

《人工智能导论》2026

智能科学与技术学院

第二次作业-机器学习

1. 对数几率回归(Logistic Regression) [25pt]

在本题目中，我们考虑将对数几率回归扩展至多分类问题，在多分类问题中进行预测时，如已知给定 $\mathbf{x}$ 时前 $N-1$ 个类别的后验概率，则可直接得到第 N 个类别的后验概率。因此，分别将前 $N-1$ 个类别的后验概率和第 N 个类别的后验概率进行比较，构建 $N-1$ 组对数几率，分别用 $N-1$ 组线性模型进行预测：

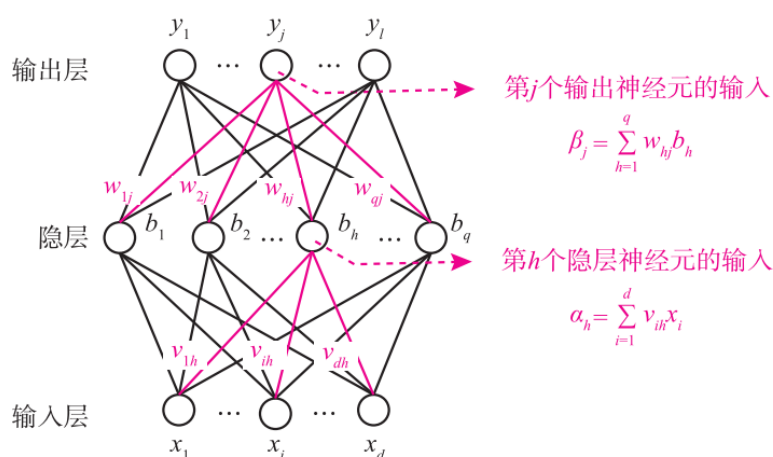
$$\begin{aligned}\ln \frac{P(y = 1 | \mathbf{x})}{P(y = N | \mathbf{x})} &= \mathbf{w}_1^\top \mathbf{x} + b_1, \\ \ln \frac{P(y = 2 | \mathbf{x})}{P(y = N | \mathbf{x})} &= \mathbf{w}_2^\top \mathbf{x} + b_2, \\ &\vdots \\ \ln \frac{P(y = N-1 | \mathbf{x})}{P(y = N | \mathbf{x})} &= \mathbf{w}_{N-1}^\top \mathbf{x} + b_{N-1}.\end{aligned}$$

请通过极大似然估计的方法构建目标函数，给出该多分类对率回归模型的优化目标（以最小化的形式），为了简化符号，你可以将偏移项 b 合并到分类器权重中，比如令 $\boldsymbol{\beta}_n = (\mathbf{w}_n, b_n)$ 。

2. 神经网络(Neural Network) [40pt]

给定训练集 $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$ 。其中， $x_i \in \mathbb{R}^d, y_i \in \mathbb{R}^l$ ，下图给出了一个有 d 个输入、 l 个输出、 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函数和损失函数,并对其梯度进行推导，**请给出具体的推导过程，而不是直接给一个结果。**

(1) [15pt] 在二分类问题 ($l = 1$) 中,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 $(\mathbf{x}, y)$ 有

$$\ell(y, \hat{y}_1) = -[y \log(\hat{y}_1) + (1 - y) \log(1 - \hat{y}_1)].$$

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) [20pt] 当 $l > 1$ 时，网络的预测结果为 $\hat{\mathbf{y}} \in \mathbb{R}^l$ ，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 $\mathbf{y} \in$

$\{0,1\}^l$, 有 $y_i = 1, y_j = 0, j \neq i$ 。对于一个训练样本 $(\mathbf{x}, \mathbf{y})$, 交叉熵损失函数 $\ell(\mathbf{y}, \hat{\mathbf{y}})$ 的定义如下:

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

多分类问题一般使用 SoftMax 函数作为输出层激活函数, 计算公式为:

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

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

(3) [5pt] 分析在二分类问题中使用 SoftMax 激活函数和 Sigmoid 激活函数的联系与区别。

3. 链式法则的矩阵形式 [35pt]

假设有两个隐层的神经网络(隐层神经元数目分别为 q_1 和 q_2), 记网络输入为 $X \in \mathbb{R}^{m \times d}$, 其中 m 为样本数, d 为特征维度. 样本标记为 $Y \in \mathbb{R}^{m \times l}$, 表示 m 个样本的标记维度为 l 。第 i 层的权重矩阵为 $W_i \in \mathbb{R}^{q_i \times q_{i+1}}$, 其中 $q_0 = d$, 偏移为 $b_i \in \mathbb{R}^{1 \times q_{i+1}}$ 。隐层激活函数为 $f(x) = \sigma(x) = \frac{1}{1+e^{-x}}$, 输出层无激活函数 (即使用线性激活函数)。第 i 层的输出表示为 X_i , 网络的输入为第 0 层 $X_0 = X$ 。使用均方误差作为损失函数, 即 $\ell = \frac{1}{2ml} \|X_3 - Y\|_F^2$ 。

(1) [15pt] 请使用向量化的形式描述多层神经网络的前向传播过程,

即从输入 X 到计算出损失的过程。

(2) [20pt] 使用反向传播的方式计算各层参数的导数 $\frac{\partial \ell}{\partial w_i}, \frac{\partial \ell}{\partial b_i}$, 并写出其矩阵形式。