通过突触智能持续学习

Friedemann Zenke [*1] Ben Poole [*1] Surya Ganguli [1]

摘要。

虽然深度学习在不同的应用中带来了显著的进步，但在数据分布随着学习过程发生变化的领域，它会遇到困难。与之形成鲜明对比的是，生物神经网络不断适应变化的领域，可能是通过利用复杂的分子机械来同时解决许多任务。在这项研究中，*我们引入了智能突触*，将这种生物复杂性的一部分带入人工神经网络。每个突触随着时间的推移积累与任务相关的信息，并利用这些信息快速存储新的记忆，而不会忘记旧的记忆。在分类任务的持续学习上评估了所提出的方法，并表明它在保持计算效率的同时显著减少了遗忘。

1. 介绍

人工神经网络(ANNs)已经成为应用机器学习不可或缺的资产，在各种特定领域的任务中与人类的表现相媲美(LeCun等人，2015)。虽然最初受到生物学的启发(Rosenblatt, 1958;Fukushima & Miyake, 1982)，其潜在的设计原则和学习方法与生物神经网络有本质的区别。例如，在训练阶段在数据集上学习人工神经网络的参数，然后在部署或召回阶段静态地冻结和使用新数据。为了适应数据分布的变化，人工神经网络通常必须在整个数据集上重新训练，以避免过拟合和灾害性遗忘(Choy等人，2006; Goodfellow et al.， 2013)。

另一方面，生物神经网络表现*出持续的学习能力,在*这种*学习*中，它们获得了新的知识一生。因此，很难在学习阶段和回忆阶段之间划出一条清晰的界线。不知为何，我们的大脑已经进化到可以从非平稳数据中学习，并随时更新内部记忆或信念。虽然尚不清楚这种壮举是如何在大脑中完成的，但似乎有可能在持续学习中无与伦比的生物性能可能依赖于底层生物湿件实现的特定功能，而这些功能目前尚未在人工神经网络中实现。

也许现代人工神经网络与生物神经网络设计的最大差距之一在于突触的复杂性。在人工神经网络中，单个突触(权重)通常由单个标量描述。另一方面，个体生物突触利用复杂的分子机制，可以在不同的空间和时间尺度上影响可塑性(Redondo & Morris, 2011)。虽然这种复杂性被推测有助于记忆巩固(Fusi等人，2005;Lahiri & Ganguli, 2013;Zenke 等，2015; 齐格勒等，2015;Benna & Fusi, 2016)，很少有研究说明它如何有益于人工神经网络的学习。

在这里，我们研究了内部突触动力学在使人工神经网络学习分类任务序列中的作用。虽然简单，标量一维突触遭受catas-营养性遗忘，即网络在接受新任务训练时忘记以前学习的任务，但这个问题可以通过具有更复杂的三维状态空间的突触在很大程度上缓解。在我们的模型中，突触状态跟踪过去和当前的参数值，并保持对突触在解决过去遇到的问题方面的“重要性”的在线估计。我们的重要性度量可以在训练过程中高效和局部地计算在每个突触上，并代表每个突触对全局损失变化的局部贡献。当任务发生变化时，我们通过防止它们在未来任务中发生变化来巩固重要的突触。因此，未来任务中的学习主要是由对过去任务不重要的突触介导的，从而避免对这些过去任务的灾难性遗忘。

[*] Equal contribution [1]Stanford University. Correspondence to: Friedemann Zenke <fzenke@stanford. edu>, Ben Poole <poole@cs. stanford.edu>.

Proceedings of the 34 [th] *International Conference on Machine Learning*, Sydney, Australia, PMLR 70, 2017.

2. 之前工作

缓解灾难性遗忘的问题在以前的许多研究中都得到了解决。这些研究可以

大致分为(1)建筑学，(2)功能性和(3)结构方法。

灾难性遗忘的架构方法改变网络的架构，以减少任务之间的干扰，而不改变目标函数。最简单的架构正则化形式是冻结网络中的某些权重，以便它们保持完全相同(Razavian et al.， 2014)。稍微放松的方法降低了与原始任务共享的层的学习率，同时进行微调以避免参数的剧烈变化(Donahue et al.， 2014; Yosin-ski et al.， 2014)。使用不同非线性的方法，如ReLU、MaxOut和local winner-take-all，已被证明可以提高置换MNIST和情绪分析任务的性能(Srivastava等人，2013;goodfellow等人，2013)。此外，使用dropout注入噪声来稀疏梯度也可以提高性能(goodfellow等人，2013)。Rusu et al.(2016)最近的工作提出了更引人注目的架构变化，即在解决新任务的同时，复制上一个任务的整个网络并使用新特征进行增强。这完全防止了对早期任务的遗忘，但会导致架构的复杂性随着任务数量的增加而增加。

灾难性遗忘的函数式方法在目标中添加了正则化项，惩罚神经网络输入输出函数的变化。在Li & Hoiem (2016)中，通过使用知识蒸馏的形式将上一个任务的网络和当前网络的预测应用于新任务的数据时，鼓励相似(Hinton等人，2014)。类似地，Jung等人(2016)正则化`2最终隐藏激活之间的距离，而不是知识蒸馏惩罚。这两种正则化的方法都旨在通过使用旧任务的参数存储或计算额外的激活来保留旧任务的输入-输出映射的方面。这使得灾难性遗忘的函数式方法在计算上非常昂贵，因为它需要为每个新数据点计算通过旧任务网络的前向传递。

第三种技术，结构正则化，涉及到对参数的惩罚，鼓励它们保持与旧任务的参数接近。最近，Kirkpatrick等人(2017)提出弹性权重固合(EWC)，对新任务和旧任务的参数差异进行二次惩罚。他们在旧任务上使用了与费雪信息度量的对角线成比例的对角线权重，而不是旧参数。精确计算费雪的对角线需要对所有可能的输出标签求和，因此复杂度与输出数量呈线性关系。这限制了这种方法在低维输出空间的应用。

3. 突触框架

为了解决神经网络中持续学习的问题，我们试图构建一个简单的结构正则化器，该正则化器可以在线计算并在每个突触本地实现。具体来说，我们的目标是在解决网络过去训练过的任务时，赋予每个单独的突触“重要性”的局部度量。当训练一个新任务时，我们惩罚对重要参数的改变，以避免旧的记忆被覆盖。为此，我们开发了一类算法，跟踪*重要性度量*ωk μ，它反映了任务目标改进的过去信用L_{μ}任务μ到单个突触θ_{k}。为简洁起见，我们将术语“突触”与术语“参数”同义使用，其中包括层之间的权重以及偏差。

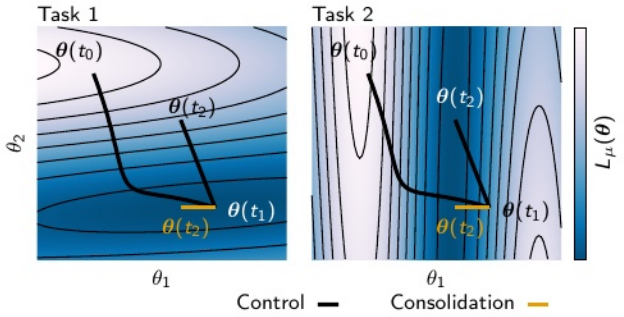

图1所示。参数空间轨迹和灾难性遗忘的示意图。实线对应训练期间的参数轨迹。左右面板对应不同任务(Task 1和Task 2)定义的不同损失函数。每个损失函数的值L_{μ}is显示为热图。任务1上的梯度下降学习在参数空间中诱导了一个从$\theta(t_0)$到$\theta(t_1)$的运动。任务2上的后续梯度下降动力学产生了参数空间中从$\theta(t_1)$到$\theta(t_2)$的运动。这最后一点以显著增加任务1的损失为代价，将任务2的损失最小化，从而导致对任务1的灾难性遗忘。然而，确实存在一个备用点$\theta(t_2)$，用橙色标记，它在*两个任务中都*实现了小的损失。在下面的文章中，我们将通过确定成分θ_2在解决任务1中比θ_1更重要，然后在解决任务2时防止θ_2发生太大变化，来展示如何找到这个替代点。这导致了一种通过巩固对解决过去任务很重要的参数的变化来避免灾难性遗忘的在线方法，同时只允许不重要的参数学习解决未来的任务。

训练神经网络的过程以参数空间中的轨迹θ(t)为特征(图1)。成功训练的壮举在于找到在所有任务上端点位于损失函数L的最小值附近的学习轨迹。让我们首先考虑一个无穷小的参数更新δ(t)在时间t时的损失变化。

在这种情况下，损失的变化很好地近似于梯度g = ∂ ∂ L θ，我们可以这样写

$$L(\boldsymbol{\theta}(t) + \boldsymbol{\delta}(t)) - L(\boldsymbol{\theta}(t)) \approx \sum_k g_k(t)\delta_k(t)\,, \qquad (1)$$

这说明每个参数变化$\delta_k(t)=\theta_k^0(t)$都贡献了$g_k(t)\delta_k(t)$对总损失变化的贡献量。

要计算通过参数空间的整个轨迹上的损失变化，我们必须对所有无穷小的变化求和。这相当于计算梯度向量场沿参数轨迹从初始点(在时间t_0)到最终点(在时间t1)的路径积分:

$$\int_C \boldsymbol{g}(\boldsymbol{\theta}(t))d\boldsymbol{\theta} = \int_{t_0}^{t_1} \boldsymbol{g}(\boldsymbol{\theta}(t)) \cdot \boldsymbol{\theta}'(t)dt. \qquad (2)$$

由于梯度是一个保守场，因此积分值等于终点和起点之间的损失之差:$L(\theta(t_1))-L(\theta(t_0))$。对于我们的方法至关重要的是，我们可以将Eq. 2分解为单个参数的总和

$$\begin{aligned}\int_{t^{\mu-1}}^{t^{\mu}} \boldsymbol{g}(\boldsymbol{\theta}(t)) \cdot \boldsymbol{\theta}'(t)dt &= \sum_k \int_{t^{\mu-1}}^{t^{\mu}} g_k(\theta(t))\theta_k'(t)dt \\ &\equiv -\sum_k \omega_k^{\mu}. \qquad (3)\end{aligned}$$

ωk μ现在有一个直观的解释，即参数对总损失变化的具体贡献。请注意，我们在第二行中引入了负号，因为我们通常对减少损失感兴趣。

在实践中，我们可以在线近似ωk μ作为梯度gk (t) = ∂ ∂θL与κ的乘积的运行和参数更新θk0(t)∂ ∂θtκ。对于具有无穷小学习率的批量梯度下降，ω_k^{μ}can可以直接解释为每个参数对总损失变化的贡献。在大多数情况下，真实的梯度被随机梯度下降(SGD)近似，导致在g_k的估计中引入噪声的近似。一个直接的后果是，近似的每参数重要性通常会高估ωk μ的真实值。

如何利用ωk μ的知识来提高持续学习?我们试图解决的问题是最小化对所有任务求和的总损失函数，L= P μ L μ，具有限制，我们无法访问我们过去训练的任务的损失函数。相反，我们在任何给定的时间都只能访问损失函数L_{μ}for单个任务μ。当最小化L_{μ} inadvertently时，会出现灾难性遗忘，导致先前任务的成本大幅增加L_{ν}withν<μ

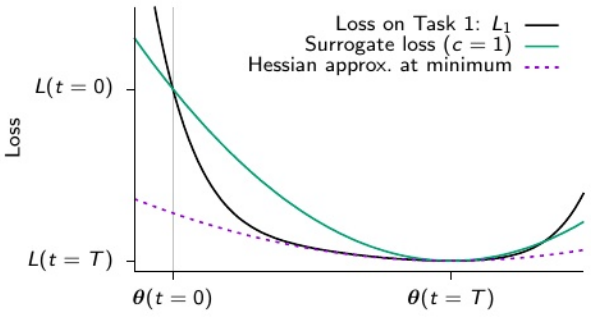

图2。学习一项任务后代理损失示意图。考虑任务1(黑色)定义的一些损失函数。选择二次代理损失(绿色)是为了精确匹配原始损失函数上下降动力学的3个方面:损失函数L(θ(0)) - L(θ(T))中的总下降，参数空间θ(0) - θ(T)中的to-tal净运动，以及在端点θ(T)实现最小值。这3个条件唯一地决定了替代二次损失，它总结了原始损失上的下降轨迹。请注意，这个代理损失不同于海森在最小值处定义的二次近似(紫色虚线)。

(图1).为了避免在训练任务μ时对所有以前的任务(ν < μ)的灾难性遗忘，我们希望避免对过去特别有影响的权重的剧烈变化。参数θ_k对于单个任务的重要性由两个量决定:1)单个参数对损失下降的贡献程度ω_k^{ν} over整个训练轨迹(参见Eq. 3)和2)它移动的距离Δν k≡$\theta_k(t^{\nu})-\theta_k(t^{\nu-1})$。为了避免重要参数的大变化，我们使用了修改后的成本函数L⊠μin，我们引入了代理损失，它近似于之前任务的损失函数的总和L_{ν}(ν<μ)。具体来说，我们使用二次代理损失，它与之前任务的成本函数具有相同的最小值，并产生相同的ω_k^{ν}over参数距离Δ_k。换句话说，如果要在替代损失而不是实际损失函数上进行学习，它将导致相同的最终参数和训练期间损失的变化(图2)。对于两个任务，这完全是通过以下二次代理损失实现的

$$\tilde{L}_{\mu} = L_{\mu} + \underbrace{c\sum_k \Omega_k^{\mu}\left(\tilde{\theta}_k - \theta_k\right)^2}_{\text{surrogate loss}} \qquad (4)$$

其中我们引入了无量纲强度参数c，即上一个任务结束时参数对应的参考权重θ⊠k=$\theta_k(t^{\mu-1})$,

以及每个参数的正则化强度:

$$\Omega_k^{\mu} = \sum_{\nu<\mu} \frac{\omega_k^{\nu}}{(\Delta_k^{\nu})^2 + \xi} \quad . \tag{5}$$

请注意，分母$(\Delta\nu_k)^2$中的项确保正则化项携带与损失l相同的单位。出于实际原因，我们还引入了一个额外的阻尼参数ξ，以在Δνk→0的情况下限制表达式。最后，c是一个强度参数，用于权衡旧记忆与新记忆。如果精确地计算路径积分(Eq. 3)，则c = 1将对应于新旧记忆的相等权重。然而，由于在计算路径积分时存在噪声(Eq. 3)，通常必须选择小于1的c来进行补偿。除非另有说明，ω_k在训练期间不断更新，而累积的重要性度量，Ωμk和参考权重，θ~，只在每个任务结束时更新。在更新Ωμk后，ω_k被设置为零。虽然我们将Eq. 4作为替代损失的动机仅在两个任务的情况下成立，但我们将通过经验证明，我们的方法在学习其他任务时具有良好的性能。

为了理解eq . 4和5的特定选择如何影响学习，让我们考虑图1中所示的示例，在该示例中我们学习两个任务。我们首先在任务1上进行训练。在时间t_1参数已经接近Task 1损失L_1的局部最小值。但是，同样的参数配置对于Task 2并没有接近最小值。因此，在没有任何额外预防措施的情况下对Task 2进行训练时，L_1损失可能会无意中增加(图1，黑色轨迹)。然而，当θ_2“记住”降低L_1很重要时，它可以在任务2的训练过程中通过保持接近其当前值来利用这种知识(图1，橙色轨迹)。虽然这几乎不可避免地会导致任务2的性能下降，但这种下降可以忽略不计，而两项任务的性能提高结合起来可能是实质性的。

这里提出的方法类似于EWC (Kirk-patrick et al.， 2017)，因为更有影响力的参数被更强烈地拉回一个参考权重，在之前的任务中获得了良好的性能。然而，与EWC相比，我们提出了一种在线计算*整个学习轨迹的重要性度量的方法*，而EWC依赖于最终参数值处Fisher信息度量对角线的点估计，必须在每个任务结束时的单独阶段进行计算。

4. 特殊情况的理论分析

在下文中，我们说明了我们的一般方法在简单和可分析的训练场景的情况下恢复了明智的Ωμk。为此，我们分析了

参数特定路径积分ω_ku及其归一化版本Ωμk (Eq.(5))在简单二次误差函数的几何尝试中的对应关系

$$E(\boldsymbol{\theta}) = \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}^*)^T \boldsymbol{H} (\boldsymbol{\theta} - \boldsymbol{\theta}^*), \tag{6}$$

以θ^*and为最小值的Hessian矩阵h进一步考虑该误差函数上的批梯度下降动力学。在小离散时间学习率的极限下，此下降动力学由连续时间微分方程描述

$$\tau \frac{d\boldsymbol{\theta}}{dt} = -\frac{\partial E}{\partial \boldsymbol{\theta}} = -\boldsymbol{H}(\boldsymbol{\theta} - \boldsymbol{\theta}^*), \tag{7}$$

其中τ与学习率有关。如果我们在时间t = 0时从初始条件θ(0)开始，则下降路径的精确解为

$$\boldsymbol{\theta}(t) = \boldsymbol{\theta}^* + e^{-\boldsymbol{H}\frac{t}{\tau}}(\boldsymbol{\theta}(0) - \boldsymbol{\theta}^*), \tag{8}$$

产生时间依赖的更新方向

$$\boldsymbol{\theta}'(t) = \frac{d\boldsymbol{\theta}}{dt} = -\frac{1}{\tau}\boldsymbol{H} e^{-\boldsymbol{H}\frac{t}{\tau}}(\boldsymbol{\theta}(0) - \boldsymbol{\theta}^*). \tag{9}$$

现在，在梯度下降动力学下，梯度服从g = τddt θ，因此(3)中的ωk μ被计算为矩阵的对角元素

—

$$\boldsymbol{Q} = \tau \int_0^{\infty} dt \, \frac{d\boldsymbol{\theta}}{dt} \frac{d\boldsymbol{\theta}}{dt}^T . \tag{10}$$

可以用Hessian H的特征基给出Q的显式公式，特别地，设λ^{α}andu^{α}denote为H的特征值和特征向量，设$d^{\alpha}=u^{\alpha}\cdot(\theta(0)-\theta^*)$为初始参数和最终参数之间的差异在第α′特征向量上的投影。然后将(9)插入(10)，对H的特征模态进行基的变换，并进行积分

$$\boldsymbol{Q}_{ij} = \sum_{\alpha\beta} \boldsymbol{u}_i^{\alpha} d^{\alpha} \frac{\lambda^{\alpha}\lambda^{\beta}}{\lambda^{\alpha} + \lambda^{\beta}} d^{\beta} \boldsymbol{u}_j^{\beta}. \tag{11}$$

注意，作为一个时间积分的稳态量，Q不再依赖于支配下降路径速度的时间常数τ。

乍一看，Q矩阵元素以复杂的方式依赖于Hessian矩阵的特征向量和特征值，以及初始条件θ(0)。为了理解这种依赖关系，我们首先考虑在随机初始条件θ(0)上求Q的平均，这样差值dα的集合构成了一组方差σ2、均值为零的iid随机变量。因此我们有了平均年龄$hd^{\alpha}d^{\beta}i=\sigma^2\delta_{\alpha\beta}$。在Q上执行这个平均值，那么就会产生

$$\langle \boldsymbol{Q}_{ij} \rangle = \frac{1}{2}\sigma^2 \sum_{\alpha} \boldsymbol{u}_i^{\alpha} \lambda^{\alpha} \boldsymbol{u}_j^{\beta} = \frac{1}{2}\sigma^2 \boldsymbol{H}_{ij}. \tag{12}$$

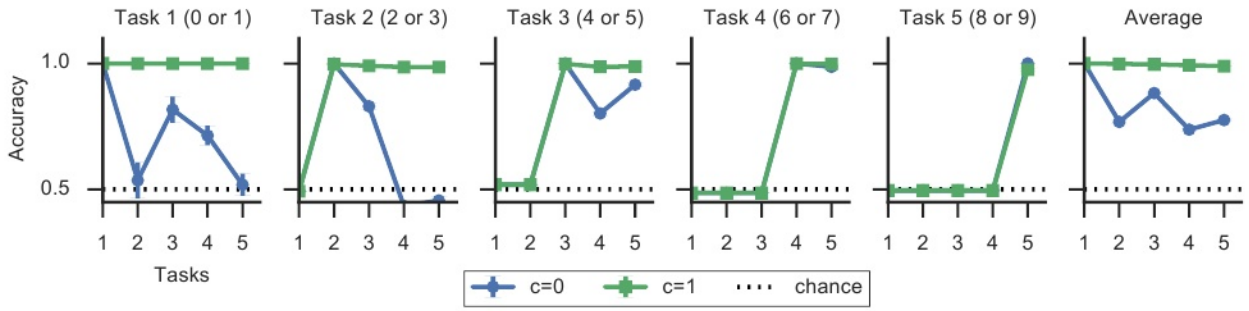

图3。分割MNIST基准的平均分类精度作为任务数量的函数。前五个面板显示了五个任务的分类精度，每个任务由两个MNIST数字组成，作为连续任务数量的函数。最右边的面板显示平均精度，它是作为过去任务的平均精度ν与ν< μ的任务精度计算的，其中μ由x轴上的任务数量给出。请注意，在这个具有多个二进制读出头的设置中，0.5的精度对应于机会水平。误差条对应于SEM (n=10)。

因此，值得注意的是，在对初始条件进行平均后，Q矩阵(只需通过将突触对之间的参数更新相关联并随时间积分即可获得)减少到黑森系数，直至决定初始条件和最终条件之间差异的比例因子。事实上，这个比例因子理论上激励了(5);(5)中的分母，在零阻尼下，ξ的平均值为σ2，从而去掉了(12)中的比例因子σ2

然而，我们感兴趣的是Q_{ij}为*单个*初始条件计算了什么。在两种情况下，Q和黑森H之间的简单关系保持不变，*而不*需要在初始条件上取平均值。首先，考虑黑森线是对角线的情况，使uαi = δαi ei，其中ei是第i个坐标向量。则α和i指标可互换，黑森的特征值为黑森的对角元:λi = Hii。则(11)化简为

$$\boldsymbol{Q}_{ij} = \delta_{ij}(d^i)^2 \boldsymbol{H}_{ii}. \tag{13}$$

在零阻尼下，(5)中的归一化再次消除了参数空间$(d^i)^2$中的运动尺度，因此归一化的Q矩阵与对角线黑森矩阵相同。在第二个场景中，考虑黑森秩为1的极限，使得λ^1是唯一的非零特征值。那么(11)化简为

$$\boldsymbol{Q}_{ij} = \frac{1}{2}(d^1)^2 \boldsymbol{u}_i^1 \lambda_1 \boldsymbol{u}_j^1 = \frac{1}{2}(d^1)^2 \boldsymbol{H}_{ij}. \tag{14}$$

因此，Q矩阵再次简化为黑森矩阵，直到一个比例因子。归一化的重要性然后成为非对角线但低秩的hessian的对角线元素。我们注意到低阶黑森是持续学习的有趣案例;误差函数中的低秩结构使突触权重空间中的许多方向不受给定任务的约束，为突触修改留下开放的多余容量，以解决未来的任务，而不会干扰旧任务的性能。

重要的是要强调，重要性的路径积分是通过沿整个学习轨迹积分信息来计算的(参见图2)。对于二次损失函数，黑森沿着这条轨迹是恒定的，因此我们发现重要性和黑森之间存在精确的关系。但对于更一般的损失函数，其中黑森沿着轨迹变化，我们不能期望在学习端点的重要性Ωμk和黑森之间有任何简单的数学对应关系，或参数灵敏度的相关度量(Pascanu & Ben-gio, 2013;马顿斯,2016;Kirkpatrick et al.， 2017)在终点。然而，在实践中，我们发现我们的重要性度量与基于此类端点估计的度量是相关的，这可能解释了它们的可比有效性，我们将在下一节中看到。

5. 实验

我们在分裂和排列MNIST上评估了我们的持续学习方法(LeCun等人，1998;goodfellow等人，2013)，以及CIFAR-10和CIFAR-100的分割版本(Krizhevsky & Hinton, 2009)。

5.1. 分裂MNIST

我们首先在一个分裂的MNIST基准上评估我们的算法。对于这个基准测试，我们将完整的MNIST训练数据集分成5个连续数字子集。这5个任务对应于学习区分从0到10的两个连续数字。我们使用了一个小型的多层感知器(MLP)，只有两个隐藏层，每个隐藏层由256个单元组成，每个单元具有ReLU非线性，以及一个标准

类别交叉熵损失函数加上我们的巩固成本项(带有阻尼参数$\xi = 1\times10^{-3}$)。为了避免由于训练过程中标签分布的变化而导致读出层数字之间串扰的复杂性，我们使用了一种多头方法，其中读出层的分类交叉熵损失仅针对当前任务中存在的数字进行计算。最后，我们使用64的小批量大小优化了我们的网络，并训练了10个epoch。为了在较少的epoch数量下实现良好的绝对性能，我们使用了自适应优化器Adam (Kingma & Ba, 2014) ($\eta = 1\times10^{-3}$，$\beta_1 = 0.9$，$\beta_2 = 0.999$)。在这个基准测试中，优化器状态在训练每个任务后被重置。

为了评估性能，我们计算了所有先前任务的平均分类精度，作为训练任务数量的函数。我们现在比较我们打开巩固动态(c = 1)的网络与关闭巩固(c = 0)的网络之间的这种性能。在训练第一个任务期间，两种情况下的巩固惩罚都是零，因为没有突触可以正则化的过去经验。当对数字“2”和“3”(任务2)进行训练时，有巩固和没有巩固的模型在任务2上都显示出接近1的精度。然而，平均而言，没有突触巩固的网络在任务1上的准确性显示出实质性的损失(图3)。相比之下，经过巩固的网络在任务1上的准确性只经历了轻微的损伤，两项任务的平均精度都保持接近1。同样，当网络看到所有的MNIST数字时，平均而言，在没有整合的情况下，前两个任务(对应于前四位数字)的准确率下降到偶然水平，而有整合的模型在这些任务上只显示出轻微的性能下降(图3)。

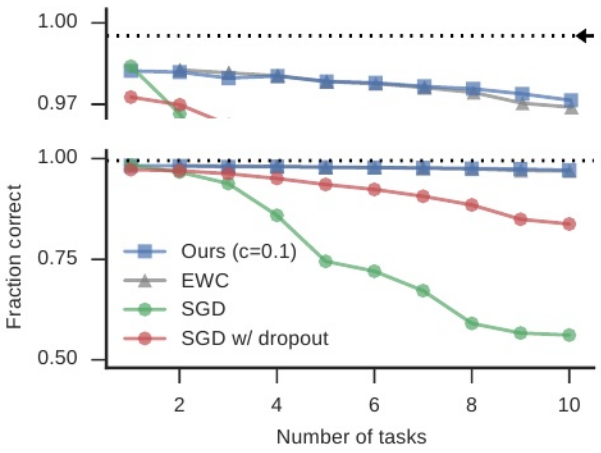

图4。从排列MNIST基准中学习到的所有任务的平均分类精度作为任务数量的函数。随着任务数量的增加，我们的方法(蓝色)和EWC(灰色，从Kirkpatrick等人(2017)提取并重新绘制)保持了很高的准确性。SGD(绿色)和隐藏层dropout为0.5的SGD(红色)的表现要差得多。最上面的面板是放大的图的上部，具有在单个任务上的初始训练精度(虚线)和同一网络在同时对所有任务进行训练时的训练精度(黑色箭头)。

5.2. 排列MNIST基准

在这个基准测试中，我们对每个任务随机地对所有MNIST像素进行不同的排列。我们训练了一个具有两个隐藏层的MLP，每个隐藏层有2000 ReLUs和softmax loss。我们使用了亚当，参数和之前一样。然而，这里我们使用$\xi = 0.1$, c = 0.1的值是通过在保留验证集上进行粗网格搜索确定的。迷你批量大小被设置为256，我们训练了20个epoch。与拆分MNIST基准测试相比，我们通过在任务之间维护亚当优化器的状态获得了更好的结果。最终的测试误差是根据MNIST测试集的数据计算的。性能由网络解决所有任务的能力来衡量。

为了建立一个比较的基线，我们首先在所有任务上按顺序训练了一个没有突触巩固(c = 0)的网络。在这种情况下，系统表现出灾难性遗忘，即它学会解决最近的任务，但是

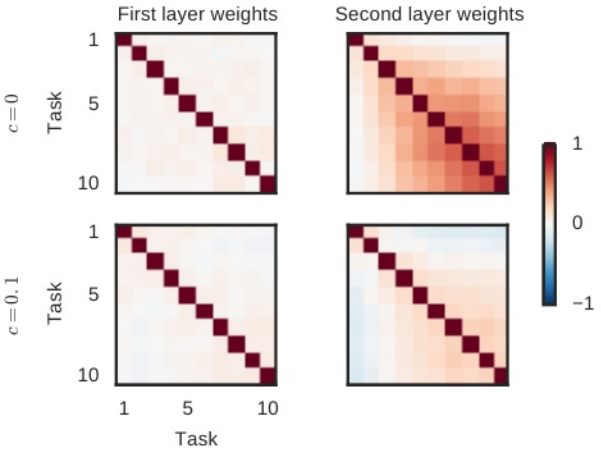

图5。在排列MNIST上，每个任务的权重重要度的相关矩阵，ωkμ。对于正常的微调(c = 0，顶部)和整合(c = 0.1，底部)，第一层权重重要性(左)在任务之间是不相关的，因为排列的MNIST数据集在输入层是不相关的。然而，随着通过微调学习到更多的任务，第二层重要性(右)变得更加与corr相关。相比之下，巩固阻止了ωk μ中的强相关性，这与用于解决新任务的不同权重的概念相一致。

快速忘记之前的任务(蓝线，图4)。与此相反，当启用突触巩固，对c > 0进行合理选择时，同一网络在任务1上保持高分类精度，同时在9个额外任务上进行训练(图4)。此外，网络学会以高精度解决所有其他任务，性能仅略差于同时在所有数据上训练的网络(图4)。这些结果在训练和验证误差中是一致的，与EWC报告的结果相当(Kirkpatrick等人，2017)。

为了更好地理解训练期间的突触动力学，我们可视化了$\omega_k\mu$across不同任务μ的成对相关性(图。5 b)。我们发现，如果没有整合，第二层隐藏层中的ωk μ在任务之间是相关的，这很可能是灾难性遗忘的原因。然而，有了巩固，这些有助于减少损失的突触集在任务之间基本上是不相关的，从而避免在更新权重以解决新任务时的干扰。

5.3. 分割CIFAR-10/CIFAR-100基准

为了评估突触巩固动力学是否也能在更复杂的数据集和更大的模型中防止灾难性遗忘，我们实验了一个基于CIFAR-10和CIFAR-100的持续学习任务。具体来说，我们训练了一个CNN(4卷积，其次是2个密集层的dropout;详见附录)。我们使用了与Adam拆分MNIST相同的多头设置($\eta = 1\times10^{-3}$，$\beta_1 = 0.9$，$\beta_2 = 0.999$,minibatch大小256)。首先，我们在完整的CIFAR-10数据集(任务1)上训练网络60个epoch，并依次在5个额外任务上训练网络，每个任务对应CIFAR-100数据集的10个连续类(图6)。为了确定最佳c，我们对参数范围$1\times10^{-3} < c < 0.1$的不同值进行了该实验。任务之间，优化器的状态被重新设置。此外，我们获得了两个特定控制案例的值。一方面，我们在所有任务上连续训练了相同的网络，c = 0。另一方面，我们在每个任务上单独从头训练相同的网络，以评估跨任务的泛化能力。最后，为了评估准确性统计波动的幅度，所有的运行都被重复n = 5次。

我们发现，在对所有任务进行训练后，经过巩固的网络在所有任务中显示出类似的验证精度，而在没有进行巩固的网络中，准确性显示出明显的年龄依赖性下降，旧任务的解决精度较低(图6)。重要的是，经过巩固训练的网络的性能总是比没有进行巩固的网络更好，除了最后一个任务。最后，当用net-比较经过巩固训练的网络在所有任务上的性能时

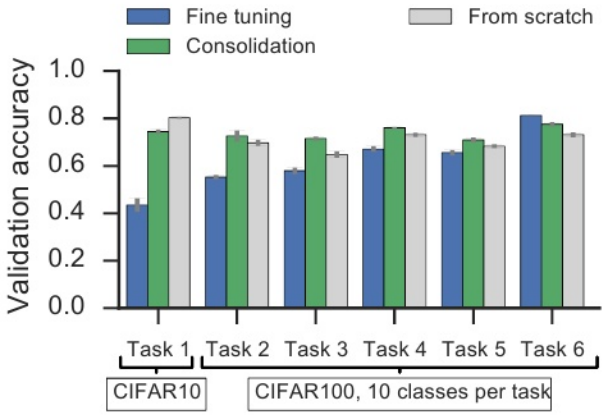

图6。分割CIFAR-10/100基准上的验证精度。蓝色:验证误差，没有合并(c = 0)绿色:验证误差，有合并(c = 0.1)。灰色:没有合并的网络，只在单一任务上从头训练。这个基准中的机会级是0.1。误差条对应SD (n=5)。

只在单一任务上从零开始训练的作品(图6;绿色vs灰色)，前者要么显著优于后者，要么产生了相同的验证精度，而这种趋势在训练精度上发生了逆转。这表明，没有整合的网络更容易过度拟合。该规则的唯一例外是任务1 CIFAR-10，这可能是由于每个类的示例数量增加了10倍。综上所述，我们发现巩固不仅保护旧记忆不被随着时间的推移慢慢遗忘，而且还允许网络在有限数据的新任务上更好地泛化。

6. 讨论

我们已经证明，通过允许单个突触估计它们对解决过去任务的重要性，可以缓解持续学习场景中常见的灾难性遗忘问题。然后通过惩罚最重要突触的变化，可以在对先前学习任务干扰最小的情况下学习新的任务。

正则化惩罚类似于Kirkpatrick等人(2017)最近引入的EWC。然而，我们的方法以在线方式计算每个突触的巩固强度，并在参数空间中计算整个学习轨迹，而对于EWC，突触重要性是离线计算的，作为指定任务损失最小的Fisher信息。尽管存在这种差异，但这两种方法在排列MNIST基准上产生了相似的性能，这可能是由于两种不同重要性度量之间的相关性。

该方法要求单个突触不简单地对应于单个标量突触权重，而是作为自己的高维动力系统。这样的高维状态使我们的每个突触能够在训练过程中智能地积累任务相关信息，并保留对先前参数值的记忆。虽然我们没有声称生物突触的行为与我们模型中的智能突触类似，但神经生物学中丰富的实验数据表明，生物突触的行为方式比主导当前机器学习模型的人工标量突触要复杂得多。从本质上讲，突触变化是否发生，以及它们是否被做成永久性的，还是留给最终的衰变，可以由许多不同的生物因素控制。例如，突触可塑性的诱导可能取决于单个突触的历史和突触状态(Montgomery & Madison, 2002)。此外，最近的突触变化可能会以小时为时间尺度衰减，除非释放特定的可塑性相关化学因子。这些化学因子被认为编码了近期变化的效价或新奇性(Redondo & Morris, 2011)。最后，最近的突触变化可以被刻板的神经活动重置，而较早的突触记忆对逆转越来越不敏感(Zhou等人，2003)。

在这里，我们引入了一个特定的高维突触模型来解决一个特定的问题:持续学习中的灾难性遗忘。然而，这表明了新的研究方向，我们在其中镜像神经生物学，赋予个体突触潜在的复杂动力学特性，可以利用这些特性智能控制神经网络中的学习动力学。从本质上讲，在机器学习中，除了增加我们网络的深度，我们可能还需要给我们的突触增加智能。

致谢。

作者感谢Subhaneil Lahiri的有益讨论。FZ得到了瑞士国家科学基金会和惠康基金会的支持。BP得到了斯坦福大学MBC IGERT奖学金和斯坦福大学跨学科研究生奖学金的支持。SG得到了Burroughs惠康、麦克奈特、西蒙斯和詹姆斯·s·麦克唐纳基金会以及海军研究办公室的支持。

参考文献。

Benna, Marcus K. and Fusi, Stefano. Computational prin-ciples of synaptic memory consolidation. *Nat Neurosci*, advance online publication, October 2016. ISSN 1097-6256. doi: 10.1038/nn.4401.

Choy, Min Chee, Srinivasan, Dipti, and Cheu, Ruey Long. Neural networks for continuous online learning and control. *IEEE Trans Neural Netw*, 17(6):1511–1531, November 2006. ISSN 1045-9227. doi: 10.1109/TNN.2006.881710.

Donahue, Jeff, Jia, Yangqing, Vinyals, Oriol, Hoffman, Judy, Zhang, Ning, Tzeng, Eric, and Darrell, Trevor. De-caf: A deep convolutional activation feature for generic visual recognition. In *International Conference in Ma-chine Learning (ICML)*, 2014.

Fukushima, Kunihiko and Miyake, Sei. Neocognitron: A Self-Organizing Neural Network Model for a Mech-anism of Visual Pattern Recognition. In *Competition and Cooperation in Neural Nets*, pp. 267–285. Springer, Berlin, Heidelberg, 1982. DOI: 10.1007/978-3-642-46466-9 18.

Fusi, Stefano, Drew, Patrick J., and Abbott, Larry F. Cas-cade models of synaptically stored memories. *Neuron*, 45(4):599–611, February 2005. ISSN 0896-6273. doi: 10.1016/j.neuron.2005.02.001.

Goodfellow, Ian J., Mirza, Mehdi, Xiao, Da, Courville, Aaron, and Bengio, Yoshua. An Empirical Investiga-tion of Catastrophic Forgetting in Gradient-Based Neu-ral Networks. *arXiv:1312.6211 [cs, stat]*, December 2013. arXiv: 1312.6211.

Hinton, Geoffrey, Vinyals, Oriol, and Dean, Jeff. Distilling the knowledge in a neural network. *NIPS Deep Learning and Representation Learning Workshop*, 2014.

Jung, Heechul, Ju, Jeongwoo, Jung, Minju, and Kim, Junmo. Less-forgetting Learning in Deep Neural Net-works. *arXiv:1607.00122 [cs]*, July 2016. arXiv: 1607.00122.

Kingma, Diederik and Ba, Jimmy. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*, Decem-ber 2014. arXiv: 1412.6980.

Kirkpatrick, James, Pascanu, Razvan, Rabinowitz, Neil, Veness, Joel, Desjardins, Guillaume, Rusu, Andrei A., Milan, Kieran, Quan, John, Ramalho, Tiago, Grabska-Barwinska, Agnieszka, Hassabis, Demis, Clopath, Clau-dia, Kumaran, Dharshan, and Hadsell, Raia. Overcom-ing catastrophic forgetting in neural networks. *PNAS*, pp. 201611835, March 2017. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.1611835114.

Krizhevsky, Alex and Hinton, Geoffrey. Learning multiple layers of features from tiny images. 2009.

Lahiri, Subhaneil and Ganguli, Surya. A memory frontier for complex synapses. In *Advances in Neural Infor-mation Processing Systems*, volume 26, pp. 1034–1042, Tahoe, USA, 2013. Curran Associates, Inc.

LeCun, Yann, Cortes, Corinna, and Burges, Christopher JC. *The MNIST database of handwritten digits*. 1998.

LeCun, Yann, Bengio, Yoshua, and Hinton, Geoffrey. Deep learning. *Nature*, 521(7553):436–444, May 2015. ISSN 0028-0836. doi: 10.1038/nature14539.

Li, Zhizhong and Hoiem, Derek. Learning without forget-ting. In *European Conference on Computer Vision*, pp. 614–629. Springer, 2016.

Martens, James. *Second-order optimization for neural net-works*. PhD thesis, University of Toronto, 2016.

Montgomery, Johanna M. and Madison, Daniel V. State-Dependent Heterogeneity in Synaptic Depression be-tween Pyramidal Cell Pairs. *Neuron*, 33(5):765–777, February 2002. ISSN 0896-6273. doi: 10.1016/ S0896-6273(02)00606-2.

Pascanu, Razvan and Bengio, Yoshua. Revisiting natural gradient for deep networks. *arXiv preprint arXiv:1301.3584*, 2013.

Razavian, Ali Sharif, Azizpour, Hossein, Sullivan, Josephine, and Carlsson, Stefan. Cnn features off-the-shelf: an astounding baseline for recognition. In *Pro-ceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 806–813, 2014.

Redondo, Roger L. and Morris, Richard G. M. Making memories last: the synaptic tagging and capture hypoth-esis. *Nat Rev Neurosci*, 12(1):17–30, January 2011. ISSN 1471-003X. doi: 10.1038/nrn2963.

Rosenblatt, Frank. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

Rusu, Andrei A., Rabinowitz, Neil C., Desjardins, Guillaume, Soyer, Hubert, Kirkpatrick, James, Kavukcuoglu, Koray, Pascanu, Razvan, and Hadsell, Raia. Progressive Neural Networks. *arXiv: 1606.04671 [cs]*, June 2016. arXiv: 1606.04671.

Srivastava, Rupesh K, Masci, Jonathan, Kazerounian, Sohrob, Gomez, Faustino, and Schmidhuber, Juergen. Compete to Compute. In Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Sys-tems 26*, pp. 2310–2318. Curran Associates, Inc., 2013.

Yosinski, Jason, Clune, Jeff, Bengio, Yoshua, and Lipson, Hod. How transferable are features in deep neural net-works? In *Advances in neural information processing systems*, pp. 3320–3328, 2014.

Zenke, Friedemann, Agnes, Everton J., and Gerstner, Wul-fram. Diverse synaptic plasticity mechanisms orches-trated to form and retrieve memories in spiking neu-ral networks. *Nat Commun*, 6, April 2015. doi: doi: 10.1038/ncomms7922.

Zhou, Qiang, Tao, Huizhong W., and Poo, Mu-Ming. Re-versal and Stabilization of Synaptic Modifications in a Developing Visual System. *Science*, 300(5627):1953– 1957, June 2003. doi: 10.1126/science.1082212.

Ziegler, Lorric, Zenke, Friedemann, Kastner, David B., and Gerstner, Wulfram. Synaptic Consolidation: From Synapses to Behavioral Modeling. *J Neurosci*, 35(3): 1319–1334, January 2015. ISSN 0270-6474, 1529-2401. doi: 10.1523/JNEUROSCI.3989-14.2015.