learning speed. Although a saturation of the increase can be seen, greater neuronal redundancy still yields faster learning speed. Therefore, these figures support our analytical results: in the case of a fixed homogeneous decoder, neuronal redundancy maximizes learning speed.

Fixed non-homogeneous decoder. The question remains whether it is necessary for FD and FA to be distributed uniformly, so we assume that the values (Z_{1i},Z_{2i}) are randomly sampled from the probability distribution $P(Z_1,Z_2)$ to make FD and FA non-homogeneous, i.e., FDs are non-uniformly distributed, and FAs are different for each neuron. In the case of a non-homogeneous decoder, the necessary and sufficient conditions to maximize learning speed are also the following self-averaging properties:

$$\begin{aligned}&\frac{1}{N}Z_{1i}^2=\frac{1}{N}Z_{2i}^2\Leftrightarrow\int_{-\infty}^{\infty}dZ_1P(Z_1)Z_1^2=\int_{-\infty}^{\infty}dZ_2P(Z_2)Z_2^2\\&\Leftrightarrow\mathrm{Var}(Z_1)+(\mathrm{Mean}(Z_1))^2=\mathrm{Var}(Z_2)+(\mathrm{Mean}(Z_2))^2\end{aligned} \quad (8)$$

and

$$\begin{aligned}&\frac{1}{N}Z_{1i}Z_{2i}=\int_{-\infty}^{\infty}dZ_1dZ_2P(Z_1,Z_2)Z_1Z_2=\\&0\Leftrightarrow\mathrm{Cov}(Z_1,Z_2)-\mathrm{Mean}(Z_1)\mathrm{Mean}(Z_2)=0,\end{aligned} \quad (9)$$

where $P(Z_1)$ and $P(Z_2)$ are marginalized distributions. Figures 3A and 3D show distributions of $\mathbf{Z}$ that satisfy equations (8) and (9). $\mathbf{Z}$ is randomly sampled from unimodal and bimodal Gaussian distributions in Figures 3A and 3D, respectively. Because these

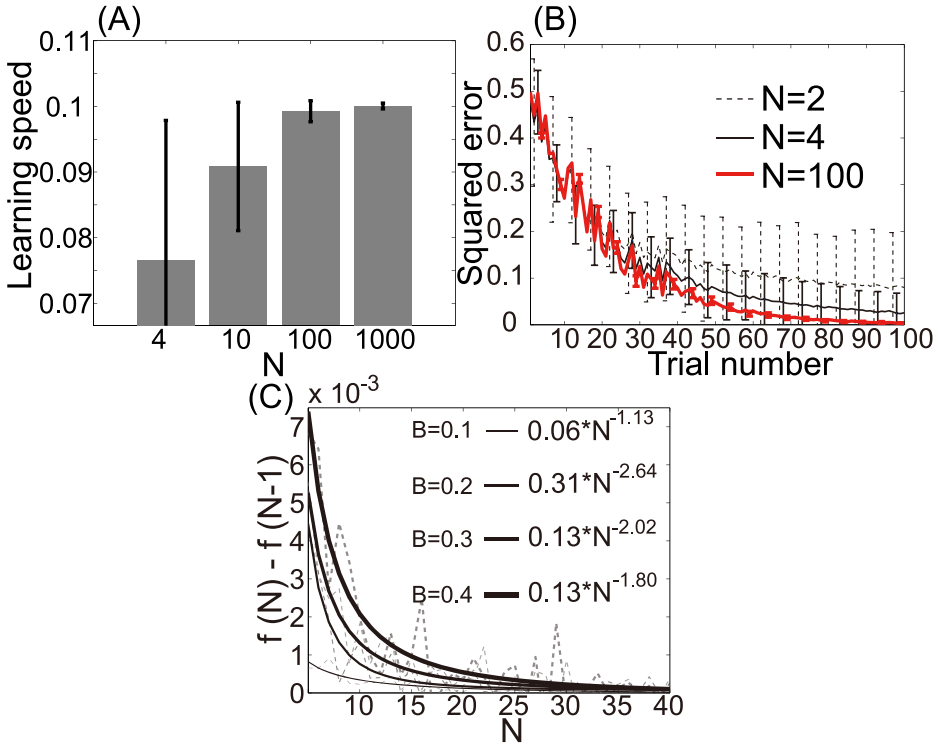

Figure 2. Relationship between learning speed and neuronal redundancy ($K=8$). (A): Learning speed when $N=4,10,100$, or 1000. The bar graph and error bars depict sample means and standard deviations, both of which are calculated using the results of randomly sampled sets of 1000 φ values. (B): Learning curves when $N=2,4$, or 100. These curves and error bars show averaged values and standard deviations of errors. (C): Relationship between learning speed and the number of model neurons when $B=0.1,0.2,0.3$, or 0.4. The horizontal axis represents the number of neurons N and the vertical axis represents $f(N)-f(N-1)$, where $f(N)$ is the learning speed when the number of neurons is N. Dotted and solid lines denote the average learning speed and power functions fitted to the values, respectively.
doi:10.1371/journal.pcbi.1002348.g002

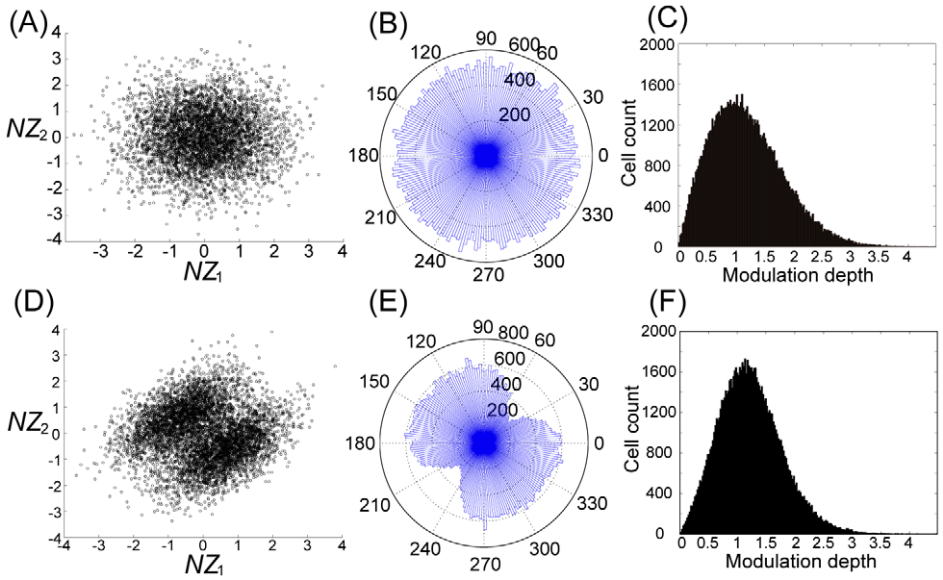

Figure 3. Network properties when $P(Z_1,Z_2)$ satisfies equations (8) and (9). (A): Scatter plot of $N\mathbf{Z}$ when Z_{1i} and Z_{2i} are randomly sampled from a unimodal Gaussian distribution ($i=1,\ldots,N$). (B), (C): Histogram of preferred direction and modulation depth when $\mathbf{Z}$ is randomly sampled as shown in (A). (D): Scatter plot of $N\mathbf{Z}$ when (Z_{1i},Z_{2i}) are randomly sampled from a bimodal Gaussian distribution. (E), (F): Histograms of preferred direction and modulation depth when $\mathbf{Z}$ is randomly sampled as shown in (D).
doi:10.1371/journal.pcbi.1002348.g003

figures show the non-uniformity in both FD and FA, neuronal redundancy maximizes learning speed regardless of these non-uniformities.

Distribution of preferred directions. Some neurophysiological studies have suggested that the distribution of PD is a skewed bimodal [15–17], but other neurophysiological studies have suggested that the distribution of PD is uniform [6]. We investigated whether our results were consistent with the results of these neurophysiological studies. Figures 3B and 3E depict the distribution of preferred directions (PDs) that results when $\mathbf{Z}$ is randomly sampled as shown in Figures 3A and 3D, respectively, with PDs calculated as $\mathrm{PD}_i = \arg\max_\theta A_i(\theta)$ (see the *Methods* section). Figures 3B and 3E show that both a skewed bimodal distribution and a uniform distribution can be observed when $P(Z_1,Z_2)$ satisfies equations (8) and (9), suggesting that our hypothesis is consistent with the results of previous neurophysiological experiments.

Figures 3C and 3F show the distribution of modulation depth, which is calculated as $m_i = \max_\theta |A_i(\theta)|$ (see the *Methods* section). Our results suggest that the distribution of modulation depth is skewed.

Adaptable decoder. We have analytically elucidated the relevance of neuronal redundancy to learning speed only when $\mathbf{Z}$ is fixed, but the question remains of whether neuronal redundancy can maximize learning speed even when $\mathbf{Z}$ is adaptable. In this case, it is analytically intractable to calculate learning speed, so we used numerical simulations. Figure 4A shows the learning speed when $N = 2,4,10,100$, or 1000 in the case of an adaptable decoder. Although there was no significant difference in learning speed between the cases in which $N = 100$ and $N = 1000$, neuronal redundancy maximized learning speed even if the decoder was adaptable. Figure 4B, which shows the learning curve when $N = 2,4$, or 100, also supports the maximization.

Importance of neuronal redundancy

Although we have revealed that neuronal redundancy maximizes learning speed when $T = 2$, it is important to verify that the effect is caused by the neuronal redundancy, i.e., the dimensional gap between N and T, and not simply the number of neurons N. In this section, we investigate this question by varying both N and T while assuming that each component of $\mathbf{t}$ is randomly sampled from a Gaussian distribution.

Figures 5A and 5B show the learning speed and the learning curve produced when $N = T = 10,50$, and 100 with a fixed non-homogeneous decoder. If N alone were important for maximizing learning speed, learning speed would be faster when $N = T = 100$ than when $N = T = 10$ or $N = T = 50$. However, the results shown in these figures support the opposite conclusion, i.e., learning speed becomes slower when $N = T = 100$ compared to the other cases. This result suggests that the number of neurons alone is not important for maximizing learning speed.

Figures 5C and 5D show the learning speed and learning curve produced when $T = 10,50$, or 100 with $N = 50$ and a fixed non-homogeneous decoder. If neuronal redundancy were important, the learning speed would be faster when $T = 10$ than when $T = 50$ or $T = 100$. These figures support this hypothesis; learning speed increased when $T = 10$ compared to the other cases. Taken together, these results indicate that the important factor for maximizing learning speed is in fact neuronal redundancy and not simply the number of neurons.

In addition, we investigated whether neuronal redundancy or neuron number is important when $\mathbf{Z}$ is adaptable. In this case, we only show learning curves because learning speed cannot be exponentially fitted, which makes it impossible to calculate learning speed. Figures 5E and 5F show the learning curves calculated when $N = T = 10,50$, or 100 and $T = 10,50$, or 100 with $N = 50$. These figures show the same results as the case when $\mathbf{Z}$ is fixed; even when $\mathbf{Z}$ is adaptable, the important factor for maximizing learning speed is neuronal redundancy, not simply the number of neurons.

Generality of our results

The generality of our results should be investigated because we analyzed only linear and feed-forward networks, but neurophysiological experiments have suggested the existence of recurrent connections [25] and nonlinear neural activation functions [26]. Also, only a linear rotational perturbation task was considered, so we need to investigate whether our results hold when the constrained tasks are nonlinear because, in fact, motor cortex neurons solve nonlinear tasks. The neurons send motor commands and control muscles whose activities are nonlinearly determined: muscles can pull but cannot push. Using numerical simulations, we show that neuronal redundancy maximizes learning speed, even when the neural network includes recurrent connections (Figure S1), when it includes nonlinear activation functions (Figure S2), and when the task is nonlinear (Figure S3).

In addition, we used only deterministic gradient descent, so the generality regarding the learning rule needs to be investigated. In

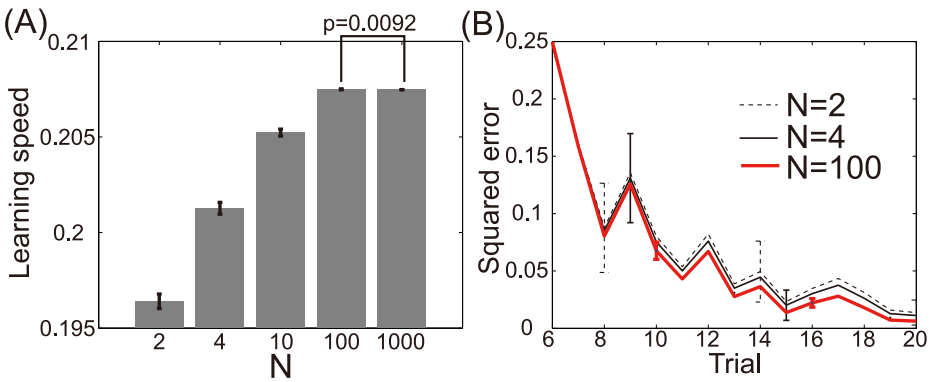

Figure 4. Relationship between learning speed and neuronal redundancy when the decoder is adaptable ($K = 8$). (A): Bar graphs and error bars depict sample means and standard deviations both of which are calculated using the results from 1000 sets of $\mathbf{Z}^0$ values. (B): Learning curves when $N = 2,4$, or 100. These curves and error bars show averaged values and the standard deviations of the errors.
doi:10.1371/journal.pcbi.1002348.g004

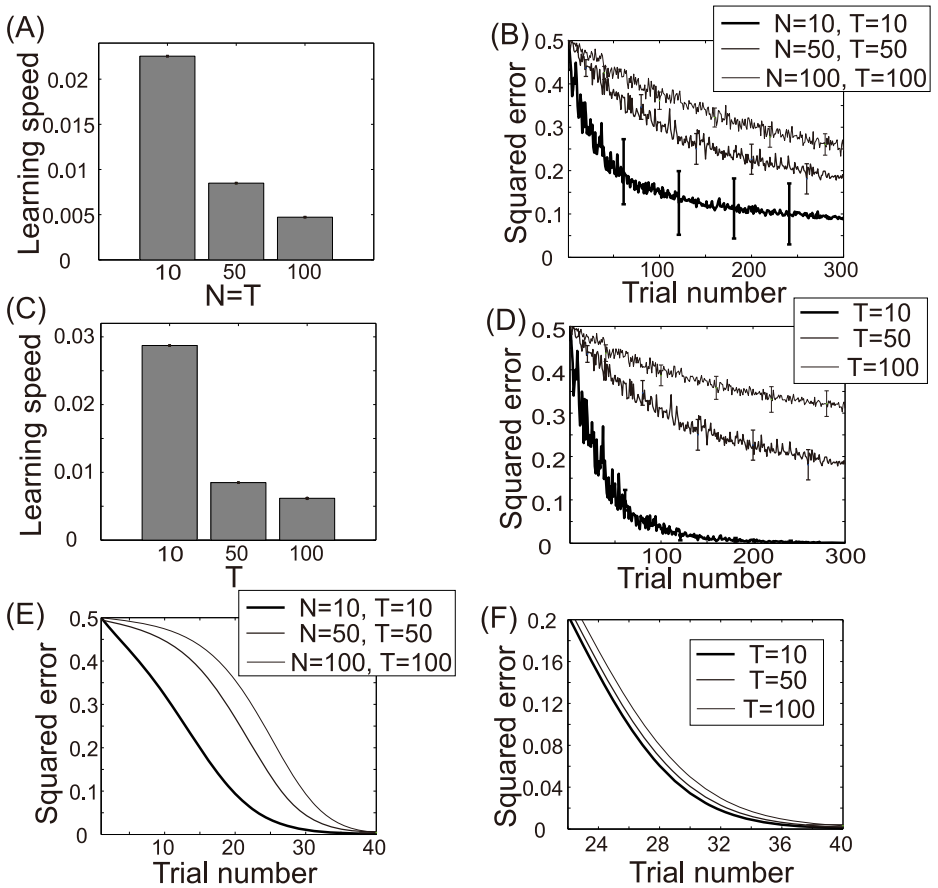

Figure 5. Importance of neuronal redundancy ($K=1$). (A): Learning speed when $N=T=10$, $N=T=50$, or $N=T=100$, where N and T are the number of neurons and constrained tasks, respectively. The bar graphs and error bars depict the sample means and standard deviations, both of which are calculated using the results of 1000 sets of $\mathbf{Z}^0$ values. (B): Learning curves when $N=T=10$, $N=T=50$, or $N=T=100$. These curves and error bars show the average values and the standard deviations of the errors. (C): Learning speed when $T=10,50$, or 100, and $N=50$. The bar graphs and error bars depict the sample means and the standard deviations, both of which are calculated using the results of 1000 sets of $\mathbf{Z}^0$ values. (D): Learning curves when $T=10,50,100$, and $N=50$. These curves and error bars show the average values and the standard deviations of the errors. (E): Learning curves calculated when $N=T=10$, $N=T=50$, or $N=T=100$ and decoder $\mathbf{Z}$ is adaptable. (F): Learning curves calculated when $T=10,50$, or 100; $N=50$; and the decoder $\mathbf{Z}$ is adaptable.
doi:10.1371/journal.pcbi.1002348.g005

fact, previous studies have suggested that stochastic gradient methods are more biologically relevant than deterministic ones [27,28]. Analytical and numerical calculations confirm that our results are invariant even when the learning rule is stochastic (Figure S4). Our results therefore have strong generality.

Activity noise and plasticity noise. Although our results have strong generality, there is still an open question regarding the robustness of noise: does neuronal redundancy maximize learning speed even in the presence of neural noise? Actually, neural activities show trial-to-trial variation [29], and the neural plasticity mechanism also includes trial-to-trial fluctuations [7]. This section investigates the relationships between neuronal redundancy, learning speed, and neural noise.

Figures 6A and 6D show the variance of learning curves when $\sigma_a=0,0.1,0.2,0.3,0.4,0.5$ and $\sigma_p=0,0.05,0.1,0.15,0.2$, respectively, with $N=4,10,100$, or 1000 and σ_a and σ_p representing the standard deviations of activity noise and plasticity noise, respectively. The definition of the variance is $\frac{1}{\mathrm{Trial}}\sum_{t=1}^{\mathrm{Trial}}\mathrm{Var}(E^t)$, which is a measure of the stability of learning. Examples of learning curves are shown in Figures 6B, 6C, 6E, and 6F. These figures show that neuronal redundancy enhances the stability of learning by eliminating the influences of activity and plasticity noise. Neuronal redundancy therefore not only maximizes learning speed but also facilitates robustness in response to neural noise.

Shape of the generalization function. In many situations, learning in one context is generalized to different contexts, such as different postures [30], different arms [31], and different movement directions [19–21], with the degree of generalization depending on the task. In this study, we define the generalization function as the degree of generalization to different movement directions. The performance of reaching towards $\theta_{k(t)}$ is generalized to that of reaching towards θ, and the degree of this generalization is determined by the generalization function $f(\theta-\theta_{k(t)})$. In visuomotor rotation adaptation, the generalization function is narrow in the direction metric [21]. In contrast, the generalization function is broad in force field adaptation [19,20]. To investigate the generality of our results with respect to various kinds of tasks, it is necessary to investigate the relationships between neuronal redundancy, learning speed, and the shape of the generalization function.

Figure 7 shows the relationship between the shape of the generalization function and learning speed. Figures 7A and 7B show the learning speed and learning curve calculated when the generalization function is broad (Figure 7C). Figures 7D and 7E

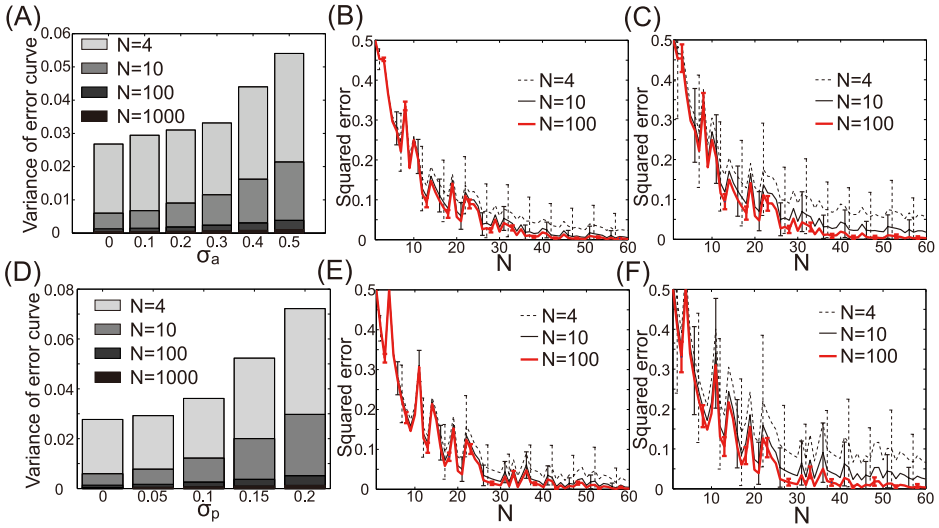

Figure 6. Relationship between neuronal redundancy and neural noise ($K=8$). (A): Variance of the learning curve when $N=4,10,100$, or 1000 and $\sigma_a=0,0.1,0.2,0.3,0.4,0.5$. The bar graphs show the average values of randomly sampled sets of 1000 φ values. (B): Learning curves calculated when $N=4,10$, or 100, and $\sigma_a=0.0$. These curves and error bars show the average values and the standard deviations of the errors. (C): Learning curves calculated when $N=4,10$, or 100, and $s_a=0.5$. (D): Variance of the learning curve when $N=4,10,100$, or 1000 and $\sigma_p=0,0.05,0.1,0.15,0.2$. (E): Learning curves when $N=4,10$, or 100, and $\sigma_p=0.0$. (F): Learning curves calculated when $N=4,10$, or 100, and $\sigma_p=0.2$.
doi:10.1371/journal.pcbi.1002348.g006

show the learning speed and learning curve calculated when the generalization function is narrow (Figure 7F). Although these figures show that narrower generalization results in a slower learning speed, neuronal redundancy maximizes learning speed independently of the shape of the generalization function.

Discussion

We have quantitatively demonstrated that neuronal redundancy maximizes learning speed. The larger the dimensional gap grows between the number of neurons and the number of constrained tasks, the faster learning speed becomes. This maximization does not depend on whether the PD distribution is unimodal or bimodal, the decoder is fixed or adaptable, the network is linear or nonlinear, the task is linear or nonlinear, or the learning rule is stochastic or non-stochastic. Additionally, we have shown that neuronal redundancy has another important functional role: it provides robustness in response to neural noise. Furthermore, neuronal redundancy maximizes learning speed in a manner independent of the shape of the generalization function. These results strongly support the generality of our results.

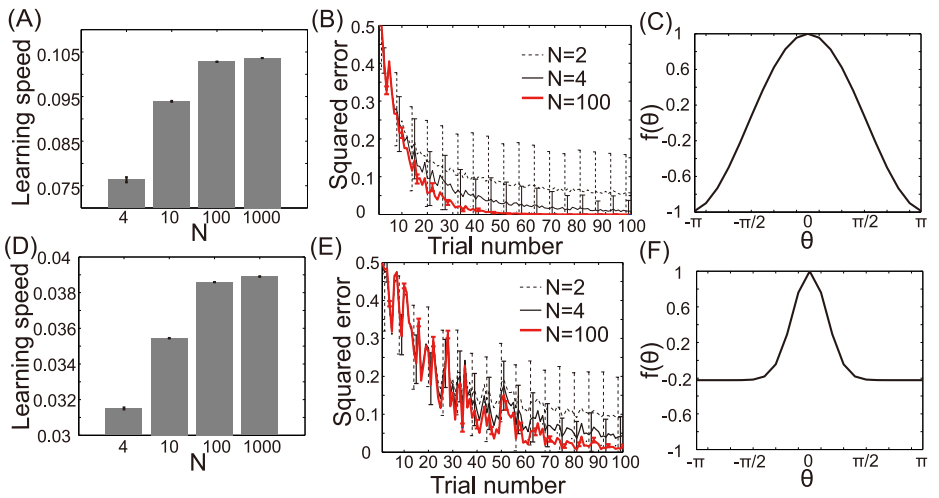

Figure 7. Relationship between neuronal redundancy, learning speed, and the shape of the generalization function ($K=8$). (A): Learning speed when $N=4,10,100$, or 1000, and $\alpha=0.1$. The bar graphs and error bars depict sample means and standard deviations, both of which are calculated using the results of randomly sampled sets of 1000 **j** values in the case of a broad generalization function. (B): Learning curves calculated when $N=2,4,100$, and $\alpha=0.1$. These curves and error bars show the average values and standard deviations of the errors. (C): The generalization function with $\alpha=0.1$. (D): The learning speed when $N=4,10,100$, or 1000, and $\alpha=10$. Bar graphs and error bars depict the sample means and standard deviations when the generalization function is narrow ($\alpha=10$). (E): Learning curves calculated when $N=2,4,100$, and $\alpha=10$. (F): The generalization function with $\alpha=10$.
doi:10.1371/journal.pcbi.1002348.g007

Neuronal redundancy maximizes learning speed because only T equalities, $\mathbf{x}=\mathbf{t}$, need to be satisfied, and N-dimensional neural activity $\mathbf{A}$ is adaptable ($N \gg T$). This dimensional gap yields the large $(N-T)$ dimensional subspace of $\mathbf{A}$ in which the T equalities are satisfied. The more N increases, the greater the fraction of the subspace becomes: $\lim_{N\to\infty}\frac{N-T}{N}\to 1$. Neuronal redundancy, rather than the number of neurons, thus enables $\mathbf{A}$ to rapidly reach a single point in the subspace. This interpretation likely applies even in the cases of an adaptable decoder, recurrent connections, a nonlinear network, a nonlinear task, and a stochastic learning rule. Furthermore, this interpretation is supported by the results shown in Figure 5; the bigger $(N-T)$ grows, the faster learning speed becomes.

At first glance, our results may seem inconsistent with the results of Werfel et al. [18], who concluded that learning speed is inversely proportional to N. In their model, because they considered the single-layer linear model, N is the same as the number of input units, which is defined as $T(=M)$ in the present study. A similar tendency can be observed in Figure 5; the more T increases, the slower learning speed becomes. We calculated the optimal learning rate and speed as shown in Text S1, and confirmed that learning speed is inversely proportional to T. Thus, our results are consistent with Werfel's study and additionally suggest that neuronal redundancy maximizes learning speed.

Neuronal redundancy plays another important role: generating robustness in response to neural noise (Figure 6). Because neuronal redundancy has the same meaning as overcompleteness, its functional role is the same as the robustness of overcompleteness in the face of perturbations in signals [32]. This additional functional role further supports our hypothesis that neuronal redundancy is a special neural basis on which to maximize learning speed. For example, if we increase the learning rate B in a non-redundant network, the learning speed approaches the maximal speed in a redundant network in which the learning rate is fixed to B. As shown in Figure 6, however, a non-redundant network is not robust with respect to neural noise. Furthermore, neuronal redundancy minimizes residual errors when the neural network includes synaptic decay [7] (see the *Methods* section and Figure S5). Thus, neuronal redundancy represents a special neural basis for maximizing learning speed while minimizing residual error and maintaining robustness in response to neural noise.

Methods

Model definition

Our study assumed the following task: participants move their arms towards one of K radially distributed targets. If the $k(t)$th target is presented in the tth trial, the neural network model receives the input $\mathbf{t}_{k(t)}=(\cos\theta_{k(t)},\sin\theta_{k(t)})^T$ ($k(t)\in 1,\ldots,K$, $t=1,\ldots,Trial$), where $\theta_{k(t)}=2\pi\frac{k(t)}{K}$. The input units project to neurons (hidden units), the activities of which are determined by

$$\mathbf{A}^t(\theta_{k(t)})=\mathbf{W}^t\mathbf{t}_{k(t)}+\sigma_a\xi^t, \tag{10}$$

where $\mathbf{W}^t\in\mathbb{R}^{N\times 2}$ is synaptic weight in the tth trial, σ_a is the standard deviation of neural activity noise, $\xi^t\in\mathbb{R}^{N\times 1}$ denotes independent normal Gaussian random variables, and N is the number of neurons (Figure 1). The ith neuron has a PD given by $\mathrm{PD}_i=\arctan\frac{W_{i2}}{W_{i1}}$ and a modulation depth $m_i=\sqrt{(W_{i1})^2+(W_{i2})^2}$, where $A_i(\theta_{k(t)})=m_i\cos(\theta_{k(t)}-PD_i)$, this cosine tuning having been reported by many neurophysiological studies.

The neural population generates a force of $\mathbf{F}^t_{k(t)}$ through a decoder matrix $\mathbf{Z}\in\mathbb{R}^{M\times N}$:

$$\mathbf{F}^t_{k(t)}=\mathbf{Z}\mathbf{A}^t(\theta_{k(t)}), \tag{11}$$

where M is the number of outputs, which, in most cases, is set to 2. When $\mathbf{Z}$ is fixed and homogeneous, the $(1,i)$th and $(2,i)$th components of $\mathbf{Z}$ are defined as $Z_{1i}=\frac{1}{N}\cos\varphi_i$ and $Z_{2i}=\frac{1}{N}\sin\varphi_i$, respectively, where division by N is used for scaling and FD φ_i is randomly sampled from a uniform distribution ($i=1,\ldots,N$). When $\mathbf{Z}$ is fixed and non-homogeneous, (Z_{1i},Z_{2i}) is randomly sampled from a probability distribution $P(Z_1,Z_2)$ and divided by N. As a result, the neural network generates a final hand coordinate $\mathbf{x}^t_{k(t)}\in\mathbb{R}^{M\times 1}$:

$$\mathbf{x}^t_{k(t)}=\mathbf{R}\mathbf{F}^t_{k(t)}=\mathbf{R}\mathbf{Z}\mathbf{W}^t\mathbf{t}_{k(t)} \tag{12}$$

which means that $\mathbf{F}^t_{k(t)}$ is perturbed by a rotation $\mathbf{R}=\begin{pmatrix}\cos\phi & -\sin\phi\\ \sin\phi & \cos\phi\end{pmatrix}$ which assumes a visuomotor rotation or curl force field. Rotational perturbations are assumed because many behavioral studies have used them. Because we discuss only the endpoint of the movement, we refer to $\mathbf{x}^t_{k(t)}$ as the motor command. The constrained tasks are those that the neural network generates $\mathbf{x}_{k(t)}$ toward $\mathbf{t}_{k(t)}$, i.e., $\mathbf{x}_{k(t)}=\mathbf{t}_{k(t)}$, which means the number of constrained tasks T is the same as M. We used T instead of M in the following sections.

If the error occurs between $\mathbf{t}$ and $\mathbf{x}$, synaptic weights $\mathbf{W}^t$ are adapted to reduce the squared error, which is defined as $E^t=\frac{1}{2}(\mathbf{t}_{k(t)}-\mathbf{x}^t_{k(t)})^T(\mathbf{t}_{k(t)}-\mathbf{x}^t_{k(t)})=\frac{1}{2}(\mathbf{e}^t_{k(t)})^T\mathbf{e}^t_{k(t)}$, based on a gradient descent method

$$\mathbf{W}^{t+1}=A\mathbf{W}^t+NB\mathbf{Z}^T\mathbf{R}^T\mathbf{e}^t_{k(t)}\mathbf{t}^T_{k(t)}+\sigma_p\zeta^t, \tag{13}$$

where A is the synaptic decay rate, B is the learning rate (B is set to 0.2 in most parts of the present study), σ_p is the strength of synaptic drift, and $\zeta^t\in\mathbb{R}^{N\times 2}$ denotes normal Gaussian random variables. Since each component of $\mathbf{Z}$ is $\mathcal{O}(\frac{1}{N})$, multiplying B by N allows trial-by-trial variation of both $\mathbf{A}$ and $\mathbf{W}$ to be $\mathcal{O}(1)$. As shown in Text S1, the optimal learning rate η^* is $\mathcal{O}(N)$ ($\eta^*=NB^*$), suggesting that we consider a quasi-optimal learning rate. It should be noted that our results hold whether the learning rate is optimal or quasi-optimal because the results do not depend on B. It should also be noted that the amount of variation in $\mathbf{W}$ does not explicitly depend on N.

Learning curve

Equation (13) yields the following update rule of squared error:

$$\begin{aligned}E^{t+1}=\frac{1}{2}(\mathbf{e}^{t+1})^T\mathbf{e}^{t+1}=\frac{1}{2}(\mathbf{e}^t+(1-A)(A\mathbf{I}-B\mathbf{\Lambda})^{-1}\mathbf{t})^T\\ (A\mathbf{I}-B\mathbf{\Lambda})^T(A\mathbf{I}-B\mathbf{\Lambda})(\mathbf{e}^t+(1-A)(A\mathbf{I}-B\mathbf{\Lambda})^{-1}\mathbf{t}),\end{aligned} \tag{14}$$

where $\mathbf{\Lambda}=N\mathbf{R}\mathbf{Z}\mathbf{Z}^T\mathbf{R}^T$, and I denotes the identity matrix. At first, we assume a case in which $K=1$ for simplicity. Because $\mathbf{\Lambda}$ is symmetric, $A\mathbf{I}-B\mathbf{\Lambda}$ can be decomposed as $A\mathbf{I}-B\mathbf{\Lambda}=\mathbf{V}^T(A\mathbf{I}-B\lambda)\mathbf{V}$, where each row of $\mathbf{V}$ is one of the eigenvectors ($\mathbf{V}^T\mathbf{V}=\mathbf{I}$) and each diagonal component of a diagonal matrix λ is one of the eigenvalues of $\mathbf{\Lambda}$. This decomposition transforms equation (14) into the simple form

$$E^{t+1} = \frac{1}{2}(v_1^{t+1})^2 + \frac{1}{2}(v_2^{t+1})^2 = \frac{1}{2}(A-B\lambda_1)^2(v_1^t + (1-A)\frac{s_1}{A-B\lambda_1})^2 + \frac{1}{2}(A-B\lambda_2)^2(v_2^t + (1-A)\frac{s_2}{A-B\lambda_2})^2, \quad (15)$$

where $\mathbf{v}^t = (v_1^t, v_2^t)^T = \mathbf{V}\mathbf{e}^t$ and $\mathbf{s} = (s_1^t, s_2^t)^T = \mathbf{V}\mathbf{t}$. This recurrence formula yields the analytical form of the learning curve:

$$v_i^t = (A-B\lambda_i)^t(v_i^0 - (1-A)\frac{\lambda_i}{1-\lambda_i}s_i) + (1-A)\frac{A-B\lambda_i}{1-(A-B\lambda_i)}s_i \ (i=1,2). \quad (16)$$

Equation (16) requires that the larger the eigenvalues become, the faster the learning speed becomes and the smaller the residual error becomes (Figure S5). Because

$$\mathbf{\Lambda} = \begin{pmatrix} \frac{1}{N}\sum_{i=1}^N \cos^2\varphi_i & \frac{1}{N}\sum_{i=1}^N \cos\varphi_i \sin\varphi_i \\ \frac{1}{N}\sum_{i=1}^N \cos\varphi_i \sin\varphi_i & \frac{1}{N}\sum_{i=1}^N \sin^2\varphi_i \end{pmatrix} \quad (17)$$

whose component is $\mathcal{O}(1)$, simple algebra gives the analytical form of the eigenvalues,

$$\lambda = \frac{1}{2}(N\sum_{i=1}^N Z_{1i}^2 + N\sum_{i=1}^N Z_{2i}^2 \pm \sqrt{(N\sum_{i=1}^N Z_{1i}^2 - N\sum_{i=1}^N Z_{2i}^2)^2 + 4(N\sum_{i=1}^N Z_{1i}Z_{2i})^2}), \quad (18)$$

which are also $\mathcal{O}(1)$, suggesting that learning speed does not depend explicitly on N. Because the learning speed is determined by the smaller eigenvalue, the necessary and sufficient conditions to maximize learning speed, or to maximize the smaller eigenvalue, are

$$N\sum_{i=1}^N Z_{1i}^2 = N\sum_{i=1}^N Z_{2i}^2 \Leftrightarrow \frac{1}{N}\sum_{i=1}^N \cos\varphi_i^2 = \frac{1}{N}\sum_{i=1}^N \sin\varphi_i^2 \quad (19)$$

and

$$N\sum_{i=1}^N Z_{1i}Z_{2i} = 0 \Leftrightarrow \frac{1}{N}\sum_{i=1}^N \cos\varphi_i \sin\varphi_i = 0. \quad (20)$$

What kind of conditions can simultaneously satisfy equations (19) and (20)? The only answer is sufficient neuronal redundancy, i.e., $N\to\infty$, because sufficient neuronal redundancy enables self-averaging properties to exist in a neural network, i.e.,

$$\frac{1}{N}\sum_{i=1}^N \cos^2\varphi_i = \int_0^{2\pi} d\varphi P(\varphi)\cos^2\varphi = \frac{1}{2}, \quad (21)$$

$$\frac{1}{N}\sum_{i=1}^N \sin^2\varphi_i = \int_0^{2\pi} d\varphi P(\varphi)\sin^2\varphi = \frac{1}{2}, \quad (22)$$

and

$$\frac{1}{N}\sum_{i=1}^N \cos\varphi_i \sin\varphi_i = \int_0^{2\pi} d\varphi P(\varphi)\cos\varphi\sin\varphi = 0, \quad (23)$$

where $P(\varphi)$ is the probability distribution in which FDs are randomly sampled. Conversely, if equations (21), (22), and (23) are satisfied in all of the sets of randomly sampled FDs, the number of neurons needs to satisfy $N\to\infty$ because the fluctuation of Monte Carlo integrals is $\mathcal{O}(1/\sqrt{N})$ [24]. Therefore, to maximize learning speed, the necessary and sufficient condition is sufficient neuronal redundancy.

The above analytical calculations hold even when $K>1$. Equation (13) yields the recurrence equation of the squared error:

$$E_{k(t+1)}^{t+1} = \frac{1}{2}(\mathbf{e}_{k(t+1)}^{t+1})^T\mathbf{e}_{k(t+1)}^{t+1} = \frac{1}{2}(\mathbf{e}_{k(t+1)}^t - B\mathbf{\Lambda}\mathbf{e}_{k(t)}^t)^T(\mathbf{e}_{k(t+1)}^t - B\mathbf{\Lambda}\mathbf{e}_{k(t)}^t), \quad (24)$$

where A is set to 1 for simplicity. Using $\mathbf{\Lambda} = \mathbf{V}^T\lambda\mathbf{V}$, this equation can be written as

$$(v_{i,k(t+1)}^{t+1})^2 = (v_{i,k(t+1)}^t - B\lambda_i\cos(\theta_{k(t+1)} - \theta_{k(t)})v_{i,k(t)}^t)^2 (i=1,2). \quad (25)$$

The larger the eigenvalue becomes, the faster learning speed becomes if $v_{i,k(t+1)}^t$ and $v_{i,k(t)}^t\cos(\theta_{k(t+1)} - \theta_{k(t)})$ have the same sign, or if $v_{i,k(t+1)}^t \times \cos(\theta_{k(t+1)} - \theta_{k(t)})v_{i,k(t)}^t > 0$. This inequality is appropriate if the equality $\mathbf{v}_{k(t+1)}^T\mathbf{v}_{k(t)} = \mathbf{e}_{k(t+1)}^T\mathbf{V}^T\mathbf{V}\mathbf{e}_{k(t)} = C\cos(\theta_{k(t+1)} - \theta_{k(t)})$ can be proved, where C is a positive constant. To prove this equality, let us assume that in the 1st trial after the rotational perturbation $\mathbf{R}$ is applied, output can be written as $\mathbf{x}_{k(t)} = \mathbf{R}\mathbf{t}_{k(t)}$ because the neural network can generate accurate outputs if there is no perturbation. In this case,

$$\mathbf{e}_{k(t+1)}^T\mathbf{V}^T\mathbf{V}\mathbf{e}_{k(t+1)} = \mathbf{e}_{k(t+1)}^T\mathbf{e}_{k(t)} = \mathbf{t}_{k(t+1)}^T(\mathbf{I}-\mathbf{R})^T(\mathbf{I}-\mathbf{R})\mathbf{t}_{k(t)} = 2(1-\cos\phi)\cos(\theta_{k(t+1)} - \theta_{k(t)}), \quad (26)$$

where $2(1-\cos\phi)$ is a positive constant. Thus, the larger λ_i becomes, the faster learning speed becomes even when $K>1$; analytical calculations show that neuronal redundancy maximizes learning speed even when $K>1$.

Fixed non-homogeneous decoder

When $\mathbf{Z}$ is fixed and non-homogeneous, i.e., $\mathrm{Mean}(Z_1) = \mu_1$, $\mathrm{Var}(Z_1) = \sigma_1^2$, $\mathrm{Mean}(Z_2) = \mu_2$, $\mathrm{Var}(Z_2) = \sigma_2^2$, and $\mathrm{Cov}(Z_1, Z_2) = \sigma_{cov}$, the necessary and sufficient conditions for maximizing learning speed are given by the following equations:

$$\sigma_1^2 + \mu_1^2 = \sigma_2^2 + \mu_2^2 = \sigma^2, \quad (27)$$

$$\sigma_{cov} - \mu_1\mu_2 = 0, \quad (28)$$

with neuronal redundancy assumed. Equations (27) and (28) can be satisfied when, for example,

$$P(\mathbf{Z}')=\sqrt{\frac{1}{(2\pi)^2|\sum|}}\exp\left(-\frac{1}{2}\mathbf{Z}'^{T}\sum^{-1}\mathbf{Z}'\right), \tag{29}$$

(shown in Figure 3A with $\sum=\begin{pmatrix}1.1 & 0\\ 0 & 1.1\end{pmatrix}$ and $N=100000$), or

$$P(\mathbf{Z}')=0.5\sqrt{\frac{1}{(2\pi)^2|\sum|}}\exp\left(-\frac{1}{2}(\mathbf{Z}'-\mu_1)^{T}\sum^{-1}(\mathbf{Z}'-\mu_1)\right)+ 0.5\sqrt{\frac{1}{(2\pi)^2|\sum|}}\exp\left(-\frac{1}{2}(\mathbf{Z}'-\mu_2)^{T}\sum^{-1}(\mathbf{Z}'-\mu_2)\right), \tag{30}$$

(shown in Figure 3D with $\sum=\begin{pmatrix}0.7 & 0.4\\ 0.4 & 0.7\end{pmatrix}$, $\mu_1=(\sqrt{0.4},-\sqrt{0.4})$, $\mu_2=-\mu_1$ and $N=100000$), where $\mathbf{Z}'=(Z_1,Z_2)^T$. Because the learning rate of motor commands is determined by $B\sigma^2$ (see the following section), σ is determined based on the results of behavioral studies [33]. We cannot analytically calculate the general class of $P(Z_1,Z_2)$ and the distributions of PDs satisfying equations (27) and (28), but broad classes of those distributions can satisfy these equations because the classes include even asymmetric distributions, e.g., when $\sum=\begin{pmatrix}0.7 & \sqrt{0.08}\\ \sqrt{0.08} & 0.9\end{pmatrix}$, $\mu=(\sqrt{0.4},-\sqrt{0.2})$.

Learning rule of decoder Z

When $\mathbf{Z}$ is adaptable, this is also adapted to minimize the squared error:

$$\mathbf{Z}^{t+1}=\mathbf{Z}^{t}+B_Z\mathbf{R}^T\mathbf{e}^{t}_{k(t)}\mathbf{t}^{T}_{k(t)}\mathbf{W}^{t}, \tag{31}$$

where $\mathbf{Z}^0$ is set to η^0/N, η^0 is a normal Gaussian random variable, and B_Z is set to 0.1 in the Adaptable Decoder section and 0.05 in the Importance of Neuronal Redundancy section. This learning rule corresponds to back-propagation [34].

High dimensional tasks

In the Importance of Neuronal Redundancy section, the neural network generates the output $\mathbf{x}\in\mathbb{R}^{T\times 1}$, which is determined by

$$\mathbf{x}^{t}=\mathbf{Z}^{t}\mathbf{W}^{t}\mathbf{t} \tag{32}$$

for the tth trial. An initial value of $\mathbf{Z}^0$ is randomly sampled from the normal Gaussian distribution and divided by N for scaling. The input $\mathbf{t}$ is randomly sampled from the normal Gaussian distribution and is normalized to satisfy $\mathbf{t}^T\mathbf{t}=1$ to avoid the effect of this value on learning speed. In addition, we used a fixed value of $\mathbf{t}$ because the generalization function (see the following section) strongly depends on T, i.e., $\mathbf{t}^t=\mathbf{t}$. It should be noted that learning speed does not explicitly depend on T because learning speed is determined only by the minimum eigenvalue of $N\mathbf{Z}\mathbf{Z}^T$.

The generalization function and the update rule for motor commands

Equation (13) yields the following update rule for motor commands:

$$\mathbf{x}^{t+1}_{k(t+1)}=A\mathbf{x}^{t}_{k(t+1)}+B\mathbf{R}\mathbf{Z}\mathbf{Z}^T\mathbf{R}^T\mathbf{e}^{t}_{k(t)}\mathbf{t}^{T}_{k(t)}\mathbf{t}_{k(t+1)}. \tag{33}$$

If equations (27) and (28) (or (22) and (23)) are satisfied, equation (33) can be written as

$$\mathbf{x}^{t+1}_{k(t+1)}=A\mathbf{x}^{t}_{k(t+1)}+B\sigma^2 f(\theta_{k(t+1)}-\theta_{k(t)})\mathbf{e}^{t}_{k(t)}, \tag{34}$$

where the cross term of $\mathbf{t}^T_{k(t)}$ and $\mathbf{t}_{k(t+1)}$ determines the generalization function $f(\theta_{k(t+1)}-\theta_{k(t)})$, e.g., $f(\theta_{k(t+1)}-\theta_{k(t)})=\cos(\theta_{k(t+1)}-\theta_{k(t)})$, if we define $\mathbf{t}_{k(t)}=(\cos\theta_{k(t)},\sin\theta_{k(t)})^T$. We set B and σ^2 to satisfy $B\sigma^2=0.2$. It should be noted that equation (34) corresponds to a model for sensorimotor learning that can explain the results of behavioral experiments [35], suggesting that our hypothesis is consistent with the results of behavioral experiments.

Because the shape of the generalization function depends on the task, we need to confirm the generality of our results with regard to the shape of the generalization function. To simulate various shapes of generalization functions, we used the von-Mises function

$$t_{k(t)}(\theta)=\frac{1}{Z_I}\left(\exp(\alpha\cos(\theta_{k(t)}-\mu_i))-\frac{1}{N_I}\sum_{n_I=1}^{N_I}\exp(\alpha\cos(\theta-\mu_{n_I}))\right), \tag{35}$$

where α, μ_i, and N_I are the precision parameter, the preferred direction of the ith input unit, and the number of input units, respectively. The normalization factor Z_I is determined to make $\mathbf{t}^T_{k(t)}\mathbf{t}_{k(t)}=1$ to avoid the influence of this value on the learning speed, where $\mathbf{t}=(t_1,...,t_{N_I})^T$. This normalization permits us to investigate the influence of the shape of the generalization function alone on learning speed. The larger the value of α, the sharper the shape of the generalization function becomes. We set N_I to 100 throughout this study.

Numerical simulation procedure

We conducted 100 baseline trials with $\phi=0$ and $K=8$ to identify the baseline values of $\mathbf{W}$. The initial value of $\mathbf{W}$, $\mathbf{W}^0$, was set to 0. After these trials, 100 learning trials were conducted using $\phi=\frac{\pi}{3}$ and $K=8$. Learning speed b was calculated by fitting the exponential function $\hat{E}^t=a\exp(-bt)+c$ to E^t. All the figures denote b which was obtained only in learning trials. The present study calculated learning speed and learning curves by averaging the results of 1000 sets of baseline and learning trials, each set including an identical target sequence that was randomly sampled, and each set using different FD values.

For all of the statistical tests, we used the Wilcoxon sign rank test. It should be noted that the p-value was indicated only if the value was significantly different from 0; no statistically significant differences were detected.

Supporting Information

Figure S1 Relationship between learning speed, neuronal redundancy, and adaptable recurrent connections ($K=8$). (A): Learning speed when $N=4,10,50,100$ and $B_M=0,0.025,0.05,0.075,0.1$. The whiter the color, the faster the learning speed. (B): Learning curves obtained when N = 10, 50, or 100 and $B_M=0.025$. These curves show the average values of 1,000 randomly sampled sets of φ. Error bars represent the

standard deviations of the errors. (C): Learning curves obtained when $B_M = 0,0.05,0.1$ and $N = 10$. These curves and error bars show average values and standard deviations. (D): Variance of the learning curve when $B_M = 0,0.05,0.1$ and $N = 100$ ($K = 8$). These variances are average values from 1,000 randomly sampled sets of φ.
(EPS)

Figure S2 Relationship between learning speed and neuronal redundancy in the case of a nonlinear neural network ($K = 8$). (A): Learning speed when N = 10, 50, 100, and 1000. The bar graphs and error bars depict sample means and standard deviations, both of which are calculated using the results of 1,000 randomly sampled sets of φ values. (B): Learning curves obtained when $N = 4,10$, or 100. These curves and error bars show average values and the standard deviations of the errors.
(EPS)

Figure S3 Relationship between learning speed and neuronal redundancy when the neural network includes nonlinear muscle units ($K = 8$). (A): The bar graphs and error bars depict sample means and standard deviations, both of which were calculated using the results of 1,000 randomly sampled sets of **C** values. (B): Learning curves obtained when $N = 10$ or 100. These curves and error bars show average values and the standard deviations of the errors.
(EPS)

Figure S4 Relationship between learning speed and neuronal redundancy in the case of weight perturbation and node perturbation ($K = 8$). (A): Learning speed when $N = 4,10,100$, or 1000, with weight perturbation as the learning rule. The bar graphs and error bars depict sample means and standard deviations, both of which are calculated using the results of 1,000 randomly sampled sets of φ. (B): Learning curves obtained when $N = 4,10$, or 100, with weight perturbation as the learning rule. These curves and error bars show the average values and the standard deviations of the errors. (C): Learning speed when $N = 4,10,100$, or 1000, with node perturbation as the learning rule. The bar graphs and error bars depict sample means and standard deviations, both of which are calculated using the results of 1,000 randomly sampled sets of φ. (D): Learning curves obtained when $N = 4,10$, or 100, with node perturbation as the learning rule. These curves and error bars show average values and the standard deviations of the errors.
(EPS)

Figure S5 Relationship between residual error, learning speed, and neuronal redundancy with synaptic decay included ($K = 8$). (A): Residual error when $A = 0$. The bar graphs and error bars denote sample means and standard deviations, both of which are calculated using the results of 1,000 randomly sampled sets of φ values. (B): Learning speed when $A = 0$. The bar graphs and error bars depict sample means and standard deviations. (C): Learning curves obtained when $N = 4,10$, and 100 and $A = 0$. These curves and error bars show average values and standard deviations. (D): Residual error when $A = 0.005$. (E): Learning speed when $A = 0.005$. (F): Learning curve when $A = 0.005$. (G): Residual error when $A = 0.01$. (H): Learning speed when $A = 0.01$. (I): Learning curve when $A = 0.01$.
(EPS)

Text S1 Generality of our results. This file contains the detailed descriptions of *Generality of our results* section.
(PDF)

Acknowledgments

We thank D. Nozaki, Y. Sakai, Y. Naruse, K. Katahira, T. Toyoizumi, and T. Omori for their helpful discussions.

Author Contributions

Conceived and designed the experiments: KT. Performed the experiments: KT. Analyzed the data: KT. Contributed reagents/materials/analysis tools: KT MO. Wrote the paper: KT MO.

References

1. Barlow H (2001) Redundancy reduction revisited. Network 12: 241–253.
2. Olshausen BA, Field DJ (1996) Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature 381: 607–609.
3. Lewicki MS (2002) Efficient coding of natural sounds. Nat Neurosci 5: 356–363.
4. Gutnisky D, Dragoi V (2008) Adaptive coding of visual information in neural populations. Nature 452: 220–224.
5. Salinas E (2006) How behavioral constraints may determine optimal sensory representations. PLoS Biol 4: 2383–2392.
6. Li CS, Padoa-Schioppa C, Bizzi E (2001) Neuronal correlates of motor performance and motor learning in the primary motor cortex of monkeys adapting to an external force field. Neuron 30: 593–607.
7. Rokni U, Richardson AG, Bizzi E, Seung HS (2007) Motor learning with unstable neural representations. Neuron 54: 653–666.
8. Narayanan NS, Kimchi EY, Laubach M (2005) Redundancy and synergy of neuronal ensembles in motor cortex. J Neurosci 25: 4207–4216.
9. Bernstein N (1967) The coordination and regulation of movements. London: Pergamon.
10. Gribble PL, Mullin LI, Cothros N, Mattar A (2003) Role of cocontraction in arm movement accuracy. J Neurophysiol 89: 2396–2405.
11. Latash ML, Scholz JP, Schoner G (2002) Motor control strategies revealed in the structure of motor variability. Exerc Sport Sci Rev 30: 26–31.
12. Thoroughman KA, Shadmehr R (1999) Electromyographic correlates of learning an internal model of reaching movements. J Neurosci 19: 8573–8588.
13. Latash ML (2000) The organization of quick corrections within a two-joint synergy in conditions of unexpected blocking and release of a fast movement. Clin Neurophysiol 11: 975–987.
14. Yang JF, Scholz JP, Latash ML (2007) The role of kinematic redundancy in adaptation of reaching. Exp Brain Res 176: 54–69.
15. Scott SH, Gribble PL, Cabel DW (2001) Dissociation between hand motion and population vectors from neural activity in motor cortex. Nature 413: 161–165.
16. Kurtzer I, Pruszynski JA, Herter TM, Scott SH (2006) Nonuniform distribution of reach-related and torque-related activity in upper arm muscles and neurons of primary motor cortex. J Neurophysiol 96: 3220–3230.
17. Naselaris T, Merchant H, Amirikian B, Georgopoulos AP (2006) Large-scale organization of preferred directions in the motor cortex. I. motor cortical hyperacuity for forward reaching. J Neurophysiol 96: 3231–3236.
18. Werfel J, Xie X, Seung S (2005) Learning curves for stochastic gradient descent in linear feedforward networks. Neural Compt 17: 2699–2718.
19. Thoroughman KA, Shadmehr R (2000) Learning of action through adaptive combination of motor primitives. Nature 407: 742–747.
20. Donchin O, Francis JT, Shadmehr R (2003) Quantifying generalization from trial-by-trial behavior of adaptive systems that learn with basis functions. J Neurosci 23: 9032–9045.
21. Krakauer JW, Pine ZM, Ghilardi MF, Ghez C (2000) Learning of visuomotor transformations for vectorial planning of reaching trajectories. J Neurosci 20: 8916–892.
22. Lewicki MS, Sejnowski TJ (2000) Learning Overcomplete Representations. Neural Comput 12: 337–365.
23. Hidetoshi N (2001) Statistical Physics of Spin Glasses and Information Processing: An Introduction. Oxford University Press.
24. Bishop CM (2006) Pattern Recognition and Machine Learning. Springer.
25. Capaday C, Ethier C, Brizzi L, Sik A, van Vreewijk C (2009) On the nature of the intrinsic connectivity of the cat motor cortex: evidence for a recurrent neural network topology. J Neurophysiol 102: 2131–2141.
26. Tsodyks M, Markram H (1997) The neural code between neocortical pyramidal neurons depends on neurotransmitter release probability. Proc Natl Acad Sci U S A 94: 719–723.
27. Seung HS (2003) Learning in spiking neural networks by reinforcement of stochastic synaptic transmission. Neuron 40: 1063–1073.
28. Fiete IR, Fee MS, Seung HS (2007) Model of birdsong learning based on gradient estimation by dynamic perturbation of neural conductances. J Neurophysiol 98: 2038–2057.
29. Lee D, Port NL, Kruse W, Georgopoulos AP (1998) Variability and correlated noise in the discharge of neurons in motor and parietal areas of the primate cortex. J Neurosci 18: 1161–1170.

30. Shadmehr R, Mussa-ivaldi FA (1994) Adaptive representation of dynamics during learning of a motor task. J Neurosci 14: 3208–3224.
31. Criscimagna-Hemminger SE, Donchin O, Gazzaniga MS, Shadmehr R (2003) Learned dynamics of reaching movements generalize from dominant to nondominant arm. J Neurophysiol 89: 168–176.
32. Simoncelli EP, Freeman WT, Adelson EH, Heeger DJ (1992) Shiftable multiscale transforms. IEEE Trans Info Theory 38: 587–607.
33. Cheng S, Sabes PN (2007) Calibration of visually guided reaching is driven by error-corrective learning and internal dynamics. J Neurophysiol 97: 3057–3069.
34. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by backpropagating errors. Nature 323: 533–536.
35. van Beers RJ (2009) Motor learning is optimally tuned to the properties of motor noise. Neuron 63: 406–417.