人工智能-知识梳理与面试真题

监督学习中的单一模型

目录 (Table of Contents)

K Nearest Neighbour

算法： 给一个点 x_0, 我们在训练集中找到 K 个最邻近的点，根据这 K个点的分类来决定 x_0 的类别。

度量：如果特征是连续的，选择 Euclidean, Manhattan or Minkowski 度量；如果特征是分类变量，选择 Hamming 度量。

K 的选择：如果是2分类问题，K 一般选取奇数。K 值越小，噪音对结果的影响越大；K 值越大，计算量越大，甚至可能导致分类完全错误。交叉验证是选取 K 值的有效方法。

优点：

- 无变量算法，无需对数据的分布做任何假设。
- K 值固定时，无需训练模型
- 依赖局部信息，适应各种数据复杂分布
- 算法简单易实现

缺点:

- 应用算法之前，需对所有特征做标准化处理 $x' = \frac{x - x_{min}}{x_{max} - x_{min}}$
- 计算时需要很大的内存
- 计算时间复杂度高

- 易于受维度灾难的影响
- K 值得选取会影响模型结果

Naive Bayes

算法：在训练集中算出每一个类别的概率 $P(Y)$, 和每一个特征的条件概率 $P(X_i|Y)$。然后给定一个点 x_0，他的特征分别是 $\{X_1, X_2, ..., X_n\}$, 算出他属于每个类别的条件概率 $P(Y=y_k|X_1, X_2, ..., X_n)$, 找到最大概率对应的类。

目标函数：最大化后验概率

$$\max_y P(Y=y)\prod_i P(X_i|Y=y)$$

优点：

- 模型训练速度快，分类快
- 对不相关的特征值不敏感
- 可以处理连续和离散的特征
- 可以处理实时数据

缺点:

- 各特征必须是相互独立的

Logistic Regression

算法：根据训练集的情况，用一个逻辑函数来建模分类。

目标函数：最大化条件概率

$$\max_W \sum_l \log P(Y^l|X^l, W)$$

优化算法: Limited-memory BFGS 算法（Quasi-Newton 方法）

正则化: 逻辑回归容易产生过拟合，特别当数据稀疏或者高维时。 减少过拟合的一个方法是正则化，也就是在目标函数中加入一个惩罚函数，目标函数变为：

$$\max_W \sum_l \log P(Y^l|X^l, W) - \rho(W)$$

本质上 $\rho(W) = \lambda\|W\|_0$, 近似于 $\rho(W) = \lambda\|W\|_1$, 也就是 L_1 正则化 (Lasso Regression)。当 $\rho(W) = \frac{\lambda}{2}\|W\|_2^2$时，是 L_2 正则化 (Ridge Regression)。

L_1与L_2正则化的比较:

L_1正则化	**L_2正则化**

L_1**正则化**	L_2**正则化**
$\lambda\|W\|_1$	$\frac{\lambda}{2}\|W\|_2^2$
无解析解	有解析解
计算效率底	计算效率高
稀疏性强	稀疏性弱
有特征选择的功能	无特征选择的功能

优点：

- 当数据集是单一决策边界时，逻辑回归表现很好
- 数据集的决策边界无需平行于坐标轴
- 正则化的逻辑回归有较小的方差，不易于过拟合

缺点：

- 逻辑回归要求决策边界必须是线性的
- 如果数据中有过多的异常值，逻辑回归会表现得比较差

Support Vector Machine

算法：根据训练集，建立两个超平面，使得它们之间的距离尽可能大，进而找到最大间隔超平面，从而进行分类。

目标函数：最大化间隔（两个超平面的距离）

$$\max \frac{2}{\|w\|^2} = \min \frac{1}{2}\|w\|^2$$

为减少过拟合，加入变量 ϵ，新的目标函数是：

$$\min \frac{1}{2}\|w\|^2 + C(\sum_i \epsilon_i)$$

过拟合： C 值增大时，间隔的宽度增加，模型不易过拟合。此时，模型偏差增加，方差减小。

优点：

- 可分类高维数据

缺点：

- 映射后的决策边界必须是线性的
- 计算时间复杂度高
- 过多噪音时表现不好

Decision Tree

算法：根据训练集的数据特征，创建一个模型来学习决策规律。

目标函数：每一次分割时，最大化信息增益

$$\max \;\; IG\;(D_p, f) = I(D_p) - \sum_{j=1}^{m} \frac{N_j}{N_p} I(D_j)$$

信息：符号 w 的信息是

$$I(w) = -log\; p(w)$$

香农熵：给定一个概率分布 $P = (p_1, p_2, ..., p_n)$, 那么这个概率所携带的信息，就叫做 P 的熵

$$I_E(P) = -\sum_{i=1}^{n} p_i * log\; p_i$$

信息增益：特征 T 与其所有可能值得熵的差

$$IG\;(p, T) = I_E(p) - \sum_{j=1}^{n} p_j I_E(p_j)$$

目标函数：每一次分割时，最大化信息增益

$$\max \;\; IG\;(D_p, f) = I(D_p) - \sum_{j=1}^{m} \frac{N_j}{N_p} I(D_j)$$

其中 f 是用来分割的特征， D_p 是所有的父节点，D_j 是所有的子节点， I 是熵的度量函数，N_p 是父节点的数据总数， N_j 是第 j 个子节点的数据总数。

二分树的目标函数：

$$\max \;\; IG\;(D_p, f) = I(D_p) - \frac{N_{left}}{N_p} I(D_{left}) - \frac{N_{right}}{N_p} I(D_{right})$$

熵的度量函数：常见有3种，吉尼指数 (I_G), 熵 (I_E), 分类误差 (I_C)。 对于一个点 t, $p(i|t)$ 是属于类别 c 的数据的个数，那么

1. 当度量函数是熵时，模型是最大化互信息

$$I_E(t) = -\sum_{i=1}^{c} p(i|t)\; log\; p(i|t)$$

2. 当度量函数是吉尼指数时，模型是最小化分类错误的概率

$$I_G(t) = \sum_{i=1}^{c} p(i|t)(1 - p(i|t)) = 1 - \sum_{i=1}^{c} p(i|t)^2$$

3. 当度量函数是分类误差时，模型是最小化分类错误

$$I_E(t) = 1 - \max\{p(i|t)\}$$

优点：

- 简单易懂，可以可视化
- 几乎不需要过多的数据预处理

- 算法时间复杂度低
- 可以同时处理 numerical data and categorical data
- white box model, 容易解释预测结果
- 即使数据有一些噪音，模型表现也会很好

缺点：

- 容易过拟合
- 不稳定
- 不善于处理不平衡数据集

面试真题

1. 朴素贝叶斯算法的假设前提？
2. 逻辑回归中 softmax 函数是什么？
3. 如何减少逻辑回归中的过拟合问题？
4. 什么是维度灾难？ 为什么 KNN 算法易受维度灾难影响？
5. 决策树的优化函数是什么？
6. 解释一下熵和基尼系数？
7. 使用不同的度量函数，决策树的优化函数有什么不同？ 模型结果有什么不同？
8. 如何减少决策树的过拟合？
9. 支持向量机的优化问题是什么？
10. 解释一下 KKT Condition？

作业

1. 二分类逻辑回归的分布函数是什么？
2. 如果想减少支持向量机的过拟合，要如何调整参数？
3. KNN 中 K 值一般如何选取？