迈向量子多分类的极致准确率：迹距离二叉树 AdaBoost 分类器

Xin Wang，[1] Yabo Wang，[2]，* 和 Rebing Wu[1]，†[1]清华大学自动化系，
中国北京，*100084*[2]西南交通大学数学学院，中国成都，*611756*

arXiv:2602.02120v1 [quant-ph] 2 Feb 2026

我们提出一种迹距离二叉树 AdaBoost（TTA）量子多分类分类器，这是一种实用的量子多分类流程，将量子感知降维与集成学习相结合，以提升可训练性和资源效率。TTA 在每个内部节点选择能最大化平均量子态之间迹距离的二分划分，从而构建层次化二叉树；每个节点都训练一个由浅层变分量子基学习器组成的二元 AdaBoost 集成。TTA 将本质上难以区分、迹距离较小的类别差异限制在较小的节点专属数据集中，并通过 AdaBoost 组合弱浅层学习器，使模型容量分布在多个小型子模型中，而非集中于一个深层电路，从而在不牺牲泛化能力的前提下，缓解贫瘠高原和优化失败问题。实验结果表明，TTA 在量子和经典基线方法中取得了最高测试准确率（≈100%），对常见量子误差具有鲁棒性，并实现了累计层数达 10000 层、参数量为 0.2M 的聚合系统，该系统由多个浅层电路构成。我们的结果具有实证基础，且可在近期平台上实现，为可扩展的量子多分类机器学习提供了一条资源高效的路径。

I. 引言

量子机器学习（QML） [1] 已成为一种颇具前景的范式，它利用叠加和纠缠等量子特性，有望超越经典方法 [2] ，带来表征 [3] 或泛化优势 [4]，应用于分类等机器学习任务。近年来，理论分析和实验均表明，在某些情形下，量子机器学习模型能够利用有限数据实现良好的泛化 [5, 6], ，这促使人们研究其在图像分类等实际任务中的应用。

然而，尽管参数化量子电路（PQC）具有强大的泛化能力，但其在量子机器学习模型中的可训练性严重限制了实际应用价值。具体而言，浅层参数化量子电路的表达能力有限，且可能存在指数多个性能不佳的局部极小值 [7, 8], ，使得收敛到全局最优解的可能性很低。虽然有人提出通过过参数化（增加电路深度）来缓解这一问题 [9]，但过深的电路不可避免地会遭遇贫瘠高原现象：损失函数及其相对于电路参数的梯度都会呈指数级集中 [10, 11], ，导致大规模量子机器学习模型实际上无法训练。雪上加霜的是，近端量子设备上的噪声会随电路深度累积，进一步降低计算保真度 [12]。因此，参数化量子电路的设计似乎陷入了金发姑娘困境：电路既不能太浅，也不能太深，而必须限制在狭窄的中间范围内，才能保持可训练性。结果，量子机器学习模型似乎无法通过扩大模型规模来提升性能，这与经典大型语言模型形成对照语言模型 [13]。

为了将量子机器学习模型扩展到更大规模，我们采用 AdaBoost 方法 [14, 15] ，将多个浅层量子机器学习模型集成为一个大规模量子机器学习模型。AdaBoost 的核心思想是将多个弱基分类器组合起来；这些分类器在训练集上的表现仅略好于随机猜测，从而构成一个能够正确预测训练集中大多数数据的强分类器，进而提升训练性能。首先，AdaBoost 只要求基分类器在训练集上的性能优于随机猜测，而无需达到全局最优解。因此，一方面，浅层量子机器学习模型陷入局部极小值不再是关键问题（只要这些局部极小值对应的表现优于随机猜测）；另一方面，选择浅层量子机器学习模型也能避免贫瘠高原现象。此外，浅层模型可防止量子噪声因层层累积而过度放大。再者，AdaBoost 只要求基分类器的表现略好于随机猜测。这一弱学习条件使量子机器学习基学习器不再依赖精心设计的参数化量子电路结构，从而简化电路设计并提升鲁棒性。此外，从拉德马赫复杂度的角度来看 [16], ，通过 AdaBoost 构成的强分类器，与作为基分类器的量子机器学习模型一样，具有出色的泛化性能。

尽管如此，AdaBoost 仅能为二分类提供可训练性保证，而无法为更实用的多分类设置提供保证。对于二分类，AdaBoost 强分类器的训练误差上界会随弱分类器数量增加而指数级下降，而多分类问题则不存在类似的保证。因此，对于现实世界中的多分类数据集，以量子机器学习模型为基分类器的多类AdaBoost，其改进仍然有限 [17]。一种自然的思路是将二分类的训练保证扩展到

* wangyabo@swjtu.edu.cn
† rbwu@tsinghua.edu.cn

多分类问题，即将多分类问题化简为多个二分类子问题。将多分类问题化简为二分类的标准策略主要包括：（i）按位聚合，将类别标签编码为二进制形式，并使用多个二元分类器，每个分类器负责预测一个比特；最终类别由预测比特解码得出 [18]；（ii）一对其余（OVR），为每个类别训练一个分类器，将该类别视为正类、其余所有类别视为负类，并根据针对给定测试样本置信度最高的分类器进行预测；（iii）一对一，为每对类别构建一个分类器，每次只处理两个类别，并通过投票聚合其判定 [19]。我们在附录中对这三种化简方法作了更详细的介绍。A。然而，将这些方法应用于量子机器学习时，并未利用量子特有的信息。具体而言，类别量子态固有的可区分性（例如，期望状态之间的迹距离）会显著影响量子分类器的可训练性 [20]。量子态几乎无法区分的二分类子问题，对任何量子分类器而言都很难；在这种情况下，即使是二分类AdaBoost，也可能无法使二元分类器达到 100% 的训练准确率，而这一问题可能沿化简过程传递，导致多分类性能下降。

为解决这些问题，我们提出一种新颖的量子多分类器：迹距离二叉树AdaBoost（TTA）分类器。该方法将多类别训练集划分为正负子集，再对正负子集递归地施加相同的二元划分，从而构建数据集二叉树，直至每个叶节点仅包含单一类别的样本。与以往将多类别归约为二元的方法不同，TTA分类器利用了量子特有的性质，即两个类别期望状态之间的迹距离会影响量子机器学习模型的训练性能。因此，每一次划分都在所有可能的类别组合中寻求迹距离最大的组合，以促进量子机器学习模型的训练。随后，我们将AdaBoost方法应用于每一个二元分类器以增强可训练性，从而使整个多类别任务获得更优的训练性能。借助量子机器学习模型强大的泛化能力，整体分类器在多类别任务上保持出色的预测性能。在此，TTA分类器在量子机器学习模型中实现了累计电路深度最高约10,000层、参数规模达0.2M，避免了单一量子机器学习模型在多类别任务上所呈现的“U形曲线” [21]，并展示了大规模量子机器学习的能力。

为了直观比较，我们在图像和合成基准上评估 TTA。在 MNIST 数据集上，我们将其与以下方法进行比较：(1) 单一量子多分类分类器，(2) 量子多分类 AdaBoost 分类器，(3) 使用按位 AdaBoost 的量子分类器，(4) 经典多分类 ResNet50，以及 (5) 经典多分类 ViT-Small。在合成数据集上，我们还进一步与一对一和一对其余方法进行比较。最终结果汇总于表 I 中。表中突出显示了最佳训练准确率、最佳测试准确率，以及达到最佳测试准确率时的最小模型参数量。综合这些指标，与其他量子和经典方法相比，TTA 分类器始终以最少的参数实现最高的预测准确率。

II. 预备知识

本节简要概述本研究的关键基础。在第 IIA 小节中，我们介绍监督量子机器学习的设定。在第 IIB 小节中，我们总结了作为整个流程构建模块的二分类AdaBoost算法。

A. 监督量子机器学习

在监督量子机器学习中，我们用 $\mathcal{X}$ 表示输入空间，用 $\mathcal{Y}$ 表示标签空间，并令 $\mathcal{D}$ 为定义在 $\mathcal{X}$ 和 $\mathcal{Y}$ 上的未知联合分布。输入空间 $\mathcal{X}$ 统一定义为量子态密度矩阵空间：对于原生量子数据，量子态 ρ 直接属于 $\mathcal{X}$；对于经典数据 $\boldsymbol{x}$，我们通过编码映射准备其对应的量子态 $\rho(\boldsymbol{x}) \in \mathcal{X}$。训练集 $S = \{(\rho^{(m)}, y^{(m)})\}_{m=1}^{M}$ 包含 M 个样本，每个样本均从 $\mathcal{X} \times \mathcal{Y}$ 中独立抽取。单个量子机器学习模型的输出定义为可观测量 O 在经参数化量子电路 $U(\boldsymbol{\theta})$ 演化后的状态上的期望值：$h(\rho, \boldsymbol{\theta}) = \mathrm{Tr}\left[OU(\boldsymbol{\theta})\rho U^{\dagger}(\boldsymbol{\theta})\right]$。在训练集 S 上学到的假设为 $h_S(\rho) = h(\rho, \boldsymbol{\theta}^*)$，其中 $\boldsymbol{\theta}^*$ 是训练过程中选定的参数。本文选择第一个量子比特上的泡利 Z 算符作为可观测量，记作 Z_1。参数化量子电路 $U(\boldsymbol{\theta})$ 的结构如图 2 所示。

在训练过程中，我们通常采用可微损失函数 L 作为优化目标，以指导模型参数的学习。为了评估模型的预测性能，我们使用逐样本风险函数 $r(\cdot,\ \cdot)$ 来衡量模型预测与真实标签之间的差异，并将假设 h_S 在数据集 $S' = \{(\rho'^{(m)},\ y'^{(m)})\}_{m=1}^{M}$ 上的经验误差定义为

$$\widehat{R}_{S'}(h_S) = \frac{1}{M}\sum_{m=1}^{M} r(h_S(\rho'^{(m)}), y'^{(m)}).$$

当 $S' = S$ 时，这就是训练误差，而 $1 - R_S(h_S)$ 是训练准确率；当数据集 S' 包含从与训练集 S 相同的分布中抽取的完全未见数据时，我们将 $R_{S'}(h_S)$ 称为测试误差。在训练集 S 上学习得到的假设 h_S 的预测误差

表 I. 各种方法在基准数据集上的性能比较（均值 ± 标准差）。M 表示 1，000，000。

数据集	模型 / 方法	训练准确率	测试准确率	参数数量
MNIST	单分类器（量子，无聚合）	89.05±0.36%	88.73±0.43%	0.00048±0.00000 M
	多类AdaBoost（量子，聚合）	96.77±0.20%	96.05±0.16%	0.24480±0.00000 M
	按位 AdaBoost（量子，聚合）	99.46±0.04%	97.30±0.07%	0.24000±0.00000 M
	ResNet50（经典，无集成）	**100.00±0.00%**	97.48±0.64%	22.0837±0.00000 M
	小型 ViT（经典，无集成）	**100.00±0.00%**	98.53±0.18%	25.5570±0.00000 M
	TTA（量子、集成、我们的方法）	**100.00±0.00%**	**98.71±0.11%**	**0.23433±0.01269 M**
合成	一对一（量子，聚合）	75.75±2.17%	76.52±2.04%	0.03607±0.00353 M
	一对其余（量子，聚合）	**100.00±0.00%**	**100.00±0.00%**	0.05256±0.00390 M
	TTA（量子、集成、我们的方法）	**100.00±0.00%**	**100.00±0.00%**	**0.02405±0.00255 M**

分布 $\mathcal{D}$ 定义为:

$$R(h_S) = \mathop{\mathbb{E}}_{(\rho,y)\sim\mathcal{D}}[r(h_S(\rho), y)].$$

预测误差无法直接测量；在实践中，通常使用测试误差作为预测误差的替代指标。机器学习的根本目标是确保模型在训练数据上表现良好时，在未见数据上也能表现良好，这一性质称为泛化能力。

B. 二分类AdaBoost

二分类AdaBoost（见算法 1）是一种自适应提升算法，通过迭代训练多个弱分类器并将其组合成一个强分类器来提升训练性能，并具有理论训练误差保证。在训练集 t上训练第 $S = \{(\rho^{(m)}, y^{(m)})\}_{m=1}^{M}$个基分类器时，样本 $(\rho^{(m)}, y^{(m)})$ 的权重为 $\boldsymbol{w}_t(m)$。我们采用合页损失 [22]，使第 t个基分类器的预测结果更接近标签:

$$\mathcal{L}_t(\boldsymbol{\theta}; S) = \sum_{m=1}^{M} \boldsymbol{w}_t(m) \max\left\{0, 1 - y^{(m)} h(\rho^{(m)}, \boldsymbol{\theta})\right\}. \tag{1}$$

训练完成后，我们得到第 t 个基分类器 $h_S^{(t)}(\rho) = h(\rho, \boldsymbol{\theta}^*)$，其中 $\boldsymbol{\theta}^*$ 是在训练过程中选定的参数。我们将 $h_S^{(t)}$ 的误差记录为 $\epsilon_t = \sum_{m=1}^{M} \boldsymbol{w}_t(m) \mathbb{1}\left(y^{(m)} \neq h_S^{(t)}(\rho^{(m)})\right)$，其中 $\mathbb{1}$ 是指示函数：条件为真时取值为 1，否则取值为 0。使用前 t 个基分类器，我们得到训练集 S上的当前训练误差上界：$\gamma_t = \exp\left[-2\sum_{i=1}^{t}(1/2 - \epsilon_i)^2\right]$. 由此可见，只要第 t 个基分类器的误差 $\epsilon_t < 1/2$，整体误差上界就会下降； ϵ_t 与 $1/2$ 之间的差距越大，整体误差上界下降得越快。此外，随着 AdaBoost 训练轮次 t 增加，误差上界会呈指数级下降。

算法 1： 二元量子 AdaBoost

输入： 训练数据 $S = \left\{\left(\rho^{(m)}, y^{(m)}\right)\right\}_{m=1}^{M}$，
其中 $\rho^{(m)}$ 是量子态（密度矩阵），$y^{(m)} \in \{-1, +1\}$；
最大提升轮数 $T \in \mathbb{N}$；
提前停止容差 $\tau = 0.005$。

输出： 最终集成 AdaBoost 二元分类器 $H_S(\cdot)$

初始化均匀权重：$\boldsymbol{w}_1(m) \leftarrow \frac{1}{M}$ 对于 $m = 1、\ldots\ldots、M$

对于 $t \leftarrow 1$ **到** T，**执行**

- 训练基分类器 $h_S^{(t)}$：关于 S 的 $\rho \mapsto \{-1, +1\}$ $\boldsymbol{w}_t$
- 计算加权误差: $\epsilon_t \leftarrow \sum_{m=1}^{M} \boldsymbol{w}_t(m) \cdot \mathbb{I}\left(y^{(m)} \neq h_S^{(t)}(\rho^{(m)})\right)$
- // 若效果不优于随机猜测，则终止
- **如果** $\epsilon_t \geq 1/2$ **则**
  - 中断; // 结束提升
- **end**
- 计算界: $\gamma_t \leftarrow \exp\left(-2\sum_{i=1}^{t}\left(\frac{1}{2} - \epsilon_i\right)^2\right)$
- **如果** $\gamma_t < \tau$ **则**
  - 中断; // 置信度足够
- **end**
- 更新系数: $\alpha_t \leftarrow \frac{1}{2}\log\frac{1-\epsilon_t}{\epsilon_t}$
- 计算归一化因子: $Z_t \leftarrow 2\sqrt{\epsilon_t(1-\epsilon_t)}$
- **对于** $m \leftarrow 1$ **到** M **执行**
  - 更新权重: $\boldsymbol{w}_{t+1}(m) \leftarrow [\boldsymbol{w}_t(m)$ 指数 $\left(-\alpha_t y^{(m)} h_S^{(t)}(\rho^{(m)})\right)]/Z_t$
- **end**

end

$H_S(\rho) = \operatorname{sgn}\left(\sum_{t=1}^{T} \frac{\alpha_t}{\|\boldsymbol{\alpha}\|_1} h_S^{(t)}(\rho)\right)$，其中 $\|\boldsymbol{\alpha}\|_1 = \sum_{t=1}^{T} \alpha_t$

返回 H_S

得到误差 ϵ_t后，我们可以推导出当前分类器的系数 α_t，并将训练样本的权重更新为 $\boldsymbol{w}_{t+1}$，提高在第 t轮中被错误分类的样本的权重，同时降低被正确分类的样本的权重，使得第（$t+1$）个基分类器专

注于此前各轮中被错误分类且权重较高的样本。最终的集成 AdaBoost 分类器定义为 $H_S(\rho) = \mathrm{sgn}(\sum_{t=1}^{T} \alpha_t h_S^{(t)}(\rho)/\|\boldsymbol{\alpha}\|_1)$，其中 T是 AdaBoost 的总轮数，$\boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_T]^\top$是由 T 个基分类器的系数组成的向量。使用 $\|\boldsymbol{\alpha}\|_1$ 进行归一化的原因是，这能确保sgn() 内的项取值范围为$[-1, 1]$ ，且是 $h_S^{(t)}$ 的凸组合。这保证了整个 AdaBoost 方法也具有与单个量子机器学习模型 [16]相同的泛化能力。

在AdaBoost训练过程中，有三个停止条件：(1) 当第 t个基分类器训练完成后，训练误差上界 γ_t降至0.005以下时，已达到预设目标，因此训练终止。(2) 当训练轮数达到预设的最大轮数时，训练结束；(3) 当第 t 个基分类器的误差 $\epsilon_t > 1/2$时，由于无法满足弱基分类器的基本要求，训练结束。我们将条件(1)称为收敛情况，将条件(2)和(3)称为不收敛情况。

III. Trace DistanceBinaryTree

本节介绍TTA框架。第IIIA 小节介绍基于迹距离最大化的数据集划分策略。第IIIB 小节描述训练和推理流程。

A. 数据集划分

在量子二分类问题中，当两个类别的平均量子态之间的迹距离非常小时，量子机器学习模型很难正确区分它们 [20]。此外，另一个直观认识是，训练集中的数据越多，量子机器学习模型就越难以正确分类所有样本。极端情况下，对两个样本进行分类显然比对数千个样本进行分类容易。

受这些观察的启发，我们旨在构建二元成员分类器时实现两个目标：(1) 尽可能增大正类和负类平均态之间的迹距离；(2) 对于数据集中正类和负类平均态之间迹距离较小的情况，尽量减少两个类别中的样本数量。

基于这两项原则，我们构建了如图 1 所示的 TTA分类器。对于一个 K 类训练集，且各类别的样本数量大致均衡，我们首先将这 K 个类别划分为大小大致相等的两组（即，当 $K/2$ 为偶数时每组包含 K 个类别，或者当 $\lfloor K/2 \rfloor$和 $\lceil K/2 \rceil$个类别时，当 K 为奇数）。划分的目标是最大化所得两组的平均量子态之间的迹距离。随后，将这一过程

算法2： 基于迹距离

输入： 平衡数据集 $S = \{(\rho^{(m)}, y^{(m)})\}_{m=1}^{M}$，包含 $|K| \geqslant 2$ 个类别的标签集 K

输出： 所有二分划分的列表（配对数据集划分）

在所有内部节点处

RecurSplit(S_{cur}, K_{cur})**:**

- **如果** $|K_{\mathrm{cur}}| \leq 1$ **则**
  - **返回** $\varnothing$; // 不再进行处理
- **否则如果** $|K_{\mathrm{cur}}| = 2$ **则**
  - 令 k_1, k_2 为 K_{cur}中的两个类别;
  - $S_- \leftarrow \{(\rho^{(m)}, y^{(m)}) \in S_{\mathrm{cur}} : y^{(m)} = k_1\}$;
  - $S_+ \leftarrow \{(\rho^{(m)}, y^{(m)}) \in S_{\mathrm{cur}} : y^{(m)} = k_2\}$;
  - **返回** $\{(S_-, S_+)\}$;
- **end**
- **else**
  - // 步骤1： 计算每个类别的平均量子态
    每个类别的状态
  - **对每个**类别 k 属于 K_{cur} **do**
    - 计算 $\rho_k = 1/M_k \sum_{m_k=1}^{M_k} \rho^{(m_k)}$;
  - **end**
  - // 步骤 2： 寻找使.....最大化的二分划分
    标签集 K 中的迹距离 $_{\mathrm{cur}}$
  - $(K_-, K_+) \leftarrow$ MaxBinarySplit($\{\bar{\rho}_k\}_{k \in K_{\mathrm{cur}}}$);
  - // 步骤 3： 按划分后的类别
  - $S_- \leftarrow \left\{(\rho^{(m)}, y^{(m)}) \in S_{\mathrm{cur}} : y^{(m)} \in K_-\right\}$;
  - $S_+ \leftarrow \left\{(\rho^{(m)}, y^{(m)}) \in S_{\mathrm{cur}} : y^{(m)} \in K_+\right\}$;
  - // 步骤 4： 递归调用和输出
    本次划分的
  - $\mathcal{P}_- \leftarrow$ RecurSplit(S_-, K_-);
  - $\mathcal{P}_+ \leftarrow$ RecurSplit(S_+, K_+);
  - **return** $\{(S_-, S_+)\} \cup \mathcal{P}_- \cup \mathcal{P}_+$;
- **end**

主程序:

$\mathcal{P} \leftarrow$RecurSplit(S, K);

对于 $(S_-, S_+) \in \mathcal{P}$ **执行**

- S_-: $y^{(m)} \leftarrow -1$, S_+: $y^{(m)} \leftarrow +1$;

end

return $\mathcal{P}$

递归地应用于每个所得子集。详细算法见算法2。

具体而言，该算法会递归地对一个包含 K 个类别的数据集进行二分：每一步拆分后，都会递归处理两个子集，直到每个子集至多包含一个类别，并记录每次拆分。通过这种分解，迹距离最小的两个类别会尽可能地被推向叶节点。这样可以确保，在样本量较大的节点处，正类与负类之间的迹距离最大；而在正类与负类之间迹距离较小的节点处，样本数量则尽可能少。将这一方法与 AdaBoost 方法相结合，以提高每个基分类器的可训练性，我们最终可以在实验条件下实现 100% 的训练准确率。此外，

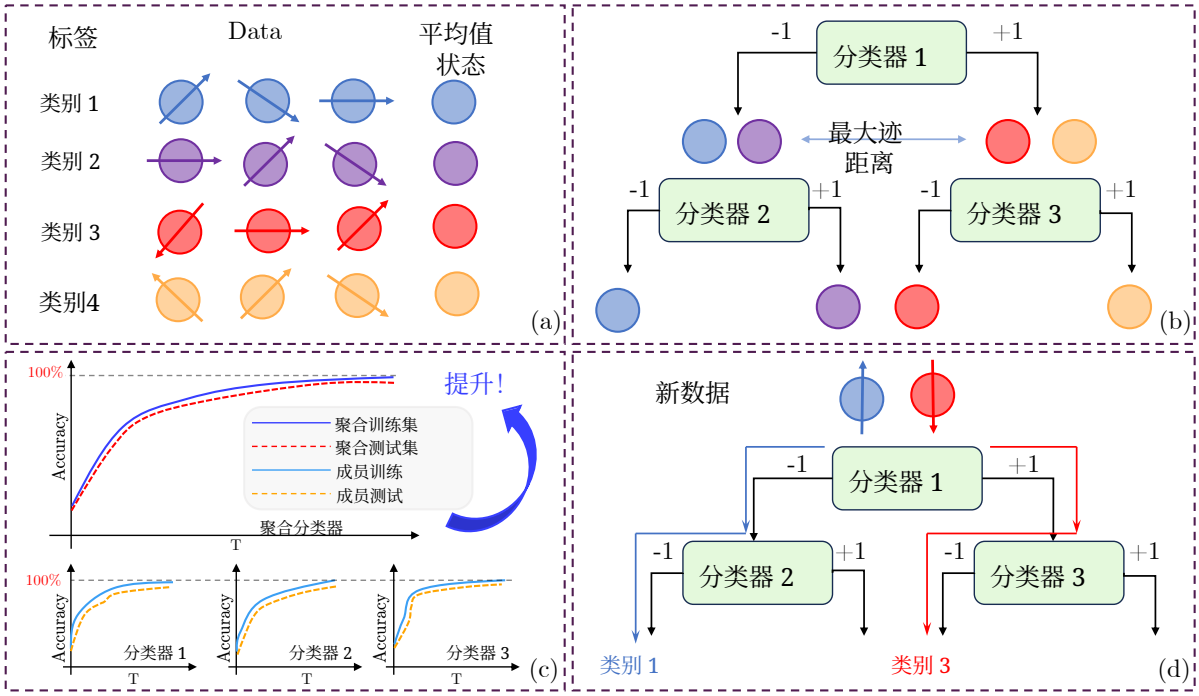

图 1. 迹距离二叉树 AdaBoost（TTA）多分类器示意图。(a) 不同类别的量子态数据及其平均态。(b) 通过选择使两个可能类别的平均态之间迹距离最大的划分，构建二叉数据集树；每个内部节点都包含一个二元成员分类器。(c) AdaBoost 将每个成员的训练准确率提升至接近 100%，并结合量子机器学习的泛化能力，得到近乎完美的聚合预测。(d) 推理沿着节点决策从根节点前进至叶节点，从而生成最终的类别标签。

由于量子机器学习模型具有很强的泛化能力，测试准确率也接近 100%。

为便于说明，来看一个具体示例：假设类别集合为 $\{1,2,3,4,5\}$，且已知所有成对平均态的迹距离。(1) 选择迹距离最大的划分（例如，$\{1,2,3\}$ 与 $\{4,5\}$），并记录该划分（S_{123}，S_{45}）。(2) 对 S_{123}（现在的 $K=3$）递归应用此过程；再次假设迹距离最大的划分为 $\{1,2\}$ 与 $\{3\}$，于是记录（S_{12}，S_3）。(3) 对 S_{45}（$K=2$）递归：由于只剩两个类别，记录（S_4，S_5），不再继续划分。(4) 对 S_{12}（$K=2$）递归：再次记录（S_1，S_2）并停止。(5) 对 S_3（$K=1$）递归：由于只剩一个类别，无需再进行划分或输出。最终，划分集合为

$$\{(S_{123},S_{45}),(S_{12},S_3),(S_4,S_5),(S_1,S_2)\}$$

总计进行 $K-1=4$ 次划分。对于列表中的每个子集，都会分配新的二元标签：按照惯例，索引左侧所示的类别被分配为 -1（负类），右侧的类别则被分配为 $+1$（正类）。例如，在 (S_{123},S_{45}) 中，类别 1、2 和 3 被分配为 -1，而类别 4 和 5 被分配为 $+1$。

该算法的一个重要组成部分是 MaxBinarySplit 过程，它用于在所有可能的组合中识别具有最大迹距离的二元划分。当类别数量 K 较少时，可以采用蛮力法，枚举所有 $\binom{K}{\lfloor K/2 \rfloor}$ 可能的组合，以找到具有最大迹距离的划分。然而，随着 K 的增加，该方法的计算复杂度变得难以承受。为此，我们在附录B 中提出了一种贪心启发式算法，该算法将 K 个类别划分为 $\mathcal{O}(K^2 \log K)$ 次操作。

B. 训练与推理

获得划分列表 $\mathcal{P}$ 后，我们可以并行训练 $\mathcal{P}$ 中所有二分类子集上的 Ad-aBoost 分类器，最终形成一个多类别分类器。对数据集进行划分会生成二叉树结构，相应地，对这些二分类子集进行分类的分类器也构成一棵二叉树。在推理过程中，未标记样本从分类器树的根节点出发，并根据各节点的输出向下遍历：如果预测标签为 -1，样本就进入左子分类器；如果预测标签为 $+1$，则进入右侧分类器。该过程持续进行，直到样本到达叶节点，此时将与该叶节点对应的类别作为该样本的预测类别。

如图所示2，由 TTA 构建的整体多分类器称为聚合分类器，而

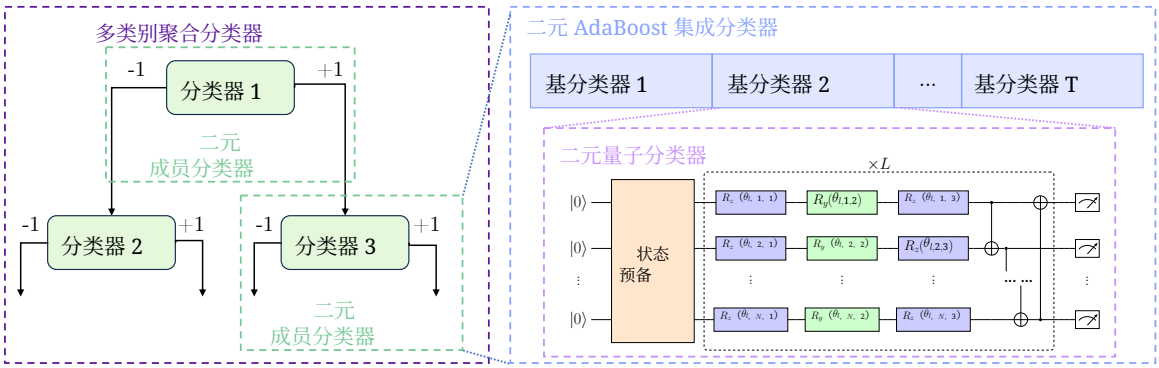

图 2. 多分类聚合分类器示意图，该分类器由二分类AdaBoost成员分类器集成构建。每个成员都是由二元量子基分类器组成的 AdaBoost 集成：数据被编码或制备为量子态，经由浅层参数化电路（由单量子比特 R_z、 R_y 和 R_z 旋转门以及环形拓扑 CNOT 门构成的层）演化，并进行测量（在第一个量子比特上测量泡利 Z），以得到二元输出。

二元树各节点上的二元分类器称为成员分类器。每个成员分类器本质上都是一种集成分类器，它使用二元 AdaBoost 方法组合多个弱基分类器。每个基分类器都由一个包含 N 个量子比特、深度为 L 层的量子电路实现。每层由单量子比特 R_z、 R_y和 R_z旋转门，以及按环形拓扑排列的 CNOT 门组成。通过测量算符 Z_1 得到输出，该算符是作用于第一个量子比特的泡利Z算符，由此构成一个二元分类模型。对新数据进行预测时，数据会输入多个量子电路；随后，AdaBoost 方法对各量子电路的输出进行集成，生成成员分类器的预测结果。最后，将所有成员分类器的预测结果聚合起来，得到给定输入的最终多分类预测。

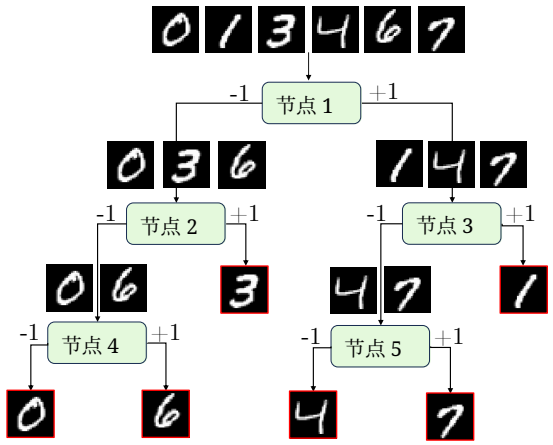

图 3. MNIST 数据集的迹距离树。每个节点都是一个二元成员分类器，红色节点表示叶节点。

IV. 经典数据上的分类

本节评估 TTA分类器在经典数据集上的表现。小节 IVA 介绍 MNIST 数据集和实验设置。小节 IVB 展示了与各种量子和经典分类器的性能比较。小节 IVC 详细分析了 TTA分类器的行为。小节 IVD 详细介绍所比较的方法。

A. MNIST 数据集

在本实验中，我们使用 MNIST 数据集的一个子集 [23]，其中包含数字 0、3、6、1、4 和 7，构成一个 6 类分类问题。我们将 MNIST 图像调整为 16×16 像素。训练集每类包含 2000 个样本，总计 12000 个训练样本，而测试集包含6093个样本。我们通过最大化负类与正类平均态之间的迹距离来构建二叉树，构建结果如图3所示。树中的每个节点代表一个成员分类器，各节点对应的负类与正类平均态之间的迹距离值列于表II中。

表 II. MNIST TTA树中各节点的负类（左）与正类（右）平均量子态之间的迹距离。

Node	类别	迹距离；	训练样本
节点 1	{0,3,6} 与 {1,4,7}	0.5672	12000
节点 2	{0,6} 对 {3}	0.5892	6000
节点 3	{4,7} 对 {1}	0.7366	6000
节点 4	{0} 对 {6}	0.6109	4000
节点 5	{4} 对 {7}	0.5593	4000

为训练TTA分类器，我们将调整尺寸后的图像振幅编码为一个8量子比特量子态，并使用图2所示的参数化量子电路进行处理，电路深度为 $L = 20$ 层，每层包含24个可训练参数，因此每个基分类器共有480个参数。每个基分类器量子电路的初始参数均独立采样自标准高斯分布。我们使用第一个量子比特上的泡利Z算符作为测量算符，并采用式（1）定义的合页损失作为损失函数。训练采用批量大小为200 ，使用Adam优化器 [24] ，学习率为0.005。我们将最大训练轮次限制为100，并采用早停策略：当训练误差低于0.5且连续10个训练轮次未再下降时停止训练，并将早停前达到最小训练误差时对应的参数选为最终参数。后续章节将展示该早停策略的有效性及其降低训练成本的能力。

B. 性能

为验证TTA分类器的优越性，我们在所选MNIST子集上将其与多种量子分类器和经典分类器进行了全面比较，结果见图4。

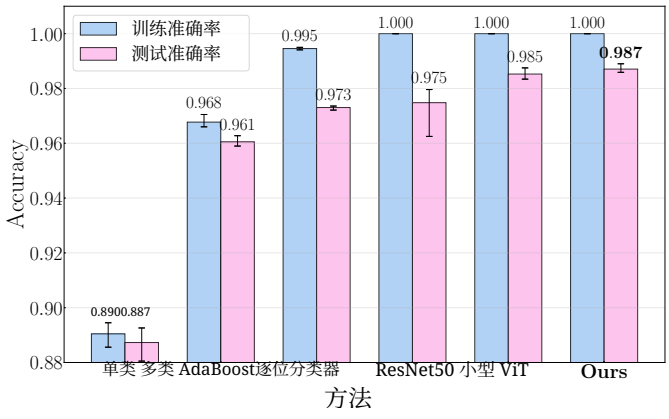

图 4. TTA分类器与其他方法在所选 MNIST子集上的训练和测试准确率比较。其他方法包括单量子多类分类器、量子多类 AdaBoost 分类器、量子按位聚合分类器，以及 ResNet50分类器和小型ViT分类器等经典神经网络。与所有其他方法相比，TTA分类器的训练和预测性能最佳。误差条上的数字表示 5 次运行的平均值，误差条表示这 5 次运行中的最大值和最小值。

1. 准确率比较

我们将TTA分类器与以下量子方法进行了比较：（1）单一量子多类分类器 [25]，（2）基于量子多类模型的多类 AdaBoost 分类器 [17, 26], 以及（3）一种按位聚合分类器，它使用按位编码将多分类问题转化为多个二元分类问题（每个二分类问题均使用二分类 AdaBoost以增强可训练性）。实验结果表明，尽管这两种Multi
与单一量子多分类器相比，AdaBoost集成分类器和按位聚合分类器的训练性能有所提升；但只有 TTA分类器达到了 100% 的训练准确率，因此测试准确率表现最佳。

我们还将 TTA分类器与经典深度学习中的代表性模型进行了比较，包括：(1) ResNet [27]（残差网络，可有效解决深度网络中的梯度消失问题）；(2) ViT [28]（视觉Transformer，这一里程碑式模型成功将Transformer架构应用于计算机视觉任务）。尽管这些经典分类器也能达到 100% 的训练准确率，但得益于量子机器学习出色的泛化能力，TTA分类器在测试准确率方面表现最佳。

2. 效率比较

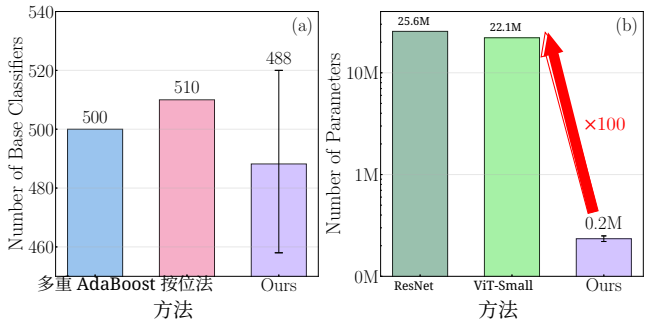

图 5. (a) 不同量子方法所使用的基分类器数量比较。(b) TTA分类器与经典神经网络的参数量比较。误差条上的数字表示 5 次运行的平均值，误差条表示最大值和最小值。

为进一步评估 TTA 的效率，我们从两个维度进行了比较分析：(1) 相较于量子分类器，比较 TTA 与其他量子方法所使用的 AdaBoost 基分类器总数；(2) 相较于经典算法，比较 TTA 与经典分类器的参数量。图5中的结果表明，

与按位聚合分类器和多类 AdaBoost 分类器相比，TTA 平均使用的基分类器数量最少，同时所需参数量比经典模型少约两个数量级。具体而言，TTA分类器总共使用约 500 个基分类器，每个基分类器包含 480 个参数（20 层， × 24 每层参数），总参数量约为 0.2M。然而，在实际中直接训练一个具有 10,000 层的 8 量子比特量子电路并不可行，因为会受到贫瘠高原和噪声效应的影响，这进一步凸显了通过迹距离二叉树构造来分配模型容量的重要性及其优势。

C. TTA分类器的详细信息

1. 随着基分类器数量增加，准确率趋近于 *100%*

我们分析了图 6 中随着基分类器数量增加，各成员分类器和聚合分类器的训练准确率与测试准确率的变化。由于 AdaBoost 算法的特点，随着基分类器数量增加，各成员分类器的训练误差呈指数级下降，训练准确率趋近于 1。这进而使聚合分类器的训练准确率呈指数级上升并达到 1。得益于底层量子机器学习模型优异的泛化性能，以及 AdaBoost 保持泛化特性的能力，测试准确率始终接近训练准确率；当训练准确率达到 1 时，测试准确率也非常接近 1。

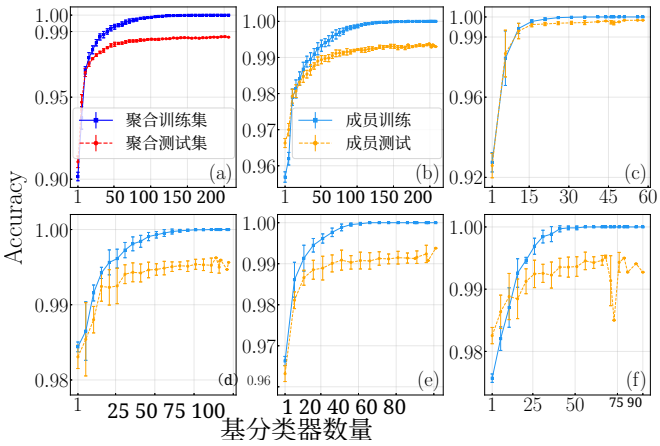

图 6. (a) 随着基分类器数量增加，聚合分类器的训练和测试准确率趋近于 1。(b)-(f) 随着基分类器数量增加，各成员分类器的训练和测试准确率趋近于 1。曲线表示 5 次运行结果的平均值，误差条表示 5 次运行中的最大值和最小值。

在实验设计中，我们测量训练和测试准确率：从 1 个基分类器开始，此后每增加 5 个基分类器（即在 1、6、11、16.....个时）测量一次每个成员分类器的准确率。此外，我们还会在每个成员分类器收敛时，进行一次最终测量，记录训练和测试准确率。由于所有实验均采用独立训练并重复 5 次，基分类器误差的差异会导致各次运行中收敛所需的基分类器数量有所不同。对于聚合分类器中，我们在这些固定的测量点聚合所有成员分类器的结果。当某个成员分类器比其他分类器更早收敛时，后续聚合测量将使用其最终结果。例如，如果成员分类器 A 在 21 个基分类器时收敛，而成员分类器 B 在 36 个基分类器时收敛，那么测量点 26 的聚合结果将使用 A 的最终结果（在 21 时）与 B 在 26 时的结果相结合。

2. 弱基分类器产生强集成

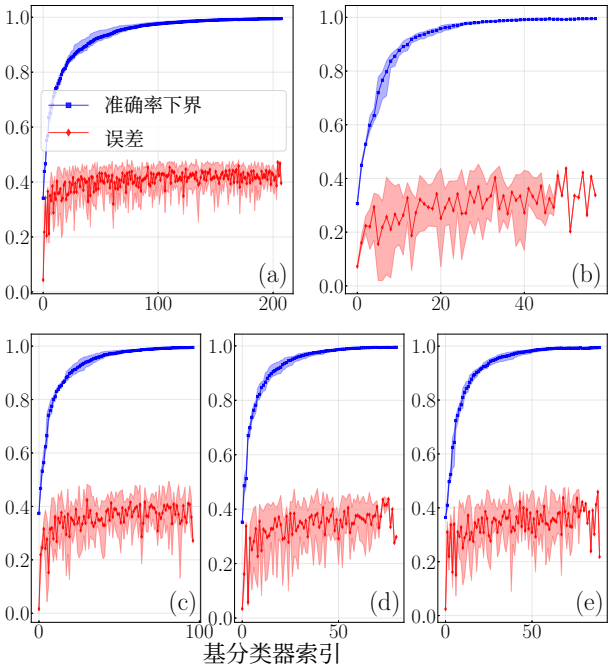

图 7. (a)-(e) 对应于图 6(b)-(f) 中的五个成员分类器，展示了每个基分类器的误差，以及整个 AdaBoost 成员分类器截至该基分类器的准确率下界。实线表示 5 次独立运行的平均结果，阴影区域表示这 5 次独立运行中的最大值和最小值。

我们进一步分析了图中每个基分类器的误差以及 AdaBoost 成员分类器的训练准确率下界。7红色曲线表明，各基分类器在重加权训练集上的误差稳定在约 0.4。尽管如此，训练准确率下界（按 1 减去 AdaBoost 保证的误差上

界计算）仍以指数速度趋近于 1。所有成员分类器均成功达到训练准确率下界超过 0.995 的收敛条件。

3. 迹距离和样本量影响收敛

图8 展示了在 5 次独立运行中，每个成员分类器达到收敛所需的 AdaBoost 基分类器数量。如表 II所示，根节点（节点 1）的迹距离最小（0.5672），训练集最大（12,000 个样本），因此达到收敛所需的基分类器数量最多。相比之下，节点 3 的迹距离最大（0.7366），分类任务更容易，因此所需的基分类器数量更少。比较节点 1 和节点 5，尽管它们的迹距离相近（0.5672 对 0.5593），但由于节点 5 的训练集较小（4,000 对 12,000 个样本），所需的基分类器显著更少。这些结果验证了迹距离和样本量都会影响收敛速度，并支持我们基于迹距离构建二叉树的策略。

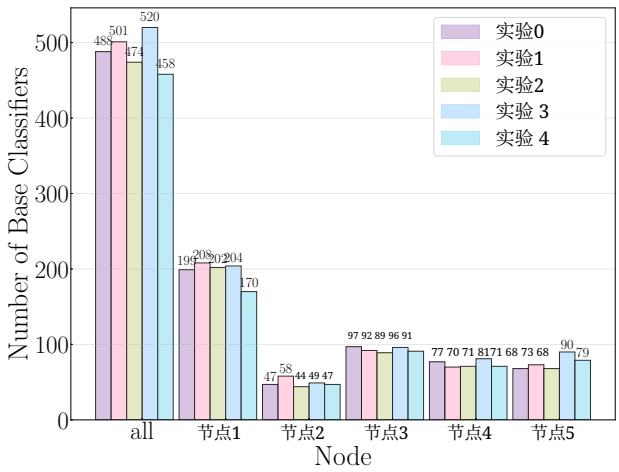

图 8. 5 个节点（成员分类器）在 5 次独立运行中达到误差上界小于 0.005 所需的基分类器数量。

D. 对比方法详情

1. 按位聚合分类器

按位聚合方法为每个类别分配唯一的二进制编码，并针对每个位位置训练一个分类器，从而将多类分类转化为二分类。对于一个有 K 个类别的问题，类别标签被编码为长度为 $\lceil \log_2(K) \rceil$ 的二进制数。随后，针对每个位位置分别训练一个二元分类器，其中该位为 0 的样本构成负类，该位为 1 的样本构成正类。为确保与 TTA分类器进行公平比较，我们还对每个成员分类器采用二分类 AdaBoost，构建一个更易训练的聚合分类器。

表 III. 按位聚合分类器中每个成员分类器的负类（左）和正类（右）平均量子态之间的迹距离。

位类别	迹距离；训练样本	
位 1 {0,1,3} 对比 {4,6,7}	0.5220 12000	
位 2 {0,1,4} 对比 {3,6,7}	0.4184	12000
位 3 {0,4,6} 对比 {1,3,7}	0.5500	12000

具体而言，在选定的 MNIST 子集中，由于最大的标签为 7，因此需要使用长度为 3 的二进制字符串对数据集中的标签进行编码，从而将 6 类问题转换为 3 个二分类问题。以 la-

以标签 6 为例，其二进制表示为 110。按照从最高位（左）到最低位（右）的位顺序，标签 6 在第一个和第二个成员分类器中属于正类，在第三个成员分类器中属于负类。三个成员分类器中正类与负类平均态之间的迹距离如表III所示。可以看出，在这三种划分情形下，平均态的迹距离都小于 TTA 划分方法中的距离，这表明这些二分类任务平均而言更难。

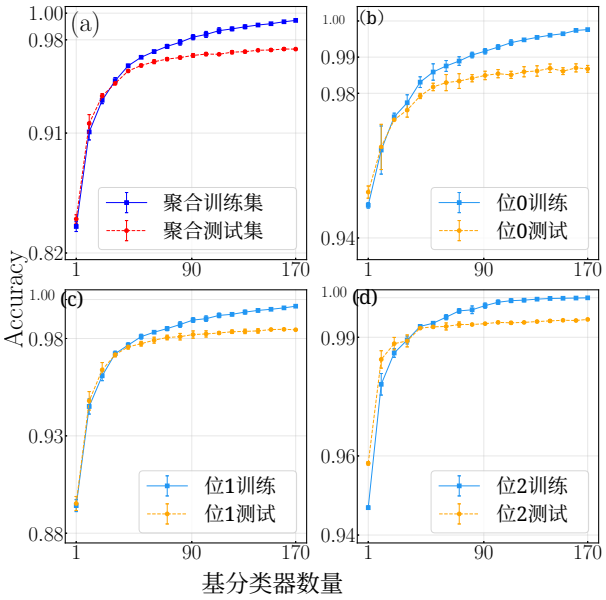

图9。（a）按位聚合分类器的训练和测试准确率。（b）—（d）每个按位成员分类器的训练和测试准确率。曲线表示5次独立运行的平均值，误差条表示5次独立运行中的最大值和最小值。

为确保比较公平，我们将按位聚合的基分类器数量设置为与 TTA分类器大致相当；后者平均使用 488 个基分类器。我们将每个成员分类器的 AdaBoost 最大轮数设为 170，因此按位聚合总共使用 510 个基分类器，略多于 TTA分类器。

然而，值得注意的是，按位聚合中的每个基分类器都必须使用完整的 12,000 个样本进行训练；而 TTA分类器仅使用 12,000 个样本训练节点 1，节点 2–5 则使用规模较小的子集进行训练，样本数为 4,000 至 6,000（见表 II）。这一差异使按位聚合的训练难度显著增加。其他所有训练设置均与 TTA分类器相同。如图 9所示，尽管随着基分类器数量增加，训练性能有所提升，但成员分类器仍难以达到较高的训练准确率（100%）。这是因为每个成员分类器都必须处理 12,000 个样本，且未依据迹距离进行合理划分。因此，尽管底层二分类模型原则上具有良好的泛化能力，按位聚合的测试性能仍因这些训练困难而较差。

2. 单量子多类分类器

对于在选定 MNIST子集上训练的单量子多类分类器，我们为每个类别 k 设计测量算子 $\Pi_k = |k\rangle\langle k|$，以处理多类分类。模型参数通过在数据集 $S = \{(\rho^{(m)}, y^{(m)})\}_{m=1}^{M}$ 上最小化交叉熵损失进行优化，损失函数定义如下：

$$\begin{aligned}
&\mathcal{L}(S;\boldsymbol{\theta}) = \frac{1}{M}\sum_{m=1}^{M} \ell(\rho^{(m)}, y^{(m)};\boldsymbol{\theta}),\\
&\ell(\rho^{(m)}, y^{(m)};\boldsymbol{\theta}) = -\sum_{k=1}^{K} \boldsymbol{y}_k^{(m)} \ln\left(\frac{e^{h_k(\rho^{(m)},\boldsymbol{\theta})}}{\sum_{j=1}^{K} e^{h_j(\rho^{(m)},\boldsymbol{\theta})}}\right),\\
&h_k(\rho^{(m)},\boldsymbol{\theta}) = \mathrm{Tr}\left[\Pi_k U(\boldsymbol{\theta})\rho^{(m)}U^{\dagger}(\boldsymbol{\theta})\right],
\end{aligned} \tag{2}$$

其中，$\boldsymbol{y}_k^{(m)}$ 表示与样本 m的真实标签 $y^{(m)}$ 对应的独热编码向量中的第 k个元素。具体而言，当 $y^{(m)} = l$时，即样本 m 属于类别 l时，有 $\boldsymbol{y}_l^{(m)} = 1$，其余位置均为 0。

为探究单量子多类分类器仅通过增加训练轮次能否显著提高训练准确率，我们将训练轮次设为1000，并关闭早停。为确保实验比较公平，优化器、批量大小和学习率等其他设置均与TTA分类器保持一致。实验结果如图所示。

10，从中可以观察到，在 5 次独立运行中，即使

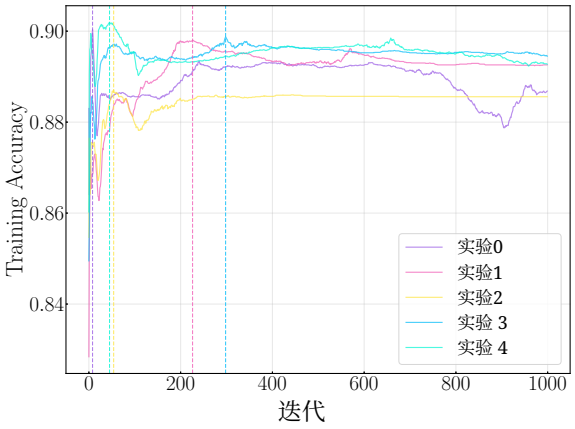

图 10. 用于多类分类的单个量子机器学习模型的训练准确率。竖线表示在 1000 个训练轮次中取得的最高训练准确率结果。

经过 1000 个训练轮次，取得最佳训练准确率的训练轮次也出现在前 400 轮内，这表明单纯增加训练迭代次数无法显著提高单量子多类分类器的训练准确率。

此外，由于 TTA分类器 总共使用约 10,000 个量子电路层，而过高的表达能力会导致贫瘠高原，因此训练一个具有 10,000 层的单个多分类量子机器学习分类器并不切实际。

3. 多类 *AdaBoost* 分类器

先前的工作 [17, 26]已将多类 AdaBoost 引入量子机器学习，以提高可训练性。然而，与二分类AdaBoost不同，多类 AdaBoost 方法在理论上并不能保证：只要每个基分类器在重加权训练集上的误差率低于某个阈值，整体集成 AdaBoost 的训练误差就会逐渐收敛至 0。因此，在多类 AdaBoost 中单纯增加基分类器的数量，并不一定能达到 TTA分类器 所实现的 100% 训练准确率。

训练第 t个多类别基分类器时，我们在加权训练集 $S = \{(\rho^{(m)}, y^{(m)})\}_{m=1}^{M}$ 上采用交叉熵损失函数，如下所示：

$$L(S,\boldsymbol{\theta}) = \sum_{m=1}^{M} \boldsymbol{w}_t(m)\ell(\rho^{(m)}, y^{(m)}, \boldsymbol{\theta})$$

其中，$\boldsymbol{w}_t(m)$是多类别 Ad- aBoost 在训练第 m个基分类器时分配给第 t个样本的权重，而 $\ell(\rho^{(m)}, y^{(m)}, \boldsymbol{\theta})$ 的定义与公式（2）相同。

同样，为确保比较公平，我们将多类AdaBoost配置为使用数量相近的基

分类器，与采用平均488个基分类器的TTA分类器相当。我们将整个多类AdaBoost的基分类器数量上限设为500，这超过了TTA分类器所使用基分类器的平均数量。此外，我们对早停策略作如下调整：当某个基分类器的训练误差小于 $5/6$，且连续10个训练轮次的训练误差均未下降时，就停止训练。这是因为多类AdaBoost只要求基分类器的训练误差小于 $(K-1)/K$(详见附录B ）。除优化器、批量大小和学习率等设置外，其余设置均与TTA分类器保持一致，以确保实验比较公平。

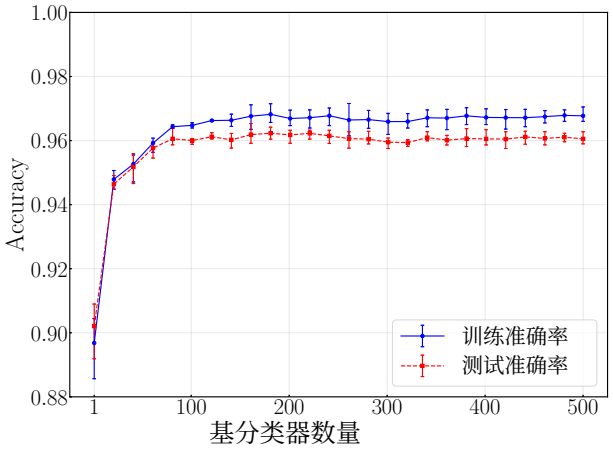

图11. 使用多类AdaBoost得到的训练和测试结果。可以观察到，各分类器的训练准确率均未达到1，而测试准确率则始终接近训练准确率。曲线表示5次独立运行结果的平均值，误差条表示这5次独立运行中的最大值和最小值。

实验结果如图11所示。随着基分类器数量增加，训练准确率最初有所提升，但超过某一阈值后便趋于平稳；与此同时，测试准确率始终接近训练准确率，这证实了量子机器学习模型具有很强的泛化能力。这一局限源于多类AdaBoost缺乏理论保证：与二分类AdaBoost不同，后者的训练误差会随着基分类器数量增加而呈指数下降，而多类AdaBoost并无此类保证，因此无法仅通过增加基分类器来进一步提升可训练性。

4. 经典神经网络

为了评估 TTA分类器，我们将其与两种经典深度学习模型进行比较：ResNet 和视觉 Transformer（ViT）。由于我们的数据集由 16×16 单通道图像组成，因此我们首先将其插值为 224×224图像，这在经典深度学习中是一种标准做法

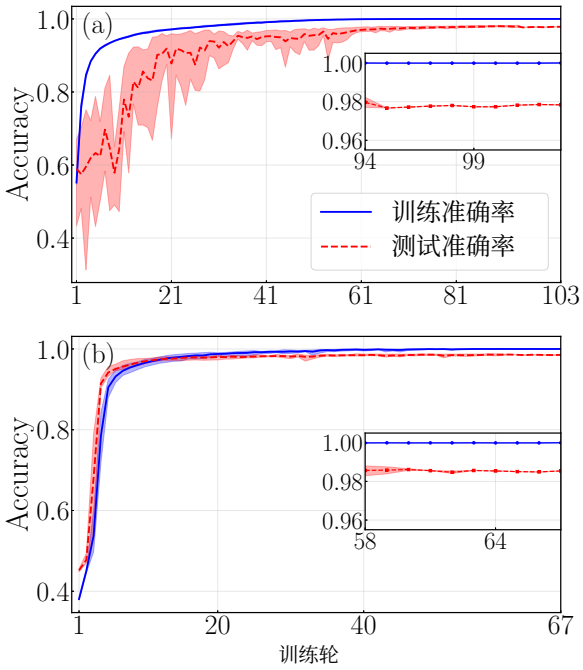

图12. (a) ResNet和(b) ViT的训练与测试结果。曲线表示5次独立运行结果的平均值，阴影表示这5次独立运行中的最大值和最小值。插图展示了最后10个训练轮次。

。然后，我们只需修改这两种模型的输入通道数。对于ResNet，我们将第一个卷积层的输入通道数从 3（RGB 图像的默认值）改为 1，同时保持其他所有架构参数不变。对于 ViT，我们同样修改了图像块嵌入层，使其接受单通道输入，而非默认的 3 通道 RGB 输入，其他所有架构参数均保持不变。

在训练过程中，我们采用交叉熵损失函数作为优化目标，并选择随机梯度下降优化器 [29]进行参数更新，主要是因为在实践中，随机梯度下降通常能带来更优的模型泛化性能。两个神经网络均从头开始训练，未使用预训练权重。在训练配置方面，我们将最大训练轮次设为 1000，并制定了明确的早停策略：一旦模型在训练集上达到 100% 的准确率，便终止训练。为确保实验比较公平，其他超参数设置（如批量大小、学习率等）均与 TTA 分类器完全一致。所有实验均随机重复 5 次。

实验结果如图所示。12从结果中可以看出，尽管经典神经网络相对容易在训练集上达到 100% 的准确率，但由于其泛化能力略逊于量子机器学习模型，最终分类性能也略微

逊于我们提出的 TTA分类器。由于分类器在数据集上达到 100% 的准确率后便停止训练，因此图表根据实验中的最大训练轮次数进行了对齐。此外，我们在附录中解释了C为何选择 SGD优化器而非 Adam优化器；在相同学习率下，ViT 使用 Adam优化器的表现不如使用 SGD。

V. 量子数据分类

本节展示 TTA 分类器在量子数据上的有效性。第VA 小节介绍 ANNNI 模型和量子相分类任务。第VB 小节展示性能结果。第VC 小节对训练和测试准确率进行可视化。

A. ANNNI模型

我们考虑一个 ANNNI 模型 [30] ，其包含 N 个量子比特，哈密顿量定义为

$$H = -\left(\sum_{i=1}^{N-1} X_i X_{i+1} - \kappa \sum_{i=1}^{N-2} X_i X_{i+2} + h \sum_{i=1}^{N} Z_i\right).$$

根据 κ 和 h的取值，该哈密顿量的基态呈现出三种不同的量子相：（i）反相（标签为 0）、（ii）铁磁相（标签为 1）和（iii）顺磁相（标签为 2）。相边界由两条临界线确定： $h_I(\kappa) \approx \frac{1-\kappa}{\kappa}\left(1-\sqrt{\frac{1-3\kappa+4\kappa^2}{1-\kappa}}\right)$ 和$h_C(\kappa) \approx 1.05\sqrt{(\kappa-0.5)(\kappa-0.1)}$。在我们的实验中，我们使用这些相标签对基态进行分类。

6量子比特相图及对应的迹距离树如图所示。 13。我们从每个相中随机选取2000个量子态作为训练集，另选取1000个量子态作为测试集，因此总计有6000个训练样本和3000个测试样本。各节点的类别分配、正类与负类平均状态之间的迹距离，以及所涉及的训练样本数量汇总于表 IV。

ANNNI模型上的TTA分类器训练流程遵循第 IVA 节中针对经典数据所述的相同方案。具体而言，我们使用图 2 所示的参数化量子电路架构，电路深度为 $L = 20$，采用第一个量子比特上的泡利Z算符作为观测量，以合页损失作为损失函数，批量大小设为200，使用学习率为0.005的Adam优化器，并采用相同的早停策略。所有实验均重复5次。

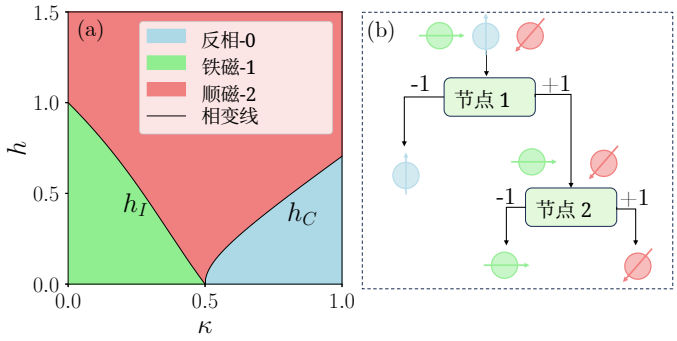

图13。（a）ANNNI模型的三个相；（b）ANNNI相数据集的迹距离树。对于节点1，将反铁磁相视为负类，而将铁磁相和顺磁相视为正类。对于节点2，将铁磁相视为

负类，而顺磁相被视为正类。

表 IV. 相位 TTA 树中各节点的负类（左）和正类（右）平均量子态之间的迹距离。

Node	类别	迹距离；	训练样本
节点 1 {0}	对比 {1,2}	0.8573	6000
节点2 {1}	与 {2}	0.7521	4000

B. 性能

1. 随着基分类器数量增加，准确率趋近于 *100%*

遵循第 IVC节中相同的测量策略，我们分析了基分类器数量增加时的训练准确率和测试准确率。如图 14所示，成员分类器和聚合分类器的训练准确率都趋近于 1；由于量子机器学习模型具有很强的泛化能力，测试准确率也始终接近训练准确率。

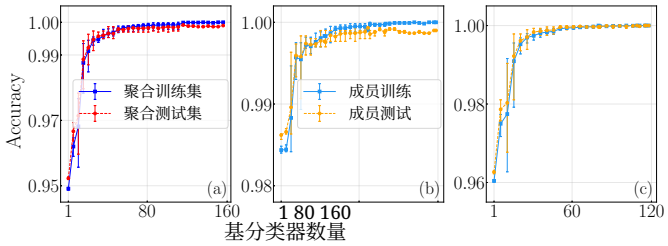

图 14. (a) 随着基分类器数量增加，训练和测试准确率均趋近于 1。(b)-(c) 单个成员分类器也呈现类似趋势。线条表示 5 次独立运行结果的平均值，误差条表示这 5 次独立运行结果中的最大值和最小值。

2. 相图上的近乎完美分类

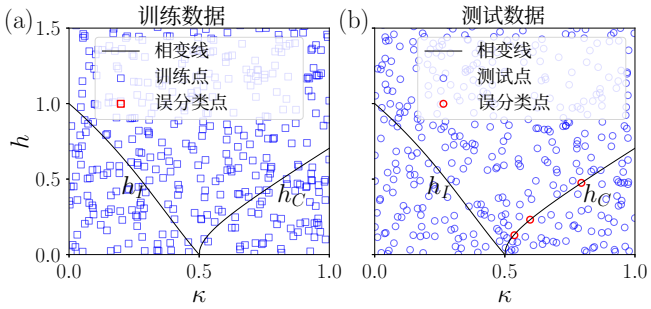

图 15. TTA 在量子相分类问题上的训练和预测结果。蓝色点表示正确分类的样本，红色点表示误分类的样本。在 (a) 中，我们成功地将全部训练数据（6000 个点）正确分类。在 (b) 中，3000 个测试样本中只有 3 个被误分类，而且这些误差都集中在分类决策边界附近。

为了更直观地展示分类性能，我们从 5 次独立运行中选取测试误差最小的一次实验结果，并在相平面上可视化分类结果，如图 15所示。实验结果表明，TTA分类器在训练集上实现了完美分类，6000 个训练样本全部被正确分类。在测试集上，3000 个测试样本中只有 3 个被误分类，且这些误分类样本都位于不同量子相之间的决策边界附近。这一现象充分体现了 TTA分类器的有效性：(1) 在训练集上的完美性能证明了迹距离二叉树构建策略和 AdaBoost集成学习的有效性；(2) 测试集上的高准确率（99.9%）展现了量子机器学习模型出色的泛化能力；(3) 误分类样本集中在决策边界附近的现象符合分类理论的预期，表明 TTA分类器能够准确识别不同量子相之间的本质差异，而相变临界区域仅存在细微的不确定性。

C. AdaBoost 中样本权重的行为

为了深入分析 AdaBoost 算法的学习机制，我们进一步考察训练过程中样本权重的动态变化。AdaBoost 算法的核心思想在于自适应地调整训练样本的权重，从而逐步提升集成分类器的性能。具体而言，在算法的初始阶段，所有训练样本都被赋予相同的权重，以体现对每个样本的公平对待。随着迭代过程继续进行，被当前分类器误分类的样本会获得更高的权重，因此在后续基分类器的训练中受到更多关注。

这种权重重分配机制使算法能够逐步聚焦于那些难以正确分类的“困难样本”，最终显著提升整体分类性能。

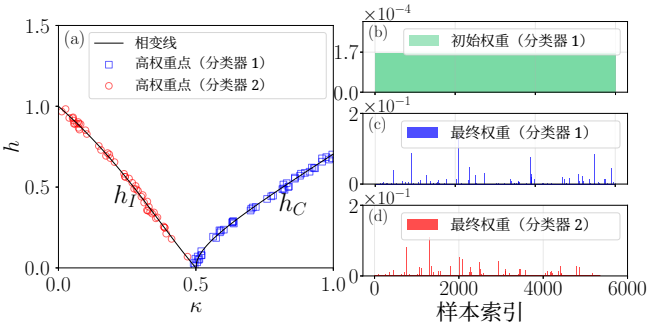

图 16. (a) 两个成员分类器的高权重训练样本都集中在决策边界附近。(b) 成员分类器 1 的初始均匀权重分布（成员分类器 2 的初始分布也均匀）。(c)-(d) 分别为成员分类器 1 和成员分类器 2 的最终权重分布。

在 ANNNI 量子相分类实验中，TTA分类器包含两个成员分类器，分别对应图13(b)所示迹距离树中的两个节点。利用图15中展示的相同实验结果，我们提取每个成员分类器收敛时最终基分类器对应的样本权重分布，并在图 κ-h 相平面中标出权重较高的训练样本，见图 16。

实验结果清楚地展示了 AdaBoost 算法的智能学习特性：权重最高的训练样本（即最难分类的样本）主要集中在不同量子相之间的决策边界附近。这一现象具有深刻的物理意义和扎实的算法理论基础：（1）从物理学角度来看，位于相变边界附近的量子态通常具有更复杂的纠缠结构和更微妙的量子关联，因此在分类任务中更难以分类；（2）从机器学习角度来看，决策边界附近的样本天然具有更高的分类不确定性，因此分类器需要投入更多“注意力”才能准确识别它们。

VI. 合成数据上的分类

本节系统地测试 TTA分类器在各种噪声环境下的鲁棒性。第VIA 节介绍合成分类数据集。第VIB 节展示了不同噪声类型下的性能结果。

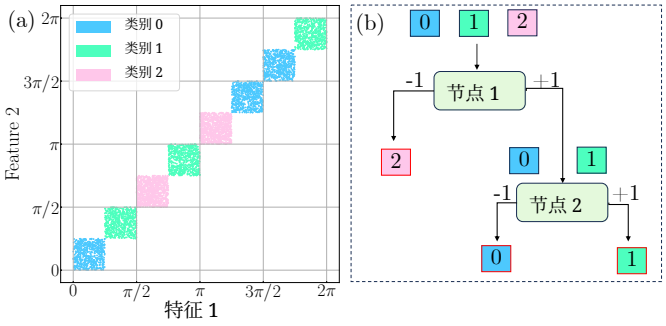

图 17。（a）合成数据集可视化。（b）合成数据集的迹距离树。

类型的噪声。第VIC 节将 TTA分类器与一对一和一对其余方法进行比较。

A. 合成分类数据集

为系统评估量子噪声对TTA分类器的影响，我们构建了一个合成数据集。具体而言，我们将区间$[0, 2\pi]$ 均匀划分为 8 个等长子区间，并在每个子区间内（例如 $[0, \pi/4]$、 $[\pi/4, \pi/2]$等）根据均匀分布随机生成 d维向量数据。随后，我们将从这 8 个不同子区间生成的 d维数据点随机分配到三个类别（标签为 0、1、2）中，从而构建一个用于三分类问题的合成数据集。这种数据生成方法确保不同类别之间具有一定程度的可分性，同时保持足够的复杂性，以验证噪声环境下TTA分类器的鲁棒性。二维数据可视化如图 17（a）所示。

在实验设计中，我们随机生成4维数据，并使用4个量子比特开展实验。每个数据点都通过 R_y 编码门编码为量子态。该数据集每个类别包含2000个训练样本和1000个测试样本，因此总计有6000个训练样本和3000个测试样本。基于最大化正类与负类平均态之间迹距离的原则，我们构建了迹距离二叉树，其结构如图17（b）所示，不同类别平均量子态之间的迹距离值见表V。

TTA分类器的训练过程遵循第 IVA 小节和第 VA小节中所述的相同方案。具体而言，我们采用图 2 所示的参数化量子电路架构，电路深度为 $L = 20$，以第一个量子比特上的泡利Z算符作为可观测量，以合页损失作为损失函数，批量大小设为200，采用 Adam优化器 [24]，学习率为0.005，并采用相同的提前停止策略。

表 V. 合成分类数据集 TTA树中每个节点的负类（左）与正类（右）平均量子态之间的迹距离。

Node	类别	迹距离；训练样本	
节点 1	{2} 与 {0,1}	0.6700	6000
节点 2	{0} 对比 {1}	0.5913	4000

B. 噪声下的性能

在量子计算中，噪声不可避免，并会影响量子态演化和测量结果。我们考虑三种典型的量子噪声模型。

1. 噪声模型

广义振幅阻尼噪声。 广义振幅阻尼噪声 [31] 描述了量子系统与有限温度环境之间的相互作用，同时考虑能量耗散和热激发过程。这种噪声由四个 Kraus 算符描述：

$$E_0 = \sqrt{p}\begin{pmatrix} 0 & \sqrt{\gamma} \\ 0 & 0 \end{pmatrix}, E_2 = \sqrt{1-p}\begin{pmatrix} \sqrt{1-\gamma} & 0 \\ 0 & 1 \end{pmatrix}$$
$$E_1 = \sqrt{p}\begin{pmatrix} 1 & 0 \\ 0 & \sqrt{1-\gamma} \end{pmatrix} E_3 = \sqrt{1-p}\begin{pmatrix} 0 & 0 \\ \sqrt{\gamma} & 0 \end{pmatrix},$$

其中， $\gamma \in [0, 1]$ 是阻尼参数， $p \in [0, 1]$是与环境温度相关的激发概率。当 $p = 1$ 时，该噪声会退化为标准振幅阻尼噪声。

退极化噪声。 退极化噪声 [31]是最常见的量子噪声模型之一，用于描述量子态向最大混合态退化的过程。对于单量子比特系统，退极化噪声的 Kraus 算符为：

$$E_0 = \sqrt{1 - \frac{3p}{4}}I, \quad E_1 = \sqrt{\frac{p}{4}}\sigma_x$$
$$E_2 = \sqrt{\frac{p}{4}}\sigma_y, \qquad E_3 = \sqrt{\frac{p}{4}}\sigma_z$$

其中 $p \in [0, 1]$ 是噪声强度参数， I 是单位矩阵，而 σ_x, σ_y， σ_z 是泡利矩阵。这种噪声以 p 的概率随机对量子态施加泡利算子，并以 $1 - p$ 的概率保持量子态不变。

重置错误噪声。 重置错误噪声 [31]描述了量子比特被意外重置为固定态的过程，这种情况经常发生在实际的量子设备中。该噪声对应的 Kraus 算符为：

$$E_0 = \sqrt{1-p}I,$$
$$E_1 = \sqrt{p}|0\rangle\langle 0|, \quad E_2 = \sqrt{p}|0\rangle\langle 1|,$$

其中，$p \in [0,1]$ 为重置概率。该噪声会以概率p 强制将量子比特重置为 $|0\rangle$ 状态，并以概率 $1-p$保持量子态不变。

在我们的实验中，对于退极化噪声，我们将噪声强度参数设为 $p = 0.1$ 。对于广义振幅阻尼噪声，我们将 $\gamma = 0.05$ 和 $p = 0.05$设为相应值。对于重置错误噪声，我们将 $p = 0.05$设为相应值。这些参数值反映了实际量子设备中的典型噪声水平。

2. 结果与分析

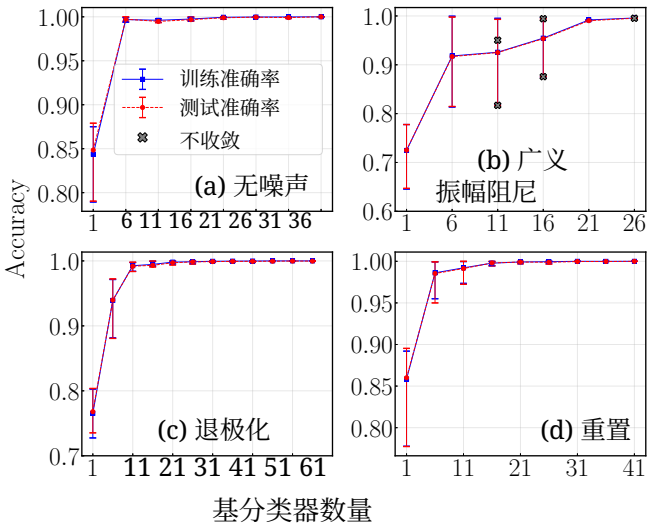

图 18. 不同条件下 TTA 分类器在合成分类数据上的分类结果：(a) 无噪声情况；(b) 广义振幅阻尼噪声情况，其中灰色点表示部分成员分类器因其基分类器的训练误差大于 0.5 而未能收敛，并直接退出训练；(c) 退极化噪声情况；(d) 重置错误情况。曲线显示 5 次独立运行的平均值，误差条表示最小值/最大值；灰色点表示未能收敛的分类器的最终结果。

图18展示了不同噪声条件下 TTA分类器的分类性能。在无噪声条件下（图 18（a）），TTA分类器稳定收敛，训练和测试准确率均达到 1.0，展现了该算法的理想性能。在广义振幅阻尼噪声下（图 18（b）），部分成员分类器由于基分类器训练误差超过 0.5 而无法收敛（图中以灰色点表示），导致整体性能有所下降，但仍保持了相对较高的准确率。相比之下，退极化噪声（图18（c））和重置错误噪声（图18（d））虽然会降低收敛速度并增加所需的基分类器数量，但仍可实现接近 1 的分类准确率。

这一现象与 AdaBoost 算法的理论预期一致：只要基分类器的性能略高于随机猜测，该算法就能保证收敛，而噪声主要影响收敛速率，而非最终性能。值得注意的是，即使在收敛失败的广义振幅阻尼噪声条件下（图18（b）），TTA分类器仍展现出良好的鲁棒性，为其在实际量子设备上的应用提供了重要的实践支持。

3. 电路深度对振幅阻尼噪声的影响

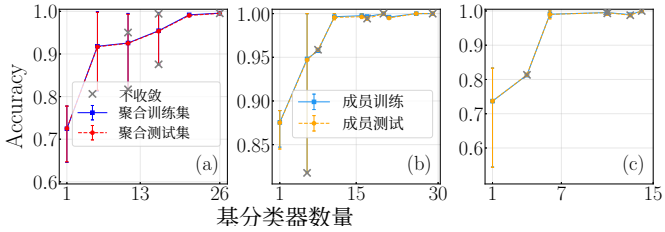

图 19. 在电路层为 $L = 20$ 的广义振幅阻尼噪声条件下，(a) 尽管并非全部 5 次试验都完全收敛至完美准确率，但其中一些试验的训练准确率和测试准确率仍接近 1。(b)-(c) 单个成员分类器表现出类似的行为。曲线表示 5 次独立运行的平均值，误差条表示最小值/最大值，灰色点表示未能收敛的分类器的最终结果

如图所示。18(b)所示，在广义振幅阻尼噪声下，TTA分类器在五次独立运行中均未能收敛。进一步考察（图19）发现，迹距离树中的两个成员分类器（如图17(b)所示）在每次运行中都未能收敛：部分分类器的准确率接近0.99，而其他分类器甚至未达到0.9。为缓解这一问题，我们将量子电路深度从 $L = 20$ 降低至 $L = 10$。如图20所示，降低电路深度后，随着基分类器数量增加，所有实验中的每个成员分类器均达到100%的训练准确率和测试准确率，TTA分类器也实现了完美收敛。这一改进有两个原因：首先，较浅的电路显著减少了噪声累积；其次，即使只有10层，TTA分类器仍可作为弱分类器（即训练误差低于0.5）发挥作用，从而保留AdaBoost的理论收敛保证。

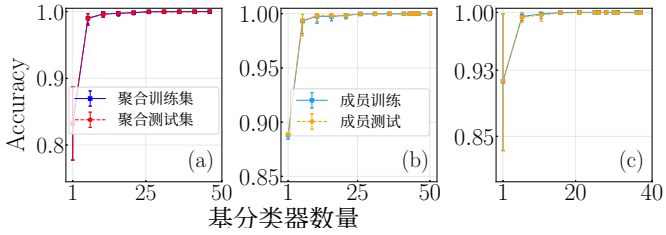

图 20. 在广义振幅阻尼噪声条件下，电路层为 $L = 10$时，(a) 随着基分类器数量增加，训练和测试准确率均趋近于 1。(b)-(c) 单个成员分类器也呈现类似趋势。曲线表示 5 次独立运行的平均值，误差条表示最小值/最大值；灰色点表示未能收敛的分类器的最终结果

C. 与一对一和一对其余的比较

1. 分类器复杂度比较

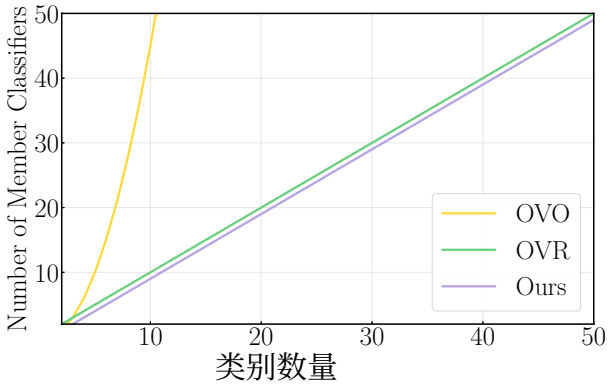

图 21. 随着类别数量 K 增加，TTA分类器、一对其余分类器和一对一分类器所需的成员分类器数量比较。

为展示TTA分类器的有效性及其在训练资源消耗方面的优势，我们将其与主流的一对一（OVO）和一对其余（OVR）方法进行比较。TTA分类器采用完全二叉树结构，其中每个内部节点有两个子节点，分别代表正类和负类。对于一棵具有 K 个叶节点（对应 K 个类别）的树，恰有 $K-1$ 个内部节点，因此只需 $K-1$ 个成员分类器。相比之下，OVR方法为每个类别构建一个单独的分类器（将该类别作为正类，其余类别作为负类），总共需要 K 个成员分类器。OVO方法为每一对类别训练一个分类器，因此需要$K(K-1)/2$ 个成员分类器。如图21所示，各方法所需的成员分类器数量随类别数量而变化。值得注意的是，随着类别数量增加，OVO所需的分类器数量迅速增长，而TTA分类器则保持始终比一对其余分类器少需要一个成员分类器。

2. 类别划分的迹距离分析

表VI和VII给出了一对其余与一对一分类器的类别划分，以及正类平均态与负类平均态之间相应的迹距离。TTA树划分（$\{2\}$对$\{0,1\}$，随后 $\{0\}$对$\{1\}$）在所有可能的二叉树配置中取得了最大的迹距离。考虑另一种方案，其第一个节点为 $\{1\}$对$\{0,2\}$：尽管随后在 $\{0\}$与 $\{2\}$ 之间的划分产生的迹距离为0.8379，但初始划分在6000个训练样本下的迹距离仅为0.4406，使收敛明显更加困难。类似地，若从 $\{0\}$对$\{1,2\}$ 开始，再分离类别1和类别2，则两个分类器的迹距离均低于TTA划分。因此，TTA 划分策略有效地平衡了分类难度与样本分布，从而使训练更加高效。

表 VI. OVR 归约方法中各成员分类器的负类（左）和正类（右）平均量子态之间的迹距离。

分类器	迹距离；训练样本	
类别 0 与其余类别	0.6498	6000
类别 1 与其余类别	0.4406	6000
类别 2 与其余类别	0.6700	6000

表 VII. 一对一归约方法中各成员分类器的负类（左）与正类（右）平均量子态之间的迹距离。

分类器	迹距离；训练样本	
类别 0 与类别 1	0.5913	4000
类别 0 与类别 2	0.8379	4000
类别 1 与类别 2	0.5739	4000

3. 性能比较

如图所示22，OVO方法的表现明显逊于TTA分类器。这是因为OVO会针对每一对类别分别训练一个分类器，从而得到彼此完全独立的二元分类器，无法利用数据集中的全局结构信息。因此，最终预测依赖于最大置信度聚合，这会增加推理不确定性并降低抗噪鲁棒性，导致准确率和稳定性下降。

相比之下，OVR方法达到了与我们的方法相当的分类准确率。然而，进一步考察基分类器效率后发现，该方法存在显著的资源浪费。例如，当类别 1 为正类、$\{0,2\}$ 为负类时，其迹

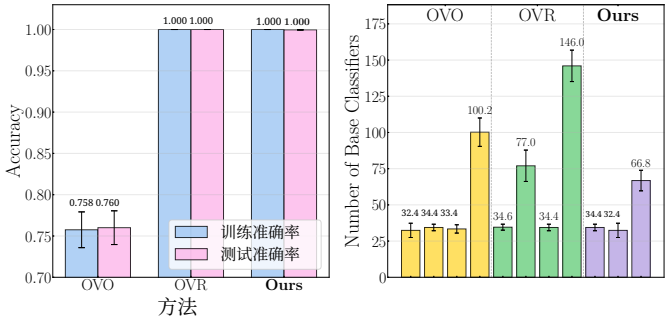

图22. TTA分类器与一对一和OVR方法在训练和测试准确率以及所用基分类器数量方面的比较：(a) 训练和测试准确率比较；(b) 基分类器数量比较。柱上的数字表示5次独立运行的平均值，误差条表示这些运行中的最大值和最小值。

距离相对较小。结果，相应的二分类AdaBoost分类器需要使用显著更多的基分类器，才能达到相同的训练误差收敛界。

与这两种替代方案相比，我们的迹距离树划分策略在理论和实践上都具有显著优势。通过采用二叉树结构递归划分多分类问题，每个内部节点都会对类别进行分组，以最大化平均量子态迹距离，从而确保更好的类别可分性，同时使每个二元任务的难度保持在可控范围内。此外，这种层次结构能够自然地捕捉全局类别关系，实现高效的信息传递和决策，无需为每个类别或类别对构建冗余的基分类器。实验结果表明，TTA分类器在不牺牲分类准确率的前提下，大幅减少了基分类器数量和所需的训练资源。这种兼具高效性和可扩展性的特点，使其尤其适用于量子计算资源有限的实际应用场景。

VII. 编码的重要性

在前文中，我们展示了 TTA 分类器在多类别任务中，无论是处理经典数据还是量子数据，都能达到 100% 的训练准确率和近 100% 的测试准确率。这一出色性能得益于采用浅层基分类器电路以避免贫瘠高原，并结合多个浅层基分类器来逼近全局最优解。因此，TTA 在训练过程中克服了贫瘠高原和虚假的局部极小值，同时保持了良好的泛化能力。然而，仍存在一个根本性限制：如何将经典数据编码为量子态。以往研究表明，常用的编码方法，包括角度编码 [32]和振幅编码 [20]，都存在固有局限，可能导致量子机器学习模型的性能不佳。此外，文献 [20] 表明，随着两个类别的平均态之间的迹距离减小，训练会变得越来越困难，预测也会趋向于随机猜测。

为研究编码对 TTA 训练和测试的影响，我们选取了以下二元数据集，其中“Twocurves”“隐藏流形”和“线性可分”由 PennyLane [33] 直接提供：

- CIFAR10 [34]：一个广泛使用的图像分类基准数据集，包含 60,000 32×32张彩色图像，分属 10 个类别（飞机、汽车、鸟、猫、鹿、狗、青蛙、马、船和卡车）。我们选取类别 0（飞机）和 2（鸟）进行二分类。

- FashionMNIST [35]：一个由 Zalando 的商品图像构成的数据集，包含 70,000 28×28 张灰度图像，分属 10 个时尚类别（T 恤/上衣、裤子、套头衫、连衣裙、外套、凉鞋、衬衫、运动鞋、包和踝靴）。我们选取类别 1（裤子）和 9（踝靴）进行二分类。

- 双曲线：这些数据来自基准测试 [33,36]，，其中两个一维类别流形通过固定次数为 $D = 5$ 的低阶傅里叶级数嵌入到 $d = 10$ 维空间中。两个类别之间以恒定偏移量 $\Delta = 0.1$ 分隔，并加入高斯噪声，从而构成一个可控基准，用于探究神经网络在复杂数据几何结构上的性能。

- 隐藏流形：这些数据使用 [37] 提出的隐藏流形方法生成，来自基准测试 [36]，。输入在低维流形上生成，由一个随机初始化的简单神经网络赋予标签，然后投影到高维环境空间中。我们具体采用该框架中的基准测试，其中环境空间的维度固定为 $d = 2$，流形维度固定为 $m = 6$。

- 线性可分：这些数据源自基准测试 [36] 中的经典“果蝇”线性可分问题示例 [38]，。输入从二维超立方体中均匀采样，并由一个与向量 $\boldsymbol{w} = [1, 1]^T$ 正交的超平面进行分类。超平面周围设置一个无数据间隔 $\Delta = 0.02$，以确保所有数据点 $\boldsymbol{x}$ 满足 $|\boldsymbol{w}^\top \boldsymbol{x}| > \Delta$。

表VIII总结了二元数据集（括号内为选定的类别标签）、编码方法（角度编码通过将数据映射到单量子比特门来实现 R_y，R_x）、数据维度（括号内为量子比特数量）、两个类别的平均量子态之间的迹距离（括号内为每类训练样本数量），以及最终

表 VIII. 训练集中两个类别的平均态之间的迹距离对 AdaBoost 结果的影响。

数据集（类别）	编码维度 方法	（量子比特）	平均值 迹距离	集成准确率（训练/测试）	终止 原因
CIFAR10 (0-2)	振幅	256 (8)	0.0760(2000) 7		
FashionMNIST (1-9)	振幅	256 (8)	0.8343 (2000)	100.00±0.01% / 99.91±0.06%	
双曲线	振幅	10 (4)	0.3856 (120)	99.83±0.33% / 100.00±0.00%	
隐藏流形	振幅	2 (1)	0.1285 (120)	70.00±3.16% / 41.67±1.83%	
隐藏流形	角度	2 (1)	0.2344 (120)	79.08±7.06% / 67.33±1.70%	
线性可分	振幅	2 (1)	0.0746 (120)	71.58±5.09% / 42.33±3.27%	
线性可分	角度	2（1）	0.4023（120）	100.00±0.00% / 99.67±0.67%	

AdaBoost 集成的训练和测试准确率（均值 ±标准差），以及每个基分类器的停止准则： ■表示训练误差超过 0.5； ■表示已达到基分类器数量上限（500）；而 ■ 表示训练误差的改善幅度低于 0.005。

如表 VIII所示，当平均迹距离较大时（例如 FashionMNIST），即使使用 4000 个训练样本（每个类别 2000 个），AdaBoost 也能实现完美拟合。对于迹距离中等的数据集，五次独立运行中的收敛行为各不相同。有时训练误差下界会降至 0.005 以下，而有时最终基分类器的训练误差仍高于 0.5。当迹距离较小时，集成通常会达到基分类器数量的上限（500 个），或完全无法收敛。值得注意的是，即使对于迹距离中等且仅有 240 个训练样本（每个类别 120 个）的数据集，达到 100% 的训练准确率仍能带来超过 99% 的测试准确率，这凸显了量子机器学习方法强大的泛化能力。

值得注意的是，一些研究指出，振幅编码带来的局限性或许可以通过角度编码来缓解。我们的结果揭示了更为复杂的情况。对于线性可分数据集，将振幅编码切换为角度编码后，平均状态之间的迹距离从 0.0746 增至 0.4023，使 AdaBoost 达到 100% 的训练准确率，平均测试准确率达到 99.67%。相比之下，对于隐藏流形数据集，角度编码得到的迹距离仅为

0.2344，这对于 AdaBoost 提升可训练性而言仍然不足。这些发现表明，通过改变编码方法来提高可训练性既不简单，也并非普遍有效。

VIII. 早停的影响

在本节中，我们展示早停策略的效率。尽管TTA分类器看似需要训练大量基础分类器（因而需要许多参数化量子电路），但早停可确保每个参数化量子电路通常只需训练略多于10轮即可收敛，远少于传统分类器通常所需的数百轮训练，从而大幅降低总体训练成本。

以 FashionMNIST数据集为例（见表 VIII），我们固定第一个基分类器的初始化参数，并比较使用和不使用早停的训练结果。所有其他超参数——优化器、批量大小和学习率——均保持不变。在每一轮中，我们选择实现最低训练误差的参数，作为 t 个基分类器的最终参数。因此，即使从完全相同的初始参数开始，每一轮所选的最优参数也可能不同，从而导致训练误差存在差异。这些差异会在集成中逐步传播：它们会影响分配给每个基分类器的权重，以及随后对训练样本的重新加权，最终导致整体性能出现差异，即使所有基分类器都采用相同的初始化参数。

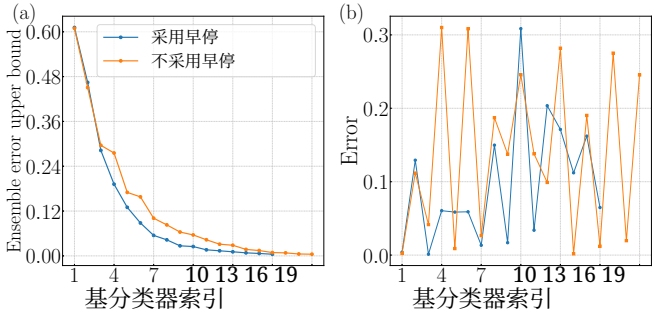

图 23. 使用和不使用早停时，(a) 集成训练误差上界与 (b) 第 t 个基分类器训练误差的比较。

我们比较了在以下两种条件下，第 t 个基分类器对应的集成训练误差上界：采用早停和不采用早停。直观而言，人们可能会预期，让每个基分类器进行更多训练轮次会降低训练误差，从而使集成更快达到收敛阈值（例如 0.005）。然而，如图23(a) 所示，结果恰恰相反：采用早停时，集成训练误差上界更早降至收敛阈值以下。具体而言，不采用早停时需要 19 个基分类器，而采用早停时只需 16 个。仔细观察图23(b) 可知，尽管在前两轮中，不采用早停时各基分类器的训练误差略低，但过度

训练会导致后续轮次中的样本权重分布出现大幅波动。这些波动使新的基分类器更难区分重新加权的数据，增加了优化难度并抬高了训练误差。因此，即使不采用早停，后续基分类器的质量也会下降，需要更多分类器才能使集成收敛。由此可见，恰当采用早停不仅能减少整体训练资源消耗，还能提升集成的整体性能。

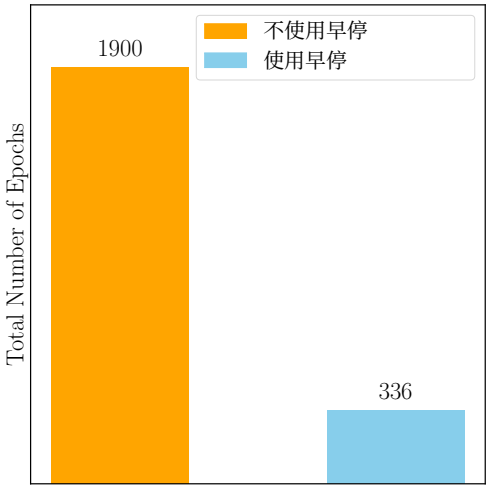

图 24. 达到相同集成训练误差上界所需的训练轮次总数，比较使用早停与不使用早停的情况。

实验结果表明，无论是否采用早停，两种集成在训练集和测试集上均达到了 100% 的准确率。为更直观地展示早停带来的资源节省，图 24比较了两种设置下所需的训练轮次总数。不采用早停时，19 个基分类器总共消耗了 1900 个训练轮次；采用早停后，仅需 336 个训练轮次，减少了约 5。65× 的计算开销。

附录 D提供了所有 16 个（采用早停）和 19 个（不采用早停）基分类器训练误差随训练轮次变化的更多细节。

IX. 结论

我们提出了 TTA 分类器，这是一种紧凑且实用的策略，将量子感知约简规则（迹距离最大化）与浅层变分电路的逐节点 AdaBoost 相结合，以应对多分类量子机器学习中的关键可训练性限制。TTA 在经典和量子基准测试中均实现了 100% 的训练准确率和近乎完美的测试准确率，同时显著降低了整体电路深度和参数预算；此外，它在现实量子噪声下也表现出明确的鲁棒性。结合可扩展的启发式方法（贪心划分、早停、并行节点训练）和广泛的实证验证，TTA 为在近期量子设备上部署多分类量子分类器提供了实用蓝图，并推动了进一步的硬件演示。

代码可用性

本文的代码可在 https://github.com/SheffieldWang/Quantum_TTA获取。该实现基于 MindSpore Quantum [39]。

致谢

本研究由量子与信息技术长三角产业创新中心—MindSpore Quantum 开放基金资助，并得到国家自然科学基金委员会（项目编号：62173201）和量子科学与技术创新计划（项目编号：2021ZD0300200）的支持。

[1] J. Biamonte、P. Wittek、N. Pancotti、P. Rebentrost、N. Wiebe 和 S. Lloyd，量子机器学习，Nature **549**，195（2017）。[2] P.-L. Zheng、J.-B. Wang 和 Y. Zhang，利用费米子神经网络实现高效且量子自适应的机器学习，Phys. Rev. Appl. **20**，044002（2023）。[3] H.-Y. Huang 等，从实验中学习的量子优势，Science **376**，1182（2022）。[4] M. C. Caro、E. Gil-Fuster、J. J. Meyer、J. Eisert 和 R. Sweke，参数化量子电路中依赖编码的泛化界，Quantum **5**，582（2021）。[5] M. C. Caro、H.-Y. Huang、M. Cerezo、K. Sharma、A. Sornborger、L. Cincio 和 P. J. Coles，基于少量训练数据的量子机器学习泛化，Nat. Commun. **13**，4919（2022）。

[6]X. Wang 和 R. Wu，《监督量子机器学习的紧致泛化界》，arXiv 预印本 arXiv:2510.24348（2025）。[7]X. You 和 X. Wu，《量子神经网络中的指数多的局部极小值》，载于*第38届国际机器学习会议论文集*（PMLR，美国纽约州纽约市，2021年）。[8] E. R. Anschuetz 和 B. T. Kiani，《量子变分算法充斥着陷阱》，Nat. Commun. **13**，7760（2022）。[9] M. Larocca、N. Ju、D. García-Martín、P. J. Coles 和 M. Cerezo，《量子神经网络中的过参数化理论》，《自然·计算科学》**3**，542（2023）。

[10]M. Cerezo、A. Sone、T. Volkoff、L. Cincio 和 P. J. Coles，《浅层参数化量子电路中依赖代价函数的贫瘠高原》，Nat. Commun. 12，1791（2021）。[11] M. Ragone、B. N. Bakalov、F. Sauvage、A. F. Kemper、C. Ortiz Marrero、M. Larocca 和 M. Cerezo，《深层参数化量子电路贫瘠高原的李代数理论》，Nat. Commun. 15，7172（2024）。[12] S. Wang 、 E. Fontana 、 M. Cerezo 、 K. Sharma 、 A. Sone、L. Cincio 和 P. J. Coles，《变分量子算法中的噪声诱导贫瘠高原》，Nat. Commun. 12，6961（2021）。[13] W. X. Zhao、K. Zhou、J. Li、T. Tang、X. Wang、Y. Hou、Y. Min、B. Zhang、J. Zhang、Z. Dong 等，《大语言模型综述》，arXiv 预印本 arXiv:2303.18223（2023）。[14] Y. Freund 和 R. E. Schapire，《在线学习的决策理论推广及其在提升中的应用》，载于《欧洲计算学习理论会议》（Springer，西班牙巴塞罗那，1995年），第23—37页。[15] J. Zhu、H. Zou、S. Rosset、T. Hastie 等，《多类AdaBoost》，Stat. Interface 2，349（2009）。[16] M. Mohri、A. Rostamizadeh 和 A. Talwalkar，《机器学习基础》，第2版（麻省理工学院出版社，美国马萨诸塞州剑桥，2018年）。[17] Y. Wang、X. Wang、B. Qi 和 D. Dong，《量子AdaBoost的监督学习保证》，Phys. Rev. Appl. 22，054001（2024）。[18] T. G. Dietterich 和 G. Bakiri，《通过纠错输出码解决多类学习问题》，J. Artif. Intell. Res. 2，263（1994）。[19] T. Hastie 和 R. Tibshirani，《通过成对耦合进行分类》，载于《神经信息处理系统进展》，第10卷（MIT Press，美国科罗拉多州丹佛，1997年）。[20] X. Wang、Y. Wang、B. Qi 和 R. Wu，《振幅编码对量子分类的局限性》，arXiv 预印本 arXiv:2503.01545（2025）。[21] Y. Du、Y. Yang、D. Tao 和 M.-H. Hsieh，《量子神经网络在多类分类中的问题依赖能力》，Phys. Rev. Lett. 131，140601（2023）。[22] L. Rosasco 、 E. De Vito 、 A. Caponnetto 、 M. Piana 和 A. Verri，《损失函数都相同吗？》，Neural Comput. 16，1063（2004）。[23] Y. LeCun、L. Bottou、Y. Bengio 和 P. Haffner，《应用于文档识别的基于梯度的学习》，Proc. IEEE 86，2278（1998）。[24] D. P. Kingma 和 J. Ba，《Adam：一种用于随机优化的方法》，arXiv 预印本 arXiv:1412.6980（2014）。[25] M. Schuld、A. Bocharov、K. M. Svore 和 N. Wiebe，《以电路为中心的量子分类器》，Phys. Rev. A 101，032308 (2020)。[26] Q. Li、Y. Huang、X. Hou、Y. Li、X. Wang 和 A. Bayat，《用于变分量子浅层电路分类器的集成学习误差缓解》，Physical Review Research 6，013027（2024）。[27] K. He、X. Zhang、S. Ren 和 J. Sun，《用于图像识别的深度残差学习》，载于《IEEE 计算机视觉与模式识别会议论文集》（IEEE，美国内华达州拉斯维加斯，2016），第770–778 页。[28] A. Dosovitskiy，《一幅图像胜过 16×16 个词：面向大规模图像识别的 Transformer》，arXiv 预印本 arXiv:2010.11929（2020）。[29] P. Zhou、J. Feng、C. Ma、C. Xiong、S. C. H. Hoi 等，《从理论上理解为什么随机梯度下降在深度学习中的泛化优于 Adam》，载于《第 33 届国际机器学习会议论文集》（PMLR，2020）。[30] W. Selke，《ANNNI模型：理论分析与实验应用》，Physics Reports 170，213（1988）。[31] M. A. Nielsen 和 I. L. Chuang，《量子计算与量子信息》（剑桥大学出版社，英国剑桥，2000）。[32] G. Li、R. Ye、X. Zhao 和 X. Wang，《参数化量子电路中的数据编码集中度》，载于《第 35 届神经信息处理系统大会论文集》（Curran Associates, Inc.，美国路易斯安那州新奥尔良，2022）。[33] J. Bowles、S. Ahmed 和 M. Schuld，《优于经典方法？量子机器学习模型基准测试的微妙艺术》，arXiv 预印本 arXiv:2403.07059（2024）。[34] A. Krizhevsky 和 G. Hinton，《从微型图像中学习多层特征》，技术报告（多伦多大学，2009）。[35] H. Xiao、K. Rasul 和 R. Vollgraf，《Fashion-MNIST：用于基准测试机器学习算法的新型图像数据集》，arXiv 预印本 arXiv:1708.07747（2017）。[36] S. Buchanan、D. Gilboa 和 J. Wright，《深度网络与多重流形问题》，arXiv 预印本 arXiv:2008.11245（2020）。[37] S. Goldt、M. Mézard、F. Krzakala 和 L. Zdeborová，《数据结构对神经网络学习影响的建模：隐藏流形模型》，Physical Review X 10，041044（2020）。[38] F. Rosenblatt，《感知机：一种用于大脑中信息存储与组织的概率模型》，Psychological Review 65，386（1958）。[39] X. Xu、J. Cui、Z. Cui、R. He、Q. Li、X. Li、Y. Lin、J. Liu、W. Liu、J. Lu 等，《MindSpore Quantum：用户友好、高性能且兼容人工智能的量子计算框架》，arXiv 预印本 arXiv:2406.17248（2024）。

A. 其他多分类到二分类归约方法示意图

A.1. 按位聚合

该方法为每个类别分配唯一的二进制编码，并针对每个位位置分别训练一个分类器，从而将多类分类转化为二分类。对于 $K = 4$，类别标签 $\{0, 1, 2, 3\}$ 被编码为两位二进制数：00、01、10 和 11。随后，针对每个位位置分别训练一个二元分类器。

考虑首位分类器：它预测编码的首位是 0 还是 1。在训练过程中，类别 0 和 1 的样本（编码分别为 00 和 01，首位均为 0）被归为负类，而类别 2 和 3 的样本（编码分别为 10 和 11，首位均为 1）则构成正类。对于第二位分类器，分组方式有所不同：类别 0 和 2（编码分别为 00 和 10，末位为 0）归为负类，类别 1 和 3（编码分别为 01 和 11，末位为 1）归为正类。所有分类器均独立训练。

在推理过程中，每个分类器都会预测新样本对应的位。将预测出的位拼接起来即可形成二进制编码，再对其解码以获得最终类别标签。对于 $K = 4$，只需要 $\lceil \log_2 4 \rceil = 2$ 个二元分类器。然而，这种方法没有考虑类别之间的量子可区分性（例如平均量子态之间的迹距离），并且每个分类器都必须处理整个数据集，这可能会使训练变得困难。

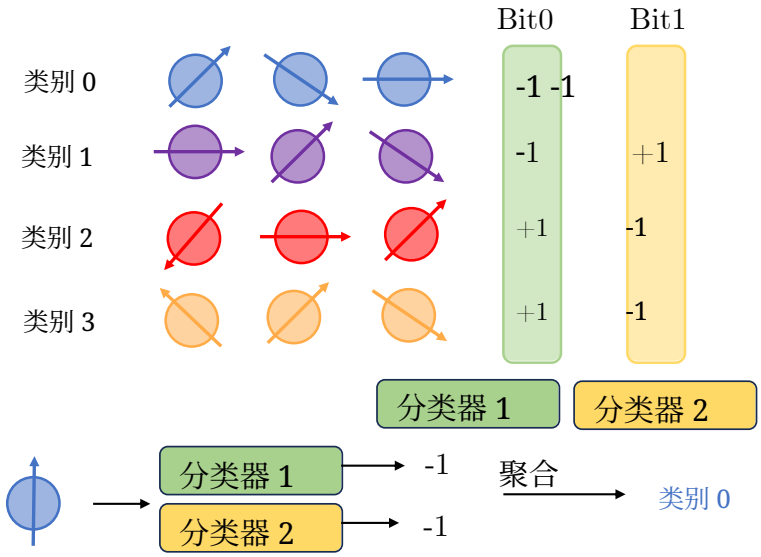

图 A.1. 按位聚合方法用于将多类分类简化为二元分类的示意图：左侧显示原始多类标签，右侧显示其二进制编码。每个二元分类器预测一个比特，最终标签由拼接后的比特解码得到。

A.2. 一对其余（OVR）

一对其余（OVR）方法是一种广泛使用的策略，用于将多类分类简化为二元任务。对于一个包含 K 个类别的问题，OVR 训练 K 个二元分类器，每个分类器区分一个类别（标记为正类）与所有其他类别（标记为负类）。在推理时，新样本会通过全部 K 个分类器，并选择其分类器输出置信度最高的类别作为预测结果。

尽管 OVR 简单易用，但它存在一些缺点。首先，每个分类器面对的都是高度不平衡的数据集：正类仅占样本的 $1/K$，这可能导致模型过拟合多数负类，从而降低判别能力。其次，最终的多类分类预测依赖于比较不同二元分类器的置信度，但这些分数通常缺乏统一标准和适当的校准，因此直接比较或汇总它们可能引入系统性偏差。第三，OVR 忽略了平均量子态之间的迹距离等量子可区分性度量，而且每个分类器都必须处理完整数据集，从而增加了训练难度。

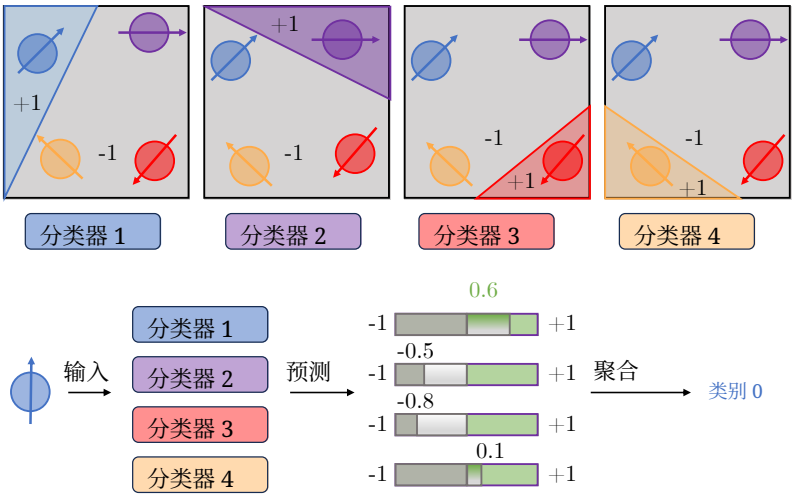

图 A.2. 一对其余法（OVR）示意图：针对每个类别，训练一个二元分类器，以区分该类别与所有其他类别。在推理过程中，将新样本输入每个成员分类器，并选择置信度最高的类别作为预测标签。

A.3. 一对一

一对一（OVO）方法是另一种广泛使用的策略，用于将多类分类简化为多个二元问题。对于一个 K 类问题，OVO 会训练 $K(K-1)/2$ 个二元分类器，每个分类器对应一对不同的类别。每个分类器仅使用其指定的两个类别的数据，并将其中一个标记为正类，另一个标记为负类。这种设计将每个分类器的任务简化为仅区分两个类别，从而得到更加平衡、更加清晰的训练分布。在推理过程中，将新样本输入所有分类器，并选择所有分类器中置信度最高的类别作为最终预测结果。

然而，一对一存在显著缺点。随着 K 增加，所需分类器的数量呈二次增长，从而大幅提高训练与推理的计算成本。此外，由于每个分类器仅看到来自两个类别的数据，它无法利用其他类别的信息。这一局限可能导致多个分类器对同一样本赋予高置信度，从而降低最终决策的可靠性。另外，分类器在面对训练期间从未遇到过的类别样本时可能表现不可预测，进一步使预测复杂化。

B. 其他算法

B.1. 寻找最大分离二元划分的算法

该算法称为 MaxBinarySplit，将当前类别集合 K_{cur} 划分为两个子集（K_1，K_2），使它们之间的平均迹距离最大化。通过利用每个类别的平均量子态ρ_k 之间的迹距离，该方法确保所得分组尽可能可区分。

步骤如下：首先，对于当前类别集 K_{cur} 中的每一对类别（i，j），计算它们的平均量子态之间的迹距离，记为 $D_{ij} = \frac{1}{2}\|\rho_i - \rho_j\|_1$。接着，找出迹距离最大的类别对，记为（$i^*$，$j^*$），并以它们初始化两个子集：$K_1 = \{i^*\}$ 和 $K_2 = \{j^*\}$。然后，对于每个剩余类别 $k \in K_{\text{cur}} \setminus \{i^*, j^*\}$，将其分配到这样的子集中：该子集的初始元素（$i^*$ 或 j^*）与 k 之间的迹距离较小。最后，输出所得划分（K_1，K_2）；该方法采用贪心策略，力求找到一种划分，使两组的平均量子态之间具有较大的迹距离。

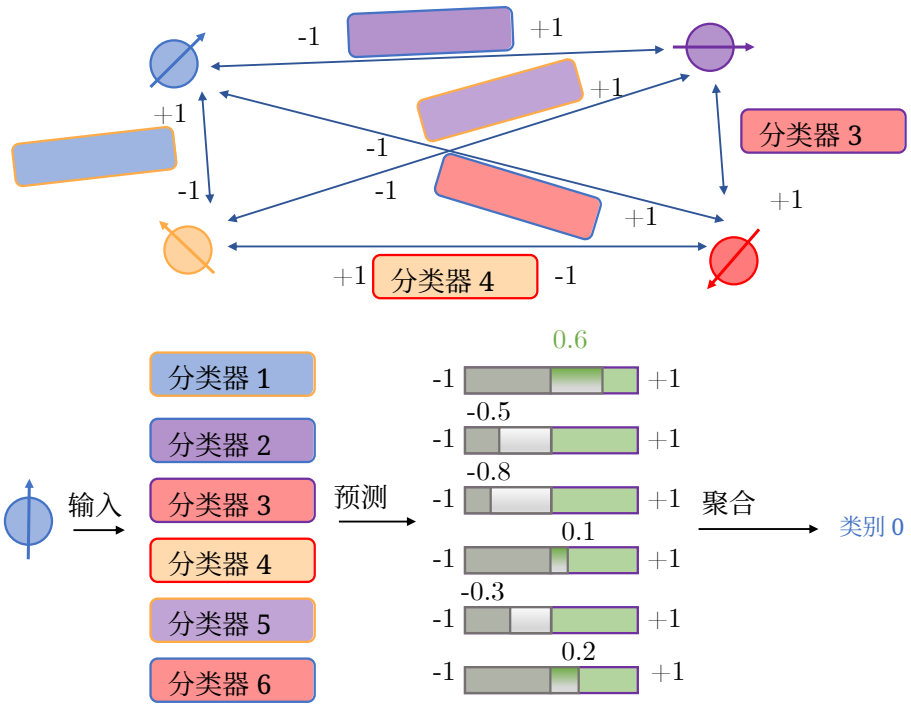

图 A.3. 一对一（OVO）方法示意图：对于每一对类别，训练一个二元分类器来区分它们。在推理过程中，将新样本输入到每个成员分类器中，并选择置信度最高的类别作为预测标签。

算法 B.1： MaxBinarySplit

输入： 当前子集 $\{\rho_k\}_{k\in K_{\text{cur}}}$ 中每个类别的平均量子态集合，类别标签集 K_{cur} **输出：** 使两组之间的平均迹距离最大化的二元划分 $(K_1,\ K_2)$

计算所有类别对之间的迹距离：

对于每个 类别 $i,\ j \in \mathcal{K},\ i<j$ **执行**

  $D_{ij} \leftarrow \frac{1}{2}\|\bar{\rho}_i - \bar{\rho}_j\|_1$;

end

选择 D_{ij} 最大的类别对：$(i^*,\ j^*) = \arg\max_{i\neq j} D_{ij}$;

$K_1 \leftarrow \{i^*\}$

$K_2 \leftarrow \{j^*\}$

对每个类别 $k \in \mathcal{K},\ k \neq i^*, j^*$ **执行**

  如果 $D_{k,i^*} < D_{k,j^*}$ **则**

    $K_1 \leftarrow K_1 \cup \{k\}$;

  else

    $K_2 \leftarrow K_2 \cup \{k\}$;

  end

end

返回(K_1, K_2)

B.2. 多类AdaBoost算法

多类AdaBoost将原始集成方法扩展到 K 类任务。与其二元版本类似，它通过迭代训练弱分类器并调整样本权重，逐步提升整体性能。然而，这种扩展不再保留训练误差上界呈指数下降的理论保证。完整算法见算法B.2。

算法 B.2： 多类别量子 AdaBoost

输入： 训练集 $S = \{(\rho^{(m)}, y^{(m)})\}_{m=1}^{M}$ 包含 K 个类别；
最大提升轮数 T

输出： 最终假设 H_S

为所有样本初始化权重：$\boldsymbol{w}_1(m) \leftarrow \frac{1}{M}$，对于 $m \in [M]$；

对于 $t = 1$ **到** T **执行**

使用权重 $\boldsymbol{w}_t$ 在 S 上训练基分类器 $h_S^{(t)}$；

计算误差：$\epsilon_t = \sum_{m=1}^{M} \boldsymbol{w}_t(m) \mathbb{1}\left(y^{(m)} \neq h_S^{(t)}(\rho^{(m)})\right)$；

如果 $\epsilon_t \geq \frac{K-1}{K}$ **那么**

中断

end

计算基分类器 $h_S^{(t)}$ 的系数：$\alpha_t \leftarrow \log \frac{1-\epsilon_t}{\epsilon_t} + \log(K-1)$；

对于 $m = 1$ **到** M **执行**

更新权重：$\boldsymbol{w}_{t+1}(m) \leftarrow \boldsymbol{w}_t(m) \exp\left(\alpha_t \mathbb{1}\left(y^{(m)} \neq h_S^{(t)}(\rho^{(m)})\right)\right)$；

end

归一化权重：$\boldsymbol{w}_{t+1}(m) \leftarrow \frac{\boldsymbol{w}_{t+1}(m)}{\sum_{m=1}^{M} \boldsymbol{w}_{t+1}(m)}$ 对于所有 m；

end

构造最终分类器：$H_S(\cdot) \leftarrow \arg\max_{k \in [K]} \sum_{t=1}^{T} \alpha_t \mathbb{1}(h_S^{(t)}(\cdot) = k)$；

返回 H_S

C. 在经典神经网络中使用随机梯度下降的原因

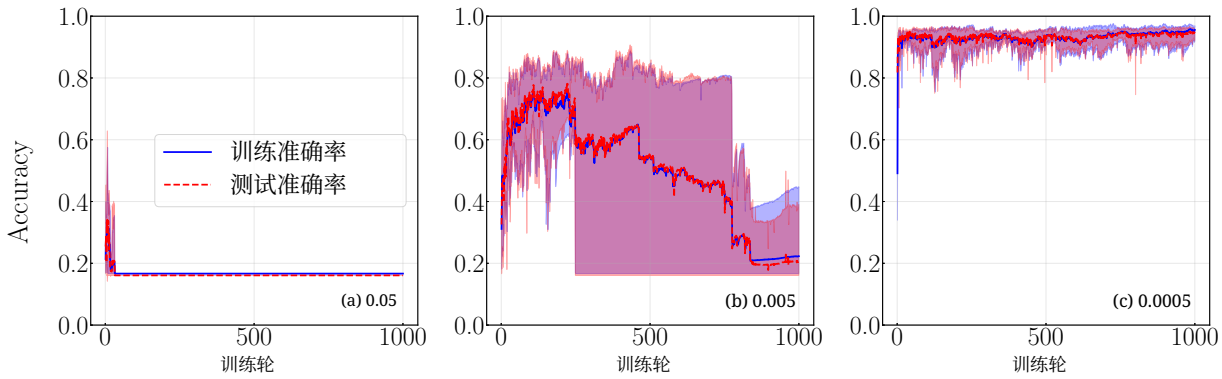

图 C.1。（a）学习率为 0.05 时 Adam 在 ViT 上的训练和测试准确率；（b）学习率为 0.005 时的训练和测试准确率；（c）学习率为 0.0005 时的训练和测试准确率。中间线表示 5 次运行的平均值，阴影区域表示这 5 次运行中的最小值和最大值。

在我们的实验中，我们使用 SGD优化器训练 ResNet 和 ViT。选择 SGD 主要有两个原因：一方面，先前的研究表明，与 Adam 相比，SGD 往往能带来更好的泛化性能 [29]；另一方面，我们发现，在我们的实验设置中，Adam优化器无法为小型 ViT 模型取得令人满意的训练结果。

具体而言，我们使用 Adam 优化器训练小型 ViT 模型，学习率分别设为 0.05、0.005 和 0.0005。最大训练轮次设为 1000；若训练准确率达到 100%，则应用早停。其他所有参数设置均与正文一致。训练结果如图 C.1 所示。当学习率设为 0.05 时，观察到一种异常现象：随着训练轮次增加，训练集和测试集上的准确率反而都下降了。当学习率设为 0.005（这也是正文所有 Adam 实验采用的设置）时，准确率起初有所上升，随后便开始下降，始终未能超过 0.9。当学习率进一步降至 0.0005 时，准确率能够稳定超过 0.9，但始终未达到 100%，且整体性能仍不如随机梯度下降。这些结果进一步

表明，在我们的实验条件下，随机梯度下降优化器比 Adam 能带来更稳定、效果更好的训练结果。

D. 提前停止策略的详细说明

正文中，我们指出，早停策略可以在不牺牲训练效果的前提下减少训练资源消耗。此处，我们通过图 D.1和 D.2，详细展示了在应用和不应用早停两种情况下，t-th 个基分类器在 FashionMNIST（类别 1 和 9）重加权样本训练集上的性能。

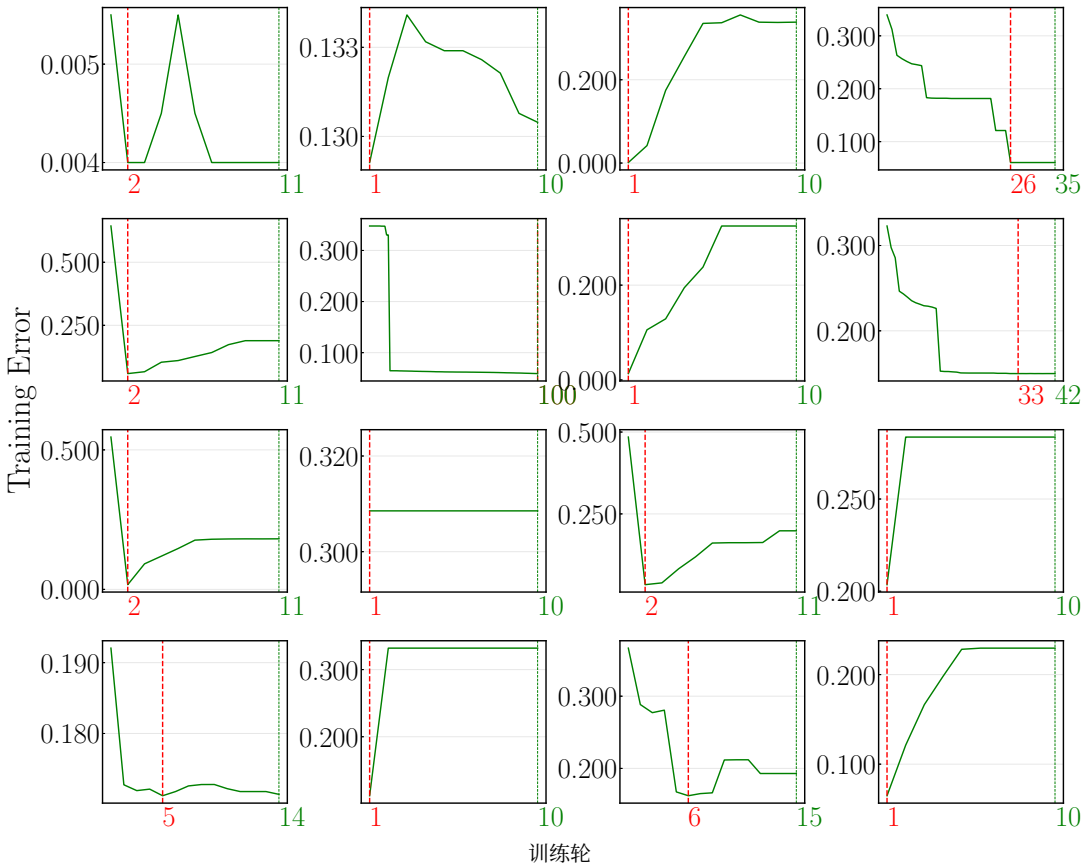

图 D.1. 应用早停时各基分类器的训练误差。红色竖线表示达到最优训练误差的训练轮次，并在该轮次选取相应参数。绿色竖线表示早停策略生效并最终停止训练的训练轮次。所有子图按照从左到右、从上到下的顺序排列，编号为 1 至 16。

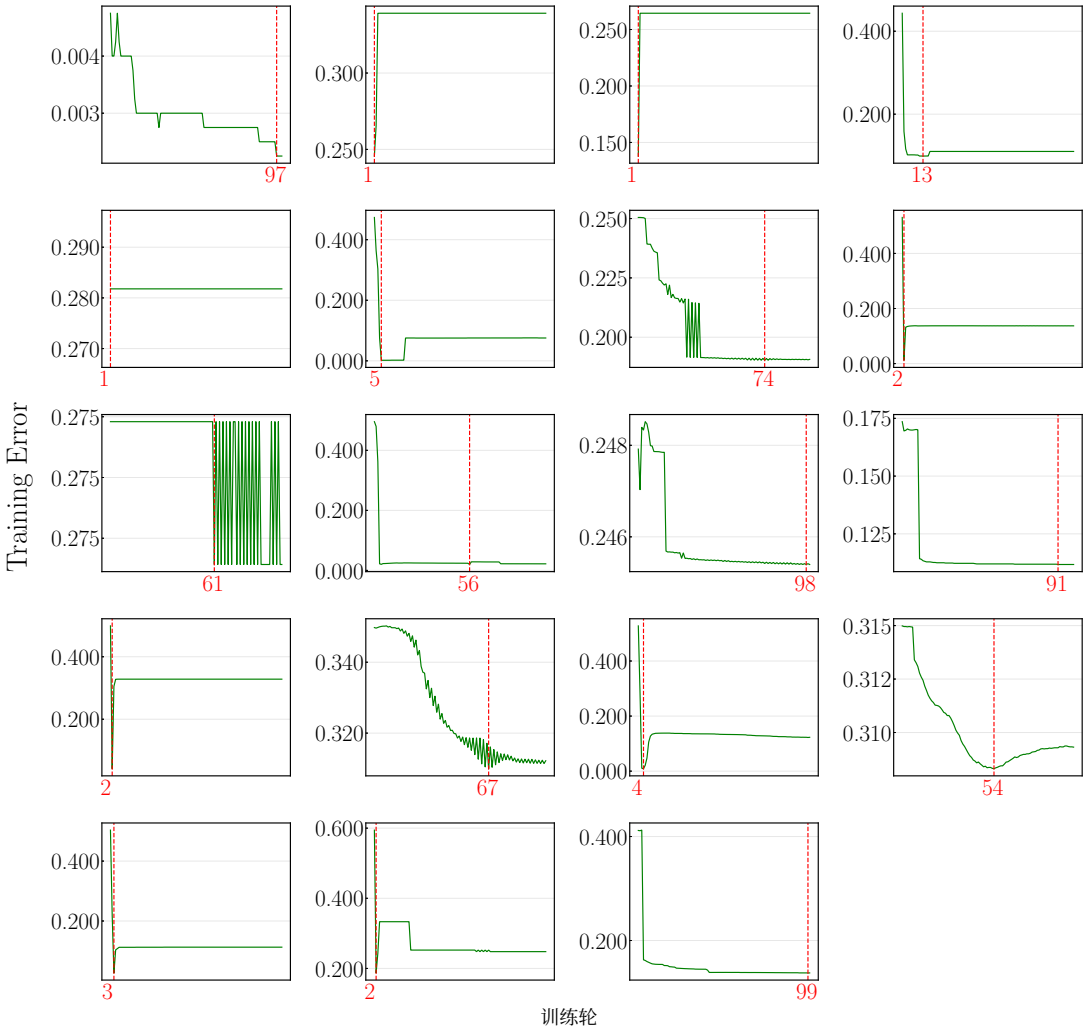

图 D.2. 未应用早停时每个基分类器的训练误差。红色竖线表示达到最优训练误差的训练轮次，并据此选取相应参数。所有子图按从左到右、从上到下的顺序排列，编号为 1 至 19。