第5章

逻辑回归

5.1 分类问题


分类问题

分类是监督学习的一个核心问题。在监督学习中，当输出变量 Y 取有限个离散值时，预测问题便成为分类问题。这时输入变量可以是离散的，也可以是连续的。监督学习从数据中学习一个分类模型或分类决策函数，称为分类器(classifier)。

分类器对新的输入进行输出的预测，称为分类，可能的输出称为类别(class)。

分类的类别为两个时，称为二分类问题。

例如，根据肿瘤的体积、患者的年龄来判断肿瘤的良性或恶性，或者根据用户的年龄、职业、存款数量来判断信用卡是否会违约。

这两个问题都是二分类问题。

分类的类别为多个时，称为多分类问题。

例如，身高 1.85m、体重 100kg 的男人穿什么尺码的 T 恤？

假设尺码有 S、M、L 3 种，那么这个问题就是多分类问题，分成 3 类。

多分类问题如何解决？

当只有两类时，如图 5-1 所示。

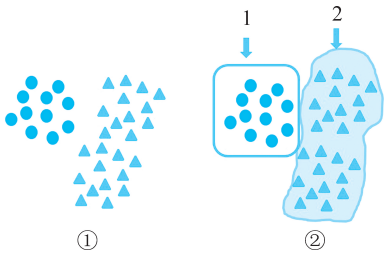

图 5-1 二分类的流程(见彩插)

1. 二分类的流程

先将蓝色圆形数据(见彩插)定义为类型 1，其余数据定义为类型 2。

只需要分类 1 次。

图 5-1 中的步骤为①→②。

2. 多分类的流程

先定义其中一类为类型 1(正类)，其余数据为类型 rest(负类)；接下来去掉类型 1 数据，剩余部分再次进行二分类，分成类型 2 和类型 rest；如果有 n 类，那

就需要分类 $n-1$ 次。

对于 n 类别,需要训练 n 个模型。

图 5-2 是多分类的流程,分类的步骤为①→②→③。

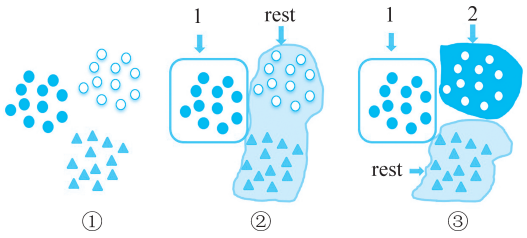

图 5-2 多分类的流程

这个方法称为一对多(one-vs-all,OVA)或一对余(one-vs-rest,OVR)。

本章主要讨论二分类问题。

Sigmoid 函数

5.2 Sigmoid 函数

5.2.1 Sigmoid 函数概述

Sigmoid 函数也叫 Logistic 函数,用于隐层神经元输出,取值范围为(0,1),它可以将一个实数映射到(0,1)区间,可以用来做二分类。在特征相差比较复杂或相差不是特别大时效果比较好。

Sigmoid 作为激活函数有以下特点:该模型的输出变量范围始终为(0,1);图像为 S 形,如图 5-3 所示。

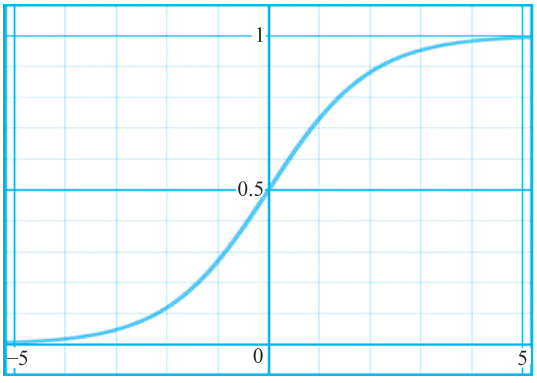

图 5-3 Sigmoid 曲线

5.2.2 Sigmoid 函数的特点

$\sigma(z)$代表一个常用的逻辑函数(logistic function),其为 Sigmoid 函数。

设

$$\sigma(z)=g(z)=\frac{1}{1+e^{-z}} \tag{5.1}$$

其中，$z=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b$。

注意，若表达式

$$h(x)=z=w_0+w_1x_1+w_2x_2+\cdots+w_nx_n+b=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b \tag{5.2}$$

则 b 可以融入 w_0，即

$$z=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} \tag{5.3}$$

从图 5-3 中可以看出，当 $\sigma(z)\geqslant 0.5$ 时，预测 $y=1$；当 $\sigma(z)<0.5$ 时，预测 $y=0$。

两者结合起来，可以得到逻辑回归的损失函数：

$$L(\hat{y},y)=-y\log(\hat{y})-(1-y)\log(1-\hat{y}) \tag{5.4}$$

其中，$\hat{y}$ 为预测值；y 为真实值。

5.2.3　Sigmoid 函数的原理

线性回归的函数 $h(x)=z=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}$，取值范围是$(-\infty,+\infty)$，而分类预测结果需要得到$[0,1]$区间的概率值，在二分类模型中，事件的概率定义为：事件发生与事件不发生的概率之比$\frac{p}{1-p}$，称为事件的发生比(the odds of experiencing an event)，其中 p 为随机事件发生的概率，p 的取值范围为$[0,1]$。对发生比取对数得到

$$\log\frac{p}{1-p} \tag{5.5}$$

而

$$\log\frac{p}{1-p}=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}=z \tag{5.6}$$

求解得到

$$p=\frac{1}{1+e^{-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}}}=\frac{1}{1+e^{-z}} \tag{5.7}$$

从式(5.7)可以知道：p 的值域为$[0,1]$。

将 z 进行逻辑变换，已知

$$g(z)=\frac{1}{1+e^{-z}} \tag{5.8}$$

则

$$g'(z)=g(z)(1-g(z)) \tag{5.9}$$

$g'(z)$的数学推导

$$\begin{aligned}g'(z)&=\left(\frac{1}{1+e^{-z}}\right)'\\&=\frac{e^{-z}}{(1+e^{-z})^2}\\&=\frac{1+e^{-z}-1}{(1+e^{-z})^2}\end{aligned}$$

$$=\frac{1}{(1+e^{-z})}\left(1-\frac{1}{(1+e^{-z})}\right)$$
$$=g(z)(1-g(z)) \tag{5.10}$$

注意：式(5.10)的第二步使用了复合函数的求导公式。


逻辑回归

5.3 逻辑回归

逻辑回归(logistic regression,LR)是经典的分类方法,也是目前应用最广泛的分类算法。逻辑回归虽然被称为回归,但其实际上是分类模型,并常用于二分类,它是分类问题的首选算法。

5.3.1 逻辑回归算法思想

假设一个二分类模型

$$\begin{cases} p(y=1 \mid \boldsymbol{x};\boldsymbol{w}) = h(\boldsymbol{x}) \\ p(y=0 \mid \boldsymbol{x};\boldsymbol{w}) = 1-h(\boldsymbol{x}) \end{cases} \tag{5.11}$$

则式(5.11)可以简化为

$$p(y \mid \boldsymbol{x};\boldsymbol{w}) = (h(\boldsymbol{x}))^{y}(1-h(\boldsymbol{x}))^{1-y} \tag{5.12}$$

逻辑回归模型的假设函数是

$$\sigma(z) = h(\boldsymbol{x}) = g(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}) \tag{5.13}$$

其中,$z=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}$。

$h(\boldsymbol{x})$的作用是,对于给定的输入变量,根据选择的参数计算输出变量等于 1 的可能性,即 $h(\boldsymbol{x})=P(y=1|\boldsymbol{x};\boldsymbol{w})$,即：当 $h(\boldsymbol{x})\geqslant 0.5$ 时,预测 $y=1$;当 $h(\boldsymbol{x})<0.5$ 时,预测 $y=0$。

其中,$\boldsymbol{x}$ 代表特征向量;g 代表一个常用的逻辑函数——S 形函数。由 5.2.3 节可以得到公式为

$$g(z) = \frac{1}{1+e^{-z}} \tag{5.14}$$

对式(5.14)求偏导得

$$g'(z) = g(z)(1-g(z)) \tag{5.15}$$

5.3.2 逻辑回归的原理

1. 逻辑回归的损失函数

损失函数又叫作误差函数,用来衡量算法的运行情况。

通过损失函数来衡量预测输出值和实际值的接近程度。

逻辑回归的损失函数为交叉熵(cross-entropy)损失函数。

设 $\hat{y}$ 为预测值,$\hat{y}=h(\boldsymbol{x})$,$y$ 为真实值,则损失函数为

$$L(\hat{y},y) = -y\log(\hat{y}) - (1-y)\log(1-\hat{y}) \tag{5.16}$$

2. 代价函数

损失函数是在单个训练样本中定义的，它衡量的是算法在单个训练样本中的表现。为了衡量算法在全部训练样本上的表现，需要定义一个算法的代价函数，损失函数只适用于单个训练样本，而代价函数是参数的总代价，所以在训练逻辑回归模型时，需要找到合适的 $\boldsymbol{w}$，使代价函数 J 的总代价降到最低，即：算法的代价函数是对 m 个样本的损失函数求和后除以 m。

$$J(\boldsymbol{w})=\frac{1}{m}\sum_{i=1}^{m}L(\boldsymbol{w})=\frac{1}{m}\sum_{i=1}^{m}(-y^{(i)}\log\hat{y}^{(i)}-(1-y^{(i)})\log(1-\hat{y}^{(i)})) \tag{5.17}$$

即

$$J(\boldsymbol{w})=-\frac{1}{m}\sum_{i=1}^{m}(y^{(i)}\log(h(\boldsymbol{x}^{(i)}))+(1-y^{(i)})\log(1-h(\boldsymbol{x}^{(i)}))) \tag{5.18}$$

当 $y=1$ 时，损失函数 $L=-\log(\hat{y})$，如果想要使损失函数 L 尽可能小，那么 $\hat{y}$ 就要尽可能大，因为 Sigmoid 函数的取值范围是(0,1)，所以 $\hat{y}$ 会无限接近于 1。

当 $y=0$ 时，损失函数 $L=-\log(1-\hat{y})$，如果想要使损失函数 L 尽可能小，那么 $\hat{y}$ 就要尽可能小，因为 Sigmoid 函数的取值范围是(0,1)，所以 $\hat{y}$ 会无限接近于 0。

3. 逻辑回归求解过程

似然函数为

$$L(\boldsymbol{w})=\prod_{i=1}^{m}P(y^{(i)}\mid\boldsymbol{x}^{(i)};\boldsymbol{w})=\prod_{i=1}^{m}(h(\boldsymbol{x}^{(i)}))^{y^{(i)}}(1-h(\boldsymbol{x}^{(i)}))^{1-y^{(i)}} \tag{5.19}$$

似然函数两边取对数，则连乘号变成了连加号：

$$l(\boldsymbol{w})=\log L(\boldsymbol{w})=\sum_{i=1}^{m}(y^{(i)}\log(h(\boldsymbol{x}^{(i)}))+(1-y^{(i)})\log(1-h(\boldsymbol{x}^{(i)}))) \tag{5.20}$$

则代价函数为

$$J(\boldsymbol{w})=-\frac{1}{m}\sum_{i=1}^{m}l(\boldsymbol{w})=-\frac{1}{m}\sum_{i=1}^{m}(y^{(i)}\log(h(\boldsymbol{x}^{(i)}))+(1-y^{(i)})\log(1-h(\boldsymbol{x}^{(i)}))) \tag{5.21}$$

即

$$J(\boldsymbol{w})=-\frac{1}{m}\sum_{i=1}^{m}(y^{(i)}\log(h(\boldsymbol{x}^{(i)}))+(1-y^{(i)})\log(1-h(\boldsymbol{x}^{(i)}))) \tag{5.22}$$

使用梯度下降法求解，权重 $\boldsymbol{w}$ 的迭代公式如式(5.23)，其中 α 为学习率，$\frac{\partial J(\boldsymbol{w})}{\partial \boldsymbol{w}}$ 为梯度。

$$\boldsymbol{w}:=\boldsymbol{w}-\alpha\frac{\partial J(\boldsymbol{w})}{\partial \boldsymbol{w}} \tag{5.23}$$

对 $J(\boldsymbol{w})$ 求偏导，得

$$\frac{\partial}{\partial w_j}J(\boldsymbol{w})=\frac{1}{m}\sum_{i=1}^{m}(h(\boldsymbol{x}^{(i)})-y^{(i)})\boldsymbol{x}_j^{(i)} \tag{5.24}$$

则式(5.23)可以转换为

$$w_j := w_j - \alpha \frac{1}{m}\sum_{i=1}^{m}(h(\boldsymbol{x}^{(i)}) - y^{(i)})\boldsymbol{x}_j^{(i)} \tag{5.25}$$

这样通过若干次迭代,就可以得到最终的 $\boldsymbol{w}$。

式(5.24)的推导过程如下。

由于

$$\begin{aligned} & y^{(i)}\log(h(\boldsymbol{x}^{(i)})) + (1-y^{(i)})\log(1-h(\boldsymbol{x}^{(i)})) \\ =& y^{(i)}\log\left(\frac{1}{1+\mathrm{e}^{-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}^{(i)}}}\right) + (1-y^{(i)})\log\left(1-\frac{1}{1+\mathrm{e}^{-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}^{(i)}}}\right) \\ =& -y^{(i)}\log(1+\mathrm{e}^{-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}^{(i)}}) - (1-y^{(i)})\log(1+\mathrm{e}^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}^{(i)}}) \end{aligned} \tag{5.26}$$

对式(5.22)求偏导得

$$\begin{aligned} \frac{\partial}{\partial w_j}J(\boldsymbol{w}) &= \frac{\partial}{\partial w_j}\left(-\frac{1}{m}\sum_{i=1}^{m}(-y^{(i)}\log(1+\mathrm{e}^{-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}^{(i)}}) - (1-y^{(i)})\log(1+\mathrm{e}^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}^{(i)}}))\right) \\ &= -\frac{1}{m}\sum_{i=1}^{m}\left(-y^{(i)}\frac{-x_j^{(i)}\mathrm{e}^{-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}^{(i)}}}{1+\mathrm{e}^{-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}^{(i)}}} - (1-y^{(i)})\frac{x_j^{(i)}\mathrm{e}^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}^{(i)}}}{1+\mathrm{e}^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}^{(i)}}}\right) \\ &= -\frac{1}{m}\sum_{i=1}^{m}(y^{(i)} - h(\boldsymbol{x}^{(i)}))x_j^{(i)} \\ &= \frac{1}{m}\sum_{i=1}^{m}(h(\boldsymbol{x}^{(i)}) - y^{(i)})x_j^{(i)} \end{aligned} \tag{5.27}$$

4. 逻辑回归的正则化

在 4.5 节中,已经讲到正则化是防止过拟合的主要方法,逻辑回归的正则化公式如下:

$$J(\boldsymbol{w}) = \frac{1}{m}\sum_{i=1}^{m}[-y^{(i)}\log(h(\boldsymbol{x}^{(i)})) - (1-y^{(i)})\log(1-h(\boldsymbol{x}^{(i)}))] + \frac{\lambda}{2m}\sum_{j=1}^{n}w_j^2 \tag{5.28}$$

这个是 L2 正则化,在代价函数后面加上了正则化项 $\frac{\lambda}{2m}\sum_{j=1}^{n}w_j^2$,即正则化系数 λ 乘以 w_j 的平方和($2m$ 是标量,可以忽略)。当 λ 的值开始上升时,方差降低了。

图 5-4 是逻辑回归分类的效果图,在图 5-4(a)中,逻辑回归没有使用正则化,模型过于强调拟合原始数据,而丢失了算法的本质——预测新数据。可以看出,若给出一个新的值使其预测,它将表现得很差,这是过拟合(overfitting)。

在图 5-4(b)中,逻辑回归使用正则化过度,也就是 λ 的值过大了,造成了欠拟合(underfitting)。

在图 5-4(c)中,逻辑回归适当进行了正则化,也就是 λ 的值正合适,适当的正则化起到了防止过拟合的作用。

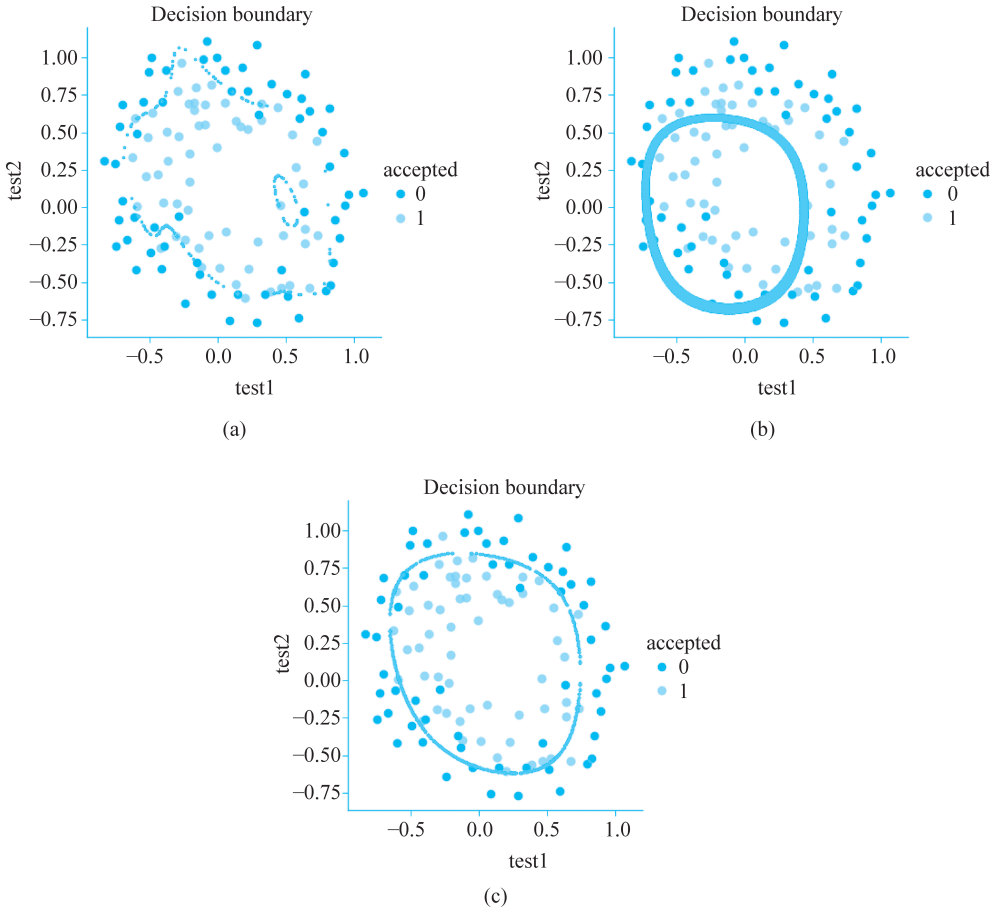

图 5-4　逻辑回归的正则化

5.4　逻辑回归算法总结

逻辑回归是经典的分类方法，也是目前应用最广泛的分类算法。其优缺点如下。

1. 优点

（1）逻辑回归模型形式简单，可解释性好，从特征的权重可以看到不同的特征对最后结果的影响。

（2）训练时便于并行化，在预测时只需要对特征进行线性加权，所以性能比较好，往往适合处理海量 ID 类特征，用 ID 类特征有一个很重要的好处，就是防止信息损失（相对于泛化的 CTR 特征），对于头部资源会有更细致的描述。

（3）资源占用小，尤其是内存。在实际的工程应用中只需要存储权重比较大的特征及其对应的权重。

（4）方便输出结果调整。逻辑回归可以很方便地得到最后的分类结果，因为输出的

是每个样本的概率分数,人们可以很容易地对这些概率分数进行划分阈值(大于某个阈值的是一类,小于某个阈值的是另一类)。

2. 缺点

逻辑回归模型也有一定的局限性,主要表现在以下 3 方面。

(1) 表达能力不强。无法进行特征交叉、特征筛选等一系列“高级操作”(这些工作都得依赖有经验的人来完成,否则会走一些弯路),因此可能造成信息的损失。这就需要开发者有非常丰富的领域经验,才能不走弯路。这样的模型迁移起来比较困难,换一个领域又需要重新提取大量的特征工程。

(2) 准确率并不是很高。因为这毕竟是一个线性模型加一个 Sigmoid 函数,形式非常简单(非常类似线性模型),很难去拟合数据的真实分布。

(3) 处理非线性数据较麻烦。逻辑回归在不引入其他方法的情况下,只能处理线性可分的数据,如果想处理非线性数据,首先对连续特征的处理需要先进行离散化(离散化的目的是引入非线性),人工分箱的方式会引入多种问题。


逻辑回归代码实现

习题

一、单选题

1. 一监狱人脸识别准入系统被用来识别待进入人员的身份,此系统一共识别 4 种不同的人员:狱警、小偷、送餐员、其他。下面(　　)学习方法最适合此种应用需求。

A. 聚类问题　　B. 二分类问题

C. 回归问题　　D. 多分类问题

2. 以下关于分类问题的说法错误的是(　　)。

A. 分类问题的输入属性必须是离散的

B. 分类属于监督学习

C. 多分类问题可以被拆分为多个二分类问题

D. 回归问题在一定条件下可被转化为多分类问题

3. 以下关于逻辑回归与线性回归问题的描述错误的是(　　)。

A. 线性回归的计算方法一般是最小二乘法,逻辑回归的参数计算方法是似然估计法

B. 逻辑回归用于处理分类问题,线性回归用于处理回归问题

C. 线性回归要求输入输出值呈线性关系,逻辑回归不要求

D. 逻辑回归一般要求变量服从正态分布,线性回归一般不要求

4. 以下关于 Sigmoid 函数的优点说法错误的是(　　)。

A. 在深层次神经网络反馈传输中,不易出现梯度消失

B. 函数处处连续,便于求导

C. 可以用于处理二分类问题

D. 可以压缩数据值到(0,1),便于后续处理

5. 逻辑回归的损失函数是(　　)。

A. MAE　　B. MSE　　C. 交叉熵损失函数　D. RMSE

6. 下面(　　)不是 Sigmoid 的特点。

A. 当 $\sigma(z)>0.5$ 时,预测 $y=-1$

B. 当 $\sigma(z)\geqslant 0.5$ 时,预测 $y=1$

C. 当 $\sigma(z)<0.5$ 时,预测 $y=0$

D. $\sigma(z)$的范围为(0,1)

7. 下列(　　)不是逻辑回归的优点。

A. 资源占用少　　B. 模型形式简单

C. 处理非线性数据较容易　　D. 可解释性好

8. 假设有 3 类数据,用 OVR 方法需要分类(　　)次才能完成。

A. 3　　B. 1　　C. 2　　D. 4

9. 以下(　　)不是二分类问题。

A. 根据一个人的身高和体重来判断他(她)的性别

B. 根据肿瘤的体积、患者的年龄来判断肿瘤的良性或恶性

C. 根据用户的年龄、职业、存款数量来判断信用卡是否会违约

D. 身高 1.85m、体重 100kg 的男人穿什么尺码的 T 恤

10. 逻辑回归通常采用(　　)。

A. L1 正则化　　B. Elastic Net 正则化

C. L2 正则化　　D. Dropout 正则化

11. 假设使用逻辑回归进行多类别分类,使用 OVR 分类法。下列说法正确的是(　　)。

A. 对于 n 类别,需要训练 $n-1$ 个模型

B. 对于 n 类别,需要训练 n 个模型

C. 对于 n 类别,只需要训练 1 个模型

D. 以上说法都不对

12. 假设你正在训练一个分类逻辑回归模型。以下(　　)是正确的。

A. 向模型中添加新特征总是会在训练集上获得相同或更好的性能

B. 将正则化引入模型中,总是能在训练集上获得相同或更好的性能

C. 在模型中添加许多新特性有助于防止训练集过拟合

D. 将正则化引入模型中,对于训练集中没有的样本,总是可以获得相同或更好的性能

二、多选题

1. 以下(　　)是正确的。

A. 如果您的模型拟合训练集,那么获取更多数据可能会有帮助

B. 使用一个非常大的训练集使模型不太可能过拟合训练数据

C. 在构建学习算法的第一个版本之前,花大量时间收集数据是一个好主意

D. 逻辑回归使用了 Sigmoid 激活函数

2. 下面(　　)是分类算法。

A. 根据肿瘤的体积、患者的年龄来判断肿瘤的良性或恶性

B. 根据用户的年龄、职业、存款数量来判断信用卡是否会违约

C. 身高 1.85m、体重 100kg 的男人穿什么尺码的 T 恤

D. 根据房屋大小、卫生间数量等特征预估房价

三、判断题

1. 逻辑回归的激活函数是 Sigmoid。(　　)

2. 逻辑回归分类的精度不够高,因此在业界很少用到这个算法。(　　)

3. Sigmoid 函数的范围是(−1,1)。(　　)

4. 逻辑回归的特征一定是离散的。(　　)

5. 逻辑回归算法资源占用小,尤其是内存。(　　)

6. 逻辑回归的损失函数是交叉熵损失。(　　)

参考文献

[1] HOSMER D W, LEMESHOW S, STURDIVANT R X. Applied logistic regression[M]. New Jersey: Wiley New York,2000.

[2] NG A. Machine learning[EB/OL]. Stanford University, 2014. https://www.coursera.org/course/ml.

[3] 李航. 统计学习方法[M]. 2 版. 北京: 清华大学出版社,2019.

[4] HASTIE T, TIBSHIRANI R, FRIEDMAN J. The elements of statistical learning[M]. New York: Springer,2001.

[5] BISHOP C M. Pattern recognition and machine learning[M]. New York: Springer,2006.

[6] BOYD S, VANDENBERGHE L. Convex optimization[M]. Cambridge: Cambridge University Press, 2004.

[7] TIBSHIRANI R. Regression selection and shrinkage via the lasso[J]. Journal of the Royal Statistical Society Series B, 1996, 58(1): 267-288.