2024 秋季高级机器学习
习题一参考答案

2025.1.17

一. (30 points) **机器学习导论复习题 (前九章)**

1. (10 points) 给定包含 m 个样例的数据集 $D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_m, y_m)\}$, 其中 $y_i \in \mathbb{R}$ 为 $\boldsymbol{x}_i$ 的实数标记, $\boldsymbol{x}_i = (x_{i1}; x_{i2}; \cdots; x_{id}) \in \mathbb{R}^d$。针对数据集 D 中的 m 个示例, 教材 3.2 节所介绍的 "线性回归" 模型要求该线性模型的预测结果和其对应的标记之间的误差之和最小:

$$\begin{aligned}(\boldsymbol{w}^*, b^*) &= \frac{1}{2}\underset{(\boldsymbol{w},b)}{\arg\min}\sum_{i=1}^{m}\left(f\left(\boldsymbol{x}_i\right) - y_i\right)^2 \\ &= \frac{1}{2}\underset{(\boldsymbol{w},b)}{\arg\min}\sum_{i=1}^{m}\left(y_i - \left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right)\right)^2,\end{aligned} \tag{1}$$

即寻找一组权重 $(\boldsymbol{w}, b)$, 使其对 D 中示例预测的整体误差最小。定义 $\boldsymbol{y} = [y_1; \ldots; y_m] \in \mathbb{R}^m$, 且 $\boldsymbol{X} = \left[\boldsymbol{x}_1^\top; \boldsymbol{x}_2^\top; \cdots; \boldsymbol{x}_m^\top\right] \in \mathbb{R}^{m\times d}$, 线性回归的优化过程可以使用矩阵进行表示:

$$\begin{aligned}(\boldsymbol{w}^*, b^*) &= \underset{\boldsymbol{w},b}{\arg\min}\frac{1}{2}\left(\boldsymbol{X}\boldsymbol{w} + \mathbf{1}_m b - \boldsymbol{y}\right)^\top\left(\boldsymbol{X}\boldsymbol{w} + \mathbf{1}_m b - \boldsymbol{y}\right) \\ &= \underset{\boldsymbol{w},b}{\arg\min}\frac{1}{2}\left\|\boldsymbol{X}\boldsymbol{w} + \mathbf{1}_m b - \boldsymbol{y}\right\|_2^2,\end{aligned} \tag{2}$$

其中, $\mathbf{1}_m \in \mathbb{R}^m$ 为元素全为 1 、长度为 m 的向量。在实际问题中, 我们常常会遇到示例相对较少, 而特征很多的场景。在这类情况中如果直接求解线性回归模型, 较少的示例无法获得唯一的模型参数, 会具有多个模型能够" 完美" 拟合训练集中的所有样例。此外, 模型很容易过拟合。为缓解这些问题, 常引入正则化项 $\Omega(\boldsymbol{w})$, 通常形式如下:

$$\boldsymbol{w}^*_{\text{Ridge}}, b^*_{\text{Ridge}} = \underset{\boldsymbol{w},b}{\arg\min}\frac{1}{2}\left\|\boldsymbol{X}\boldsymbol{w} + \mathbf{1}_m b - \boldsymbol{y}\right\|_2^2 + \lambda\Omega(\boldsymbol{w}), \tag{3}$$

其中, $\lambda > 0$ 为正则化参数。正则化表示了对模型的一种偏好, 例如 $\Omega(\boldsymbol{w})$ 一般对模型的复杂度进行约束, 因此相当于从多个在训练集上表现同等预测结果的模型中选出模型复杂度最低的一个。考虑岭回归问题, 即设置正则项 $\Omega(\boldsymbol{w}) = \|\boldsymbol{w}\|_2^2$。

(1) (3 points) 请证明对于任何矩阵 $\boldsymbol{X} \in \mathbb{R}^{m\times d}$, 下式均成立

$$\left(\boldsymbol{X}\boldsymbol{X}^\top + \lambda\boldsymbol{I}_m\right)^{-1}\boldsymbol{X} = \boldsymbol{X}\left(\boldsymbol{X}^\top\boldsymbol{X} + \lambda\boldsymbol{I}_d\right)^{-1}. \tag{4}$$

(2) (7 points) 请给出岭回归的最优解 $\boldsymbol{w}^*_{\text{Ridge}}$ 和 b^*_{Ridge} 的闭式解表达式, 并使用矩阵形式表示, 分析其最优解和原始线性回归最优解 $\boldsymbol{w}^*_{\text{LS}}$ 和 b^*_{LS} 的区别。

2. (10 points) 教材 4.2 节中给出度量样本集合纯度的常用指标, 从而衍生出决策树属性选择的常用准则。假设决策树分类问题中标记空间 $\mathcal{Y}$ 的大小为 $|\mathcal{Y}|$，训练集 D 中第 k 类样本所占比例为 $p_k(k=1,2,\ldots,|\mathcal{Y}|)$ 。请回答以下问题:

(1) (3 points) 信息熵 $\mathrm{Ent}(D)$ 定义如下

$$\mathrm{Ent}(D)=-\sum_{k=1}^{|\mathcal{Y}|}p_k\log_2 p_k, \tag{5}$$

请证明信息熵的上下界为

$$0\leq \mathrm{Ent}(D)\leq \log_2|\mathcal{Y}|, \tag{6}$$

并给出等号成立的条件。

(2) (3 points) 除信息熵外，教材中也介绍了基尼指数衡量纯度，定义如下

$$\sum_{k=1}^{|\mathcal{Y}|}\sum_{k'\neq k}p_k p_{k'}=1-\sum_{k=1}^{|\mathcal{Y}|}p_k^2, \tag{7}$$

由于在决策树叶结点中使用包含样例最多的类别作为其预测结果，因此也可使用误分类错误率

$$1-\max_k p_k, \tag{8}$$

作为衡量指标。请给出二分类问题（$|\mathcal{Y}|=2$，正类所占比例为 p, 负类为 $1-p$）下三种衡量标准的表达式。

(3) (4 points) 在 ID3 决策树的生成过程中, 需要计算信息增益以生成新的结点。设离散属性 a 有 V 个可能取值 $\left\{a^1,a^2,\cdots,a^V\right\}$ ，请参考教材 4.2.1 节相关符号的定义证明:

$$\mathrm{Gain}(D,a)=\mathrm{Ent}(D)-\sum_{v=1}^{V}\frac{|D^v|}{|D|}\mathrm{Ent}\left(D^v\right)\geq 0, \tag{9}$$

即信息增益非负。

3. (10 points) 给定训练集 $D=\{(\boldsymbol{x}_1,\boldsymbol{y}_1),(\boldsymbol{x}_2,\boldsymbol{y}_2),\ldots,(\boldsymbol{x}_m,\boldsymbol{y}_m)\}$. 其中 $\boldsymbol{x}_i\in\mathbb{R}^d,\boldsymbol{y}_i\in\mathbb{R}^l$ 表示输入示例由 d 个属性描述，输出 l 维实值向量. 教材图 5.7 给出了一个有 d 个输入神经元、l 个输出神经元、q 个隐层神经元的多层神经网络，其中输出层第 j 个神经元的阈值用 θ_j 表示，隐层第 h 个神经元的阈值用 γ_h 表示。输入层第 i 个神经元与隐层第 h 个神经元之间的连接权为 v_{ih} ，隐层第 h 个神经元与输出层第 j 个神经元之间的连接权为 w_{hj} 。记隐层第 h 个神经元接收到的输入为 $\alpha_h=\sum_{i=1}^{d}v_{ih}x_i$, 输出层第 j 个神经元接收到的输入为 $\beta_j=\sum_{h=1}^{q}w_{hj}b_h$ ，其中 b_h 为隐层第 h 个神经元的输出。

不同任务中神经网络的输出层往往使用不同的激活函数和损失函数，本题介绍几种常见的激活和损失函数，并对其梯度进行推导。

(1) (3 points) 在二分类问题中 ($l=1$), 标记 $y\in\{0,1\}$ ，一般使用 Sigmoid 函数作为激活函数，使输出值在 $[0,1]$ 范围内，使模型预测结果可直接作为概率输出. Sigmoid 函数的输出一般配合二元交叉熵损失函数使用，对于一个训练样本 $(\boldsymbol{x},y)$ 有

$$\ell\left(y,\hat{y}_1\right)=-\left[y\log\left(\hat{y}_1\right)+(1-y)\log\left(1-\hat{y}_1\right)\right], \tag{10}$$

记 $\hat{y}_1$ 为模型将样本判断为正例的预测概率，请计算 $\frac{\partial \ell(y,\hat{y}_1)}{\partial \beta_1}$。

(2) (5 points) 当 $l > 1$，网络的预测结果为 $\hat{\boldsymbol{y}} \in \mathbb{R}^l$，其中 $\hat{y}_i$ 表示输入被预测为第 i 类的概率。对于第 i 类的样本，其标记 $\boldsymbol{y} \in \{0,1\}^l$，有 $y_i = 1, y_j = 0, j \neq i$。对于一个训练样本 $(\boldsymbol{x}, \boldsymbol{y})$，交叉熵损失函数 $\ell(\boldsymbol{y}, \hat{\boldsymbol{y}})$ 的定义如下

$$\ell(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\sum_{j=1}^{l} y_j \log \hat{y}_j, \tag{11}$$

在多分类问题中，一般使用 Softmax 函数作为输出层激活函数，其计算公式如下

$$\hat{y}_j = \frac{e^{\beta_j}}{\sum_{k=1}^{l} e^{\beta_k}}, \tag{12}$$

易见 Softmax 函数输出的 $\hat{\boldsymbol{y}}$ 符合 $\sum_{j=1}^{l} \hat{y}_j = 1$，所以可以直接作为每个类别的概率. Softmax 函数输出一般配合交叉熵损失函数使用，请计算 $\frac{\partial \ell(\boldsymbol{y},\hat{\boldsymbol{y}})}{\partial \boldsymbol{\beta}}$。

(3) (2 points) 分析在二分类中使用 Softmax 激活函数和 Sigmoid 激活函数的联系与区别。

解：

1. **(1)**

   对于公式的证明直接利用矩阵的性质，由于

   $$\mathbf{X}\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{X} = \mathbf{X}\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{X},$$

   即

   $$\mathbf{X}\left(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_d\right) = \left(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I}_m\right)\mathbf{X}.$$

   左右两边各乘上相同的矩阵：

   $$\left(\boldsymbol{X}\boldsymbol{X}^\top + \lambda\boldsymbol{I}_m\right)^{-1}\boldsymbol{X}\left(\boldsymbol{X}^\top\boldsymbol{X} + \lambda\boldsymbol{I}_d\right)\left(\boldsymbol{X}^\top\boldsymbol{X} + \lambda\boldsymbol{I}_d\right)^{-1}$$

   $$= \left(\boldsymbol{X}\boldsymbol{X}^\top + \lambda\boldsymbol{I}_m\right)^{-1}\left(\boldsymbol{X}\boldsymbol{X}^\top + \lambda\boldsymbol{I}_m\right)\boldsymbol{X}\left(\boldsymbol{X}^\top\boldsymbol{X} + \lambda\boldsymbol{I}_d\right)^{-1}$$

   即：

   $$\left(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I}_m\right)^{-1}\mathbf{X} = \mathbf{X}\left(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}_d\right)^{-1}.$$

   (2)

   对于岭回归问题闭式解的推导过程类似原始线性回归。定义岭回归的目标函数：

   $$g(\mathbf{w}, b) := \frac{1}{2}\left\|\mathbf{X}\mathbf{w} + \mathbf{1}_m b^\top - \mathbf{y}\right\|_2^2 + \lambda\|\mathbf{w}\|_2^2.$$

   针对模型变量 $\mathbf{w}$ 求偏导，并令偏导为 0：

   $$\left(\mathbf{X}^\top\mathbf{X} - \frac{1}{m}\mathbf{X}^\top\mathbf{1}_m\mathbf{1}_m^\top\mathbf{X} + 2\lambda\mathbf{I}_d\right)\mathbf{w} - \left(\mathbf{X}^\top - \frac{1}{m}\mathbf{X}^\top\mathbf{1}_m\mathbf{1}_m^\top\right)\mathbf{y} = 0.$$

则有

$$\mathbf{w}^*_{\text{Ridge}} = \left(\mathbf{X}^\top\mathbf{X} - \frac{1}{m}\mathbf{X}^\top\mathbf{1}_m\mathbf{1}_m^\top\mathbf{X} + 2\lambda\mathbf{I}_d\right)^{-1}\left(\mathbf{X}^\top - \frac{1}{m}\mathbf{X}^\top\mathbf{1}_m\mathbf{1}_m^\top\right)\mathbf{y},$$

简化为

$$\mathbf{w}^*_{\text{Ridge}} = \left(\mathbf{X}^\top\mathbf{H}\mathbf{X} + 2\lambda\mathbf{I}_d\right)^{-1}(\mathbf{H}\mathbf{X})^\top\mathbf{y}.$$

同理，有

$$b^*_{\text{Ridge}} = \frac{1}{m}\left(\mathbf{1}_m^\top\mathbf{y} - \mathbf{1}_m^\top\mathbf{X}\mathbf{w}^*_{\text{Ridge}}\right).$$

对于岭回归和原始线性回归的解，能够发现这两个模型权重 w 有所不同，偏移项 b 的形式是一致的（依赖于 w 的最优解）。在岭回归的最优解中，主要的区别在于式 $\mathbf{w}^*_{\text{Ridge}}$ 的第一项在矩阵求逆的过程中增加了 $2\lambda I_d$。新增的一项能够避免矩阵的特征值趋于 0，使得 $X^\top HX + 2\lambda I_d$ 矩阵的特征值至少大于 2λ，从而方便矩阵的求逆操作。岭回归方法也能够看做具有高斯先验的线性回归模型。

2. **(1)**

观察式 (4.1) 中信息熵的形式，由于 $0 \le p_k \le 1$，因此，$-\log_2 p_k > 0$，从而 $\text{Ent}(D) \ge 0$。若所有样本属于同一类，如 $p_1 = 1$，且 $\{p_k\}_{k>1} = 0$ 时 $\text{Ent}(D) = 0$。考虑到 $\log_2 x$ 为凹函数，使用 Jensen 不等式

$$\log_2|\mathcal{Y}| = \log_2\left(\sum_{k=1}^{|\mathcal{Y}|}\frac{p_k}{p_k}\right)$$

$$\ge \sum_{k=1}^{|\mathcal{Y}|} p_k \log_2\frac{1}{p_k} = -\sum_{k=1}^{|\mathcal{Y}|} p_k\log_2 p_k = \text{Ent}(D)$$

当 $\{p_k = \frac{1}{|\mathcal{Y}|}\}_{k=1}^{|\mathcal{Y}|}$ 时（即 $|\mathcal{Y}|$ 类各类概率相等时），取得等号。

(2)

基尼指数衡量了从数据集中两次抽样的样本属于不同类别的比重，因此数据越纯，两次抽样的来自于异类的概率越小，基尼指数越小。而对于误分类错误率，衡量了样本数量最多的类别在整体数据集中的占比，比重越大，则数据越纯，误分类错误率越低。

将不同衡量标准转化为二分类的形式，假设正类所占比例为 p，则负类为 $1-p$，此时误分类错误率、基尼指数和信息熵的形式变化为

$$1 - \max(p, 1-p), \quad 2p(1-p), \quad -p\log_2 p - (1-p)\log_2(1-p)$$

给定 $0 \le p \le 1$，各指标关于 p 的变化趋势如 Figure 1,2,3 所示。可以看出，三种曲线形状相似，当 $p = 0$（所有数据均为负类）或 $p = 1$（所有数据均为正类）时，各指标取值为 0，纯度最高；而当 $p = 0.5$（正负类样本数量相同）时，此时各指标达到最大值，纯度最低。此外，基尼指数和信息熵的曲线更加平滑且可导，误分类错误率是一条折线，在 $p = 0.5$ 时不可导，因此基尼指数和信息熵更多用于数值优化中。三种函数存在包含顺序，即信息熵完全在基尼指数上方，基尼指数完全在误分类错误率上方。一般而言，信息熵和基尼指数对 p 的变化比误分类错误率更为敏感。

(3)

本题从随机变量熵的角度推导信息增益的性质。定义随机变量 Y 为类别标记，其分布为数据集 D 中类别的经验分布。使用 $H(Y)$ 表示 Y 的信息熵，定义与公式 (1) 相同。

定义随机变量 A 为属性 a 的取值，D^v 是在属性 a 上取值为 a^v 的样本集合，随机变量 A 的分布为数据集 D 中属性 a 的经验分布 (D^v 占比)。对于给定 $A = a^v$ 条件下随机变量 Y 的信息熵即为 D^v 的信息熵，权重 $\frac{|D^v|}{|D|}$ 可以看作 A 取值为 a^v 的概率。使用 $H(Y \mid A)$ 表示条件随机变量的信息熵，也称条件熵，定义为

$$H(Y \mid A) = -\sum_{A} P(A) \left(\sum_{Y} P(Y \mid A) \log_2 P(Y \mid A) \right) \tag{1}$$

$$= -\sum_{Y} \sum_{A} P(Y, A) \log_2 P(Y \mid A) \tag{2}$$

即对 A 所有可能的取值取期望。基于前面的定义，可将信息增益表示为

$$\mathrm{Gain}(D, a) = \mathrm{Ent}(D) - \sum_{v=1}^{V} \frac{|D^v|}{|D|} \mathrm{Ent}(D^v) \tag{3}$$

$$= H(Y) - \sum_{v=1}^{V} P(A = a^v) H(Y \mid A = a^v) \tag{4}$$

$$= H(Y) - H(Y \mid A) \tag{5}$$

其中公式 (3) 到公式 (5) 的过程参照上述定义，而公式 (4) 到公式 (5) 使用了全概率展开。综上，信息增益相当于随机变量 Y 的信息熵 $H(Y)$ 和其在给定随机变量 A 的条件下信息熵 $H(Y \mid A)$ 之差。

$$H(Y) - H(Y \mid A) = -\sum_{Y} P(Y) \log_2 P(Y) + \sum_{Y} \sum_{A} P(Y, A) \log_2 P(Y \mid A) \tag{6}$$

$$= -\sum_{Y} P(Y) \log_2 P(Y) + \sum_{Y} \sum_{A} P(Y, A) \log_2 \frac{P(Y)P(A)}{P(Y, A)} \tag{7}$$

$$\geq -\log_2 \sum_{Y} \sum_{A} P(Y, A) \frac{P(Y)P(A)}{P(Y, A)} \tag{8}$$

$$= -\log_2 \sum_{Y} \sum_{A} P(Y)P(A) = -\log_2 1 = 0 \tag{9}$$

其中公式 (7) 在分子上不同乘了 $P(A)$，而公式 (8) 使用了 Jensen 不等式。

3. **(1)**

首先计算 BCE 损失函数针对预测 $\hat{y}_1$ 的导数

$$\frac{\partial \ell}{\partial \hat{y}_1} = -\frac{y}{\hat{y}_1} + \frac{1 - y}{1 - \hat{y}_1}$$

然后计算预测值 $\hat{y}_1$ 针对 Sigmoid 函数的导数

$$\frac{\partial \hat{y}_1}{\partial \beta_1} = \sigma(\beta_1)(1 - \sigma(\beta_1))$$

所以有

$$\frac{\partial \ell}{\partial \beta_1} = \frac{\partial \ell}{\partial \hat{y}_1}\frac{\partial \hat{y}_1}{\partial \beta_1} = \frac{\hat{y}_1 - y}{\hat{y}_1(1-\hat{y}_1)}\sigma(\beta_1)(1-\sigma(\beta_1))$$

(2)

对交叉熵损失函数求导如下

$$\frac{\partial \ell}{\partial \hat{y}_j} = -\frac{y_j}{\hat{y}_j},$$

$$\frac{\partial \ell}{\partial \hat{\boldsymbol{y}}} = -\boldsymbol{y} \odot \hat{\boldsymbol{y}}^{-1}$$

其中 $\odot$ 表示元素分别相乘。然后对 Softmax 函数进行求导，记 $Z = \sum_j e^{\beta_j}$，在 $i = j$ 的情况下，有

$$\begin{aligned}\frac{\partial \hat{y}_i}{\partial \beta_j} &= \frac{\partial e^{\beta_i}}{\partial \beta_i}Z - e^{\beta_i}\frac{\partial Z}{\partial \beta_i}Z^2 \\ &= \frac{e^{\beta_i}Z - e^{\beta_i}e^{\beta_i}}{Z^2} \\ &= \frac{e^{\beta_i}}{Z}\left(1 - \frac{e^{\beta_i}}{Z}\right) \\ &= \hat{y}_i(1-\hat{y}_i)\end{aligned}$$

对于 $i \neq j$ 的情况，类似的有

$$\begin{aligned}\frac{\partial \hat{y}_i}{\partial \beta_j} &= \frac{\partial e^{\beta_i}}{\partial \beta_j}Z - e^{\beta_i}\frac{\partial Z}{\partial \beta_j}Z^2 \\ &= -\frac{e^{\beta_i}e^{\beta_j}}{Z^2} \\ &= -\hat{y}_i\hat{y}_j\end{aligned}$$

综上

$$\frac{\partial \hat{y}_i}{\partial \beta_j} = \begin{cases}\hat{y}_i(1-\hat{y}_i), & i = j, \\ -\hat{y}_i\hat{y}_j, & i \neq j\end{cases}$$

写成矩阵形式有

$$\frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{\beta}} = \operatorname{diag}(\hat{\boldsymbol{y}}) - \hat{\boldsymbol{y}}\hat{\boldsymbol{y}}^\top$$

结合交叉熵损失函数的导数，有

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\beta}} = \frac{\partial \ell}{\partial \hat{\boldsymbol{y}}}\frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{\beta}} = -[\boldsymbol{y} \odot \hat{\boldsymbol{y}}^{-1}]\left[\operatorname{diag}(\hat{\boldsymbol{y}}) - \hat{\boldsymbol{y}}\hat{\boldsymbol{y}}^\top\right]$$

其中 $\operatorname{diag}(\cdot)$ 将向量转化为对角矩阵。

(3)

在二分类情况下，假设网络输出层有两个神经元（$l = 2$），输入分别为 β_0 和 β_1。基于 Softmax 函数得到正类的输出后可以做如下变换

$$\hat{y}_1 = \frac{e^{\beta_1}}{e^{\beta_0} + e^{\beta_1}} = \frac{1}{1 + e^{\beta_0 - \beta_1}} = \sigma(\beta_1 - \beta_0)$$

$$\hat{y}_0 = 1 - \sigma(\beta_1 - \beta_0)$$

所以 Softmax 函数可以看成 Sigmoid 函数在多分类问题中的一个推广。

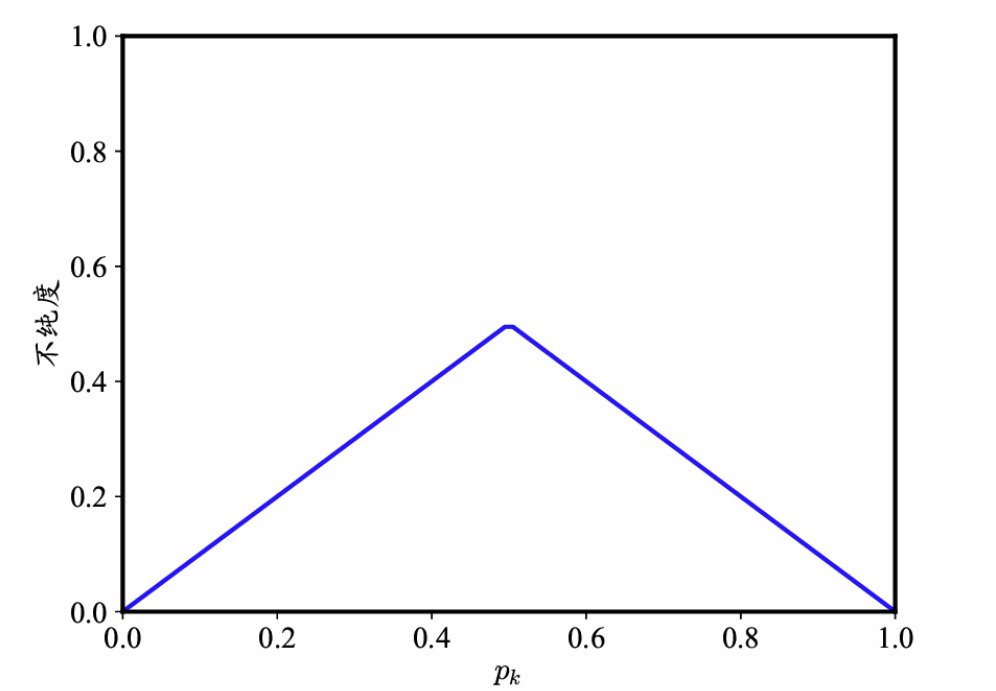

Figure 1: 误分类错误率

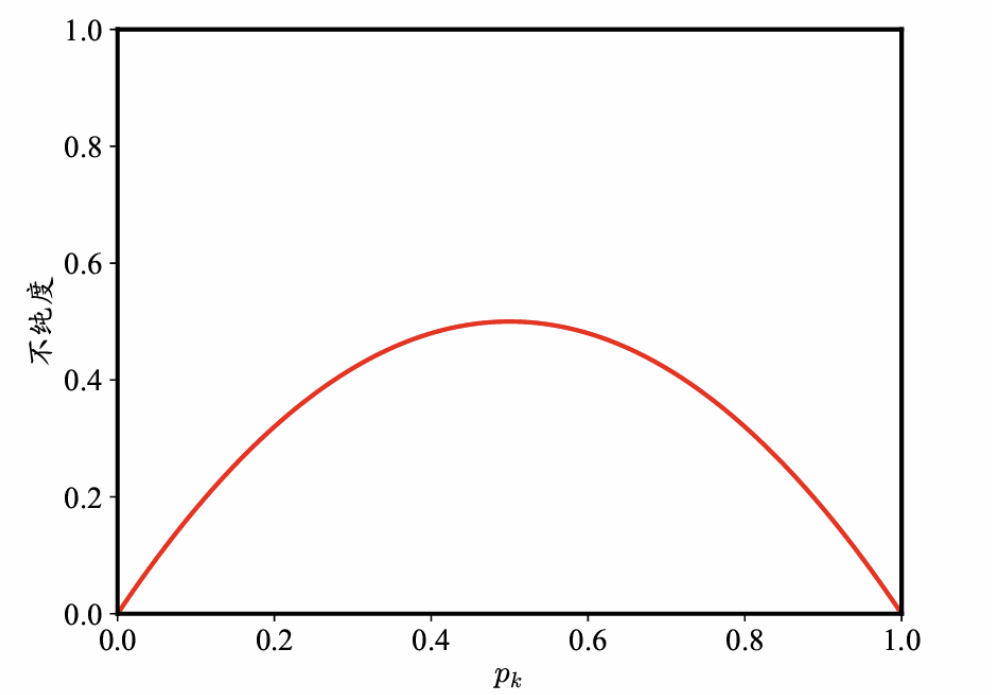

Figure 2: 基尼指数

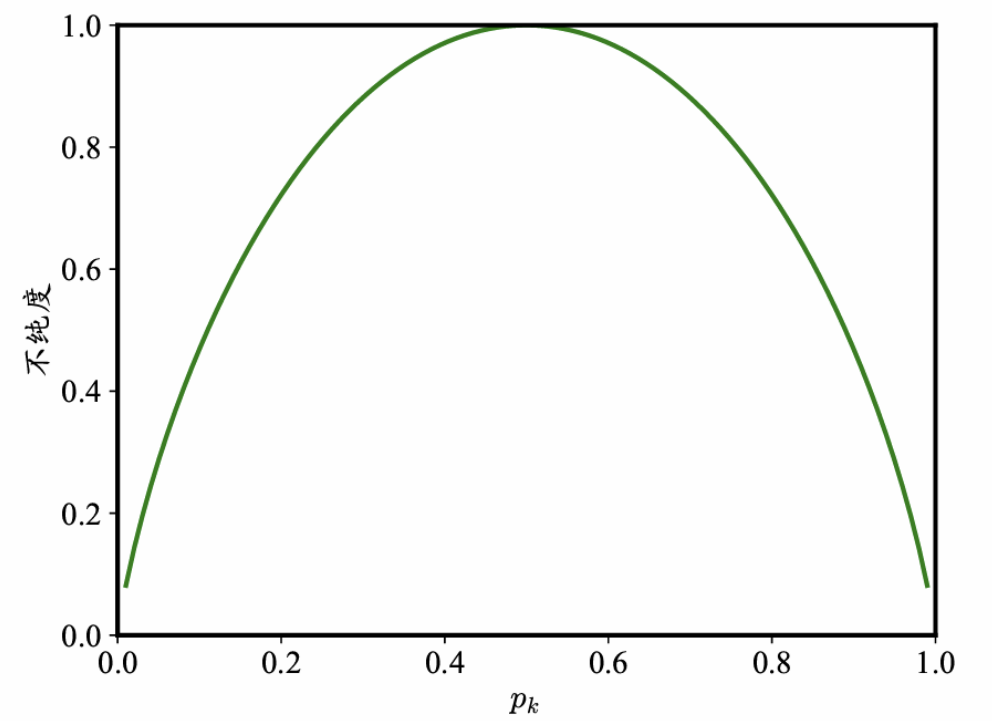

Figure 3: 信息熵

二. (30 points) **PCA 降维**

教材 10.3 节介绍了主成分分析 (Principal Component Analysis, PCA) 方法对数据进行降维。本题考察 PCA 相关的线性代数基础知识以及基本操作。给定 d 维空间中 m 个样本构成的矩阵为

$$X = [x_1^\top; \ldots; x_m^\top] \in \mathbb{R}^{m\times d}, \tag{13}$$

$\hat{X} \in \mathbb{R}^{m\times d}$ 为 X 中心化后得到的矩阵。根据教材 10.3 节讨论，严格的协方差矩阵具有 $\frac{1}{m-1}$ 因子，由于常数对本题分析结果无影响，所以在本题的讨论中忽略该常数因子。

1. (6 points) $\hat{X}^\top\hat{X}$ 和 $\hat{X}\hat{X}^\top$ 为什么是半正定矩阵？二者的特征值有什么联系？受此启发，请思考当特征维度远大于样本个数时 ($d \gg m$)，使用特征值分解求解 PCA 应如何执行将更加高效？
2. (6 points) 奇异值分解定义如下：

   令 $\hat{X} \in \mathbb{R}^{m\times d}$，则存在正交矩阵 $U \in \mathbb{R}^{m\times m}$ 和 $V \in \mathbb{R}^{d\times d}$ 使得：

   $$\hat{X} = U\Sigma V^\top, \tag{14}$$

   其中

   $$\Sigma = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix}, \tag{15}$$

   并且 $\Sigma_1 = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_r)$，其对角线元素按数值大小排序：

   $$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0, \quad r = \mathrm{rank}(\hat{X}), \tag{16}$$

   当矩阵 $\hat{X}$ 的秩 $r = \mathrm{rank}(\hat{X}) < h$ 时，奇异值分解可以进行截尾，从而可简化为：

   $$\hat{X} = U_r\Sigma_r V_r^\top, \tag{17}$$

   式中

   $$U_r = (u_1, u_2, \ldots, u_r), V_r = (v_1, v_2, \ldots, v_r), \Sigma_r = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_r), \tag{18}$$

   这种奇异值分解方式，被称为薄奇异值分解 (Thin SVD)。

   在实现 PCA 时，往往使用奇异值分解 (SVD) 而非特征值分解求解。请说明奇异值与特征值的关系，如果可以获得 $\hat{X}$ 的奇异值分解，应如何使用 PCA 对 $\hat{X}$ 进行降维？请分析使用 SVD 求解 PCA 相比于使用特征值分解求解 PCA 的优势。
3. (8 points) PCA 的一个重要步骤是将误差路径最小化重构误差，请说明为什么在最小化重构误差之前需要对数据进行中心化。
4. (10 points) 针对以下样本矩阵 (包含 5 个示例, 每个示例 2 维)，请对其进行主成分分析，将样本降至二维，并写出详细计算过程。

   $$X^\top = \begin{pmatrix} 3 & 4 & 4 & 6 & 3 \\ 2 & 3 & 2 & 3 & 0 \end{pmatrix} \tag{19}$$

解：

1. $\hat{\mathbf{X}}^\top\hat{\mathbf{X}}$ 和 $\hat{\mathbf{X}}\hat{\mathbf{X}}^\top$ 均为半正定矩阵。

   证明：$\hat{\mathbf{X}}^\top\hat{\mathbf{X}}$ 显然是对称矩阵，对于 $\forall \mathbf{x}\in\mathbb{R}^d$，有

   $$\mathbf{x}^\top\hat{\mathbf{X}}^\top\hat{\mathbf{X}}\mathbf{x}=(\hat{\mathbf{X}}\mathbf{x})^\top(\hat{\mathbf{X}}\mathbf{x})\geq 0$$

   所以，$\hat{\mathbf{X}}^\top\hat{\mathbf{X}}$ 是半正定矩阵。同理可证 $\hat{\mathbf{X}}\hat{\mathbf{X}}^\top$ 为半正定矩阵。

   $\hat{\mathbf{X}}^\top\hat{\mathbf{X}}$ 和 $\hat{\mathbf{X}}\hat{\mathbf{X}}^\top$ 具有相同的非零特征值。

   证明：令 $\hat{\mathbf{X}}^\top\hat{\mathbf{X}}$ 的特征值和特征向量分别为 $\lambda,\mathbf{x}$。根据特征值的定义，有：

   $$\hat{\mathbf{X}}^\top\hat{\mathbf{X}}\mathbf{x}=\lambda\mathbf{x}$$

   两边同时乘以 $\hat{\mathbf{X}}$，有：

   $$\hat{\mathbf{X}}\hat{\mathbf{X}}^\top(\hat{\mathbf{X}}\mathbf{x})=\lambda(\hat{\mathbf{X}}\mathbf{x})$$

   显然，λ 同时为 $\hat{\mathbf{X}}\hat{\mathbf{X}}^\top$ 的特征值，对应的特征向量为 $\hat{\mathbf{X}}^\top\mathbf{x}$。

   当 $d\gg m$ 时，计算 $\hat{\mathbf{X}}\hat{\mathbf{X}}^\top\in\mathbb{R}^{m\times m}$（时间复杂度为 $O(dm^2)$）并进行特征值分解的成本低于 $\hat{\mathbf{X}}^\top\hat{\mathbf{X}}\in\mathbb{R}^{d\times d}$（时间复杂度为 $O(d^2m)$）。因此当维度较大时，直接对协方差矩阵进行特征值分解具有较大的计算开销。由于 $\hat{\mathbf{X}}\hat{\mathbf{X}}^\top$ 的非零特征值与 $\hat{\mathbf{X}}^\top\hat{\mathbf{X}}$ 的非零特征值相同，可先计算 $\hat{\mathbf{X}}\hat{\mathbf{X}}^\top$ 的特征值 $\{\lambda_i\}$ 和特征向量 $\{\mathbf{x}_i\}$，并通过 $\hat{\mathbf{X}}^\top\mathbf{x}_i$ 得到 $\hat{\mathbf{X}}^\top\hat{\mathbf{X}}$ 特征值对应的特征向量。

2. 对 $\hat{\mathbf{X}}$ 的薄奇异值分解 $\hat{\mathbf{X}}=U_r\Sigma_rV_r^\top$ 取转置，可得

   $$\hat{\mathbf{X}}^\top=V_r\Sigma_rU_r^\top. \tag{1}$$

   分别计算 $\hat{\mathbf{X}}\hat{\mathbf{X}}^\top$ 和 $\hat{\mathbf{X}}^\top\hat{\mathbf{X}}$，得：

   $$\hat{\mathbf{X}}\hat{\mathbf{X}}^\top=U_r\Sigma_rV_r^\top V_r\Sigma_rU_r^\top=U_r(\Sigma_r\Sigma_r)U_r^\top, \tag{2}$$

   $$\hat{\mathbf{X}}^\top\hat{\mathbf{X}}=V_r\Sigma_rU_r^\top U_r\Sigma_rV_r^\top=V_r(\Sigma_r\Sigma_r)V_r^\top. \tag{3}$$

   最后一项恰好对应协方差矩阵的特征值分解。所以，令 $\{\lambda_i\}_{i=1}^r$ 为 $\hat{\mathbf{X}}^\top\hat{\mathbf{X}}$ 的非零特征值，当 λ_i 对应的特征向量与 σ_i 对应的右奇异向量相同时，$\lambda_i=\sigma_i^2$。更进一步，对 $\hat{\mathbf{X}}$ 右乘 V_r，可得

   $$\hat{\mathbf{X}}V_r=U_r\Sigma_r. \tag{4}$$

   与式 (3) 以及 PCA 的求解方法比较，V_r 恰好是 PCA 的降维矩阵，而 $U_r\Sigma_r$ 为 PCA 的降维结构结果。所以，使用 SVD 求解 PCA，就是对中心化后的样本矩阵 $\hat{\mathbf{X}}$ 进行 SVD 分解，然后选定降维的维度 d'，对 SVD 分解进行 d' 截尾

   $$\hat{\mathbf{X}}=U_{d'}\Sigma_{d'}V_{d'}^\top. \tag{5}$$

   则 $V_{d'}$ 就是 PCA 中的降维矩阵，$U_{d'}\Sigma_{d'}$ 就是降维后的结果。在实际工程中，使用 SVD 求解的原因主要是降低计算的复杂度，因为使用特征值分解需要先计算协方差矩阵，当样本个数以及维度非常大时，这一步的计算复杂度很高。但通过随机化优化算法大幅降低时间复杂度，比如在 scikit-learn 中，其使用随机方法的时间复杂度为 $O(nd^2)+O(nd'^2)$，d' 为主成分的个数。

3. 从最小化重构误差的角度考虑，假设投影后得到的新坐标系为 $\{w_1,w_2,\ldots,w_d\}$，其中两两向量正交。x_i 在 d' 维坐标系中的投影为 $z_i=(z_{i1},z_{i2},\ldots,z_{id'})$，重构的坐标 x_i' 为：

   $$x_i'=\sum_{j=1}^{d'}z_{ij}w_j+\sum_{j=d'+1}^{m}b_jw_j. \tag{1}$$

最小重构误差就是最小化：

$$J = \frac{1}{m}\sum_{i=1}^{m} \|x_i - x_i'\|_2^2. \tag{2}$$

令 J 对 z_{ij} 求偏导并将其置为 0 得到：

$$z_{ij} = x_i^\top w_j. \tag{3}$$

令 J 对 b_j 求偏导并令偏导为 0，同时利用 $w_i^\top w_j = 0, i \neq j$ 这一正交条件，可以得到：

$$b_j = \bar{x}^\top w_j, \tag{4}$$

其中 $\bar{x} = \frac{1}{m}\sum_{i=1}^{m} x_i$ 代表样本均值。由此可得：

$$x_i - x_i' = \sum_{j=d'+1}^{d} \left((x_i - \bar{x})^\top w_j\right) w_j. \tag{5}$$

可以看出，中心化的过程即是 $x_i - \bar{x}$ 对应的操作。将式 (5) 中 $x_i - \bar{x}$ 用 $\tilde{x}$ 取代并代回式 (2) 中，可以得到：

$$J = \sum_{j=d'+1}^{d} w_j^\top \tilde{\mathbf{X}}^\top \tilde{\mathbf{X}} w_j. \tag{6}$$

重构思路也可以从另外一个角度解释。PCA 希望在 d 维空间中寻找 d' 维子空间，该子空间上的任意一点都可以通过式 (1) 的形式表示出来。对其进一步简化可写成如下的形式：

$$x_i' = \sum_{j=1}^{d'} z_{ij} w_j + b, \tag{7}$$

其中，b 是一个与子空间相关的常数。但 PCA 的结果只能控制 $\{w_i\}_{i=1}^{d'}$，但这样的子空间随 b 的变化有无数个，且对于每一个固定的 b，$\{w_i\}_{i=1}^{d'}$ 都能使得重构误差最小，那么就需要考虑 b 的最优值。显然，该子空间需要穿过样本均值 $\bar{x}$，因为可以考虑将样本降到零维，很容易证明，降到 0 维后（即使用单一向量表示所有样本），将所有样本重构成 $\bar{x}$ 会使重构误差最小。因此可用 $\bar{x}$ 确定子空间的基点。经过中心化后，$\bar{x}$ 就变成了原点，子空间如果穿过原点的话，b 必然为零向量。

4. 计算每列的均值：

$$\bar{x} = \begin{pmatrix} 4 \\ 2 \end{pmatrix}.$$

将样本矩阵减去均值后得到中心化数据矩阵：

$$X_{\text{centered}}^T = X^T - \bar{x} = \begin{pmatrix} -1 & 0 & 0 & 2 & -1 \\ 0 & 1 & 0 & 1 & -2 \end{pmatrix}.$$

协方差矩阵的计算公式为：

$$\Sigma = X_{\text{centered}} X_{\text{centered}}^T.$$

计算结果为：

$$\Sigma = \begin{pmatrix} 6 & 4 \\ 4 & 6 \end{pmatrix}.$$

协方差矩阵的特征值通过解以下特征方程得到：

$$\det(\Sigma - \lambda I) = 0.$$

展开得：

$$\det\begin{pmatrix} 6-\lambda & 4 \\ 4 & 6-\lambda \end{pmatrix} = (6-\lambda)^2 - 16 = \lambda^2 - 12\lambda + 20 = 0.$$

解得特征值：

$$\lambda_1 = 2, \quad \lambda_2 = 10.$$

对应的特征向量通过解 $(\Sigma - \lambda I)\alpha = 0$ 得到：

$$\alpha_1 = \begin{pmatrix} -\sqrt{2}/2 \\ \sqrt{2}/2 \end{pmatrix}, \quad \alpha_2 = \begin{pmatrix} \sqrt{2}/2 \\ \sqrt{2}/2 \end{pmatrix}.$$

我们选取全部两个特征向量作为投影方向矩阵：

$$A = \begin{pmatrix} -\sqrt{2}/2 & \sqrt{2}/2 \\ \sqrt{2}/2 & \sqrt{2}/2 \end{pmatrix}.$$

将数据投影到二维主成分空间的公式为：

$$Z = X_{\text{centered}} A.$$

其中：

$$X_{\text{centered}} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 2 & 1 \\ -1 & -2 \end{pmatrix}.$$

计算得：

$$Z = \begin{pmatrix} -1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 2 & 1 \\ -1 & -2 \end{pmatrix} \begin{pmatrix} -\sqrt{2}/2 & \sqrt{2}/2 \\ \sqrt{2}/2 & \sqrt{2}/2 \end{pmatrix} = \begin{pmatrix} \sqrt{2}/2 & -\sqrt{2}/2 \\ \sqrt{2}/2 & \sqrt{2}/2 \\ 0 & 0 \\ -\sqrt{2}/2 & 3\sqrt{2}/2 \\ -3\sqrt{2}/2 & -\sqrt{2}/2 \end{pmatrix}.$$

注：本题保留主成分数等于特征维度数，因此可以直接将投影到二维主成分空间的数据还原到原始特征空间。用公式：

$$X_{\text{reconstructed}} = ZA^T.$$

计算 A^T：

$$A^T = \begin{pmatrix} -\sqrt{2}/2 & \sqrt{2}/2 \\ \sqrt{2}/2 & \sqrt{2}/2 \end{pmatrix}.$$

计算还原矩阵：

$$X_{\text{reconstructed}} = \begin{pmatrix} \sqrt{2}/2 & -\sqrt{2}/2 \\ \sqrt{2}/2 & \sqrt{2}/2 \\ 0 & 0 \\ -\sqrt{2}/2 & 3\sqrt{2}/2 \\ -3\sqrt{2}/2 & -\sqrt{2}/2 \end{pmatrix} \begin{pmatrix} -\sqrt{2}/2 & \sqrt{2}/2 \\ \sqrt{2}/2 & \sqrt{2}/2 \end{pmatrix}.$$

通过矩阵乘法，得到还原矩阵：

$$X_{\text{reconstructed}} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 2 & 1 \\ -1 & -2 \end{pmatrix}.$$

会发现与原数据一样。

三. (40 points) **度量学习应用**

度量学习旨在学习一个适用于某个任务的距离度量，等价于为实现某个距离度量找到合适的特征变换。

1. (20 points) 教材 10.6 节介绍了马氏距离:

$$dist^2_{\text{mah}}(x_i, x_j) = (x_i - x_j)^\top M(x_i - x_j) = \|x_i - x_j\|^2_M, \tag{20}$$

在标准的马氏距离中，M 为样本协方差矩阵的逆 Σ^{-1}。而在度量学习中，M 是一个可学习的半正定矩阵 ($M \succeq 0$)，度量学习的过程可以看成一个优化 M 的过程。请回答以下问题：

   1. (6 points) 标准的马氏距离去除了变量之间的相关性，并且与量纲无关。结合 PCA 中关于协方差矩阵的相关知识，请解释马氏距离为什么有这些优点。(提示：可将协方差矩阵进行特征值分解，重写 M 及上式)
   2. (2 points) 标准马氏距离中 M 为协方差矩阵的逆，是否存在某些情况下协方差矩阵不可逆，应该如何应对这个问题?
   3. (4 points) 不同于人工设定 M，度量学习在给定目标函数的条件下优化出半正定矩阵 M。结合教材 9.3 节对距离度量的介绍，请说明马氏距离是否是标准的距离度量（是否满足距离度量的四个性质)?
   4. (8 points) 教材 3.4 节介绍的监督降维方法线性判别分析 LDA 以及 10.3 节介绍的无监督降维方法主成分分析 PCA 均可视为特殊的度量学习方法。简单来说，首先对样本进行降维，并在降维后空间中计算样本之间的欧氏距离作为距离度量。参考教材中的定义，类内散度矩阵 S_w 为每个类别的散度矩阵之和，类间散度矩阵 S_b 为每个类别与类中心的协方差矩阵。请写出 LDA 和 PCA 对应的马氏距离中的 M，并说明 LDA 和 PCA 的异同。(提示：将两种方法与度量学习进行关联)

2. (20 points) 度量学习方法一般需学习一个半正定的距离度量矩阵, 其目标函数是一个半正定规划 (Semi-Definite Programming, SDP) 问题, 是一类特殊的凸优化问题。

   注：半正定规划有以下形式

$$\min_{X \in \mathcal{S}_+} \langle C, X \rangle$$
$$\text{s.t.} \quad \langle A_k, X \rangle \le b_k, \quad k = 1, \ldots, m,$$

   其中 X, C, A_k ($k = 1, \ldots, m$) 均为 $d \times d$ 方阵。$\langle A, B \rangle = \text{tr}(A^\top B) = \sum_{i=1}^d \sum_{j=1}^d A_{ij} B_{ij}$，$\mathcal{S}_+$ 表示半正定矩阵的集合。

   本题以 LMNN 为例，探究度量学习的优化方式。

   1. (5 points) 相比于线性或二次优化，半正定优化的求解较为缓慢。请推导 LMNN 损失函数对于 M 的梯度。若要保证 M 求解后为对称矩阵，M 需要如何初始化?
   2. (10 points) 使用梯度下（5 points）降法求解 M 时，需保证 M 满足半正定约束。常见的做法是使用投影梯度下降（Projected Gradient Descent, PGD）方法在每次更新 M 时将其投影到半正定矩阵集合 $\mathcal{S}_+$ 中。半正定投影等价于求解如下问题：

$$\arg\min_{\hat{M}} \|\hat{M} - M\|_F^2 \quad \text{s.t.} \quad \hat{M} \in \mathcal{S}_+, \tag{21}$$

假设对称矩阵 M 的特征值分解为 $M = Q\Lambda Q^\top$，其中 $QQ^\top = I$ 为正交矩阵，Λ 为特征值构成的对角矩阵。请证上述问题的解为 $\hat{M} = Q\Lambda^+ Q^\top$，其中 Λ^+ 表示将 Λ 中的非负元素不变，负元素置零。

3. (5 points) 将任意半正定矩阵分解为投影矩阵，即 $M = PP^\top$，则 LMNN 可转化为关于 P 的无约束优化问题。请推导 LMNN 损失关于 P 的梯度。该问题是凸优化问题吗?

解:

1. **(1)**

首先对马氏距离的定义形式进行分解。假设（半正定）协方差矩阵具有特征值分解 $\Sigma = UDU^\top$，其中 $U \in \mathbb{R}^{d\times d}$ 为正交矩阵，$D \in \mathbb{R}^{d\times d}$ 为对角矩阵，对角元素对应矩阵 Σ 的特征值。因此有 $M = \Sigma^{-1} = UD^{-1}U^\top$。进一步整理后，可将 M 分解为 $M = LL^\top$，其中 $L = UD^{-1/2}$。

马氏距离可以写成:

$$\mathrm{dist}^2_{\mathrm{mah}}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^\top LL^\top(\mathbf{x}_i - \mathbf{x}_j) = \|L^\top(\mathbf{x}_i - \mathbf{x}_j)\|_2^2 = \|D^{-1/2}U^\top(\mathbf{x}_i - \mathbf{x}_j)\|_2^2.$$

给定 m 个样本构成的矩阵 $X = (\mathbf{x}_1^\top; \mathbf{x}_2^\top; \cdots; \mathbf{x}_m^\top) \in \mathbb{R}^{m\times d}$，上述马氏距离的计算相当于先对样本进行线性投影 $X' = XL$，然后计算变换后的欧式距离作为样本之间的距离度量。变换之后样本的协方差矩阵为 $X'^\top X' = I$。协方差矩阵非对角线元素为 0 表示过程可以去除变量之间的相关性，对角线上每一个维度的协方差都统一为 1 表示在标准化的基础上进一步缩放操作，可以去除量纲的影响。

(2)

协方差矩阵不可逆，可能是因为样本个数 m 较少，或者收集到的样本之间存在线性关系，导致协方差矩阵的秩小于特征的维度 d，导致 $\mathrm{rank}(\Sigma) \le m < d$。该问题可以通过增加样本数量加以缓解。根据第一问的分析，可以使用 PCA 算法对特征进行预处理，选择特征值比较大的部分重构协方差矩阵，从而去除特征之间的线性关系。除上述两种方法外，也可以计算协方差矩阵的伪逆矩阵（pseudo inverse）。

(3)

距离度量需要满足四个性质:

1. 非负性: $\mathrm{dist}(\mathbf{x}_i, \mathbf{x}_j) \ge 0$。
2. 同一性: $\mathrm{dist}(\mathbf{x}_i, \mathbf{x}_j) = 0 \iff \mathbf{x}_i = \mathbf{x}_j$。
3. 对称性: $\mathrm{dist}(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{dist}(\mathbf{x}_j, \mathbf{x}_i)$。
4. 直递性: $\mathrm{dist}(\mathbf{x}_i, \mathbf{x}_j) \le \mathrm{dist}(\mathbf{x}_i, \mathbf{x}_k) + \mathrm{dist}(\mathbf{x}_k, \mathbf{x}_j)$。

对称性可由马氏距离的定义得出。由于 M 是半正定矩阵，即对于任意非零向量 $\mathbf{z}$，都有:

$$\mathbf{z}^\top M\mathbf{z} \ge 0$$

因此满足非负性。但当 $\mathbf{z} \ne 0$ 时，也存在 $\mathbf{z}$，使得 $\mathbf{z}^\top M\mathbf{z} = 0$，所以不满足同一性（只有 M 是正定矩阵时才满足）。例如，对 M 进行特征值分解:

$$M = UDU^\top = (UD^{\frac{1}{2}})(UD^{\frac{1}{2}})^\top = LL^\top$$

假设 M 有 $d' < d$ 个非零特征值。则对于两个样本在通过 U 进行变换后，如在前 d' 维完全相同，但在后 $d - d'$ 维不同，两个样本虽不相等，但距离仍为 0。因此不满足同一性。

根据 Cauchy–Schwarz 不等式，可得：

$$\begin{aligned}\mathrm{dist}_{\mathrm{mah}}(\mathbf{x}_i, \mathbf{x}_j) &= \|L^\top \mathbf{x}_i - L^\top \mathbf{x}_j\|_2 = \|L^\top \mathbf{x}_i - L^\top \mathbf{x}_k + L^\top \mathbf{x}_{d'} - L^\top \mathbf{x}_j\|_2 \\ &\leq \|L^\top \mathbf{x}_i - L^\top \mathbf{x}_k\|_2 + \|L^\top \mathbf{x}_k - L^\top \mathbf{x}_j\|_2\end{aligned}$$

即：

$$\mathrm{dist}(\mathbf{x}_i, \mathbf{x}_j) \leq \mathrm{dist}(\mathbf{x}_i, \mathbf{x}_k) + \mathrm{dist}(\mathbf{x}_k, \mathbf{x}_j)$$

(4)

结合第一小问的分析，马氏距离度量相当于先对数据进行投影，在投影的基础上计算欧氏距离。PCA 和 LDA 均可视为特线的度量学习方法，基于协方差矩阵或散度矩阵获得投影，对数据降维，使降维后的数据的分散特性仍满足要求。因此，通过对 PCA 和 LDA 中投影矩阵的分析，即可得其和度量学习的关联。

使用 PCA 进行降维，优化目标为：

$$\min_W -\mathrm{tr}(W^\top \Sigma W) \quad \text{s.t. } W^\top W = I.$$

优化的投影矩阵 W 为协方差矩阵 Σ 最大的 d' 个特征值对应的特征向量。所以 PCA 的距离度量可以写成：

$$\mathrm{dist}^2_{\mathrm{PCA}}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^\top W W^\top (\mathbf{x}_i - \mathbf{x}_j) = \|W^\top \mathbf{x}_i - W^\top \mathbf{x}_j\|_2^2.$$

使用 LDA 进行降维，优化目标为：

$$\min_W -\mathrm{tr}(W^\top S_b W) \quad \text{s.t. } W^\top S_w W = I.$$

由于 S_w 是半正定类内散度矩阵，因此可基于第一小问的思路，基于特征值分解得到 $S_w = S_w^{\frac{1}{2}} S_w^{\frac{1}{2}}$。令 $V = S_w^{\frac{1}{2}} W$，则 $W = S_w^{-\frac{1}{2}} V$，优化目标变为：

$$\min_V -\mathrm{tr}(V^\top S_w^{-\frac{1}{2}} S_b S_w^{-\frac{1}{2}} V) \quad \text{s.t. } V^\top V = I.$$

此时优化问题的解是 $S_w^{-\frac{1}{2}} S_b S_w^{-\frac{1}{2}}$ 最大的 d' 个特征向量 $a_1, a_2, \ldots, a_{d'}$ 组成的矩阵 A，$W = S_w^{-\frac{1}{2}} A$。此外，可以对上述结果进行进一步的化简。根据特征值的定义，得到：

$$S_w^{-\frac{1}{2}} S_b S_w^{-\frac{1}{2}} a_i = \lambda_i a_i.$$

两边同时乘以 $S_w^{\frac{1}{2}}$，由于 $S_w^{\frac{1}{2}} a_i = w_i$，可得：

$$S_w^{-1} S_b w_i = \lambda_i w_i.$$

所以，W 为 $S_w^{-1} S_b$ 最大 d' 个特征值对应特征向量的矩阵。综上，基于 LDA 的距离度量可写为：

$$\mathrm{dist}^2_{\mathrm{LDA}}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^\top W W^\top (\mathbf{x}_i - \mathbf{x}_j) = \|W^\top \mathbf{x}_i - W^\top \mathbf{x}_j\|_2^2.$$

2. **(1)**

马氏距离 $\mathrm{dist}_M^2(x_i, x_j)$ 关于 M 的导数为：

$$\frac{\partial \mathrm{dist}_M^2(x_i, x_j)}{\partial M} = (x_i - x_j)(x_i - x_j)^\top$$

因此有：

$$\begin{aligned}\frac{\partial l_{\mathrm{lmnn}}}{\partial M} =& (1-\mu) \sum_{i,j \in N_i} \frac{\partial \mathrm{dist}_M^2(x_i, x_j)}{\partial M} \\ &+ \mu \sum_{i,j \in N_i} \sum_{l} \left(1 - \mathbb{I}(y_i = y_j)\right) a(x_i, x_j, x_l) \\ &\times \left(\frac{\partial \mathrm{dist}_M^2(x_i, x_j)}{\partial M} - \frac{\partial \mathrm{dist}_M^2(x_i, x_l)}{\partial M} \right)\end{aligned}$$

其中：

$$a(x_i, x_j, x_l) = \mathbb{I}(1 + \mathrm{dist}_M^2(x_i, x_j) - \mathrm{dist}_M^2(x_i, x_l) \geq 0)$$

由于 $(x_i - x_j)(x_i - x_j)^\top$ 都是对称矩阵，因此每一步的梯度 $\frac{\partial l_{\mathrm{lmnn}}}{\partial M}$ 同样为对称矩阵。若需要更新后的矩阵对称，则在初始化时需要将 M 初始化为对称矩阵。

(2)

证明: 不妨设 M 的特征值 $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \cdots \geq \lambda_n$，显然若 $\lambda_n \geq 0$，则 $\hat{M} = M = Q\Lambda Q^\top$ 成立。若 $\lambda_n < 0$，即存在负特征值，则可假设 $\lambda_1 \geq \lambda_2 \geq \cdots \lambda_k > 0 > \lambda_{k+1} \geq \cdots \geq \lambda_n$。令以 $\lambda_1, \lambda_2, \ldots, \lambda_k$ 所对应的特征向量为列向量的矩阵为 $Q_1 \in \mathbb{R}^{n \times k}$，以 $\lambda_{k+1}, \ldots, \lambda_n$ 所对应的特征向量为列向量的矩阵为 $Q_2 \in \mathbb{R}^{n \times (n-k)}$，则

$$[Q_1\ Q_2]^\top [Q_1\ Q_2] = \begin{bmatrix} Q_1^\top Q_1 & Q_1^\top Q_2 \\ Q_2^\top Q_1 & Q_2^\top Q_2 \end{bmatrix} = \begin{bmatrix} I_k & 0 \\ 0 & I_{n-k} \end{bmatrix} = I_n.$$

于是有

$$[Q_1\ Q_2][Q_1\ Q_2]^\top = Q_1 Q_1^\top + Q_2 Q_2^\top = I_n.$$

令投影矩阵 $P_1 = Q_1 Q_1^\top, P_2 = Q_2 Q_2^\top$。因为 $P_1 + P_2 = I$，易知对于任意 $X, P_1 X + P_2 X = X$.

另外：

$$\begin{aligned}&\|P_1 X + P_2 X\|_F = \mathrm{tr}\big((P_1 X + P_2 X)^\top (P_1 X + P_2 X)\big) \\ &= \mathrm{tr}\big((P_1 X)^\top P_1 X + (P_2 X)^\top P_2 X\big) \text{ (因为} P_1^\top P_2 = 0) \\ &= \mathrm{tr}\big((P_1 X)^\top P_1 X\big) + \mathrm{tr}\big((P_2 X)^\top P_2 X\big) \\ &= \|P_1 X\|_F + \|P_2 X\|_F.\end{aligned}$$

则：

$$\begin{aligned}\|\hat{M} - M\|_F &= \|P_1 \hat{M} - P_1 M + P_2 \hat{M} - P_2 M\|_F \\ &= \|P_1 \hat{M} - P_1 M\|_F + \|P_2 \hat{M} - P_2 M\|_F\end{aligned}$$

由于

$$P_1 M = Q_1 Q_1^\top Q \Lambda Q^\top = \sum_{i=1}^{k} \lambda_i q_i q_i^\top$$

若要使第一项最小化，则必须令

$$P_1\hat{M} = P_1M.$$

同理，

$$P_2M = \sum_{i=k+1}^{n} \lambda_i q_i q_i^\top,$$

因为 $\lambda_{k+1}, \ldots, \lambda_n < 0$，则 $-P_2M$ 为正定矩阵。

$$\|P_2\hat{M} - P_2M\|_F = \mathrm{tr}\big(\hat{M}^\top P_2^\top P_2\hat{M} + \hat{M}^\top P_2^\top(-P_2M) + (-M^\top P_2^\top)P_2\hat{M} + M^\top P_2^\top P_2M\big)$$

$$= \|P_2\hat{M}\|_F^2 + \|P_2M\|_F^2 + \mathrm{tr}\big(-\hat{M}^\top P_2M - M^\top P_2^\top\hat{M}\big)$$

$$= \|P_2\hat{M}\|_F^2 + \|P_2M\|_F^2 + 2\langle\hat{M}, -P_2M\rangle$$

因为 $\hat{M}$ 和 $-P_2M$ 都为半正定矩阵，因此 Frobenius 内积

$$\langle\hat{M}, -P_2M\rangle \geq 0.$$

所以

$$\|P_2\hat{M} - P_2M\|_F \geq \|-P_2M\|_F,$$

且仅当 $P_2\hat{M} = 0$ 时取等。

综上，

$$P_1\hat{M} = P_1M = \sum_{i=1}^{k} \lambda_i q_i q_i^\top,$$

$$P_2\hat{M} = 0,$$

因此

$$\hat{M} = P_1M + P_2M = \sum_{i=1}^{k} \lambda_i q_i q_i^\top = Q\Lambda^+Q^\top.$$

(3)

3. 类似第一小问的求解方式，由于

$$\frac{\partial \mathrm{dist}_M^2(x_i, x_j)}{\partial P} = 2(x_i - x_j)(x_i - x_j)^\top P$$

则

$$\frac{\partial \ell_{\mathrm{mnn}}}{\partial P} = (1-\mu)\sum_{i,j\in N_i} \frac{\partial \mathrm{dist}_M^2(x_i, x_j)}{\partial P} + \mu \sum_{i,j\in N_i}\sum_{l} \big(1 - \mathbb{I}(y_i = y_j)\big)a(x_i, x_j, x_l)$$

$$\cdot\left(\frac{\partial \mathrm{dist}_M^2(x_i, x_j)}{\partial P} - \frac{\partial \mathrm{dist}_M^2(x_i, x_l)}{\partial P}\right)$$

该问题不一定为凸问题，对马氏矩阵的分解将问题转变为二次，因 hinge 损失的存在使得存在马氏距离的负项，因此问题可能非凸。尽管会使得损失函数非凸，但一方面，这一分解能够去除对矩阵的约束，同时使得最终优化的结果为低秩矩阵。在实际应用中，这一非凸优化一般能够收敛到较好的结果，在某些情况下甚至达到全局最优。