机器学习

1.获取数据

2.数据基本处理

3.特征工程

4.机器学习（模型训练）

5.模型评估

监督学习

输入数据：特征值和目标值

输出数据：连续值（回归）、有限个离散值（分类）

无监督学习

输入数据：特征值

输出数据：一般是类别划分

半监督学习：同时进行监督和无监督学习

训练集包含有标记样本和未标记样本

强化学习：代理agent、动作action、环境enviroment、奖励reward、观察者observation

自动进行决策，并可以做连续决策。

评估：

分类模型评估：准确率、精确率、召回率、F1-score、AUC

回归模型评估：均方根误差、相对平方误差、平均绝对误差、相对绝对误差

拟合：

欠拟合：学习特征太少

过拟合：特征学的太多，甚至有错误的特征，最终判断出错

启动juypter步骤

1.workon ai

2.在存储代码的界面打开终端

3.输入juypter lab --allow-root

K-近临算法

定义：如何一个样本在特征空间中的K个最相似的大多数都属于一个类别，则该样本也属于

算法：求欧氏距离 取最近的K个值，则他属于这个K中占比多的

Api：sklearn.neighbors.KNsighborsClassifier(n\_neighbors = 5)

seaborn.lmplot(x第一维数据 , y第二维数据 ,data数据集 ，hue目标值，fit\_reg = False是否线性拟合)

**距离**

1欧式距离：

2哈曼顿距离：

3切比雪夫距离：最长的维度距离

4闵可夫斯基距离：

当P=1：哈曼顿距离

当P=2：欧式距离

当P—>+∞：切比雪夫距离

缺点：维度单位当作一样了，没有权重，多用于同类的多维度特征值

5标准化欧式距离： Sk：标准差

6余弦距离：

7汉名距离：两个等长字符串（二进制）。将其中一个变为另外一个所替换的最小次数。

8杰卡德距离A交B/A并B

杰卡德相似系数1-A交B/A并B

9马氏距离：正态标准化

**Kd树建立：离散数据分类**

输入：多组离散的特征值 输出：一个已知的目标值

1.取方差大的维度的中位数进行分割

2.取另一维度的中位数继续分割

分割到没有

**线性回归：连续数据预测**

输出：多组离散的n维数据 2.输出：符合所有数据的拟合函数

1.公式：

其中

2.损失函数：=X是特征值矩阵 y是真实值 w是权重矩阵

3.优化函数

3.1正规方程： X：特征值矩阵 y：目标值矩阵 无法解决拟合问题

3.2 梯度下降： 最大变化率 ：学习率（步长）

无法保证找到最小值 用于大规模数据

3.2.1全梯度下降算法FG

目标函数：对其求和再取平均值

每次更新都要计算所有数据集的全部梯度，因此计算速度很慢

无法处理超出内存容量的数据集（基于全部点的损失函数）

运行时不能增加新的样本，不能更新

3.2.2 随机梯度SG（基于随机点的损失函数）

3.2.3 小批量梯度下降算法mini-batch（基于部分随机点的损失函数）

3.2.4 随机平均梯度下降算法SAG（基于旧梯度和当前的随机点的损失函数求得平均梯度）

**逻辑回归：二分类问题**

输入：。就是线性回归的输出

3.

输出：通过激活函数（sigmod：）输出[0,1]的数，并通过阈值判断属于哪个类别

1.损失函数：对数似然损失

代表预测值 y代表真实值

完整的损失函数：

2.优化：采用梯度下降对损失函数进行优化

**决策树**

1.决策树划分依据信息增益。可取值较多的属性信息增益更大ID3

1.1信息熵：越集中、越有序，就越小

公式：=

1.2.信息增益：推到方法见图片

1.求解总信息熵（按照目标值分类）

2.求解对应属性的条件熵（按照对应属性算）

3.使用信息增益求解公式：

4.比较每一个属性的信息增益，信息增益越大，表示影响越大。

2.决策树划分依据信息增益率C4.5。 不一定是2叉树

属性固有值：

信息增益率：

3.决策树划分依据基尼值，基尼指数CART 一定是二叉树

基尼值：从数据集中随机抽取两个样本，其目标值不一样的概率，基尼值越小，样本越纯 纯度越小则越优先判断

多变量决策树OC1

评估：

样本不均衡的时候用ROC曲线 按照概率由大到小计算（TPR，FPR）依次连接

和AUC指标来评判AUC越接近1越好 越接近0.5越胡扯，接近零表示全分错了

**集成学习**

Bosting：

Bagging

GBDT

XGBoost

**聚类算法（无监督）**