MNIST 实验指导

实验概览

实验目的

1. 了解 机器学习 的基本概念，掌握神经网络的基本原理。
2. 熟悉 MNIST 数据集的特点，理解其在模式识别中的应用。
3. 掌握 TensorFlow.js 的基础知识，学会使用 JavaScript 进行机器学习模型的搭建、训练和推理。
4. 提高代码实现能力，掌握模型训练、优化和评估的流程。

实验要求

1. 了解 机器学习 的基本概念。
2. 理解 MNIST 数据集的结构及其适用场景。
3. 搭建基于 TensorFlow.js 的机器学习环境。
4. 训练并测试 手写数字识别模型，分析其性能。
5. 能够对模型进行优化，提高识别精度。

提交内容

实验文件夹里除了 `node_modules` 之外的所有内容，打成压缩包。

提交地址

https://icloud.qd.sdu.edu.cn:7777/link/0D769E6A5F2102FA16227F594D4A07EF

截止日期

4月6日

---

1. 机器学习与 MNIST 数据集

1.1 机器学习

机器学习（Machine Learning, ML） 是一种让计算机从数据中自动学习规律，并在没有明确编程指令的情况下进行预测或决策的方法。它通过算法分析数据，寻找模式，并用于解决分类、回归、聚类等问题。

机器学习主要分为：

- **监督学习**：使用带标签的数据训练模型，如图像识别、语音识别。
- **无监督学习**：在无标签数据中发现结构，如聚类分析。
- **强化学习**：通过奖励机制学习最佳策略，如游戏 AI、机器人控制。

它广泛应用于推荐系统、自动驾驶、医疗诊断等领域。

在本实验中，我们使用 **监督学习**，通过 **主成分分析** 方法对图片降维，训练一个能够识别手写数字的模型。

1.2 MNIST 数据集

MNIST（Modified National Institute of Standards and Technology） 是机器学习领域中一个著名的手写数字数据集。它包含 **60,000** 张训练图片和 **10,000** 张测试图片，每张图片为 **28×28** 像素的灰度图，表示 **0 到 9** 之间的手写数字。

MNIST 数据集的特点：

1. **格式简单**：每张图片由 **784** 个像素（28×28）构成，每个像素的取值范围是 **0 到 255**，表示灰度值。
2. **标签明确**：每张图片对应一个数字 **（0~9）**，用于训练和测试分类模型。
3. **广泛应用**：MNIST 被广泛用于 **机器学习入门**，是评估图像分类算法的经典数据集。

在实验中，我们会使用 `TensorFlow.js` 处理 MNIST 数据，并训练一个神经网络模型，实现手写数字分类任务。

---

2. 数据处理

2.1 下采样

2.1.1 简介

图片下采样（Downsampling） 是一种减少图像尺寸的方法，主要用于**降低计算量**、**减少存储需求**，同时保留主要特征。

常见的下采样方法包括：

- **邻近采样（Nearest Neighbor）**：直接选择最接近的像素，简单但可能失真。
- **双线性插值（Bilinear Interpolation）**：通过周围像素加权平均，效果较平滑。
- **池化（Pooling）**：如**最大池化（Max Pooling）**，取区域内最大值，常用于**卷积神经网络（CNN）**。

下采样广泛应用于**图像处理**、**计算机视觉**和**深度学习**，如提高神经网络的计算效率。

2.1.2 常见的池化方法

最大池化（Max Pooling）

- **方法**：在一定窗口（如 2×2）内，取**最大值**作为输出。
- **特点**：保留最显著的特征，丢弃无关信息，适用于**边缘检测、目标检测等任务**。
- **示例**（2×2 池化，步长 2）：

```
输入：
[[1, 3, 2, 4],
 [5, 6, 1, 2],
 [3, 8, 2, 6],
 [7, 2, 5, 3]]

取 2×2 区域的最大值：
[[6, 4],
 [8, 6]]
```

- **优点**：突出重要特征，减少背景噪声。
- **缺点**：可能会丢失一些全局信息。

平均池化（Average Pooling）

- **方法**：在一定窗口内计算**平均值**作为输出。
- **特点**：平滑特征，减少噪声，适用于**风格迁移、低级特征提取**。
- **示例**（2×2 池化，步长 2）：

```
输入：
[[1, 3, 2, 4],
 [5, 6, 1, 2],
 [3, 8, 2, 6],
 [7, 2, 5, 3]]

取 2×2 区域的平均值：
[[3.75, 2.25],
 [5.00, 4.00]]
```

- **优点**：保留更多背景信息，减少过拟合。
- **缺点**：可能会导致特征模糊，影响分类效果。

2.2 主成分分析（Principal Component Analysis）

2.2.1 简介

主成分分析 是一种**降维**方法，用于从高维数据中提取主要特征，同时减少数据维度，常用于**数据压缩、特征提取、可视化**等任务。它通过**找到数据的主要方向（主成分）**，使得尽可能多的信息被保留在较少的维度中。

2.2.2 算法

假设数据集有 n 个样本，每个样本有 d 维特征，**PCA** 的计算步骤如下：

1. 数据标准化

- 计算每个特征的均值，并减去均值，使数据中心化。

2. 计算协方差矩阵

- 协方差矩阵表示特征之间的相关性，公式为：

$$C = \frac{1}{n-1} X^T X$$

3. 计算特征值和特征向量

- 通过特征分解或 SVD 方法，求协方差矩阵的特征值和特征向量。
- 特征向量表示主成分的方向，特征值表示主成分的重要性。

4. 选择前 k 个主成分（最大特征值对应的特征向量）

- 选取前 k 个最大特征值对应的特征向量，作为新的低维基。

5. 数据投影到新空间

- 用选定的主成分矩阵 W 变换原始数据：

$$X' = XW$$

- 得到降维后的数据。

2.2.2 应用

- **数据降维**：如图像压缩、去除冗余信息。
- **数据可视化**：将高维数据降至 2D/3D 方便分析。
- **去噪**：保留主要特征，减少噪声干扰。

3. TensorFlow.js

3.1 TensorFlow.js 简介

TensorFlow.js（tfjs） 是 Google 开发的 **基于 JavaScript 的机器学习库**，它支持：

- 在 **浏览器** 端运行神经网络，无需服务器支持。
- 使用 **GPU 加速**，提升计算效率。
- 加载和部署 **预训练模型**，或从头训练新的模型。

3.2 TensorFlow.js 的核心功能

1. 创建张量（Tensors）：
   - 在 `tfjs` 中，数据以 **张量（Tensor）** 形式表示，如 `tf.tensor2d()` 处理二维数据。
2. 搭建神经网络：
   - 通过 `tf.sequential()` 创建模型。
   - 添加不同类型的神经网络层，如 `tf.layers.dense()`（全连接层）、`tf.layers.conv2d()`（卷积层）。
3. 训练模型：
   - 通过 `model.fit()` 进行训练，使用 MNIST 数据优化模型参数。
4. 模型推理：
   - 通过 `model.predict()` 在浏览器端进行手写数字识别。

4. MNIST 数字识别

4.1 实验简介

本次实验要求使用 TensorFlow.js 进行 MNIST 识别。

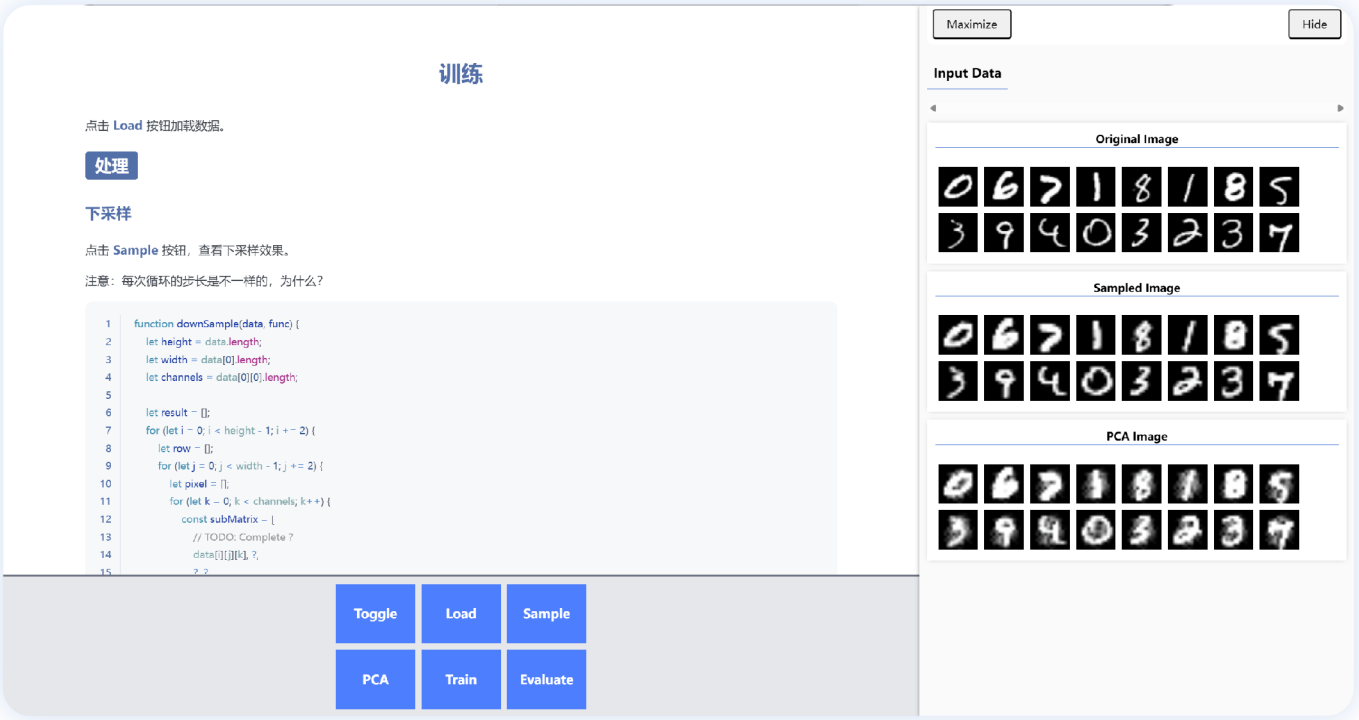
为了加速模型训练，首先对输入图片进行 下采样，减少分辨率，以降低数据的维度和计算量。接着，使用 主成分分析 对降维后的图片数据进行进一步处理，提取主要特征并去除冗余信息。最后，将经过 主成分分析 处理后的特征输入到分类模型中进行训练，以提高训练效率，同时尽可能保留关键信息，确保分类精度。

具体实验流程参考压缩包里的 `readme.md`

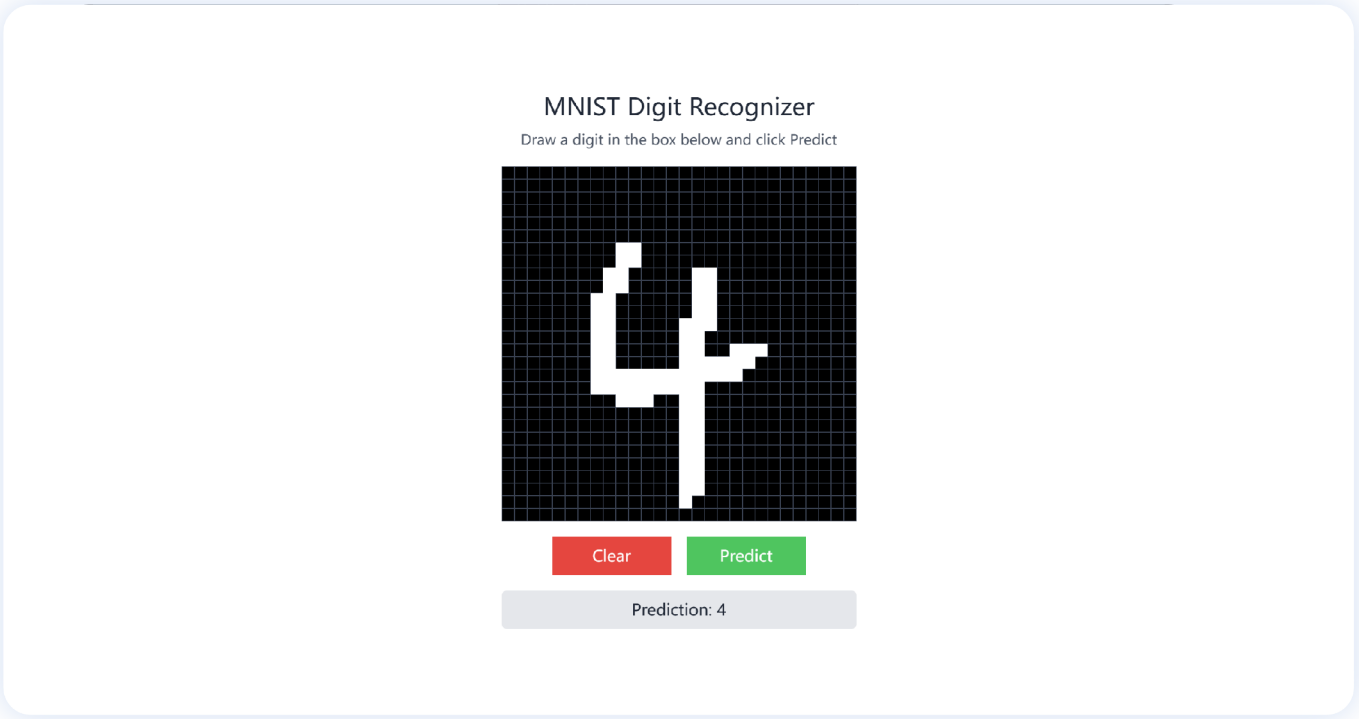
4.2 实验流程

4.2.1 数据

4.2.2 训练


训练
点击 Load 按钮加载数据。
处理
下采样
点击 Sample 按钮，查看下采样效果。
注意：每次循环的步长是不一样的，为什么？
function downSample(data, func) {
let height = data.length;
let width = data[0].length;
let channels = data[0][0].length;
let result = [];
for (let i = 0; i < height - 1; i += 2) {
let row = [];
for (let j = 0; j < width - 1; j += 2) {
let pixel = [];
for (let k = 0; k < channels; k++) {
const subMatrix = [
// TODO: Complete ?
data[i][j][k], ?,
? ?
Toggle
Load
Sample
PCA
Train
Evaluate
Maximize
Hide
Input Data
Original Image
Sampled Image
PCA Image

4.2.3 实践


MNIST Digit Recognizer
Draw a digit in the box below and click Predict
Clear
Predict
Prediction: 4