机器学习导论 习题三

学号, 姓名, 邮箱

2024 年 6 月 5 日

作业提交注意事项

1. 作业所需的 LaTeX 及 Python 环境配置要求请参考: [Link];
2. 请在 LaTeX 模板中第一页填写个人的学号、姓名、邮箱;
3. 本次作业需提交的文件与对应的命名方式为:
   (a) 作答后的 LaTeX 代码 — `HW3.tex`;
   (b) 由 (a) 编译得到的 PDF 文件 — `HW3.pdf`;
   (c) 第三题模型代码 — `p3_models.py`;
   (d) 第四题模型代码 — `p4_models.py`;
   (e) 第四题训练代码 — `p4_trainer.py`.

   请将以上文件**打包为 学号_姓名.zip** (例如 221300001_张三.zip) 后提交;
3. 若多次提交作业, 则在命名 .zip 文件时加上版本号, 例如 221300001_张三_v1.zip" (批改时以版本号最高的文件为准);
4. 本次作业提交截止时间为 **5 月 14 日 23:59:59**. 未按照要求提交作业, 提交作业格式不正确, **作业命名不规范**, 将会被扣除部分作业分数; 除特殊原因 (如因病缓交, 需出示医院假条) 逾期未交作业, 本次作业记 0 分; **如发现抄袭, 抄袭和被抄袭双方成绩全部取消**;
5. 学习过程中, 允许参考 ChatGPT 等生成式语言模型的生成结果, 但必须在可信的信息源处核实信息的真实性; **不允许直接使用模型的生成结果作为作业的回答内容**, 否则将视为作业非本人完成并取消成绩;
6. 本次作业提交地址为 [Link], 请大家预留时间提前上交, 以防在临近截止日期时, 因网络等原因无法按时提交作业.

1 [25pts] Principal Component Analysis

主成分分析是一种经典且常用的数据降维方法. 请仔细阅读学习《机器学习》第十章 10.3 节, 并根据图 10.5 中的算法内容, 完成对如下 6 组样本数据的主成分分析.

$$\mathbf{X}=\begin{bmatrix}2&3&3&4&5&7\\2&4&5&5&6&8\end{bmatrix}$$

(1) **[6pts]** 试求样本数据各维的均值、标准差.

(2) **[7pts]** 试求标准化后的样本矩阵 $\mathbf{X}_{\text{std}}$, 以及 $\mathbf{X}_{\text{std}}$ 对应的协方差矩阵.

(3) **[7pts]** 试求协方差矩阵对应的特征值, 以及投影矩阵 $\mathbf{W}^{\star}$.

(4) **[5pts]** 如果选择重构阈值 $t=95\%$, 试求 PCA 后样本 $\mathbf{X}_{\text{std}}$ 在新空间的坐标矩阵.

Solution. 此处用于写解答 (中英文均可)

(1) 设样本中心 $\bar{x}=\left(\bar{x}^{(1)},\bar{x}^{(2)}\right)^T$, 标准差 $s(x)=\left(std\left(x^{(1)}\right),std\left(x^{(2)}\right)\right)^T$, 则可算得:

$$\begin{cases}\bar{x}^{(1)}=\frac{2+3+3+4+5+7}{6}=4\\ \bar{x}^{(2)}=\frac{2+4+5+5+6+8}{6}=5\end{cases}$$

$$\begin{cases}std\left(x^{(1)}\right)=\sqrt{\frac{1}{5}\left[(2-4)^2+(3-4)^2+(3-4)^2+(4-4)^2+(5-4)^2=(7-4)^2\right]}=\frac{4\sqrt{5}}{5}\\ std\left(x^{(2)}\right)=\sqrt{\frac{1}{5}\left[(2-5)^2+(4-5)^2+(5-5)^2+(5-5)^2+(6-5)^2=(8-5)^2\right]}=2\end{cases}$$

(2) 对样本矩阵标准化处理, 得到标准后的样本矩阵如下:

$$\mathbf{X}_{\text{std}}=\begin{bmatrix}-\frac{\sqrt{5}}{2}&-\frac{\sqrt{5}}{4}&-\frac{\sqrt{5}}{4}&0&\frac{\sqrt{5}}{4}&\frac{3\sqrt{5}}{4}\\-\frac{3}{2}&-\frac{1}{2}&0&0&\frac{1}{2}&\frac{3}{2}\end{bmatrix}$$

计算样本的协方差矩阵如下:

$$\begin{aligned}\mathbf{S}=&\frac{1}{n-1}\mathbf{X}_{\text{std}}\mathbf{X}_{\text{std}}^T\\ =&\frac{1}{5}\begin{pmatrix}-\frac{\sqrt{5}}{2}&-\frac{\sqrt{5}}{4}&-\frac{\sqrt{5}}{4}&0&\frac{\sqrt{5}}{4}&\frac{3\sqrt{5}}{4}\\-\frac{3}{2}&-\frac{1}{2}&0&0&\frac{1}{2}&\frac{3}{2}\end{pmatrix}\begin{pmatrix}-\frac{\sqrt{5}}{2}&-\frac{3}{2}\\-\frac{\sqrt{5}}{4}&-\frac{1}{2}\\-\frac{\sqrt{5}}{4}&0\\0&0\\\frac{\sqrt{5}}{4}&\frac{1}{2}\\\frac{3\sqrt{5}}{4}&\frac{3}{2}\end{pmatrix}\\ =&\frac{1}{5}\begin{pmatrix}5&\frac{17}{8}\sqrt{5}\\\frac{17}{8}\sqrt{5}&5\end{pmatrix}=\begin{pmatrix}1&\frac{17}{40}\sqrt{5}\\\frac{17}{40}\sqrt{5}&1\end{pmatrix}\end{aligned}$$

(3) 计算 $\mathbf{S}$ 的特征值, 由 $|\mathbf{S}-\lambda\mathbf{I}|=0$, 有 $\begin{vmatrix}1-\lambda&\frac{17}{40}\sqrt{5}\\\frac{17}{40}\sqrt{5}&1-\lambda\end{vmatrix}=0$ 。解得:

$$\begin{cases}\lambda_1=1+\frac{17}{40}\sqrt{5}\\ \lambda_2=1-\frac{17}{40}\sqrt{5}\end{cases}$$

由 $\mathbf{S}\boldsymbol{w}_1 = \lambda_1 \boldsymbol{w}_1$, 可解得 $\boldsymbol{w}_1 = \left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right)^T$; 由 $\mathbf{S}\boldsymbol{w}_2 = \lambda_2 \boldsymbol{w}_2$, 可解得 $\boldsymbol{w}_2 = \left(-\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right)^T$. 由此可算得:

$$\mathbf{W}^{\star} = \begin{pmatrix} \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{pmatrix}$$

(4) **[5pts]** 如果选择重构阈值 $t = 95\%$, 试求 PCA 后样本在新空间的坐标矩阵.

由 (3) 问结论可得, 第一主成分的方差贡献率 $r_1 = \frac{\lambda_1}{\lambda_1 + \lambda_2} \approx 97.5\% > 95\%$, 因此进行 PCA 选取 $d' = 1$. 计算 PCA 后样本在新空间的坐标矩阵为:

$$\mathbf{X}_{\text{new}} = \boldsymbol{w}_1^T \mathbf{X}_{\text{std}} = \left(\frac{-\sqrt{10} - 3\sqrt{2}}{4}, \frac{-\sqrt{10} - 2\sqrt{2}}{8}, -\frac{\sqrt{10}}{8}, 0, \frac{\sqrt{10} + 2\sqrt{2}}{8}, \frac{3\sqrt{10} + 6\sqrt{2}}{8}\right)$$

2 [25pts] Support Vector Machines

核函数是 SVM 中常用的工具, 其在机器学习中有着广泛的应用与研究. 请仔细阅读学习《机器学习》第六章, 并回答如下问题.

(1) **[6pts]** 试判断下图 ① 到 ⑥ 中哪些为支持向量.

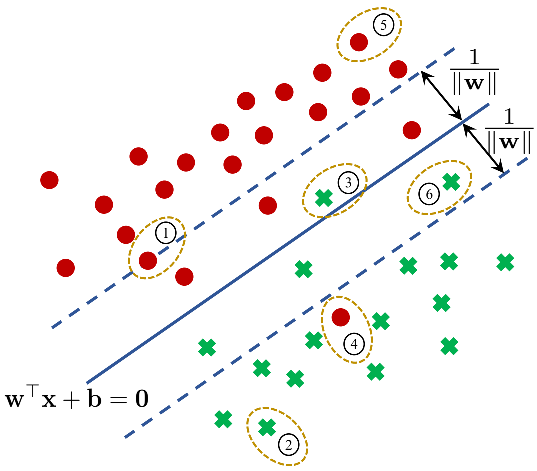

图 1: 分离超平面示意图

(2) **[5pts]** 试判断 $\kappa(\boldsymbol{x},\mathbf{z}) = (\langle \boldsymbol{x},\mathbf{z}\rangle + 1)^2$ 是否为核函数, 并给出证明或反例.

(3) **[5pts]** 试判断 $\kappa(\boldsymbol{x},\mathbf{z}) = (\langle \boldsymbol{x},\mathbf{z}\rangle - 1)^2$ 是否为核函数, 并给出证明或反例.

(4) **[9pts]** 试证明: 若 κ_1 和 κ_2 为核函数, 则两者的直积

$$\kappa_1 \otimes \kappa_2(\boldsymbol{x},\mathbf{z}) = \kappa_1(\boldsymbol{x},\mathbf{z})\kappa_2(\boldsymbol{x},\mathbf{z})$$

也是核函数. 即证明《机器学习》(6.26) 成立.

(**Hint:** 利用核函数与核矩阵的等价性.)

Solution. 此处用于写解答 (中英文均可)

(1) 按照支持向量的定义, ① ③ ④ ⑥ 是支持向量， 而 ② ⑤ 不是.

(2) $\kappa(\boldsymbol{x},\mathbf{z}) = (\langle \boldsymbol{x},\mathbf{z}\rangle + 1)^2$ 是核函数. $\kappa(\boldsymbol{x},\mathbf{z}) = (\langle \boldsymbol{x},\mathbf{z}\rangle + 1)^2 = \left(\left\langle \begin{pmatrix}\boldsymbol{x}\\1\end{pmatrix}, \begin{pmatrix}\mathbf{z}\\1\end{pmatrix}\right\rangle\right)^2$ 对应的核矩阵是半定的.

(3) 该函数不能作为核函数, 给出反例如下.

考虑一维变量, 并取数据集 $D = \{\boldsymbol{x}_1, \boldsymbol{x}_2\}$（$\boldsymbol{x}_1 = 2, \boldsymbol{x}_2 = -2$）. 此时对应的核矩阵（kernel matrix）为:

$$\mathbf{K} = \begin{bmatrix} 9 & 25 \\ 25 & 9 \end{bmatrix}$$

其行列式为 $|\mathbf{K}| = 9 \times 9 - 25 \times 25 = -544 < 0$, 说明其存在负特征值, 不为半正定矩阵. 故 $\kappa(\boldsymbol{x}, \mathbf{z}) = (\langle \boldsymbol{x}, \mathbf{z} \rangle - 1)^2$ 不是核函数.

(4) 考虑到核函数与核矩阵的充要关系, 本题等价于证明: 若矩阵 $\mathbf{A} = \{a_{ij}\}_{m\times m}, \mathbf{B} = \{b_{ij}\}_{m\times m}$ 均为半正定矩阵, 则矩阵 $\mathbf{H} = \{a_{ij}b_{ij}\}_{m\times m}$ 也为半正定矩阵. 下证明该结论. 同时因为 $\mathbf{A}$ 半正定, 由半定矩阵的性质可知, 存在 $\mathbf{C} \in \mathbb{R}^{m\times m}$ 使得 $\mathbf{A} = \mathbf{C}^\top \mathbf{C}$, 即 $a_{ij} = \sum_{k=1}^m c_{ik}c_{jk}$. 因此, 任取 $\boldsymbol{x} \in \mathbb{R}^m$, 成立:

$$
\begin{aligned}
\boldsymbol{x}^\top \mathbf{H} \boldsymbol{x} &= \sum_{i,j} a_{ij} b_{ij} x_i x_j \\
&= \sum_{i,j} \left(\sum_{k=1}^m c_{ik} c_{jk} \right) b_{ij} x_i x_j \\
&= \sum_{k=1}^m \left[\sum_{i,j} b_{ij} \left(c_{ik} x_i\right) \left(c_{jk} x_j\right) \right]
\end{aligned}
$$

同时, 因为 $\mathbf{B}$ 也为半正定矩阵, 因此对于任意 k, 成立:

$$
\sum_{i,j} b_{ij} \left(c_{ik} x_i\right) \left(c_{jk} x_j\right) \geq 0
$$

故对任意 $\boldsymbol{x} \in \mathbb{R}^m$, 成立 $\boldsymbol{x}^T \mathbf{H} \boldsymbol{x} \geq 0$, 即 $\mathbf{H}$ 也为半正定矩阵, 证毕.

3 [30pts] Basics of Neural Networks

多层前馈神经网络可以被用作分类模型. 在本题中, 我们先回顾前馈神经网络的一些基本概念, 再利用 Python 实现一个简单的前馈神经网络以进行分类任务.

[基础原理] 首先, 考虑一个多层前馈神经网络, 规定网络的输入层是第 0 层, 输入为 $\mathbf{x} \in \mathbb{R}^d$. 网络有 M 个隐层, 第 h 个隐层的神经元个数为 N_h, 输入为 $\mathbf{z}_h \in \mathbb{R}^{N_{h-1}}$, 输出为 $\mathbf{a}_h \in \mathbb{R}^{N_h}$, 权重矩阵为 $\mathbf{W}_h \in \mathbb{R}^{N_{h-1} \times N_h}$, 偏置参数为 $\mathbf{b}_h \in \mathbb{R}^{N_h}$. 网络的输出层是第 $M+1$ 层, 神经元个数为 C, 权重矩阵为 $W_{M+1} \in \mathbb{R}^{N_M \times C}$, 偏置参数为 $\mathbf{b}_{M+1} \in \mathbb{R}^C$, 输出为 $\mathbf{y} \in \mathbb{R}^C$. 网络隐层和输出层的激活函数均为 f, 网络训练时的损失函数为 $\mathcal{L}$, 且 f 与 $\mathcal{L}$ 均可微.

(1) **[5pts]** 请根据前向传播原理, 给出 $\mathbf{z}_h, \mathbf{a}_h$ $(1 \le h \le M)$ 及 $\mathbf{y}$ 的具体数学表示.

(2) **[5pts]** 结合 (2) 的表示形式, 谈谈为何要在神经网络中引入 (非线性) 激活函数 f?

[编程实践] 下面, 我们针对一个特征数 $d = 2$, 类别数为 2 的分类数据集, 实现一个结构为 “2-2-1” 的简单神经网络, 即: 输入层有 2 个神经元; 隐层仅一层, 包含 2 个神经元; 输出层有 1 个神经元; 所有层均使用 Sigmoid 作为激活函数. 此外, 我们使用 BP 算法进行神经网络的训练. **关于本题的细节介绍及具体要求, 请见附件: `p3_` 编程题说明.** 请参考编程题说明文档与附件中的代码模板, 完成下面的任务.

(3) **[15pts]** 基于 `p3_models.py`, 补全缺失代码, 实现神经网络分类器的训练与预测功能.

(4) **[5pts]** 参考《机器学习》及第一次作业中对超参数调节流程的介绍, 为 (1) 中模型设置合适的超参数 (即: 学习率与迭代轮数). **请将选择的超参数设置为调用模型时的默认参数, 并在解答区域简要介绍你的超参数调节流程.**
(提示: 可以从数据集划分方法, 评估方法, 候选超参数生成方法等角度说明).

Solution. 此处用于写解答 (中英文均可)

(1) 参考《机器学习》的表述, 令 $\mathbf{a}_0 = \mathbf{x}$, 则 $\mathbf{z}_h, \mathbf{a}_h$ $(1 \le h \le M)$ 以及 $\mathbf{y}$ 的具体形式如下:

$$
\begin{aligned}
\mathbf{z}_h &= \mathbf{W}_h^\top \mathbf{a}_{h-1}, && (1 \le h \le M) \\
\mathbf{a}_h &= f(\mathbf{z}_h + \mathbf{b}_h), && (1 \le h \le M) \\
\mathbf{y} &= f(\mathbf{W}_{M+1}^\top \mathbf{a}_M + \mathbf{b}_{M+1}).
\end{aligned}
$$

(亦存在 $\mathbf{z}_h = \mathbf{W}_h^\top \mathbf{a}_{h-1}$ 的表述; 只要结果无误, 这两种表述均视为正确.)

(2) 若不引入激活函数 (即: $f(x) = x$ 的情况), 无论神经网络的深度, 宽度如何, 输出 $\mathbf{y}$ 永远只是输入 $\mathbf{x}$ 与偏置 $\mathbf{b}$ 的线性组合. 非线性激活函数引入了非线性变换, 有利于神经网络在使用数量较少的神经元时, 依然有较强的表示及拟合能力.

(3) 请参考附件: `p3_models_sol.py`.

(4) 可通过交叉验证等方式挑选最优的超参数, 调参流程合理即可.

4 [20(+5)pts] Neural Networks with PyTorch

> 在上一题的编程实践中, 我们使用 Python 实现了一个简单的神经网络分类器. 其中, 我们根据 BP 算法中神经网络参数梯度的数学定义, 手动实现了梯度计算及参数更新的流程. 然而, 在现实任务中, 我们往往利用*深度学习框架*来进行神经网络的开发及训练. 一些常用的框架例如: PyTorch, Tensorflow 或 JAX, 以及国产的 PaddlePaddle, MindSpore. 这类框架往往支持*自动微分*功能, 仅需定义神经网络的具体结果与前向传播过程, 即可在训练时自动计算参数的梯度, 进行参数更新. 此外, 我们可以使用由框架实现的更成熟的优化器 (如 Adam 等) 来提高模型的收敛速度, 或使用 GPU 加速以提高训练效率. 如果希望在今后的学习科研中应用神经网络, 了解至少一种框架的使用方式是极为有益的.

在本题中, 我们尝试使用 PyTorch 框架来进行神经网络的开发, 完成 FashionMNIST 数据集上的图像分类任务. 与上一题考察神经网络底层原理不同, 本题考察大家阅读文档, 搭建模型并解决实际任务的能力. **关于本题的细节介绍及具体要求, 请见附件: `p4_`** 编程题说明**.** 请参考编程题说明文档与附件中的代码模板, 完成下面的任务.

(1) **[10pts]** 阅读文档, 配置 PyTorch 环境, 补全 `p4_models.py` 中神经网络的 `__init__` 与 `forward` 方法, 最终成功运行 `p4_main.py`. **请在解答区域附上运行 `p4_main.py` 后生成的 `plot.png`.**

(2) **[10pts]** 从 (1) 中生成的训练过程图片 `plot.png` 中可以看出: 模型明显出现了**过拟合**现象, 即训练一定轮次后, 训练集 loss 持续下降, 但测试集 loss 保持不变或转为上升. 请提出**至少两种**缓解过拟合的方法, 分别通过编程实现后, 在解答区域附上应用前后的训练过程图片, 并结合图片简要分析方法有效/无效的原因.
(提示: 可以考虑的方法包括但不限于: Dropout, 模型正则化, 数据增强等.)

(3) **[5pts]** (本题为附加题, 得分计入卷面分数, 但本次作业总得分不超过 100 分)
寻找最优的改进神经网络结构及训练方式的方法, 使模型在另一个未公开的测试集上取得尽可能高的分类准确率.
本题得分规则如下: *假设共有 N 名同学完成本题, 我们将这 N 名同学的模型测试集分类准确率由高到低排列, 对前 $K = \min(\lfloor N/10 \rfloor, 10)$ 名同学奖励附加题分数. 对于排列序号为 i 的同学 ($1 \le i \le K$), 得分为: $5 - \lfloor 5(i-1)/k \rfloor$.*
(提示: 你可以自由尝试修改模型结构, 修改优化器超参数等方法.)

Solution. 此处用于写解答 (中英文均可)

(1) 请参考附件: `p4_models_sol.py`. 运行后的 `plot.py` 样例如下图所示.

(2) 缓解过拟合的方法原理及实现正确, 分析合理即可. 以下给出一个通过设置 `weight_decay=0.005`, 实现 L2 正则化的效果样例.

(3) 根据实际完成情况排名赋分.

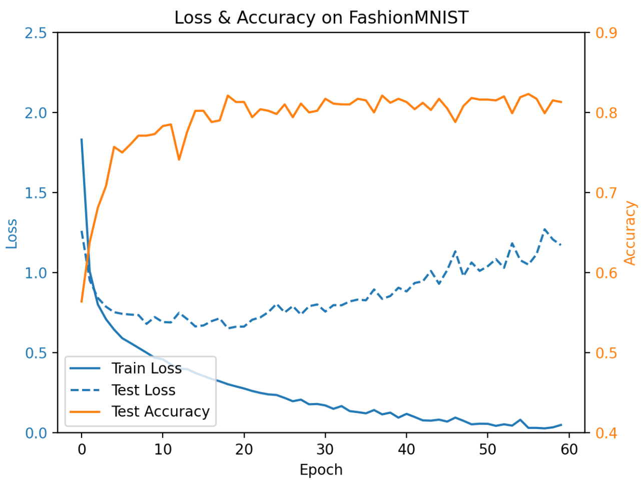

图 2: 训练过程图片样例

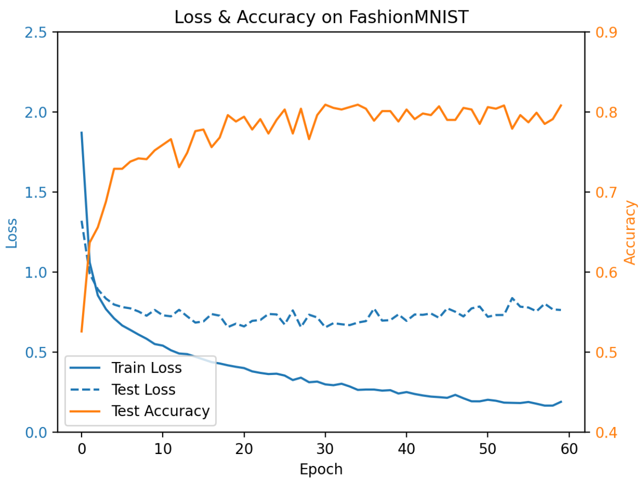

图 3: 使用 L2 正则化后的训练过程图片样例

Acknowledgments

允许与其他同样未完成作业的同学讨论作业的内容, 但需在此注明并加以致谢; 如在作业过程中, 参考了互联网上的资料, 且对完成作业有帮助的, 亦需注明并致谢.