**机器学习实验二：回归分析**

**一、 实验目标**

1. 从零开始编程实现线性回归的核心算法，理解其数学原理。
2. 掌握并对比解析解（正规方程）与迭代解（梯度下降）的求解过程与优劣。
3. 在实验中理解学习率、正则化、模型复杂度等关键因素对模型性能的影响。
4. 学会通过分析训练/测试误差、绘制收敛曲线等方式来评估和诊断模型。

**二、 实验内容与要求**

**基础实验要求（4分）：**

**任务1：线性回归 – 最小二乘法**

* **数据集**: dataset\_regression.csv
  + **下载链接**: [https://storage.googleapis.com/tf-datasets/notebooks/secondary\_school\_pe\_dataset\_regression.csv](https://www.google.com/search?q=https://storage.googleapis.com/tf-datasets/notebooks/secondary_school_pe_dataset_regression.csv) (请将文件名保存为 dataset\_regression.csv)
* **要求**:
  + 根据数据集，利用正规方程（Normal Equation）求得最小二乘解 **θ = (XᵀX)⁻¹Xᵀy**
  + 使用你得到的回归方程，自行构造5个新的测试样本点并进行预测。
  + 画出训练数据的散点图和拟合的回归直线，给出训练误差和测试误差。
* **思考**:
  + 正规方程的求解核心是哪一步？这一步在什么情况下可能会失败？

**任务2：线性回归 - 梯度下降法（批量和随机两种）**

* **数据集**: winequality-white.csv (白葡萄酒质量)
  + **下载链接**: <https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-white.csv>
* **要求**:
  + 读取数据集，将其按 4:1 的比例随机划分为训练集和测试集。**注意**：数据的第一行是表头，分隔符是分号;。
  + 编程实现批量梯度下降（BGD）或随机梯度下降（SGD）算法。
  + 训练你的线性回归模型，并记录下每次迭代后在训练集上的均方误差（MSE），输出最终模型在训练集和测试集上的MSE。
  + 可视化：画出训练过程中的 MSE 收敛曲线（横轴为迭代次数，纵轴为MSE）。

**中级实验要求（1分）：**

**任务3：超参数调优 - 学习率分析**

* **要求**:
  1. 基于任务2的代码，尝试几个不同的学习率。
  2. 为每个学习率绘制训练过程的 MSE 收敛曲线。
  3. 分析并解释：分析最佳学习率，思考学习率过大或过小分别对模型收敛过程产生了什么影响？

**高级实验要求（1分）：**

**任务4：正则化 - 岭回归**

* **要求**:
  1. 使用任务2的数据集 winequality-white.csv。
  2. 编程实现岭回归（Ridge Regression）。你可以选择解析法、批量梯度下降法或随机梯度下降法实现。
  3. 设置一个正则化参数。
  4. 训练模型并计算最终在训练集和测试集上的平均误差。
  5. （选做）尝试不同的正则化参数值，观察它对模型参数（权重）大小以及测试误差的影响。

**拓展任务：模型选择 - 多项式回归**

* **要求**:
  1. 使用任务2的数据集 winequality-white.csv。
  2. 为数据增加多项式特征，将原始线性回归模型扩展为多项式回归模型。
  3. 尝试不同的多项式阶数（例如：1, 2, 4,），并对每个阶数都训练一个模型。
  4. 可视化：将不同阶数下的拟合曲线与原始数据散点图绘制在一起。
  5. 分析并解释：哪个阶数拟合效果最好？当阶数过高时，你观察到了什么现象（过拟合/欠拟合）？