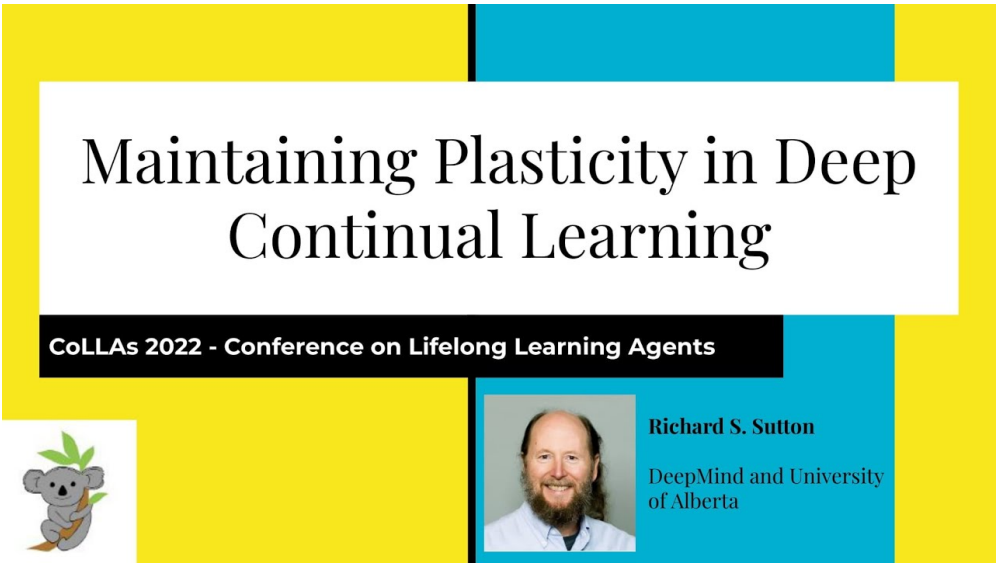
课程笔记

深度持续学习中的可塑性维持
Continual Backpropagation

cnfjlhj & DeepSeek Harness

2026 年 8 月 26 日



视频作者/频道：CoLLAs 2022 | Rich Sutton & Shibhanjan Dohare
发布日期：2022-09-05
视频时长：约 71 分钟
视频链接：`https://www.youtube.com/watch?v=p_zknyfV9fY`

目录

1　引言：持续学习与可塑性危机

1.1　演讲者与主题

这场报告来自 2022 年首届**终身学习智能体会议**（Conference on Lifelong Learning Agents, CoLLAs 2022）的主题演讲。主讲人是强化学习领域的奠基者之一 **Rich Sutton** 教授（阿尔伯塔大学、DeepMind 阿尔伯塔联合实验室），他与博士生 **Shibhanjan Dohare**（Rupam Mahmood 指导）共同呈现了关于**深度持续学习中维持可塑性**的工作。

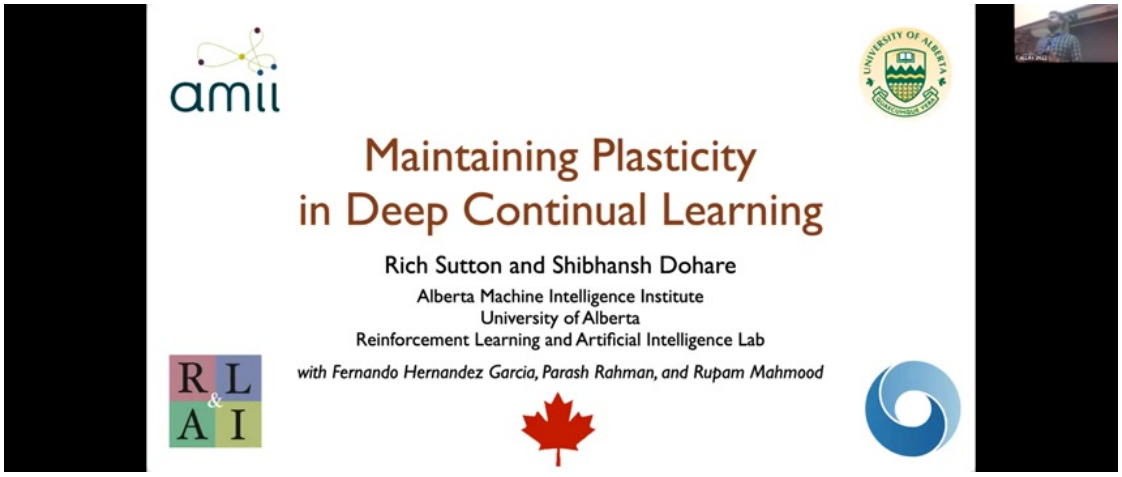


图 1: 报告开场标题页：Maintaining Plasticity in Deep Continual Learning。[1]

Sutton 在开场便定调：尽管大众将他等同于强化学习，但这场报告几乎不谈强化学习，而是聚焦于**监督学习**中的持续学习问题。他的理由很朴素——持续学习的困难在监督设定与强化设定中都存在，那么就应当先在最简单的设定下把它研究清楚。

1.2　两条核心命题

整场报告围绕两条相互呼应的命题展开：

报告的两条核心命题

命题一（诊断）：深度学习其实并不能真正胜任持续学习。随着新任务不断到来，学习速度会持续变慢，最终退化到与非深度（线性）网络相当的水平。

命题二（处方）：改进并不困难。只要我们有意识地寻找为持续学习专门设计的方法，就能让深度学习在持续问题上重新发挥作用——关键在于走出“一次性学习”给我们挖的坑。

Sutton 特别划清了一条边界：他**不考虑回放缓冲区（replay buffer）**这一类机制。在他看来，回放本身就是一种“深度学习行不通”的变相承认，是一种不够优雅的极端手段。报告所追求的，是不依赖这类极端措施、真正面向持续学习的学习算法。

1.3　可塑性：定义与术语

在进入实验之前，Sutton 反复澄清核心术语，因为他认为这些词在文献中存在冗余与混用：

[1] 视频画面时间区间：00:00:00–00:01:00。

术语统一

可塑性（plasticity）：网络继续学习的能力。一旦失去可塑性，就意味着失去了学习新事物的能力，也就无法做持续学习。
可塑性丢失（loss of plasticity / capacity loss）：学习能力的衰退。Clara Lyle、Will Dabney 等人在 ICML 提出的“capacity loss”正是同一概念的不同命名。
维持可塑性（maintaining plasticity）：在长期、持续的学习过程中始终保留学习的能力。对于 CoLLAs 这个社区而言，这应当是被优先关注的目标。

值得注意的是，在此次工作之前，学界**尚无人**在监督学习设定下对“可塑性丢失”做过直接的、系统化的检验——这本身就令人意外，也正是本报告要补上的缺口。

1.4 历史脉络：从灾难性遗忘到可塑性丢失

Sutton 把问题放进更长的历史脉络中理解，让听众意识到“深度学习不能持续学习”并非新发现，而是多方面证据的汇聚：

- **灾难性遗忘（catastrophic forgetting）**：上世纪 90 年代就被认识到，新任务的学习会覆盖旧任务的知识。
- **可塑性丢失**：心理学与认知科学文献早有大量讨论，描述学习系统逐渐丧失学习能力的现象。
- **Warm-starting 不奏效**（Ash & Adams 的工作）：先用部分数据“预热”系统，本意是让后续学习更好，结果却适得其反——预热后在新数据上反而更差，不如干脆把最初学到的东西全丢掉、用全部数据一次性重新学。
- **深度强化学习中的 primacy bias**：有工作表明，与其保留网络已学到的一切，不如保留回放缓冲区、但把网络权重全部丢弃重学，反而能维持持续学习的能力。

一个反复出现的信号

无论是 warm-starting 还是“丢弃网络重学”，这些极端的“重置”手段本身，就是 continual deep learning 行不通的明证——如果系统本身能持续学习，我们何须诉诸如此激烈的措施？

1.5 本章小结

报告从一个大胆但证据充分的诊断出发：标准深度学习在持续学习设定下会丧失可塑性，退化为线性网络的水平。Sutton 把“可塑性”定义为持续学习的能力，并梳理了从灾难性遗忘到 capacity loss 的历史脉络，指出此前缺少监督学习下的直接系统检验。接下来，报告将用一系列精心设计的实验来证实这一诊断，并最终给出一个简单有效的处方。

2 如何检验“深度学习不能持续学习”

要论证“深度学习无法持续学习”，首先得设计一个能干净地暴露该现象的测试。Sutton 讨论了两种范式：

1. **单个非平稳任务**：用一个不断变化的学习问题来观察网络能否跟上。

2. **任务序列**：给出一串不同的任务，且不向学习者透露任务已切换的任何信号。

报告选择了第二种范式，并刻意把设定压到最简——**经典的有监督分类**，使用经典数据集 ImageNet 与 MNIST。Sutton 解释这一选择时颇为坦诚：深度学习本身已经足够复杂，再叠加强化学习的额外因素只会让我们看不清究竟发生了什么，因此要把变量理清楚。后续还会引入一个更小的“缓慢变化回归”问题，用于做大量系统化实验。

为什么刻意不给任务切换信号？

报告强调，任务之间没有任何指示告知学习者“新任务开始了”。这是出于对真实持续学习场景的忠实：一个生活在世界中的智能体，并不会收到某种特殊的“概念已漂移”通知，它面对的只是源源不断、缓慢或突变的数据流。算法必须能在不依赖任务边界信息的前提下持续适应。

2.1 本章小结

检验设计遵循“极简而忠实”的原则：用经典分类、经典数据集、任务序列范式，且不提供任何任务切换信号。目的是把复杂度降到能看懂的程度，同时保留持续学习的本质——面对连续不断、无明确边界的数据流。

3 ImageNet 持续学习实验

3.1 数据集与任务构造

ImageNet 是深度学习中最经典的图像分类数据集：数百万张图像、按名词标注、标准划分下每类约 700 张图、共 1000 个类别。报告使用的是一个**降采样到** 32×32 的标准缩减版（Sutton 屏示了一张被降采样的鲨鱼图作为示例）。

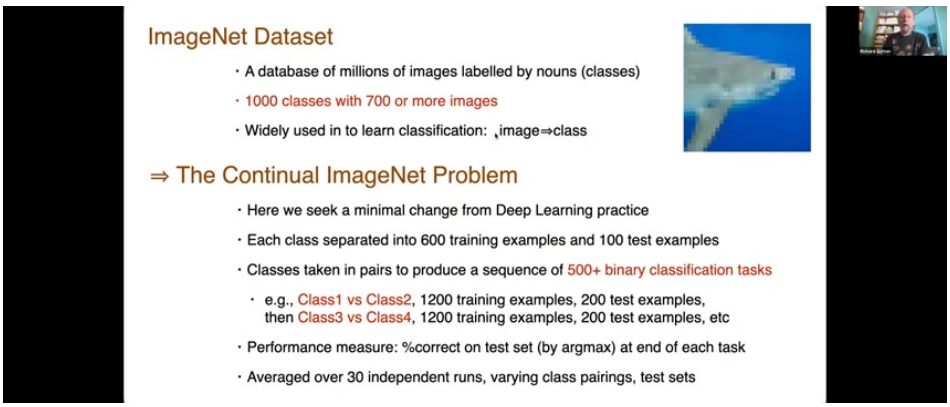


图 2: ImageNet 数据集介绍与降采样示例图。[2]

任务构造的思路是“尽量贴近标准深度学习实践、只做最小的持续化改动”：

1. 1000 个类别每类 700 张图，划分为 **600 张训练 + 100 张测试**。
2. 将类别两两配对，得到 **500 个二分类任务**（例如“类别 1 vs 类别 2”，接着“类别 3 vs 类别 4”……，顺序跨运行随机打乱）。

[2] 视频画面时间区间：00:08:39–00:09:00。

3. 每个二分类任务有 $2 \times 600 = 1200$ 个训练样本、$2 \times 100 = 200$ 个测试样本。

4. 在每个任务上训练后，用 200 个测试样本以 argmax 计算正确率，随即进入下一个任务，不告知任务已切换。

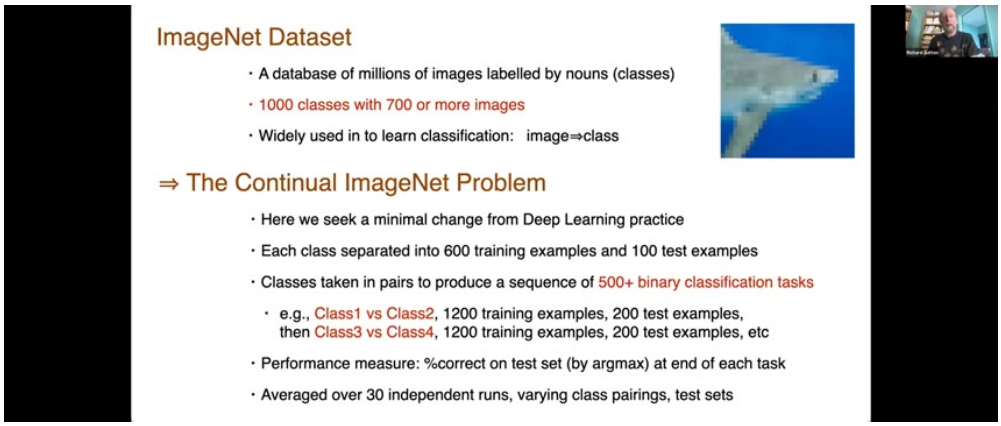


图 3: ImageNet 二分类任务序列的实验设置。[3]

3.2 网络结构与训练设置

所有 500 个任务共用**同一个网络**，权重只在第一个任务之前初始化一次，此后不再重新初始化。网络结构为：

- **3 个卷积层**（filters 宽度为 128）；
- **3 个全连接层**；
- 末端 **2 个输出的分类头**（因为是二分类）。

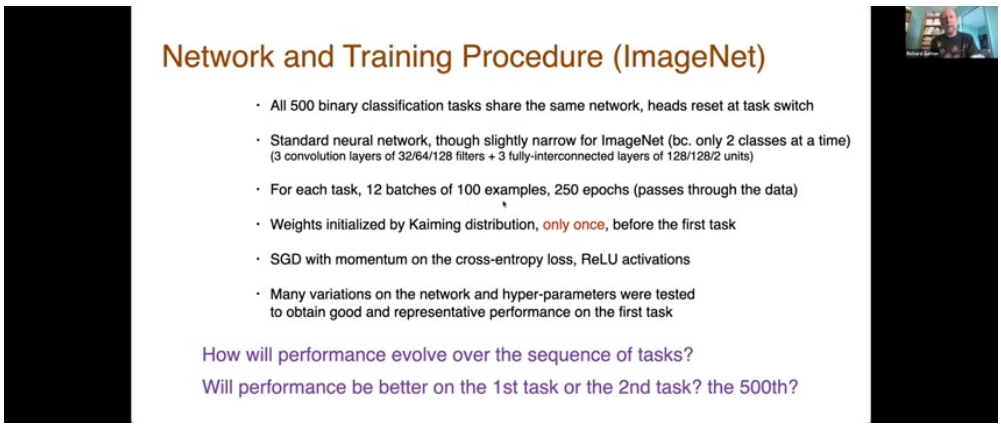


图 4: 网络结构：3 个卷积层 + 3 个全连接层 + 二分类头。[4]

训练细节：每个任务内构造 12 个 batch（每批 100 样本），做 **250 个 epoch**；优化器为带动量的 SGD；损失为交叉熵；激活函数为 ReLU。一个值得注意的处理是：**在任务切换时会重置分类头**——这是为了去掉“头部的干扰”、把分析焦点放在表征身上；后续实验中会取消这一处理。

[3] 视频画面时间区间：00:09:40–00:10:10。

[4] 视频画面时间区间：00:11:12–00:12:00。

3.3 初始性能：短期获益的假象

Sutton 先抛出一个引导性问题让听众思考：第一个任务、第二个任务、第五百个任务，性能会怎样演变？第二个任务会因前一个任务的经验而受益吗？

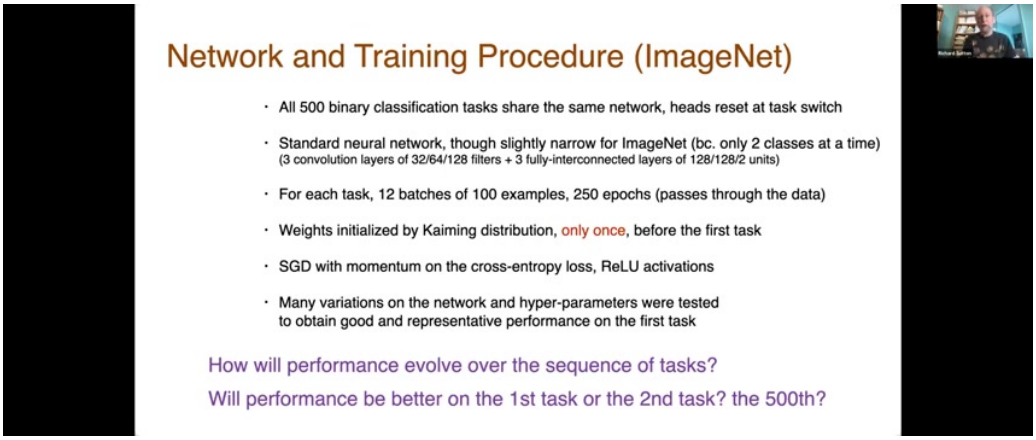


图 5: 初始若干任务的测试正确率（按步长分组）。[5]

答案是“因步长而异，好坏掺半”：

- 步长 0.01 时，第一个任务就能达到约 **89%** 正确率（随机水平为 50%）。
- 在更小的步长下，反而出现了**正向迁移**：第二个任务比第一个好，第三个更好，大约在第五个任务处达到顶峰。

Sutton 顺势教读者如何读图：横轴是任务编号，纵轴是每个任务结束时的测试正确率，阴影区域为一个标准差，并给出一条**线性基线**——即直接从像素用一个线性头分类的性能。他强调：深度网络必须显著超过这条线性基线，否则就不能算在“做深度学习”。

不要被短期获益迷惑

初始几个任务上的正向迁移只是表象。报告真正关心的是长期趋势——而长期趋势会给出截然不同的结论。

3.4 长期趋势：可塑性的灾难性丢失

把时间轴拉到 **1–2000 个任务**（为降低方差，每 50 个任务取一个数据点），结论变得清晰而残酷：

[5] 视频画面时间区间：00:12:00–00:13:00。

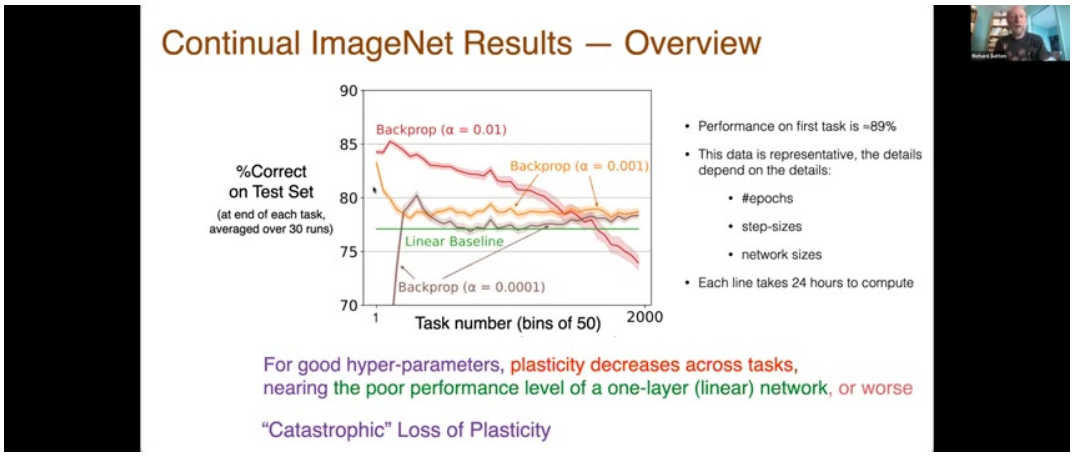


图 6: 长期性能曲线：可塑性随任务数持续下降，甚至跌破线性网络水平。[6]

ImageNet 上的核心发现

即便使用调好的超参数，网络的可塑性仍**随任务数持续下降**，最终**逼近甚至跌破单层线性网络的水平**。换成更慢的步长也无济于事——慢步长同样剧烈地丢失可塑性，只是略好于线性网络。Sutton 将红线（原本表现最好的那条）称为“可塑性的灾难性丢失”。

一个现实约束是：ImageNet 上的每条曲线需要**多处理器上约 24 小时**的人力/算力才能跑出，难以做系统的超参数扫描。因此报告随后转向更小、更适合系统化实验的 MNIST。

3.5 本章小结

ImageNet 实验证实了核心诊断：标准深度网络在长任务序列中持续丢失可塑性，退化到线性网络水平乃至更差，且调慢学习率也无法挽回。短期可见的正向迁移只是假象，长期趋势才是真相。受限于算力成本，报告转用 MNIST 做更彻底的系统化研究。

4 MNIST 置换实验

4.1 置换任务构造

MNIST 是 60000 张手写数字灰度图（28×28 像素、10 个类别）。为了让单一数据集变成任意长的任务序列，报告采用了一个经典技巧——**像素置换**（permutation）：对像素位置做一次随机打乱，就得到一个“新任务”。每一个置换对应一个不同的任务，由此可以生成任意数量的任务。

[6] 视频画面时间区间：00:14:49–00:15:49。

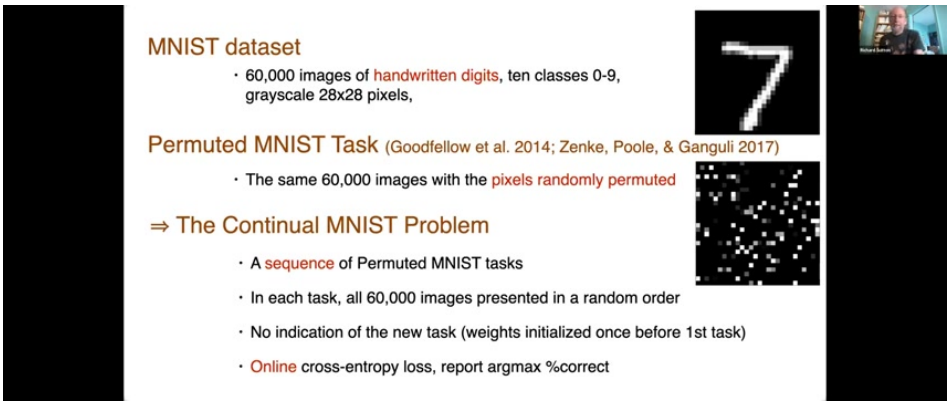


图 7: MNIST 置换任务：打乱像素位置生成新任务。[7]

关键设定：每个任务上处理全部 60000 张置换图，**不告知**任务已切换；权重只在第一个任务前初始化一次。这一技巧源自 Goodfellow 等人的工作，在持续学习社区已被广泛使用。

4.2 网络与在线性能度量

网络是一个“相当标准”的全连接网络：**4 个全连接层**，主体隐藏层宽度 **2000**，末端 10 个输出单元。MNIST 置换任务中一般不用卷积（图像已居中归一化，卷积带来的增益有限）。所有任务共用同一网络，用普通梯度下降（此问题上动量帮助不大）、ReLU 激活。

一个重要转变是**性能度量**：报告切换到对持续学习更自然的**在线性能**——即**在线交叉熵损失**，逐样本评估，而非“训练完再测”。

4.3 鳄鱼图：任务切换的尖刺

在线性能曲线呈现出一种标志性的形状：每个任务开始时，网络面对全新的置换、只能以 1/10 的概率随机猜测（性能跌到 10%），随后逐渐学会当前置换、性能爬升；任务一切换，性能又骤跌回 10%。这种“上升—骤降—上升”的锯齿被 Sutton 戏称为 **“鳄鱼图”（alligator graph）**——像鳄鱼尾巴上的一串尖刺。

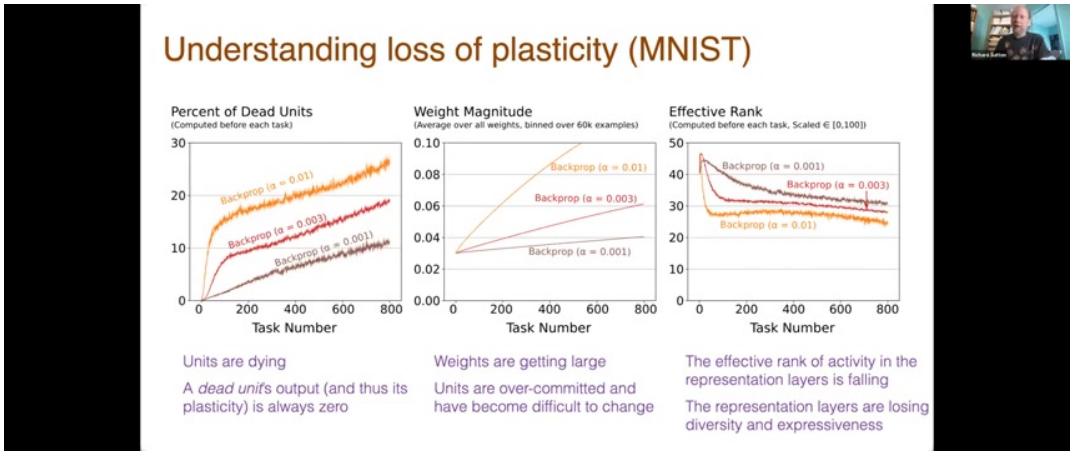


图 8: 在线性能的“鳄鱼图”：每次任务切换都带来一次性能骤降与重新爬升。[8]

[7] 视频画面时间区间：00:17:06–00:18:23。

[8] 视频画面时间区间：00:25:20–00:26:20。

为看清长期趋势，报告把每个“尖刺”压缩成一个数（尖刺上的平均性能），从而得到更平滑的长期曲线：早期任务性能确有提升，随后趋于平稳，但到第 **800 个任务**左右已经显著退化。

4.4 长期性能与网络规模

把各步长下的长期在线性能画出来后，结论与 ImageNet 一致：

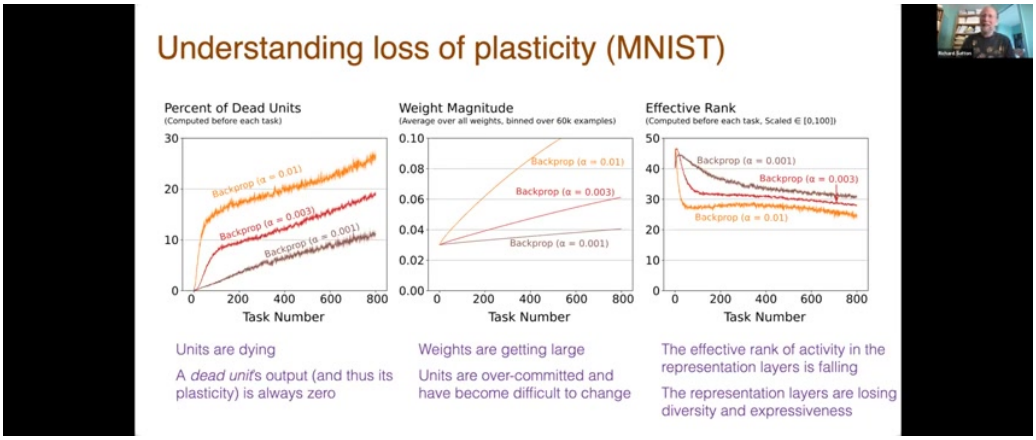


图 9: MNIST 长期在线性能：各步长下可塑性均持续下降。[9]

- 之前表现最好的红线，长期来看**持续走低**，到 800 个任务仍在下降。
- **更快的步长**：上得快、丢得也快；**更慢的步长**：轨迹相同，只是整体更慢。
- 所有步长下都存在**实质性的可塑性丢失**。

网络规模的影响同样值得记取：把隐藏层从 2000 扩到 10000，整体变慢、可塑性丢失更少（更大的网络“家底更厚”，剩得更多），但**仍在丢失**；小网络则急剧衰落。

规模只是缓兵之计

更大的网络能延缓可塑性丢失，却无法阻止它。这提示问题出在学习算法本身，而非网络容量。

4.5 本章小结

MNIST 置换实验以更低的算力成本复现并深化了 ImageNet 的结论：无论步长快慢、网络大小，标准反向传播都会持续丢失可塑性。在线性能的“鳄鱼图”直观展示了任务切换带来的反复崩塌与重建，而长期曲线证明这种退化是不可逆的系统性行为。

5 透视网络内部：可塑性丢失的机理

要理解“为什么”丢失可塑性，Sutton 带领听众钻进网络内部，追踪三个可观测的量随任务数的变化。这一节是整场报告从“现象”走向“机理”的关键过渡。

[9] 视频画面时间区间：00:23:50–00:24:30。

5.1 死亡单元（Dead Units）

第一个也是最直观的指标是**死亡单元**：如果一个单元的输出始终为零、且其梯度也始终为零，它就不再参与网络的任何可塑性——等于被“冻死”了。形式上，对单元 i，若在所有后续步骤中

$$a_i = 0 \quad \text{且} \quad \frac{\partial \mathcal{L}}{\partial a_i} = 0,$$

则该单元死亡。

- 单元符号说明：a_i 为单元 i 的输出激活；$\mathcal{L}$ 为损失；$\partial \mathcal{L}/\partial a_i$ 为损失对该单元输出的梯度。

在标准反向传播下，死亡单元的比例随任务数**稳步上升，最高约达 20%**，且随步长变化。这强烈暗示：网络正在逐渐丧失可用的计算单元。

5.2 权重幅度与过度承诺

第二个指标是**权重幅度**：随任务推进，权重的绝对值**持续增大**。其后果是隐藏单元变得**过度承诺（over-committed）**——已被既有任务“焊死”，难以再改变、难以再为后续任务生成有用的新特征。

5.3 有效秩与多样性

第三个指标是**剩余隐藏层的有效秩（effective rank）**，即这些层变化的线性独立成分数，衡量表征的**多样性**。反向传播下，有效秩从约 40 先升后降，最终回落到约 30——下降幅度虽不剧烈，但方向一致：多样性在流失。

三个指标，一个故事

死亡单元增多、权重过度膨胀、有效秩下降——三个从不同角度切入的指标共同指向同一机理：网络的可学习资源（活着的、可改变的、多样的单元）正在被持续学习过程逐步耗尽。这正是后续所有方法与诊断的对照基准。

5.4 本章小结

通过追踪死亡单元比例、权重幅度与有效秩，报告把“可塑性丢失”从黑箱现象解释为可观测的内部退化：网络中可用、可变、多样的单元在持续学习中不断减少。这三个指标也成为后续评估各种方法时的统一“体检项目”。

6 现有方法能否维持可塑性？

在确认诊断后，报告检验了一批来自深度学习文献、被认为可能缓解退化的方法，并在 MNIST 上用同样的三个内部指标做体检。

6.1 L2 正则化与 Shrink-and-Perturb

L2 正则化（权重衰减）的实质是持续地把所有权重向零收缩：

$$w \leftarrow w - \eta\big(\nabla \mathcal{L} + \lambda w\big),$$

- w：权重；η：学习率；$\nabla\mathcal{L}$：损失梯度；λ：正则强度。

直觉上，持续收缩能让权重保持“可改变”、不至于过度承诺，因此 L2 丢失可塑性的速度比反向传播慢，但仍在丢失。

Shrink-and-Perturb 在 L2 的基础上，每步再给权重注入随机噪声，以提升多样性：

$$w \leftarrow w - \eta\big(\nabla\mathcal{L} + \lambda\, w\big) + \epsilon, \qquad \epsilon \sim \mathcal{N}(0, \sigma^2).$$

- ϵ：每步注入的高斯噪声；σ：噪声强度。

这一方法表现相当好，几乎不丢失可塑性。

6.2 在线归一化与 Dropout

在线归一化类似在线版批归一化，持续对信号做平移与缩放。它在初期效果不错，但在持续学习下灾难性地丢失可塑性。**Dropout**（训练时随机置零一部分单元、迫使网络发展冗余特征）曾被提议为可塑性丢失的解药，但在这里表现比标准反向传播更差。

6.3 方法的内部诊断

用三个体检指标审视这些方法，能看清它们各自的得失：

- **L2**：权重幅度变小，但**容易杀死单元**（收缩到零 ⇒ 输出为零 ⇒ 死亡），故死亡单元偏多。
- **Shrink-and-Perturb**：既收缩又扰动，死亡单元较少，综合表现最好。
- **在线归一化**、**Dropout**：在体重幅度上方向走反（权重反而变大）；有效秩方面，L2 与 Shrink-and-Perturb 在维持多样性上也不够理想。

“对症”不等于“治本”

这些方法各自只缓解了部分症状：L2 控制了权重却牺牲了单元存活率；Shrink-and-Perturb 平衡较好但多样性仍不足。没有任何一个能同时治好死亡单元、权重膨胀、多样性流失三方面问题——这正是 Continual Backprop 登场的动机。

6.4 本章小结

既有方法各有得失：L2 抑制权重却易制造死亡单元，Shrink-and-Perturb 较均衡但多样性仍不足，在线归一化与 Dropout 甚至帮倒忙。统一的内部体检表明，没有一种现成方法能同时解决可塑性丢失的三方面症结。

7 缓慢变化回归：理想化实验平台

7.1 问题构造

为了能做大量、系统化的实验（扫描步长、激活函数、算法及其组合），报告引入了一个更小的理想化问题——**缓慢变化回归**（slowly changing regression）。它的设计直指持续学习的本质。

目标函数由一个单隐藏层网络生成，隐藏单元为**线性阈值单元（LTU）**、权重随机取 ± 1：

$$f(x) = \sum_{k=1}^{100} c_k\, \mathbf{1}\big[w_k^\top x > b_k\big], \qquad c_k,\, w_k \in \{-1, +1\} \text{ 随机.}$$

- $f(x)$：目标输出（实数，故为回归问题，用平方误差衡量）；x：二值输入；c_k：输出层权重；w_k, b_k：第 k 个 LTU 的权重与阈值；$\mathbf{1}[\cdot]$：指示函数（阈值阶跃）。

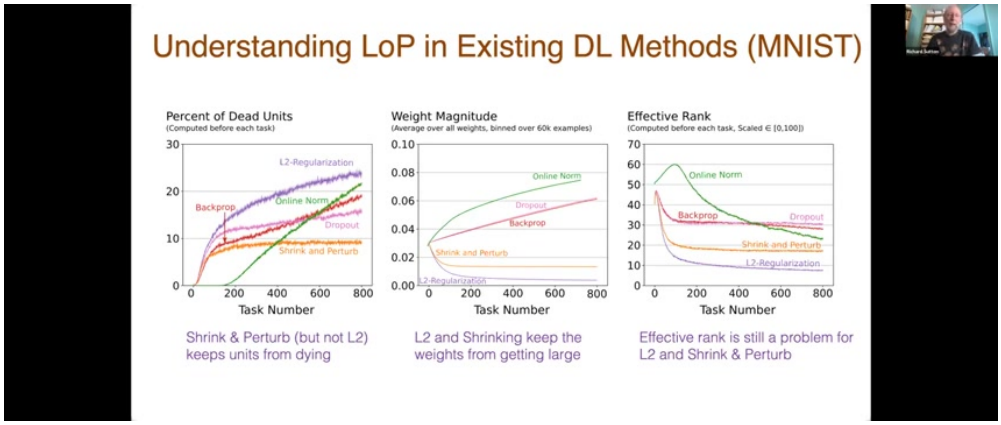


图 10: 缓慢变化回归问题的构造。[10]

输入为 **21 个二值位**，其中：1 个常数位（偏置）、5 个**独立同分布翻转**位（每个样本以 50/50 概率翻转）、其余 15 个**缓慢变化**位（每 10000 步随机挑一个翻转）。只要这 15 个位不变，目标函数就稳定；一旦某个位翻转，目标函数就随之缓慢改变——这便模拟了“缓慢的概念漂移”。

7.2 学习网络与逼近

学习网络（求解方）与目标函数结构相同（单隐藏层），但用可微激活，且**只有 5 个隐藏单元**——远小于生成目标的 100 个单元。这意味着智能体永远只能近似这个比它大的“世界”，并随着目标缓慢变化而跟踪它。

“大世界、小智能体”的设计哲学

求解网络远小于目标函数，呼应了 Sutton 后续在问答中反复强调的 **big world** 思想：智能体（心智）总是远小于真实世界，不可能在所有情形下都表现良好，只能在当下情境中做好局部跟踪。这一哲学也解释了为何报告不追求“记住一切”。

7.3 系统化实验与激活函数

在这个小问题上，团队得以系统扫描：不同步长、不同激活函数（tanh、sigmoid、ReLU）、不同算法乃至其 Adam 版本。

[10] 视频画面时间区间：00:28:50–00:29:30。

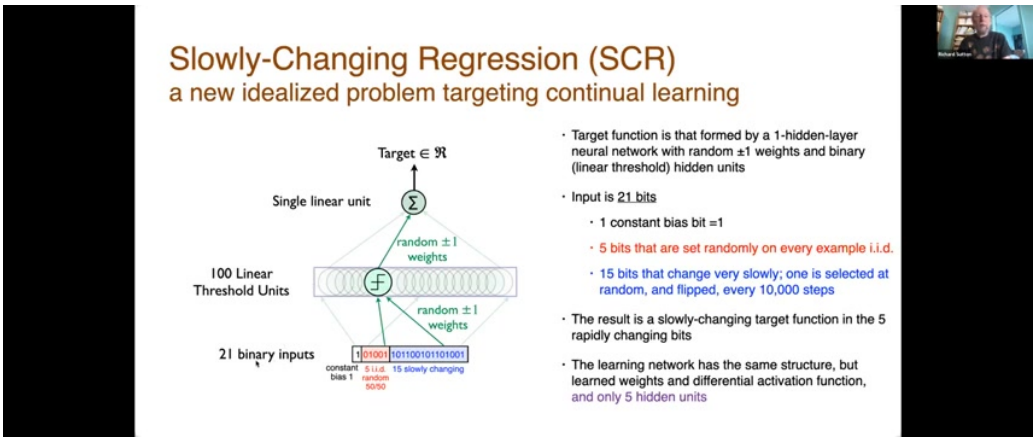


图 11: 缓慢变化回归问题的结果（与更大规模实验结论一致）。[11]

一个常被问到的疑问是：死亡神经元是否只是 ReLU 的固有缺陷、换个激活函数就能解决？报告在回归问题上专门检验了这一点，结论是否定的。

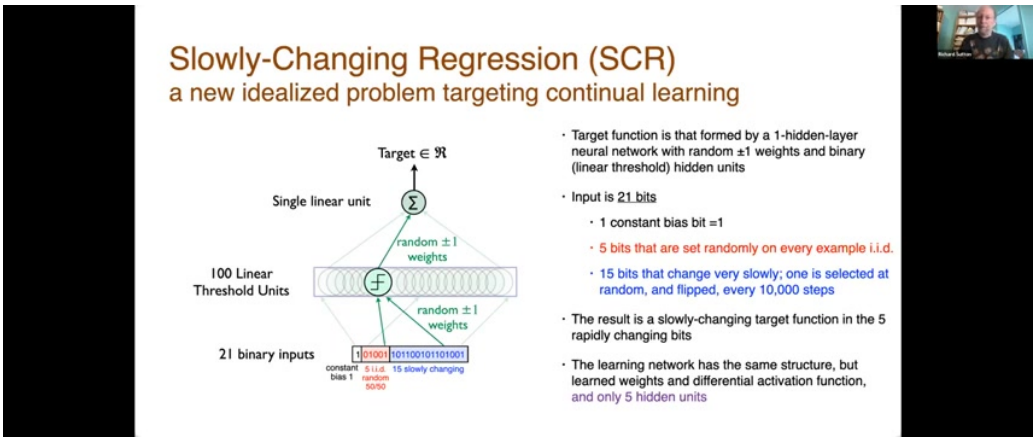


图 12: 激活函数与死亡神经元分析：更换激活函数并不能根除可塑性丢失。[12]

为什么换激活函数没用

所有常见激活函数都有**两个区域**——梯度大的区域与梯度小的区域。初始化时单元落在梯度大的区域（学得快），随着训练逐渐被推入梯度小的区域（学得慢）。只要激活函数存在这种“快/慢”两段结构，单纯更换激活函数就无法消除可塑性丢失。

7.4 本章小结

缓慢变化回归作为一个极小、可系统扫描的理想化平台，确认了更大规模实验的全部结论：标准深度学习在持续设定下丢失可塑性，且更换激活函数并不能根治。它也体现了“大世界、小智能体”的跟踪哲学，为后续算法的迭代提供了廉价而严格的试验场。

[11] 视频画面时间区间：00:32:00–00:33:00。
[12] 视频画面时间区间：00:33:40–00:34:40。

8 Continual Backprop：让反向传播持续学习

报告的后半段由 **Shibhanjan Dohare** 主讲，提出解决方案 **Continual Backprop**（持续反向传播）。

8.1 标准反向传播的持续学习缺陷

Dohare 先回顾标准反向传播的两个组成部分：**(1)** 用小随机数初始化权重（只做一次）；**(2)** 每步做梯度下降。他指出，这并非为持续学习而设计——初始化是一个只在开头发生的特殊计算，之后再不重复。一个真正面向持续学习的算法，理应在所有时刻都做类似的事情。

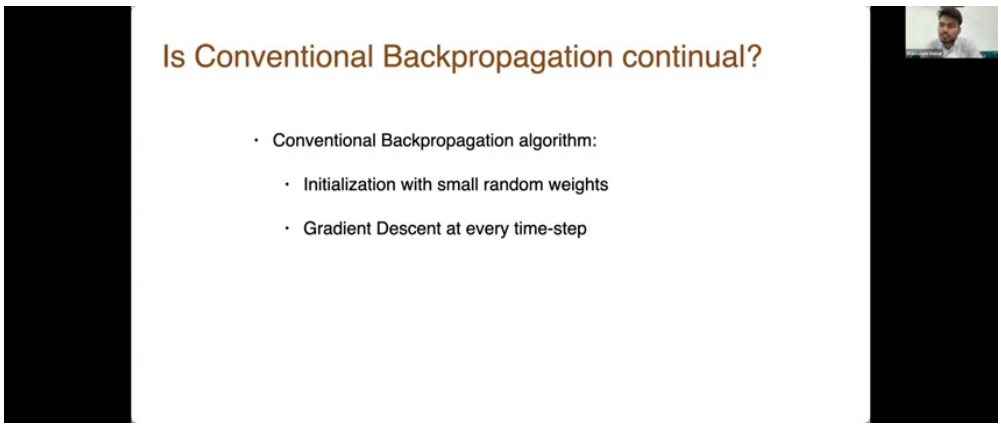


图 13: 从标准反向传播到持续反向传播的改进思路。[13]

8.2 核心思想：选择性再初始化

让反向传播“持续”起来的最简单方式，就是**偶尔重新初始化**。但整网重置会遗忘一切，在持续学习中不可取。于是 Continual Backprop 采取两步细化：

1. 只重置网络的一**部分**单元，而非整网；
2. 并且**有选择地**重置——找出最没有用的单元（例如死亡单元，或用更精细的“效用”度量），把效用最低的单元重新初始化为新的随机单元。

Continual Backprop 的一句话概括

在常规梯度下降之上，**持续地**用新的随机单元替换效用最低的旧单元——把“初始化”这件只在开头做一次的事，变成贯穿整个学习过程的常规操作。

8.3 从 Generate-and-Test 到多层推广

这一思路与 **Mahmood & Sutton** 早先提出的 **Generate-and-Test**（生成与测试）思想一脉相承：生成新单元、测试其效用、保留好的、淘汰差的。Continual Backprop 正是连接“持续学习”与“生成-测试”的桥梁。不过原始 Generate-and-Test 局限于**单隐藏层、单输出**（每个单元只有一个下游权重）。本工作将其推广到**多层、多输出（fan-out** > 1**）**的深度网络，并兼容 Adam 等现代优化器。

[13] 视频画面时间区间：00:37:22–00:38:00。

8.4 效用度量的四个要素

推广到多层时，“效用”如何定义是核心。Dohare 把它分解为四个直观要素：

1. **输出权重幅度之和**：原始 Generate-and-Test 中只有一个下游权重，效用即其幅度；多层时推广为该单元所有下游权重幅度之和 $\sum_j |w_{ij}^{\text{out}}|$。
2. **乘以平均激活**：对于 ReLU 等现代激活，单元激活的尺度差异很大，故把权重幅度之和再乘以该单元的平均激活 $\bar{a}_i$（对 LTU 这一点不重要，因其激活恒为 0/1）。
3. **考虑可变性（lability）**：该特征还能多快地被改变——这对“设计能持续适应的算法”尤为关键。
4. **贡献转移到偏置**：在移除一个单元时，把它的平均贡献转移给下游单元的偏置，以减小移除带来的扰动。

把这四点合起来，便得到一个形式上复杂、实则由简单直觉构成的效用度量。可以概念性地记为

$$u_i \;\propto\; \underbrace{\bar{a}_i}_{\text{平均激活}} \cdot \underbrace{\Big(\sum_j |w_{ij}^{\text{out}}|\Big)}_{\text{下游权重幅度}} \cdot \underbrace{\ell_i}_{\text{可变性}},$$

- u_i：单元 i 的效用；$\bar{a}_i$：其平均激活；w_{ij}^{out}：到下游单元 j 的权重；ℓ_i：该单元的可变性（改变难易程度）。效用越低，越优先被替换。

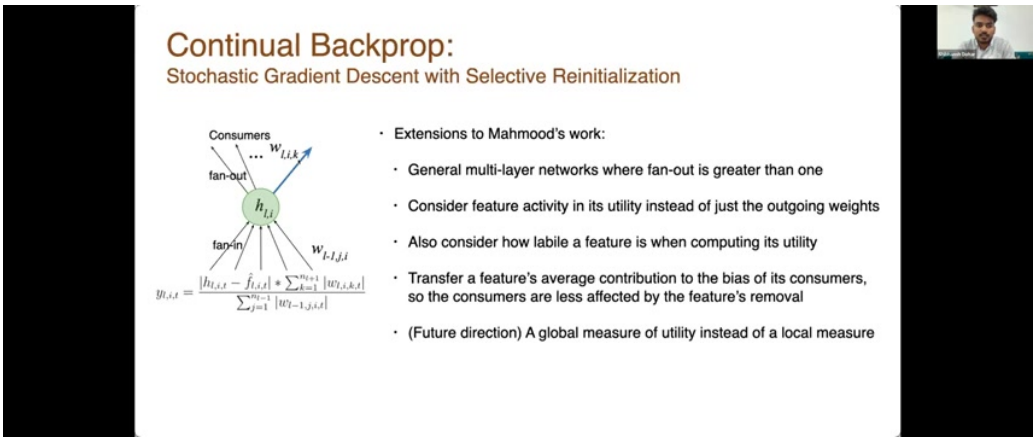


图 14: 效用度量及其“全局化”的开放方向。[14]

Dohare 也坦承当前效度是**局部的**（只看该单元自身的输入权重、输出权重、激活），并指出一个明确的开放方向：发展**全局效用**——衡量某单元对整体所表示函数的影响程度。此外，当前的“生成器”只是从初始分布重新采样，未来完全可能有更聪明的初始化方式来显著提升性能。

两个明确的改进方向

(1) 从局部效用走向全局效用；**(2)** 从“重新随机初始化”走向更优的生成器。两者都是 Generate-and-Test 这一更宽广框架下大量可能性的入口。

[14] 视频画面时间区间：00:42:33–00:43:30。

8.5 本章小结

Continual Backprop 的核心是把“一次性初始化”改造为贯穿全程的“选择性再初始化”：持续地用新随机单元替换效用最低的旧单元。它把 Mahmood & Sutton 的 Generate-and-Test 从单层单输出推广到深度网络，其效用度量由“输出权重幅度、平均激活、可变性、贡献转移”四个直觉要素构成。报告也明确指出，走向全局效用与更优生成器是下一步。

9 Continual Backprop 的实验验证

9.1 置换 MNIST：完全维持可塑性

在置换 MNIST 上，Continual Backprop（蓝线）**完全维持了可塑性**，长期性能不再下降。

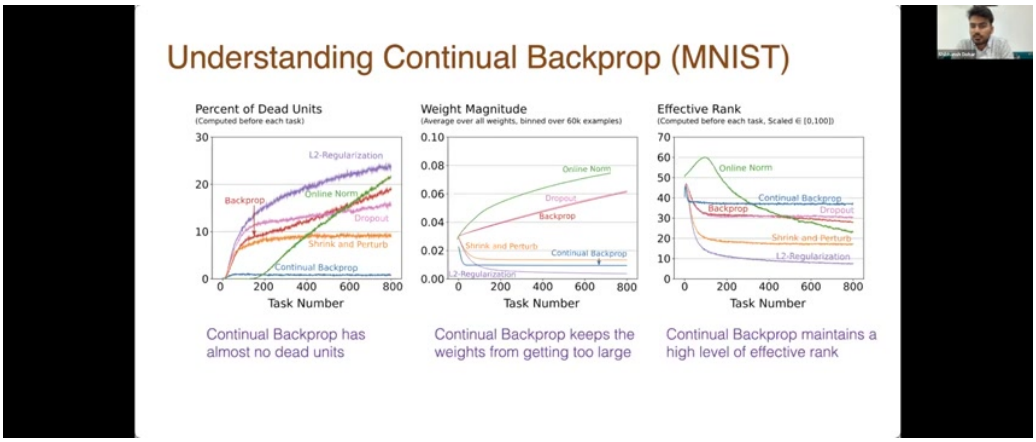


图 15: Continual Backprop 在置换 MNIST 上完全维持可塑性。[15]

其关键超参数是**替换率**（replacement rate）ρ：例如 $\rho = 10^{-6}$ 表示每步替换百万分之一的表征；在 2000 个特征下，相当于*每 500 步替换每层一个单元*——替换极其缓慢。右侧参数曲线显示，*替换率不是一个敏感参数*，在一个数量级内变化对性能影响不大。

用三个体检指标验证，Continual Backprop 全面“对症”：

- **死亡单元**：几乎为零——因为效用度量计入了平均激活，死亡单元会*立刻*被替换。
- **权重幅度**：很小——持续注入的新单元带有较小的随机权重。
- **有效秩**：很高——随机初始化的单元天然带来更好的多样性与秩。

[15] 视频画面时间区间：00:45:20–00:46:00。

9.2 ImageNet：扩展到深度卷积网络

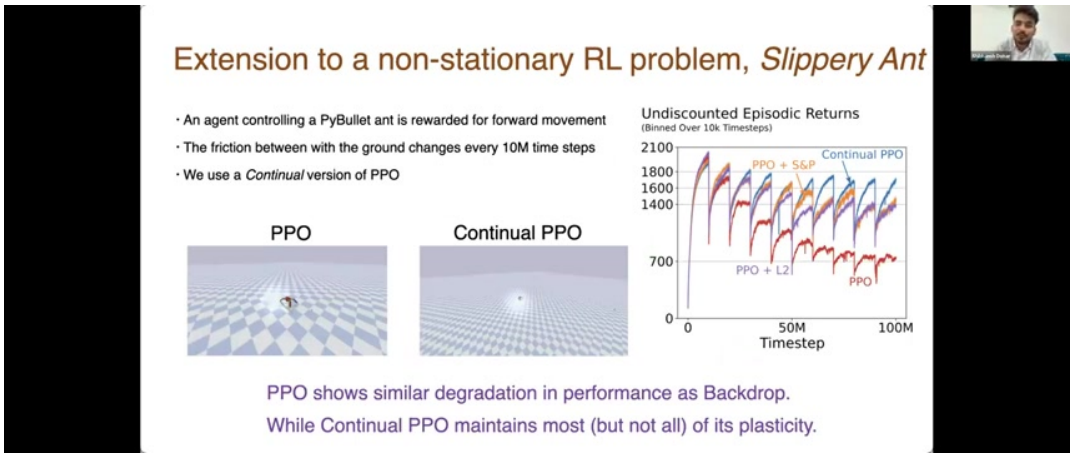


图 16: ImageNet 上 Continual Backprop 几乎完全解决可塑性丢失。[16]

在更深的卷积网络上，Continual Backprop 同样表现出色：反向传播在 2000 个任务后性能显著下跌，L2 与 Shrink-and-Perturb 有显著帮助、不再持续丢失，但达不到最初水平；Continual Backprop 则几乎完全解决了可塑性丢失，最终维持在约 **88–89%**。由于替换很慢，其增益是逐步显现的——初期各算法差异不大，随时间推移 Continual Backprop 的优势才稳定拉开。

9.3 强化学习：Slippery HalfCheetah

作为“蛋糕上的樱桃”，报告给出一个强化学习结果：**Slippery HalfCheetah**——标准 HalfCheetah（PyBullet）的变体，唯一区别是*地面与智能体之间的摩擦每 1000 万步改变一次*，智能体因前进而获得奖励。比较 PPO、带 L2 的 PPO、带 Shrink-and-Perturb 的 PPO 与 **Continual PPO**（基于 Continual Backprop 的选择性初始化）：

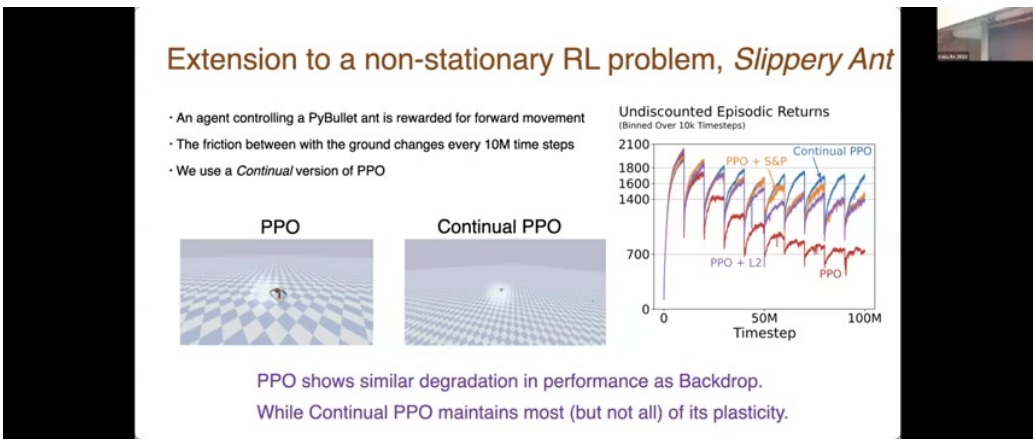


图 17: Slippery HalfCheetah 上 Continual PPO 几乎完全维持可塑性。[17]

- 标准 PPO 初期表现很好，随后性能**彻底崩塌**。

[16] 视频画面时间区间：00:46:00–00:47:30。
[17] 视频画面时间区间：00:47:16–00:48:30。

- L2 与 Shrink-and-Perturb 大幅改善，但 Continual PPO **表现最佳**，几乎完全维持可塑性（尚未完全）。
- 行为层面差异极为直观：PPO 智能体*连走路都挣扎*，而 Continual PPO 智能体*可以奔跑*。

9.4 本章小结

Continual Backprop 在三个尺度上验证有效：置换 MNIST 上完全维持可塑性并治好全部三项内部症结；ImageNet 深度卷积网络上几乎完全解决退化、稳居 88–89%；强化学习的 Slippery HalfCheetah 上 Continual PPO 表现最佳，行为上从“挣扎行走”变为“能够奔跑”。

10 总结与延伸

10.1 演讲者的结论

Dohare 在结尾归纳了整场报告的收获：为**一次性学习**优化的深度网络，在持续学习中*彻底失效*，而归一化、Dropout 等标准手段*并不能帮忙*；与之相对，**Continual Backprop** 这样简单的改动就能让它们有效得多。他也清醒地指出，当前的效用度量仍是**启发式**的，应当有更好的效用定义，尤其是推广到**循环网络**时情形会更复杂。最终的受益者很可能是**标准强化学习**——因为策略迭代、乃至先进的基于模型的架构中存在大量非平稳性来源，许多相互作用的组件都将从“维持可塑性”中受益。

10.2 问答精华

报告后的问答质量很高，值得单独成节，因为它们揭示了方法的边界与团队的研究取向。

- **在单个大规模平稳任务（如 ImageNet）上是否也更好？** 尚未在平稳设定下测试，但有可能改善——大型网络中常见大量死亡单元，Continual Backprop 或许同样有益。
- **改善可塑性与改善迁移如何区分？** Sutton 把两者看作*对立面*：可塑性丢失是指“早先所学让后续任务*更差*”，必须先解决这个“负迁移”，才有资格谈正迁移。
- **可塑性好了，稳定性（stability）呢？** 尚未探索，但思想已内置于 Generate-and-Test——可*以保护高效用单元*、阻止梯度下降改变它，从而兼顾稳定。这是明确的未来方向。
- **方法是否依赖任务边界？** *不依赖*——目标是完全持续的方法，没有“任务”概念，适用于只有缓慢概念漂移、根本没有明确任务的真实场景。
- **置换 MNIST 是否是好测试床？** Sutton 认为它*完美*适合研究可塑性丢失，因为这里不关心“记住所有任务”，只关心“当前任务表现如何”；并再次援引 **big world** 思想——心智总小于世界，无法在所有情形下都好，只能就地跟踪。
- **是否做了不确定性量化？** 没有。当前性能度量其实是“对不确定性无感的”——用交叉熵训练，却只报告 argmax 类别，没有利用预测概率的不确定性。团队承认应当更关注这一点。
- **替换率应否自适应？** 理想答案是“是”——用*元学习*自动设定替换率等参数。但由于性能对替换率不敏感（一个数量级内影响不大），目前并非紧迫问题。Sutton 表示他非常想攻克“自动为网络不同部分设定步长”的元学习问题。

- **换激活函数（如 SELU）能否绕开死亡神经元？** 不能。如前所述，只要激活函数有“快/慢”两段梯度区域，单纯更换就无法根治。
- **能否用模拟退火等非梯度方法在任务切换时“跳出去”？** 提问者暗示可利用“任务已变”的侧信道信号。Sutton 明确反对给智能体注入特殊侧信道信息，更倾向于用元学习持续地决定“现在该改变多少、以后该改变多少”——并以“今天腿累了、要换种走法，但手臂还好”作比喻，强调应按网络的不同部分自适应地调节学习。
- **不断重置会不会耗尽容量、无法永远可塑？** Sutton 不认为会“耗尽”——有限智能体在巨大世界中本就必须在“记住”与“遗忘”之间权衡；目前聚焦于遗忘/可塑性一侧，记忆一侧有待后续。
- **是否试过“反向操作”——重置最高效用的单元？** 团队做过消融。提问者联想到了经典的**最优脑外科手术**（optimal brain surgery）：移除重要成分反而可能让系统更好，虽反直觉却有意思。

10.3 我的提炼

通读全篇，我认为这场报告最值得带走的有三层认识。

第一层：诊断必须可观测

“深度学习不能持续学习”本可能停留在口号，但报告用 **ImageNet**、**MNIST**、**缓慢变化回归**三个尺度、用**死亡单元/权重幅度/有效秩**三个可观测指标，把一个模糊的“变慢”落到了可测量、可对照的体检项目上。这种“把现象变成可观测量的诊断范式”，比任何单一结论都更可迁移——它让我们能用同一把尺子去衡量未来任何声称“解决持续学习”的方法。

第二层：持续 $\neq$ 一次性 + 时间

Continual Backprop 的深刻之处不在替换本身，而在一个观念转变：**持续学习的算法应当在所有时刻做同类计算**。标准反向传播把“初始化”当作只在开头发生的一次性仪式，这本身就与“持续”相悖。把初始化从仪式变为常规操作，本质上是承认——在持续学习中，生成新候选与淘汰旧候选应当和学习一样，是一个永不停止的过程。这与生物系统的“生成—选择”循环、与 Sutton 一贯的“持续学习”世界观高度一致。

第三层：可塑性与稳定性是一体两面

问答中反复浮现的“稳定性/记忆”问题，揭示出 Continual Backprop 目前只解了持续学习的一半——它解决了“还能不能学”，却尚未系统解决“该不该忘、该记什么”。一个完整的持续学习系统，必然要在“注入新单元以保可塑性”与“保护高效用单元以保稳定性”之间达成动态平衡。报告已为这一半指明了入口（保护高效用单元、全局效用、自适应替换率），这恰恰是后续最有价值的研究方向。

10.4 开放问题与下一步

结合报告与问答，我梳理出几条清晰的后续线索：

1. **全局效用度量**：从“单元自身属性”走向“单元对整体函数的贡献”，为选择性替换提供更准确的依据。

2. **更优的生成器**：替换时不止于从初始分布重采样，而用更有针对性的初始化/生成策略。
3. **稳定性机制**：高效用单元的显式保护（冻结其梯度），与可塑性注入形成闭环，构建“可塑—稳定”的权衡控制器。
4. **元学习自适应**：自动设定替换率、以及按网络不同部分设定步长，把目前手工调参的旋钮交给元学习。
5. **循环网络与表征动态**：将方法推广到 RNN/Transformer 等带状态的架构，其中可塑性丢失的机理可能更复杂。
6. **不确定性感知**：把预测概率的不确定性纳入性能度量与替换决策，而非仅用 argmax。
7. **平稳设定下的迁移**：检验 Continual Backprop 在标准（非持续）大规模训练中是否也能因“清理死亡单元”而获益。

一句话总结

这场报告用一个清晰的诊断（标准深度学习在持续学习中丧失可塑性）、一套可观测的体检指标、以及一个简单而深刻的处方（把初始化变成持续的选择性再初始化），为“深度持续学习”指明了一条务实的研究路线。它最持久的价值，或许不在于 Continual Backprop 这个具体算法，而在于它示范了**如何把“学习的能力”本身当作一个可以被测量、被维护、被持续优化的对象**。