机器学习

作业二

一. (30 points) 线性回归

给定包含 m 个样例的数据集 $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_m, y_m)\}$，其中 $\mathbf{x}_i = (x_{i1}; x_{i2}; \cdots; x_{id}) \in \mathbb{R}^d$, $y_i \in \mathbb{R}$ 为 $\mathbf{x}_i$ 的实数标记。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平方之和最小:

$$(\mathbf{w}^*, b^*) = \underset{(\mathbf{w},b)}{\operatorname{argmin}} \frac{1}{2} \sum_{i=1}^{m} (y_i - (\mathbf{w}^\top \mathbf{x}_i + b))^2 \tag{1}$$

即寻找一组权重 $(\mathbf{w}, b)$，使其对 D 中示例预测的整体误差最小。定义 $y = [y_1; y_2; \cdots; y_m] \in \mathbb{R}^m$ 且 $\mathbf{X} = [\mathbf{x}_1^\top; \mathbf{x}_2^\top; \cdots; \mathbf{x}_m^\top] \in \mathbb{R}^{m \times d}$.

1. 请将线性回归的优化过程使用矩阵进行表示; (10 points)

2. 使用矩阵形式给出权重 $\mathbf{w}^*$ 和偏移 b^* 最优解的表达式; (10 points)

3. 针对波士顿房价预测数据集 (boston), 编程实现原始线性回归模型。请基于闭式解在训练集上构建模型, 计算测试集上的均方误差 (Mean Square Error, MSE)。请参考如下形式完成函数 linear_regression 和 MSE 的代码。除示例代码中使用到的 sklearn 库函数外, 不能使用其他的 sklearn 函数, 需要基于 numpy 实现线性回归模型闭式解以及 MSE 的计算. (10 points)

In [3]:
```python
from sklearn.datasets import load_boston
from sklearn.model_selection import train_test_split

X, y = load_boston(return_X_y=True)
trainx , testx , trainy , testy = train_test_split(X, y, test_size = 0.33, random_state =42)

def linear_regression(X_train , y_train):
''' 线性回归
    : 参数X_train: np.ndarray , 形状为(n, d), n 个d 维训练样本
    : 参数y_train: np.ndarray , 形状为(n, 1), 每个样本的标签
    : 返回: 权重矩阵W
'''

def MSE(X_train , y_train , X_test , y_test):
''' 调用linear_regression得到权重矩阵W后计算MSE
    : 参数X_train: np.ndarray , 形状为(n, d), n 个d 维训练样本
    : 参数y_train: np.ndarray , 形状为(n, 1), 每个训练样本的标签
    : 参数X_test: np.ndarray , 形状为(m, d), m 个d 维测试样本
    : 参数y_test: np.ndarray , 形状为(m, 1), 每个测试样本的标签
    : 返回: 标量, MSE 值
'''


linear_regression_MSE= MSE(trainx , trainy , testx , testy)
```

图 1: 示例代码

二. (25 points) 对率回归

信用卡欺诈检测数据集 (Credit Card Fraud Detection) 包含了 2013 年 9 月通过信用卡进行的欧洲持卡人的交易。这是一个典型的类别不平衡数据集，数据集中正常交易的标签远多于欺诈交易。请你根据附件中提供的该数据集完成以下问题:

1. 数据集共有 284807 个样本，其中只有 492 个负样本。请按照训练集和测试集比例 7:3 的方式划分数据集（使用固定的随机种子）。在训练集上分别训练对率回归 (Logistic Regression) 与决策树两类模型，并计算两者在测试集上的精度（至少包含：准确率、召回率、F1 分数、AUC）。请展示完整代码，并在同一张图上绘制两种模型的 ROC 曲线。(5 points)

2. 保持测试集不变，在训练集中对多数类（正例）进行随机下采样，分别构造 **负例: 正例**的三种比例：**1 : 10**、**1 : 100**、**1 : 200**。在这三个新的训练集上分别训练对率回归与决策树模型，并记录每个模型的精度。观察并比较这几组实验的结果，结合准确率与召回率的定义，请说明不平衡数据集对模型的影响（8 points）;

3. 除了上述第 2 问的随机欠采样的方式以外，对小类样本的“过采样”也是处理不平衡问题的基本策略。一种经典的方法为人工合成的过采样技术 (Synthetic Minority Over-sampling Technique, SMOTE), 其在合成样本时寻找小类中某一个样本的近邻, 并在该样本与近邻之间进行差值, 作为合成的新样本。请查阅相关资料，实现 SMOTE 算法中的 over sampling 函数, 以代码块的形式附于下方即可（8 points）;

```python
"""
注意:
1. 这个框架提供了基本的结构，您需要完成所有标记为 'pass' 的函数。
2. 记得处理数值稳定性问题，例如在计算对数时避免除以零。
3. 在报告中详细讨论您的观察结果和任何有趣的发现。
"""
class SMOTE(object):
    def __init__(self , X, y, N, K, random_state =0):
        self.N = N # 每个小类样本合成样本个数
        self.K = K # 近邻个数
        self.label = y # 进行数据增强的类别
        self.sample = X
        self.n_sample , self.n = self.sample.shape # 获得样本个数，特征个数

    def over_sampling(self):
        pass
```

Listing 1: SMOTE 模型接口

4. 请说明 SMOTE 算法的缺点并讨论可能的改进方案（4 points）。

三. (20 points) 决策树

二分类数据集 D 含 20 个样本，正类 8、反类 12。候选离散属性为 A 与 B，其在 D 中的类分布如下，若需对数，取底数 2：

A 的取值	正类	反类	小计
$A{=}a_1$	2	8	10
$A{=}a_2$	6	4	10
合计	8	12	20

B 的取值	正类	反类	小计
$B{=}b_1$	3	2	5
$B{=}b_2$	1	4	5
$B{=}b_3$	4	6	10
合计	8	12	20

请完成：

1. 计算数据集 D 的信息熵 $\mathrm{Ent}(D)$ (5 points)；

2. 分别计算信息增益 $\mathrm{Gain}(D,A)$、$\mathrm{Gain}(D,B)$，以及对应的增益率 $\mathrm{GainRatio}(D,A)$、$\mathrm{GainRatio}(D,B)$ (5 points)；

3. 分别给出在 ID3（最大信息增益）、C4.5（最大增益率）、CART 分类树（最大基尼指数下降）三种准则下根节点最优划分的属性名称，并简述理由（可给出近似值）(10 points)。

四. (25 points) 机器学习中的过拟合现象

本题以决策树与线性模型为例，探究机器学习中的过拟合现象。机器学习希望训练得到的模型在新样本上保持较好的泛化性能；如果在训练集上将模型训练得 “过好”，捕获了与任务无关的偶然特性，会导致泛化能力下降，即为过拟合。

1. 请简要总结决策树与线性模型的工作原理及其常用的缓解过拟合手段（5 points）；

2. 请使用 scikit-learn 实现决策树模型，并扰动决策树的最大深度 max_depth，一般来说，max_depth 的值越大，决策树越复杂，越容易过拟合，实验并比较测试集精度，讨论并分析观察到的过拟合现象等 (5 points)；

3. 对决策树算法的未剪枝、预剪枝和后剪枝进行实验比较，并进行适当的统计显著性检验 (7 points)；

4. 使用 scikit-learn 实现一个线性模型：通过调整正则化强度与训练轮数，控制模型有效复杂度，实验并比较测试集精度，讨论并分析观察到的过拟合现象等（8 points）。

注：从 UCI 机器学习库中选择 1 至 3 个数据集进行实验。