机器学习分类实验报告

1 课题综述

1.1 课题说明

成员名单及分工（按学号排序）：

学号：2151531　姓名：栾佳浩　任务：传统分类方法，手写决策树，撰写实验报告

学号：2152486　姓名：刘翌帆　任务：深度学习方法，撰写实验报告

学号：2152496　姓名：郭桉齐　任务：传统分类方法，手写朴素贝叶斯，撰写实验报告

学号：2154312　姓名：郑博远　任务：数据预处理与特征提取，撰写实验报告

1.2 课题目标

本实验旨在通过综合运用传统机器学习方法与深度学习技术，对 CIFAR-10 图像数据集进行分类任务。选取的传统机器学习方法包括决策树、随机森林、KNN、支持向量机（SVC）、朴素贝叶斯，并使用 HOG（Histogram of Oriented Gradients）和 HSV（Hue, Saturation, Value）特征进行图像特征提取。同时，深度学习方法方面，我们选择了 VGG16 与 ResNet 进行图像分类。

通过这次实验，我们旨在深入了解不同机器学习算法在图像分类任务上的表现，并对比传统方法与深度学习方法之间的优劣。特别关注的是各算法在处理 CIFAR-10 数据集中多类别图像分类时的准确性、泛化能力以及对特征的提取和表示能力。此外，我们也关注深度学习方法相对于传统方法在复杂图像分类任务上的优越性。通过对比实验结果，我们期望能够得出对于 CIFAR-10 数据集上各方法的性能差异进行深入分析。这将有助于为图像分类任务选择最佳的模型和特征提取方法提供实质性的指导，为未来类似任务的研究和应用提供有益的经验和参考。

1.3 课题数据集

CIFAR-10（Canadian Institute for Advanced Research）是一个广泛应用于图像分类任务的公共数据集，由加拿大高级研究所创建（http://www.cs.toronto.edu/~kriz/cifar.html）。该数据集包含来自 10 个不同类别的 60000 张 32×32 像素的彩色图像，每个类别包含 6000 张图像。这十个类别分别为飞机、汽车、鸟类、猫、鹿、狗、青蛙、马、船和卡车。

CIFAR-10 图像分辨率相对较低，图像内容复杂多样，涵盖了常见的自然场景和对象。由于图像尺寸较小，CIFAR-10 适用于快速验证和比较不同分类算法的性能。它也常用于深度学习模型的初步训练和调优。因此，CIFAR-10 在学术界和工业界都被广泛使用，成为研究图像分类和模式识别算法的重要基准数据集之一。

2 传统分类

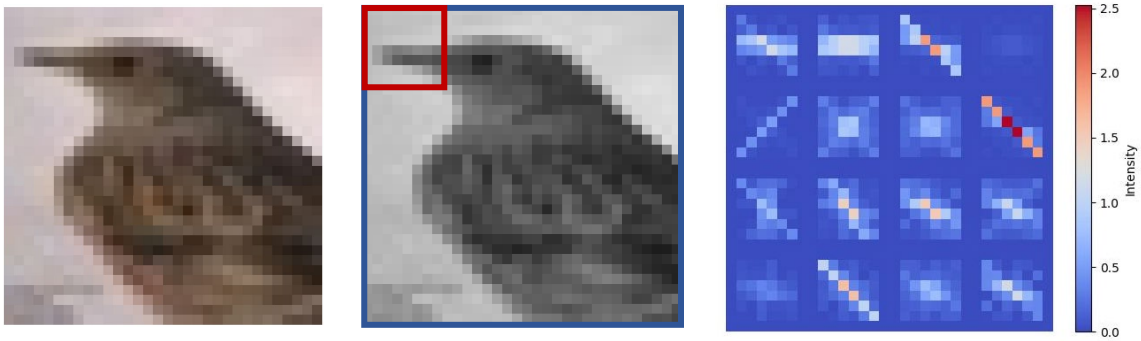
2.1 数据准备

首先访问 CIFAR-10 的官方网站（http://www.cs.toronto.edu/~kriz/cifar.html），进行数据集获取下载。下载并解压数据后可以观察到 CIFAR-10 数据集包含 60000 张分辨率为 32x32 的彩色图像，分为 10 个类别，每个类别包含 6000 张图像。其中训练集分为 5 个，各有 10000 张图像，测试集有 10000 张图像。为了方便训练，在读取数据时我们进行了 5 个批次训练集的合并。由于数据量较大，在提取计算 HOG、HSV 特征时，我们将预处理后的数据保存为 CSV 文件，这样可以在模型训练时快速读取数据，无需每次重新进行繁琐的预处理过程。

2.2 数据预处理

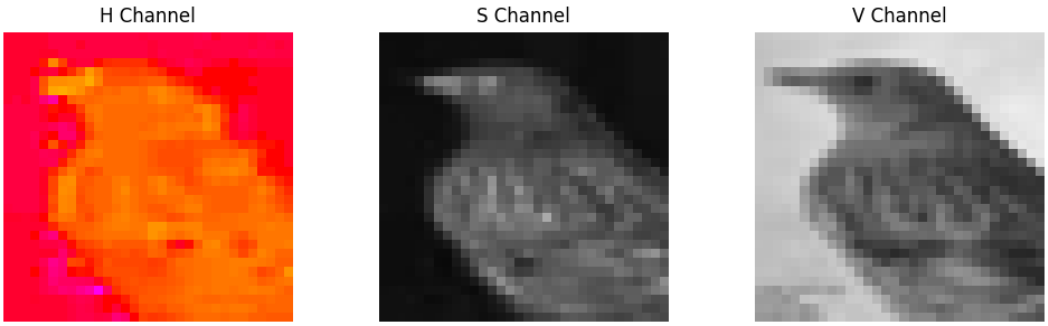
由于原本数据集中的特征量为 3×32×32，对于传统分类方法来说维数较大，因此需要进行特征提取。我们的数据预处理经过以下步骤：首先，对彩色图像进行灰度转换，利用 HOG 算法提取 160 维度的轮廓特征。其次，将彩色图像转为 HSV 颜色空间，提取 30 维度的色彩特征。这两部分特征合并成 190 维度的向量。为了确保稳定性，使用 StandardScaler 进行标准化。最后，利用 PCA 降维，保留 80%的方差，得到最终的特征向量。

2.2.1 HOG 特征提取



首先，将数据集中的 RGB 图像其转换为灰度图像。随后，应用 HOG（Histogram of Oriented Gradients）算法，其中我们尝试了不同的参数组合。我们探索了（8×8 与 4×4 pixels）不同像素块大小（block，即上方中间图的红色框线）和（4×4 与 2×2 blocks）细胞单元大小（cell，即上方中间图的蓝色框线）的可能性。最终通过实验发现，我们发现 8×8 像素的块和 4×4 像素的细胞单元的组合效果最好，此时所提取的特征数量也较少，为 160 个维度；这一选择在维持足够细节信息的同时，降低了计算负担。提取到的特征信息见上方右侧图。这个特征向量可用于训练机器学习模型，有效应用于图像分类任务。这一实验结果突显了通过调整 HOG 算法参数，我们在不同可能性之间找到最优组合，既满足了计算效率要求，又保持了对关键信息的敏感性。

2.2.2 HSV 特征提取



HSV 特征提取的过程包括将 RGB 图像转换为 HSV 颜色空间，然后对 Hue、Saturation、Value 三个通道分别进行直方图提取，每个通道提取 10 个类别，总计 30 个特征。这一步骤使得我们能够捕捉到图像中颜色的分布和强度信息，为后续的图像分析提供了更全面的视角。

将这 30 个 HSV 特征与之前提取的 160 个 HOG 特征合并，形成了一个维度为 190 的特征向量。这种综合利用 HOG 和 HSV 的特征表示方法，使得我们在特征空间中能够更好地描述图像的纹理和颜色特性，为机器学习模型提供更为丰富和全面的输入。

2.2.3 数据标准化

我们使用了 sklearn.preprocessing 中的 StandardScaler 库。该库提供了一个标准化的工具，用于对特征向量进行均值为 0、方差为 1 的标准化处理。

在训练阶段，我们使用 fit_transform()方法对训练集进行标准化，计算并应用均值和标准差；在测试阶段，使用 transform()方法对测试集进行相同的标准化，这种方式确保了在测试阶段我们不会引入关于整个数据集的额外信息，从而保持了模型对新数据的泛化性能。标准化的操作有助于提高模型的稳定性和性能，确保了特征之间的尺度一致，为机器学习模型提供更可靠的输入。

2.2.4 PCA 主成分分析

我们采用 PCA 主成分分析对特征进行降维，旨在保留数据中的主要变化，并减少数据的维度。我们选择保留累积方差达到 80%的主成分，通过 PCA 降维成功将原始特征空间的维度从 190 降至 93。通过观察模型在测试集上的性能指标，我们发现 PCA 降维后的模型，具有更高的训练效率与更好的分类效果。这表明 PCA 的应用在维持模型性能的前提下，成功减少了特征空间的复杂度，对于大规模数据集和计算资源有限的情境下尤为重要。

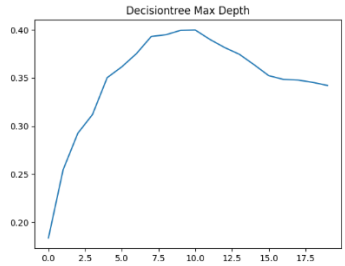
2.3 模型搭建

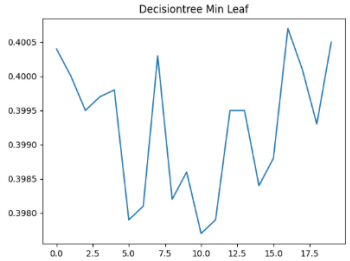
我们使用了 7 种传统分类机器学习模型，包括决策树、手写决策树、朴素贝叶斯、手写朴素贝叶斯、K 近邻（KNN）、随机森林、支持向量机（SVC）。

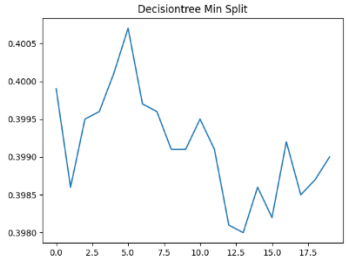
2.3.1 决策树

在决策树中，有许多超参数可以调整，其中有三个常用且影响显著的参数 max_depth、

min_samples_split、min_samples_leaf。max_depth 参数通过控制决策树的最大深度限制树的生长，有助于防止过拟合，但可能导致欠拟合。通过调整这个参数，可以平衡模型的复杂度和性能。min_samples_split 参数定义了在进行节点分裂之前，节点必须具有的最小样本数，增大这个值可以防止过拟合，使分割更加稳健。min_samples_leaf 参数定义了叶子节点必须具有的最小样本数，与 min_samples_split 类似，增大这个值有助于防止过拟合。通过如下三图的测试，纵坐标为准确率，得出 max_depth 应取 10，其余两个参数对准确率结果的影响在万分位，因此选取为默认值。

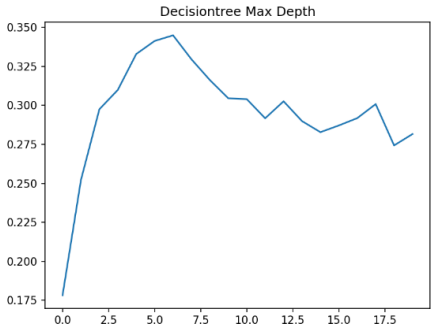






2.3.2 手写决策树

我们选择了手写和 sklearn 库中相同的 CART 决策树，可以处理 ID3 决策树处理不了的连续值特征变量。本实验只实现了最简单的 CART 决策树，并没有对该树进行剪枝或者优化。因此，对于(50000,93)的特征矩阵来说，手写的决策树无法在可接受的时间内得出结果，因此我们通过数据压缩随机抽取其中的 5000 个数据进行训练。分析 accuracy 与决策树最大深度的关系以进行超参数选择。与调库的决策树比较，我们可以发现手写的决策树由于训练数据比较少，acc 略微下降，最佳决策树深度略微减少。这可能由于没有经过特定的优化剪枝，深度增大时会发生过拟合。



2.3.3 朴素贝叶斯

朴素贝叶斯分类是一种基于贝叶斯定理的分类算法，它基于特征之间的独立性假设，利用训练数据学习类别之间的联合概率分布，并通过贝叶斯定理计算后验概率以分类。CIFAR-10 的图像数据是具有连续像素值的连续特征，因此我们使用了高斯朴素贝叶斯分类器。

2.3.4 手写朴素贝叶斯

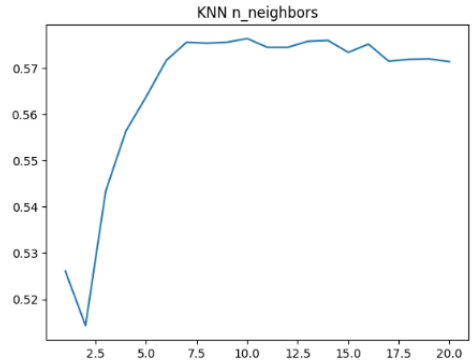
基于以上对朴素贝叶斯原理的分析，我们手写实现了朴素贝叶斯分类器，核心函数是计算后

验概率的函数，通过调用计算先验概率的函数和似然函数计算概率，然后选取后验概率最大的标签作为结果。在计算似然函数的过程中，由于特征连续，采用了高斯分布的形式，表达式为：

$$P(x|\mu,\sigma)=\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

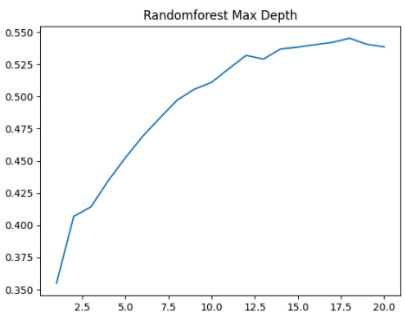
2.3.5 K 近邻

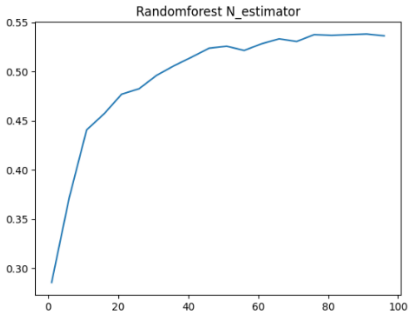
在 K 近邻中，核心参数是 n_neighbors（即通过计算样本与训练集中所有样本的距离，选择与其距离最近的 k 个训练样本，然后根据这 k 个样本的类别进行投票），其训练结果如下图所示。根据 accuracy 的大小，最终选取效果较好的超参数 n_neighbors = 10。



2.3.6 随机森林

随机森林通过构建多个决策树，通过投票或取平均值的方式进行集成。我们选取核心参数 max_depth 和 n_estimators 进行调参，max_depth 表示每棵树允许生长的最大深度，可以控制单棵决策树的复杂性。n_estimators 表示随机森林中决策树的个数，调整该参数可以在训练速度和性能之间找到平衡。由下图所示，max_depth 取 18，而 n_estimators 在大于 60 后准确值提高的幅度减小，但增加决策树个数会消耗更多的计算成本，因此综合考虑速度与性能选择 n_estimators=60。

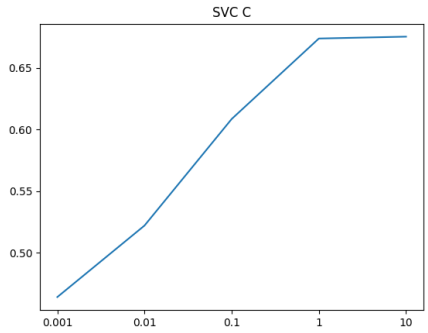




2.3.7 支持向量机

SVM 中针对分类问题的模型是 SVC，一般来说其更适用于中小型数据集，但在本项目中 SVC 的表现也不错，故纳入使用模型。SVC 有多个参数可调，我们选择了 kernel 和 C 进行调参。针对

kernel 参数我们实验了'rbf'、'li near'和'poly'三个值，根据结果的 accuracy（分别为 0.56、0.67、0.66）我们选择'rbf'核函数，其也是 SVC 最常用的核函数。针对 C 参数，我们尝试了[0.001,0.01,0.1,1,10]五个数量级，结果 accuracy 最高的是 C=1（即默认值）如下图。除此之外，SVC 存在 decision_function_shape ，它用于指定决策函数的形状，在 scikit-learn 中，该参数的取值包括'ovo'（表示采用一对一策略）和'ovr'（表示采用一对多策略）。一般而言，对于小型数据集，'ovr' 可能更有效，而对于大型数据集，'ovo' 的计算开销可能较小，因此我们选择'ovo'。



2.4 结果可视化

2.4.1 模型结果评估

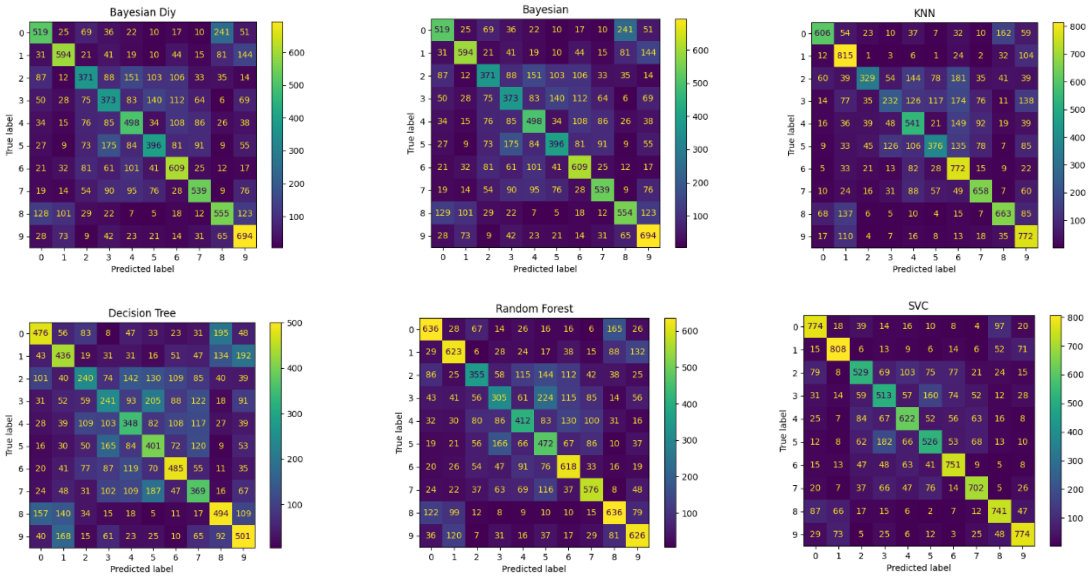
我们评估模型使用 Accuracy 和 Kappa 两种方式，Accuracy 表示准确率，即测试集中预测正确的样本数与测试集总样本数的比值。Kappa 系数是一个用于一致性检验的指标，它的计算是基于混淆矩阵的，取值为-1 到 1 之间，其值越高说明模型的预测结果与实际分类结果越一致。

表 1 各种传统分类方法在测试集上的 Accuracy 与 Kappa 值比较

	Accuracy	Kappa
决策树	0.3998	0.3331
手写决策树	0.3439	0.2710
朴素贝叶斯	0.5147	0.4608
手写朴素贝叶斯	0.5148	0.4609
K 近邻	0.5764	0.5293
随机森林	0.5259	0.4732
支持向量机	0.6740	0.6378

2.4.2 混淆矩阵

混淆矩阵是在机器学习和统计学中用于评估分类模型性能的表格，它以矩阵的形式展示模型对样本的分类结果与实际情况的对应关系。



2.5 分析和优化

在我们实验的模型中，准确率最低的模型是决策树，可能的原因是高维度的特征空间使得决策树在构建分割规则时更加复杂，容易产生过拟合而降低泛化性能。而准确率最高的模型是 SVC，其在高维空间中表现良好，可以通过核技巧能够学习复杂的非线性决策边界。

针对传统分类算法模型，我们通过调参的方式进行优化，如决策树模型和随机森林模型更改了 max_depth 参数，KNN 调整了 k 的大小，SVC 调整了核函数，均使结果有了很大的提升。除了调参的方式外，可以利用混淆矩阵的结果，考虑某类图片容易被误判为另一个特定类图片，进行特定数据预处理的优化。

3 深度学习

3.1 数据准备

与传统学习方法相同，我们在深度学习部分依然使用 CIFAR-10 数据集，不同的是在调用过程中，我们使用 torchvision 库提供的 CIFAR-10 数据集进行加载，其中包括训练集和测试集。同时，我们利用 torch 库的 DataLoader()函数创建数据加载器，用于按批次加载数据。

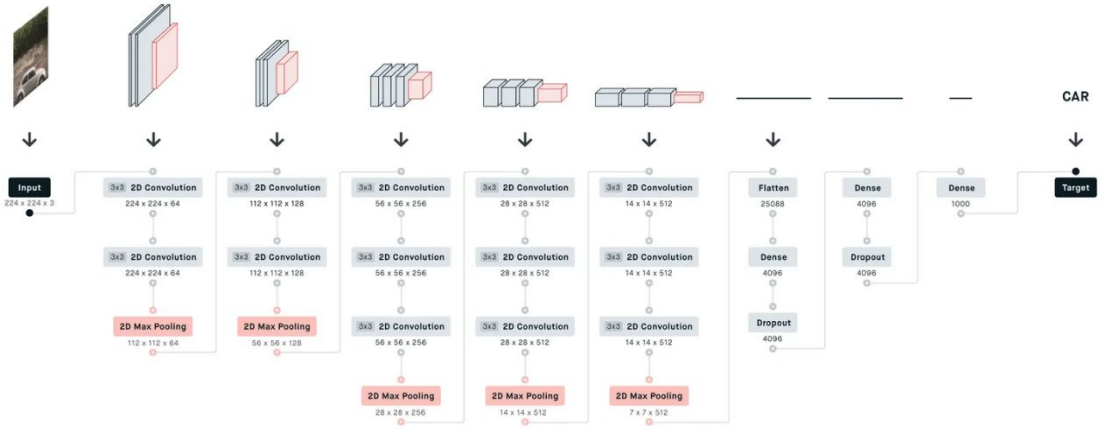
3.2 数据预处理

我们以给定的概率使用 RandomHorizontalFlip() 随机水平翻转图像以及 RandomRotation() 随机旋转图像，同时利用 ColorJitter() 改变图像的亮度、对比度和饱和度。我们还使用给定的均值和标准差对图像进行标准化处理。通过水平翻转、随机旋转和颜色变化等操作，我们向数据集引入了一些多样性，有助于提高模型的泛化能力；数据增广有助于防止过拟合和增加训练样本的多样性，模型更能够学到数据的不同变化，而不是仅仅记住特定的样本；对图像进行标准化可以确保输入模型的数据具有相似的尺度和分布，有助于模型更快地收敛。

3.3 模型搭建

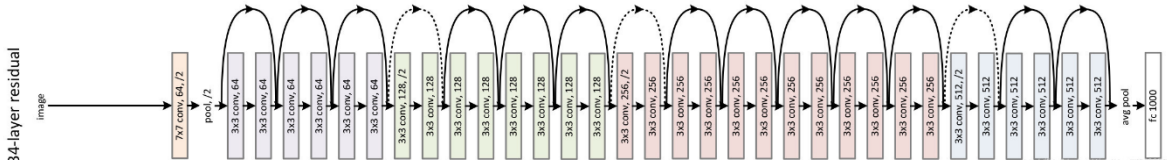
3.3.1 VGG-16 模型

VGG（Visual Geometry Group）是一个视觉几何组在 2014 年提出的深度卷积神经网络架构。VGG16 网络架构由多个卷积层和池化层交替堆叠而成，最后使用全连接层进行分类。VGG 网络被广泛应用于图像分类、目标检测、语义分割等计算机视觉任务中，并且其网络结构的简单性和易实现性使得 VGG 成为了深度学习领域的经典模型之一。



3.3.2 ResNet 模型

残差神经网络（ResNet）由微软研究院提出，其针对退化现象（Degradation）发明了快捷连接（Shortcut connection），极大的消除了深度过大的神经网络训练困难问题。ResNet 模型设计旨在克服深层神经网络梯度消失，通过残差块设计实现信息传递，同时通过批归一提高训练效果。



3.4 模型训练测试

• 批量大小（Batch Size）：不同的批量大小可能影响模型的泛化能力和训练速度。通过实验，我们选取了较为合适的 batchsize=256；

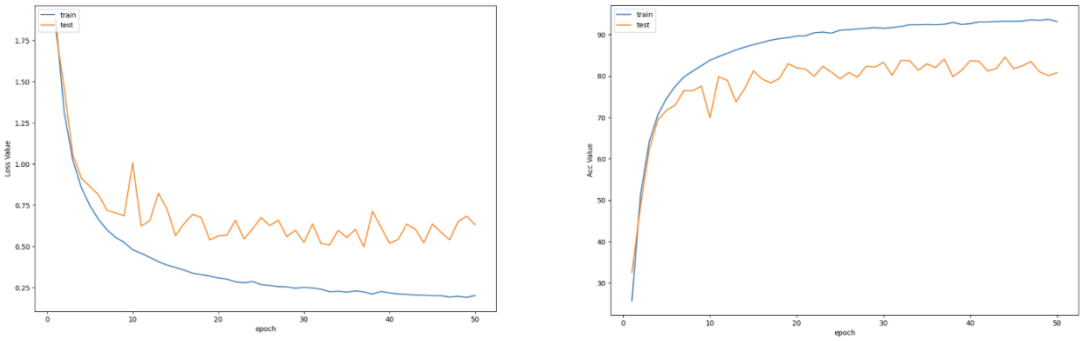
• 次数（Epoch）：适当的 epoch 数量取决于具体任务、模型复杂度和数据集规模等因素。为了更好的观察模型的 Loss、Accuracy 变化，我们选取 epoch=50，可以观察到此时模型已收敛；

• 学习率（Learning Rate）： 调整学习率可以影响模型的收敛速度和性能。通过实验，我们发现学习率在 0.001 时分类效果与速率都较佳，因此设为此值。

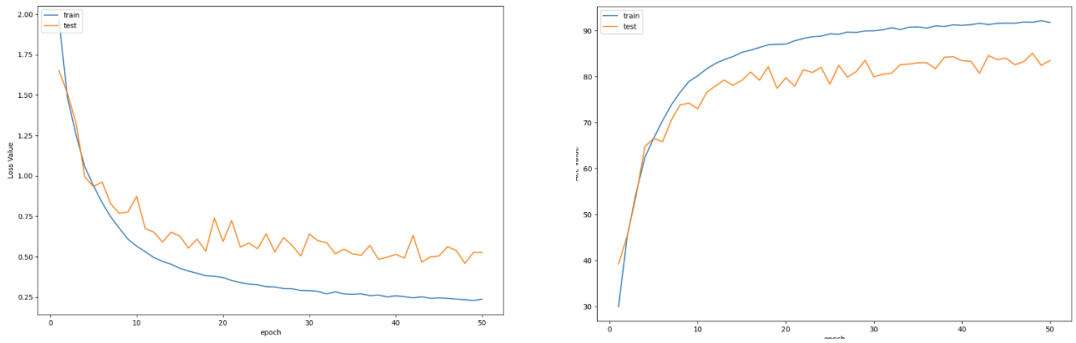
3.5 结果可视化

表 2 VGG-16 与 ResNet 在测试集上的分类准确率

种类	VGG-16	ResNet
total	81.03%	83.59%
airplane	95.30%	79.80%
automobile	95.50%	77.30%
bird	78.90%	81.20%
cat	43.60%	75.70%
deer	72.00%	86.80%
dog	88.60%	68.60%
frog	84.70%	80.70%
horse	90.70%	95.70%
ship	81.70%	92.40%
truck	81.70%	90.30%



VGG-16 模型的 Loss Curve 和 Accuracy Curve



ResNet 模型的 Loss Curve 和 Accuracy Curve

装 订 线

3.6 分析和优化

VGG16 以其简洁而深入的卷积结构而闻名，采用了连续的 3×3 卷积核和最大池化层的深层堆叠，使得网络结构清晰简单。然而，VGG16 的缺点之一是参数量较大，导致模型相对庞大，不够轻便，同时在深层网络中容易发生梯度消失问题，难以训练更深层次的网络。相比之下，ResNet 则引入了残差块的概念，通过短路连接有效地解决了梯度消失问题，使得网络可以更深地训练。ResNet 的优势在于其更好的梯度传播和网络收敛性，同时参数共享的设计使得它在训练大规模数据集时更为高效。然而，ResNet 的结构相对更为复杂，增加了网络的计算复杂度和资源需求。

针对 VGG16，优化的措施可以包括使用更轻量级的模型结构或引入一些先进的卷积结构，以减少参数数量并提高模型的计算效率。对于 ResNet，可考虑进一步优化残差块的设计，例如采用更高效的残差块变种，以提高模型在资源有限的情况下的性能。此外，可以考虑引入注意力机制或其他模块，以进一步提升模型在特定任务上的性能。

4 结语

本实验旨在通过综合运用传统机器学习方法和深度学习技术，对 CIFAR-10 图像数据集进行分类任务。该数据集包含 10 个类别的彩色图像，输入特征维度为 3×32×32，分辨率较低但内容复杂。传统方法包括决策树、随机森林、KNN、支持向量机、朴素贝叶斯，同时使用 HOG 和 HSV 特征进行图像特征提取；深度学习方法选择了 VGG16 和 ResNet。本实验主要关注各算法在多类别图像分类上的性能，并对比传统方法与深度学习方法的特点与差异。

传统分类部分中，在数据预处理阶段，我们进行了 HOG 和 HSV 特征提取、标准化、PCA 降维等一系列特征提取；在模型选择阶段，我们搭建了 7 种传统分类机器学习模型，包括决策树、朴素贝叶斯、KNN、随机森林、支持向量机等，对各模型的关键参数进行了调整和优化；结果评估我们使用了 Accuracy 和 Kappa 两种方式，通过混淆矩阵展示了模型对样本分类的性能。

深度学习部分中，我们依然使用了 CIFAR-10 数据集，对数据集进行了数据预处理，包括随机水平翻转、随机旋转、颜色变化和标准化；我们搭建了基于 VGG-16 和 ResNet 的深度学习模型，通过调节学习率、批量大小、epoch 等参数进行模型训练；结果可视化展示了训练集和测试集上的准确率和损失值，以及各个种类的预测值和图像类别对比。

对于传统方法，实验结果表明 SVC 模型在 CIFAR-10 数据集上表现最好，而决策树模型的性能较差。深度学习部分，VGG-16 和 ResNet 在测试集上均取得了不错的性能，ResNet 稍优。总而言之，本次实验拓展了我们对实际数据集的认识和处理能力，通过对不同算法的性能比较，掌握了常用的机器学习和深度学习模型的搭建和调优方法，使我们在数据科学和机器学习领域的发展中具备了更为丰富的技能和知识储备。