理解深度学习

2023 年 9 月 13 日

目录

Chapter 1

损失函数景观的嵌入原则

凝聚现象展示了非线性训练过程同层神经元的特殊行为。这个现象引出了很多有意思的问题。比如，既然训练过程要使神经网络的有效神经元数目远比实际网络的规模要小，那为什么我们不直接训练一个小网络呢？当然，我们要使用一个足够表达目标函数的小网络。在表达能力上，大网络和小网络都足以表达目标函数的情况，它们有什么相似和差异呢？显然，它们一定有差异，否则，我们一般就不会用过参数化的网络。再比如，为什么我们总观察到凝聚的方向数目在训练过程中是单调增加的?

为了理解这些问题，我们需要关注非凸优化过程中，参数在训练过程中经历的路径具有哪些特点。我们引入损失景观这个常见的概念，损失景观本质上就是损失函数 (经验风险函数)，之所以叫景观是强调这个函数在参数空间具有像地貌一般高低起伏的特性，这种特性对理解损失函数诱导的训练过程有重要的意义。前面提到的两个问题的本质问题在于，损失景观具有什么样的结构使得凝聚现象可以发生。

我们的切入点是研究不同宽度的网络的损失景观之间的联系，并发现了嵌入原则 (Embedding Principle)(Zhang, Zhang, Luo & Xu 2021), 即一个神经网络的损失景观中“包含”所有更窄神经网络损失景观的所有临界点（损失函数关于参数导数为零的点，包括鞍点、局部最优点和全局最优点等）。具体而言，这项工作发现了一类将窄网络的参数空间嵌入到任一更宽网络的参数空间中的方法，能够保证窄网络的任何临界点嵌入到宽网络后仍然是临界点并且网络的表征保持不变（作为推论，网络的输出函数和损失值也都保持不变）。通过引入这种一般的嵌入方法，我们可以发现对于任一窄网络的临界点，所有比之更宽的网络的损失景观中都包含有和该临界点具有相同输出函数的临界点，这也就是“包含”的含义。实验发现，在很大的初始化区域，神经网络的实际训练过程会经历这类由嵌入原则带来的极值点附近，这使得嵌入原则的理论对理解神经网络的训练过程具有重要意义。

1.1 不同宽度的神经网络的训练过程

我们取目标函数

$$f(x) = \sigma(2x + 0.6) - \sigma(x) + \sigma(0.5x - 0.4) - \sigma(1.5x - 0.6),$$

其中 $\sigma(x) = \mathrm{ReLU}(x)$。我们使用梯度下降训练不同宽两层神经网络，学习率为 0.001。根据相图分析，我们选择所有网络的参数的初始化满足凝聚区域中的同一位置（也就是具有相似的动力学），其中 m 为隐藏层神经元的数目。这样的初始化保证这些神经网络都处于非线性训练区域。如图 1.1所示，不同宽神经网络在训练过程中，他们损失函数会在一些相同的值附近停留比较长的时间。我们很自然地提出两个问题：

(1) 这个现象是否具有普遍性，还是一次试验所导致的巧合；

(2) 损失停驻位置具有相同的损失值是否意味着它们有一些特殊的相似性。

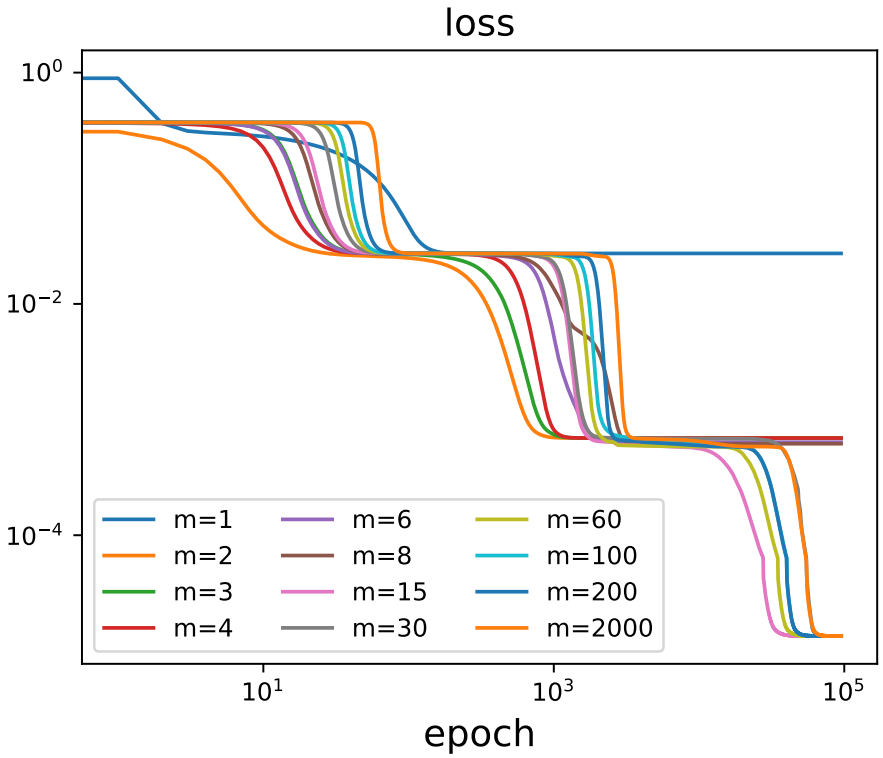


图 1.1: 小初始化下不同宽神经网络的损失图线，其中 m 代表不同网络宽度。

对于第一个问题，我们使用多次不同初始化的结果进行验证。对于一条损失函数的演化轨

迹，我们可以统计它在各个小区间经历的训练步数。如果某个损失函数在一个小区间内演化了非常多的步数，那它表现在损失函数的演化图中就是图 1.1所示的停留的平台。对于每个宽度的神经网络，我们进行了 100 次实验，然后把它们每个训练步对应的损失值合并统计，就可以得到损失函数值的分布。如图1.2所示，每一行就是一个宽度的网络所对应的损失函数值的分布，图的颜色表示损失函数值落在横坐标所对应的小区间的频率。图1.2的功能类似于物理学中的能谱图，通过能谱图中谱的离散型及有限性，可以分析损失函数的一些信息，特别是它会在哪些值发生停留，以及不同宽网络之间的关系。如图1.2所示，我们取纵坐标为网络的宽度变化，横坐标为 $\log_{10}(\text{loss})$ 的值。对于横坐标取值区间 $[-5,0]$，我们在其中等距取 200 个区间。对于每个给定的宽度，我们记录 100 次实验中落在每个区间中的损失值频数 p_i。由此我们可以得到集合 $\{p_i\}_{i=1}^{200}$，即为在 100 次实验中损失值经历某取值区间的次数。为了使较小频数的谱线也能在图中明显呈现，我们使用对数尺度进行颜色设定，具体而言，我们取 $\log_{10}\frac{p_i}{\sum_{i=1}^{200}p_i}$ 作为颜色条的取值。直观理解，对于 100 次实验中经历次数较多的区间 (损失函数存在损失停滞现象的区间), 谱线会呈现更加高亮的色彩。

图1.2展示的统计结果提供了很多有意思的信息。首先，对于同一宽度的网络，不同随机种子下，损失函数具有高度的相似性，也就是在同样的损失值处停留得比较久。其次，不同宽度的网络也具有高度的相似性，它们在类似的损失值处发生明显的停留。更有意思的地方在于，网络越宽的情况下，损失函数在更小的数值停留的概率更大。按我们统计的方式，最后一个停留的数值是训练后停止时的值。因此，从这个图可以发现，当网络越宽时，神经网络的损失值会更容易下降到更小的数值。这显然是一个非常重要的现象，这提示我们宽网络可能会带来优化上的优势。

对于第二个问题，我们考察不同宽神经网络在同一驻点位置的等效神经元特征及其网络输出。如图1.3(a)，我们展示了不同宽度的神经网络在同一个谱峰上（损失函数停留的地方）的输出函数，它们彼此非常相似。该谱峰是宽度为 2 的神经网络最小值点所对应的损失谱线。我们可以在图中看到对于其他更宽的网络，当他们的损失函数到达此损失值时，其输出函数也出现了极为相似的输出。回顾一下在凝聚现象上的讨论，大宽度的网络在发生凝聚现象时，可以等效成小网络，也就是和小网络有相同的输出。受此启发，我们进一步考虑网络的特征分布图。对于一维输入的两层神经网络

$$f_{\boldsymbol{\theta}}(\boldsymbol{x})=\sum_{j=1}^{m}a_j\sigma(\boldsymbol{w}_j\cdot\boldsymbol{x})=\sum_{j=1}^{m}a_j\sigma(w_jx+b_j), \tag{1.1}$$

为了深入探究它们的参数空间表示的细节，我们注意到对于 ReLU 激活函数，每个神经元的参数对 $(a_k,\boldsymbol{w}_k)$ 可以分离为一个单位方向特征 $\hat{\boldsymbol{w}}=\boldsymbol{w}/|\boldsymbol{w}|_2$ 和一个表示其对输出贡献的振幅 $A=|a||\boldsymbol{w}|_2$, 即 $(A,\hat{\boldsymbol{w}})$。对于一维输入，由于加入了偏置项，$\boldsymbol{w}$ 是二维的。因此，我们使用每个 $\hat{\boldsymbol{w}}$ 相对于 x 轴的 $[-\pi,\pi)$ 内的角度来表示其方向。我们对 $(A_k,\hat{\boldsymbol{w}}_k)_{k=1}^m$ 的散点图做一些小

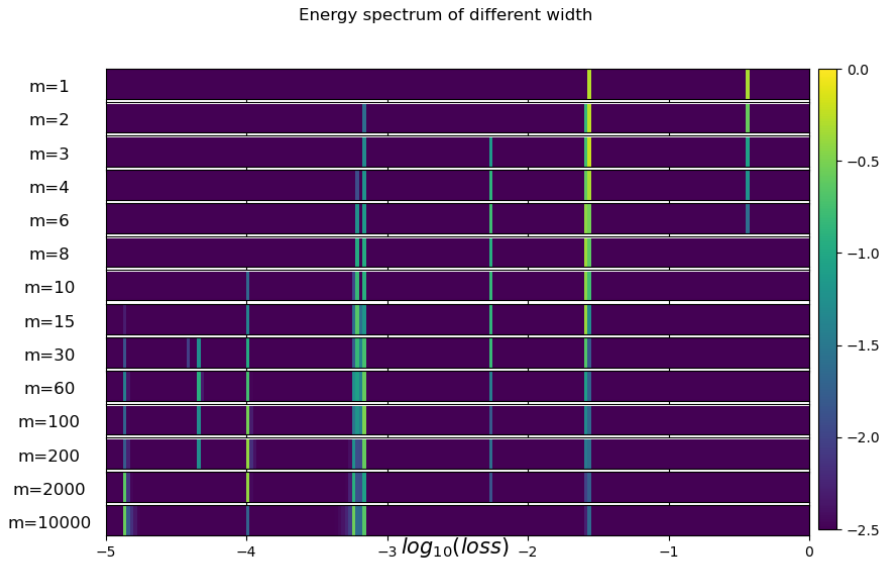


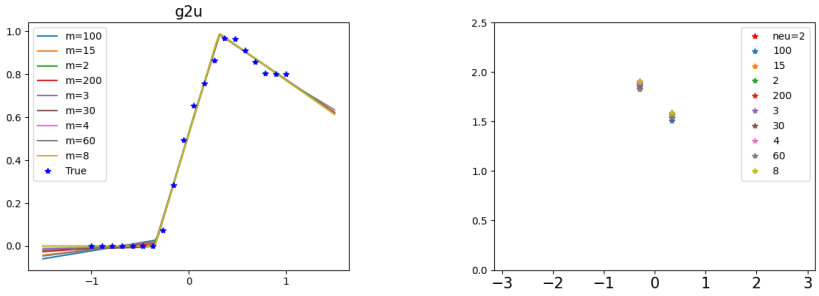
图 1.2: 不同宽的两层神经网络，100 次训练得到的损失谱图。

修正。我们将同一方向上的神经元振幅相加，作为一个等效神经元，并删去振幅接近 0 的等效神经元。如图1.3(b) 所示，不同宽网络的等效神经元具有近乎一致的特征方向及振幅。

这些实验结果将凝聚和训练过程的停留联系在了一起。因为发生凝聚，所以不同宽度的网络会具有相似的损失函数值和输出函数。损失谱图又给我们提示，更宽的网络似乎更容易优化到更小的损失函数值。下面，我们进一步分析其中的机制。

1.2 临界点

对于前面提到的现象，要理解其中的机制，我们需要明确哪些因素是关键的。训练中会发生停留的现象，显然很关键。停留意味着训练非常慢，而训练是由梯度驱动的，因此，这些停留的点应该是在某个梯度接近于零的点。我们把梯度为零的点定义为临界点或者驻点。一般临界点包括全局最小点，局部最小点，以及鞍点。直观地理解，全局最小点和局部最小点的周围都比它高，它没有下降的方向，而对于鞍点，它沿着周围的某个方向可以使损失函数继续下降。稍微严格一些，我们可以定义损失函数在这些点的二阶导，对于高维函数，二阶导是一个矩阵，定义为 Hessian 矩阵。对于全局最小点和局部最小点，他们的 Hessian 矩阵的特征值都是正的，而对于鞍点，Hessian 矩阵存在负的特征值。另外，我们说某个方向是退化的时候指



(a) 网络输出　　(b) 网络特征

图 1.3: (a) 不同宽的两层神经网络，在宽度为 2 的神经网络最小值点所对应的损失谱线处的网络输出。(b) 不同宽的两层神经网络，在宽度为 2 的神经网络最小值点所对应的损失谱线处的网络特征。

Hessian 矩阵在这个方向的特征值为 0。

因此，我们需要深入去研究临界点的性质，特别是不同宽度的神经网络他们为什么会经历看起来相似的临界点，以及这些临界点和凝聚现象有什么联系。

临界点是损失景观中非常重要的结构，它们与训练有非常密切的关系。我们希望训练能够找到全局最小点，而不是局部最小点，更进一步，我们希望能够从众多全局最小点中找到具有泛化能力的点。鞍点在训练点也有特殊的位置，如果太靠近鞍点，会使点训练过程在鞍点附近停留的时间太长，造成过低的训练效率。但鞍点也不是一无是处，由于它在某些方向是极小值点，训练的轨迹会被它吸引，同时，它在另一些方面又是极大值，也就是在这些方向它能继续下降，因此，不同初始化开始的训练轨迹可能会被它吸引到非常相近的区域后，以类似的路径持续降低损失函数，使得整个训练过程对初始化的敏感程度降低。

1.3 损失景观中的嵌入原则

理解深度神经网络的损失景观对于深度学习理论至关重要。一个重要的问题是准确量化损失景观的结构特征。这个问题很困难，因为损失景观如此复杂，几乎可以呈现出任何模式 (Skorokhodov & Burtsev 2019)。而且，它的高维度以及对数据、模型和损失的依赖性使得通过经验研究获得一般性理解非常困难。因此，尽管多年来进行了广泛研究，但仍然存在很多未解之迷。

受到凝聚现象的启发，我们证明了深度神经网络的损失景观中存在一个嵌入原则，如下所述：

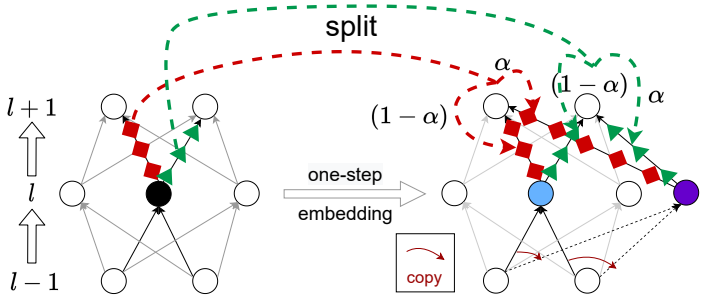


图 1.4: 一步嵌入的示意图。左侧网络中的黑色神经元分裂成右侧网络中的蓝色和紫色神经元。左侧网络中黑色神经元的红色（绿色）输出权重分裂成右侧网络中的两个红色（绿色）权重，比例分别为 α 和 $(1-\alpha)$。

嵌入原则：任何网络的损失景观都“包含”所有较窄网络的临界点。

“较窄的网络”指的是深度相同但每层宽度不大于目标神经网络。嵌入原则在某种程度上滥用了“包含”的概念，因为不同宽度的神经网络的参数空间是不同的。然而，这种包含关系在某种意义上是合理的，因为我们通过一类嵌入操作，任何较窄网络的临界点都可以嵌入到目标网络的临界点中，同时保留其输出函数不变。由于这种保持临界点的特性，我们将构造的嵌入操作称为临界嵌入（critical embedding）。

我们以一个“原则”作为结论，是因为嵌入原则是深度神经网络损失景观的一个非常普遍的特性，独立于训练数据和损失函数的选择，并且与深度神经网络的逐层架构本质相关。此外，嵌入原则与深度神经网络的训练密切相关。图 1.1和图 1.3的例子说明，在实际训练中，宽的网络会经历一些鞍点，这些鞍点和窄网络的全局最小点具有相同的输出，可以理解为，窄网络的全局最小点嵌入到宽网络中后变成鞍点。

我们首先直观地介绍一步嵌入，更严格的定义可以参考文献Zhang, Zhang, Luo & Xu (2021), Zhang, Li, Zhang, Luo & Xu (2021)。如图 1.4所示，一步嵌入是通过将任何隐藏神经元（例如左侧网络中的黑色神经元）分成两个神经元（右侧网络中的蓝色和紫色）。这两个分裂神经元的输入权重与原始黑色神经元的输入权重相同。原始黑色神经元的每个输出权重都被分为分数 α 和 $(1-\alpha)$（$\alpha \in \mathbb{R}$，一个超参数）。多步嵌入是多个一步嵌入的组合。由于每个一步嵌入可以向选择的层添加一个神经元，因此通过多步嵌入，任何更窄的网络都可以嵌入到任何更宽的网络。多步嵌入操作保持了以下的性质。首先，窄网络和宽网络的输出是一致的，其次，如果窄网络处于临界点，则嵌入得到的宽网络也处于极值点。

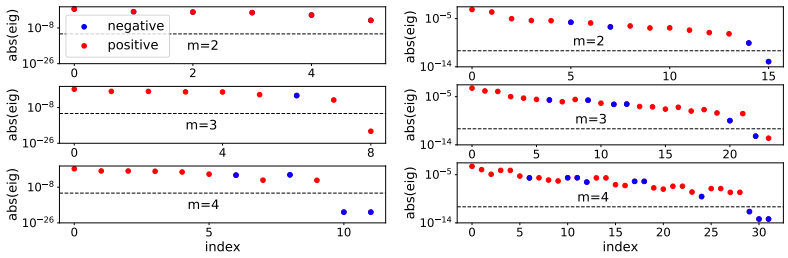
更一般的嵌入操作可以参考文献Zhang, Li, Zhang, Luo & Xu (2021)。

有了嵌入原则后，我们建立了不同宽度网络的临界点之间的联系。进一步，我们可以研究它们之间有什么差异。有两个很直接的切入点，一是研究它们的下降方向的个数，也就是二阶

导 Hessian 矩阵的负特征值个数，显然，下降的方向越多，对优化越容易；其次，我们可以研究它们的退化程度，也就是二阶导 Hessian 矩阵的零特征值个数。

首先，我们用实验来看一下当嵌入发生时，特征值如何发生变化。我们训练一个宽度为 $m_{\text{small}} = 2$ 的两层神经网络，用于学习一些从一维函数采样获得的数据（如图1.5(a) 所示）或 Iris 数据集 (Fisher 1936)（如图1.5(b) 所示）。当损失下降非常极其缓慢的时候，我们判定它处于临界点，并且验证此时的导数数值。在图1.5(a) 中，该经验性临界点处的损失函数导数的 L_1 范数约为 7.15×10^{-15}，在图1.5(b) 中约为 3.72×10^{-13}，这些值都相当小，因此可以合理的把它们当作临界点。

然后，我们通过一步或两步嵌入将此临界点嵌入到宽度为 $m = 3$ 和 $m = 4$ 的网络中。从图1.5可以明显看出，每一步嵌入都会在 Hessian 矩阵中引入一个额外的零特征值。此外，在图1.5(a) 中，对于 $m = 2$，所有特征值都是正的（红色），表明通过训练得到的临界点是一个局部或全局最小值。经过嵌入后，由于出现了负特征值（蓝色），这一点变成了一个鞍点。具体来说，在图1.5(a) 和 (b) 中，我们观察到显著的负特征值稳步增加，例如在 (a) 中从 0 到 1 再到 2，在 (b) 中从 3 到 5 再到 7，这意味着在更宽的神经网络中，从相应的临界点中逃脱的难度降低了。



(a) 一维人造数据　　(b) Iris 数据

图 1.5: 学习一维人造数据和 Iris 数据集中的临界点所对应的神经网络的 Hessian 矩阵特征值。每个子图中的 m 值表示嵌入后的神经网络宽度。每个子图中的辅助虚线是 $y = 10^{-11}$。在这些子图中，我们将处于临界点的宽度为 2 的两层神经网络中的一个神经元等分成 k 个神经元 $(k = 2, 3)$，其输入权重保持不变，但输出权重变为原神经元的 $1/k$。可以看到，在嵌入后，Hessian 矩阵的特征值中出现了负特征值，这表明了更宽的神经网络中相应临界点的变化，以及从这些临界点中逃脱的难度降低，支持了前文的观察。

其次，我们观察一下一步嵌入操作中有一个自由参数 α，因此，当一个网络嵌入到一个比它更宽一个神经元的网络时，它并不是嵌入到一个点，而是嵌入到一个一维的流形中，在这个一维流形中，所有的点都梯度都为零。因此，二阶导在这个方向上为零，即，新加了一个退化方向。对于一个宽网络中的两个临界点，如果它们分别来源于两个小网络的嵌入，且每个网络

都不能再找到更小的网络来嵌入。显然，由更小网络嵌入的临界点带来了更多的退化方向，但注意，我们并不能说这个点一定有更多的退化方向。

让我们更加不严格一些来看一下退化对训练有什么影响。考虑一个临界点所拥有的体积，可以这样认为，在这个体积内，损失函数的值和这个临界点的值的差距小于某一个非常小的数。显然，如果这个临界点的周围很平坦，那体积很大，如果非常陡峭，那体积非常小。体积越大，训练过程就越有可能会落入到这个临界点附近。最平坦的情况就是这个方向是完全平，也就是损失函数在这个方向是退化的。直观上，越小网络嵌入的临界点会有更多退化，因此，在训练过程中，训练轨迹更有可能先经历由越小网络嵌入而来的临界点。虽然这个说法极其不严格，但实验上却明显观察到这样的现象。正如在凝聚现象的章节里看到的，凝聚的方向从一个到五个逐渐增长。

现在，我们来看一下当嵌入发生时，Hessian 矩阵的特征值是如何发生变化的。Zhang, Li, Zhang, Luo & Xu (2021) 证明对一般的临界嵌入操作，大网络的 Hessian 矩阵的零特征值（退化方向），正特征值（上升方向），和负特征值（下降方向）的个数都不会下降。零特征值的退化情况我们已经讨论过了，现在我们感兴趣负特征值方向个数，因为它直接影响优化的效率。只要有负特征值方向，这个临界点就不会是局部最小点。因此，对于一个小网络的局部最小点，当它嵌入到更宽的网络时，它就有很大的概率由于增加了下降方向而变成鞍点，并且，下降方向的数目越多，训练更容易逃离该鞍点。这个理解和我们在损失图谱中看到更宽的网络更容易学到更小的损失值是一致的。

我们来总结一下实验和直观理解。在一个神经网络的损失景观中，由更小的网络的临界点嵌入得到的临界点有更多的退化和下降方向，因此，它相比于该网络的损失景观中的其它临界点，更多的退化方向使它更容易吸引训练轨迹，相比于具有相同表达能力的小网络，更多的下降方向使它更容易将损失函数下降到更小的数值。尽管神经网络的损失景观非常复杂，但它的层结构使得复杂的损失景观中具有特殊的结构，能够使神经网络的在训练中慢慢增加复杂度，并且容易下降误差，这些参数空间的理解和频率原则带来的理解是一致的。

下图是一个极端的例子来抽象上面的这些理解。一个人从初始点要走向最后的城堡，中间有各种危险。但由一个神经元构成的网络的全局最小点嵌入来的临界点充当了第一个交警，将该人引到第一个驿站，然后由两个神经元构成的网络的全局最小点嵌入来的临界点充当了第二个交警，继续将该人引到第二个驿站，实际训练，可能跳过某个临界点，比如后面直接到由四个神经元构成的网络的最小点嵌入而来的临界点，直接到最后找到城堡。这些带有结构的临界点形成了一些特殊的训练路径。

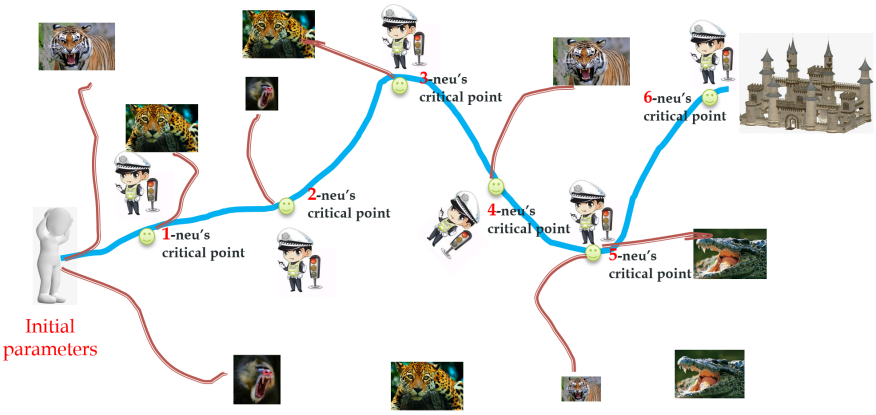


图 1.6:

1.4 简化神经网络的规模

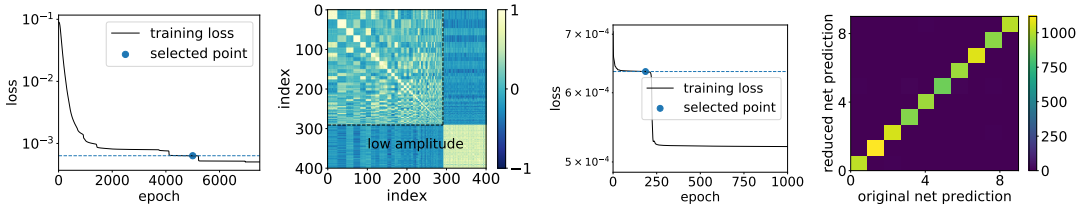
实际的训练过程中，神经网络一般不会严格处于凝聚区域，因此，我们很难观察到非常绝对的凝聚，但可以看到同层神经元会趋于凝聚，达到控制复杂度的效果。

这里我们用一个凝聚区域训练的网络来看一下当一个神经网络凝聚到一个由小网络嵌入而来的临界点时，我们可以快速简化网络的规模。

嵌入原则预测了从较窄的神经网络嵌入的临界点处，我们应该能够将这些凝聚的神经元组由一个神经元替代。这一预测在图1.7中的以下实验中得到了证实。我们在 MNIST 数据集的 1000 个训练样本上使用小初始化训练了一个宽度为 400 的两层 ReLU 神经网络 $\boldsymbol{f}_{\boldsymbol{\theta}} = \sum_{k=1}^{m} a_k \sigma(\boldsymbol{w}_k^T \tilde{\boldsymbol{x}})$ ($\tilde{\boldsymbol{x}} = [\boldsymbol{x}^\intercal, 1]^\intercal$)。在图1.7(a) 中的蓝点处，损失下降非常缓慢，很可能非常接近一个鞍点。然后，我们通过计算两个归一化输入权重的内积来检查每对神经元输入权重之间的方向相似度。

如图1.7(b) 所示，出现了 58 个神经元组（忽略了振幅非常小的神经元，后续直接移除），其中同一组内的输入权重相似度至少为 0.9。对于每个相似性组 S_{similar}，我们随机选择一个神经元 j，将其输出权重替换为 $\sum_{k\in S_{\text{similar}}} a_k \|\boldsymbol{w}_k\|_2 / \|\boldsymbol{w}_j\|_2$，并且移除组内的所有其他神经元。减少前的参数集记为 $\boldsymbol{\theta}_{\text{ori}}$，减少后的参数集记为 $\boldsymbol{\theta}_{\text{redu}}$。神经网络的宽度从 400 减少到 58。我们从 $\boldsymbol{\theta}_{\text{redu}}$ 开始训练减少后的神经网络，如图1.7所示，它在几步后停滞在与图1.7(a) 中的蓝点相同的损失值，由蓝色虚线标记，并在图1.7(c) 中用蓝点表示。然后，我们在 10000 个测试数据上比较原始模型和简化模型在相应的蓝点上的预测，如图1.7(d) 所示。对于每个格点，颜色表示该预测对的频率。具体来说，对角线元素的亮点表示两个模型之间的高预测一致性（总体约为 98.5%）。因此，简化后的宽度为 58 的神经网络的这个临界点与原始宽度为 400 的神经网

络的临界点非常匹配，清楚地证明了我们的嵌入原则与真实数据集训练的相关性。



(a) initial loss (b) orientation similarity (c) loss after reduction (d) prediction similarity

图 1.7: (a) 初始网络在 MNIST 上的训练损失。蓝点被选中进行简化。(b) 不同神经元的输入权重的归一化内积。横坐标和纵坐标表示神经元序号。位于“低振幅”区域的神经元幅度远低于其他神经元，因此被移除。(c) 简化网络的训练损失。蓝色虚线表示与 (a) 中的蓝色虚线相同的损失值。蓝点被选为代表进行比较。(d) 预测相似性。对于每个格点，颜色表示该预测对的频率。

参考文献

Fisher, R. A. (1936), 'The use of multiple measurements in taxonomic problems', *Annals of eugenics* **7**(2), 179–188.

Skorokhodov, I. & Burtsev, M. (2019), 'Loss landscape sightseeing with multi-point optimization', *arXiv preprint arXiv:1910.03867* .

Zhang, Y., Li, Y., Zhang, Z., Luo, T. & Xu, Z.-Q. J. (2021), 'Embedding principle: a hierarchical structure of loss landscape of deep neural networks', *arXiv preprint arXiv:2111.15527* .

Zhang, Y., Zhang, Z., Luo, T. & Xu, Z.-Q. J. (2021), 'Embedding principle of loss landscape of deep neural networks', *arXiv preprint arXiv:2105.14573* .