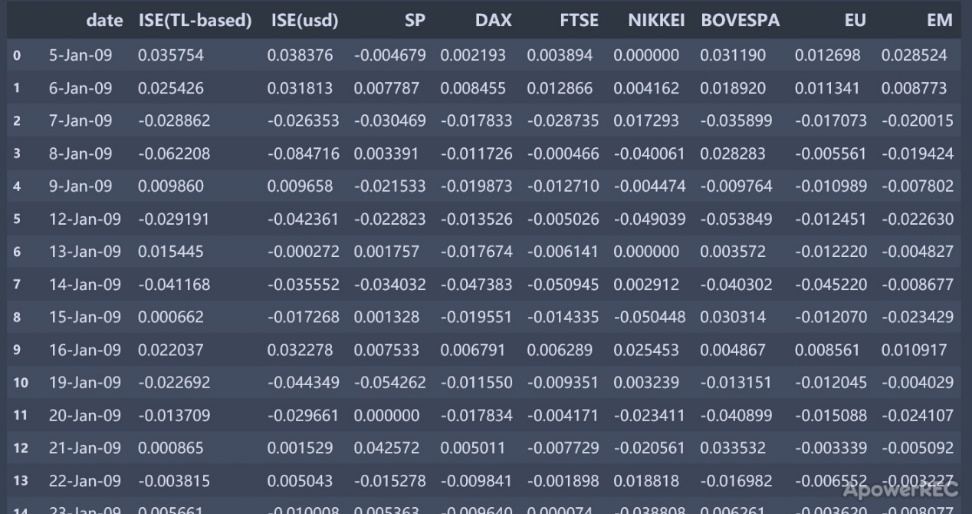
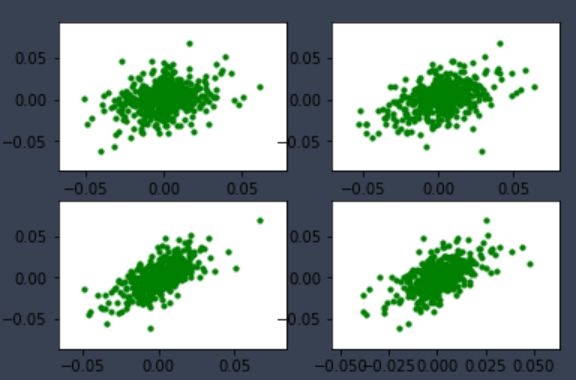
实验报告

周韧哲 181220076

任务一 回归

首先读取文件，发现文件中的数据类型为str，故经过处理后转成float。（如图）

将第二列数据作为y轴，其他八列数据为x轴，分别画出散点图

图片包含 屏幕截图

已生成极高可信度的说明

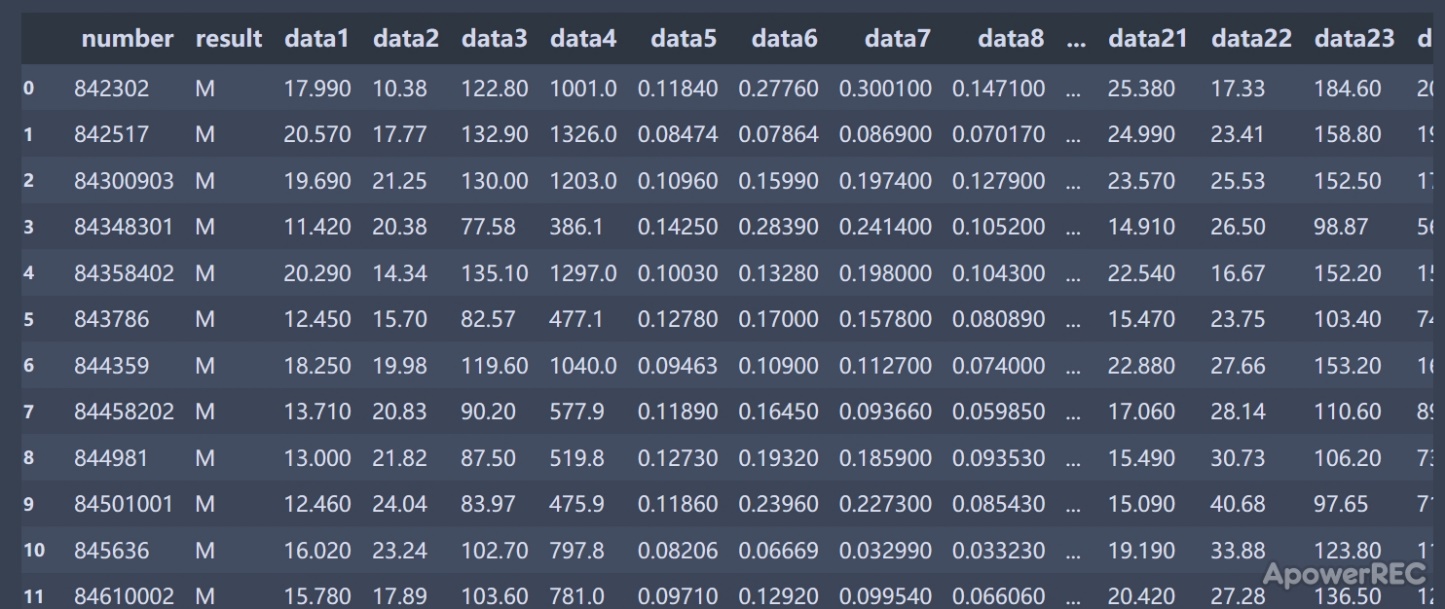
可以看出，上图中第一幅散点图两者线性相关性最强，而其余也大致呈线性，故可以用线性回归分析。

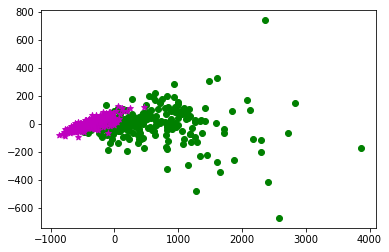
导入LinearRegression进行回归分析，导入train\_test\_split进行数据集划分，随机抽取70%的样本作为训练集，30%作为测试集，评估指标为差值均方即MSE，差值均方越小，模型效果越好。由于上述数据的较强的线性相关性，我们估计误差会比较小。得出训练误差为2.3002264298141784e-05，泛化误差为2.8215961776885618e-05。

对原数据用sklearn中的PCA进行降维，前四个的主成分比列分别为0.63500318、 0.14567663、0.09174919、0.06589358，总比例约为0.93，所以我们降成4维，用mse评估方法得出训练误差为2.6745442101005458e-05，泛化误差为2.6730928021083106e-05，比较可得降维后的误差比原误差略大。提高维数再分析可以看出误差越来越接近原误差，8维时与原误差相等。

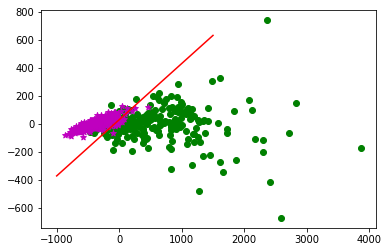
任务二 二分类

读取文件：

用PCA降维，前两个主成分的比例为0.98204467、0.01617649，总比例约为0.99，所以将数据降成两维，将标记为M的点绘成紫色，标记为B的点绘成绿色，绘制出2维图像：



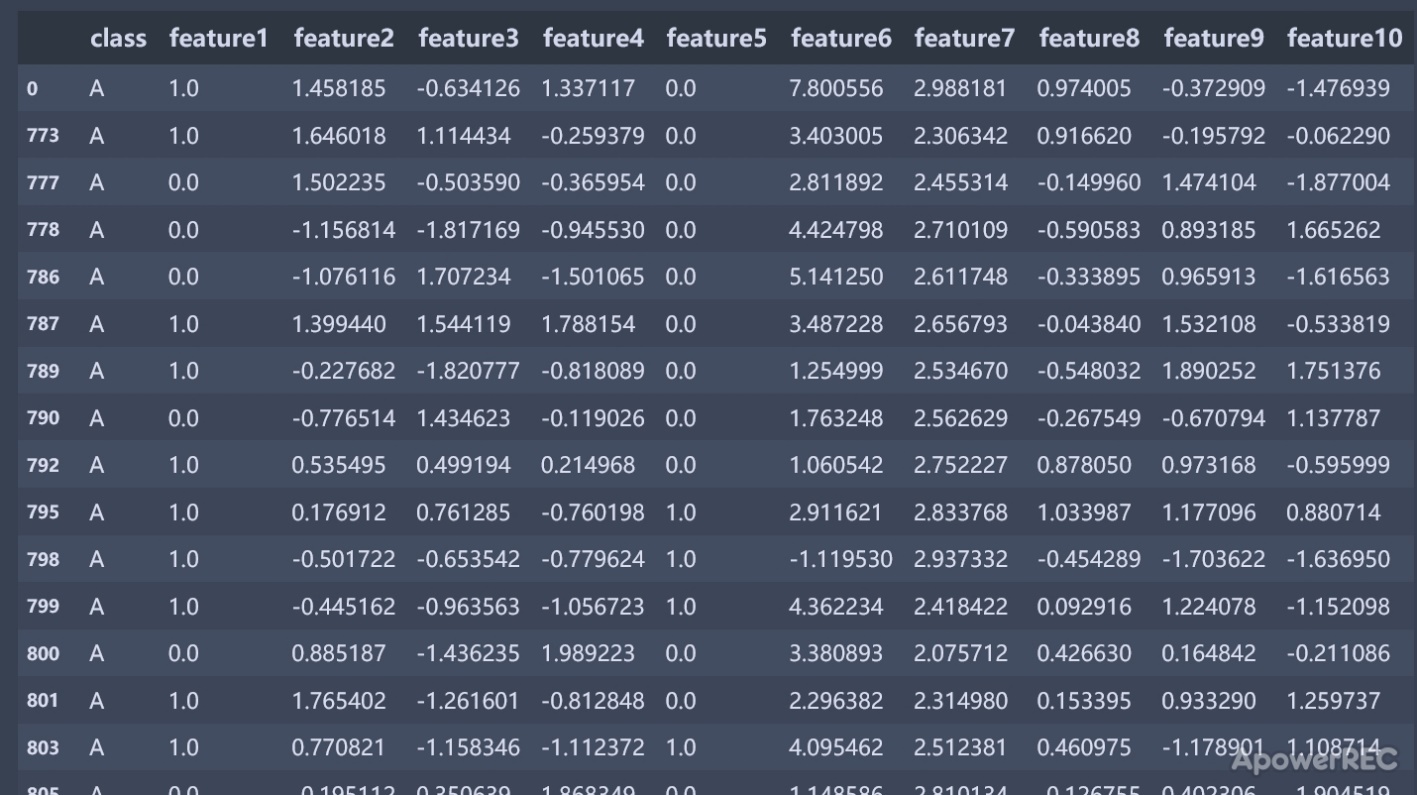
由图容易看出数据分成了两簇，故可以将数据二分类。导入LogisticRegression包进行二分类，导入train\_test\_split进行数据集划分，随机抽取70%的样本作为训练集，30%作为测试集。因为涉及癌症的模型训练样本大，但患癌症的人的比例是非常小的，这就导致如果一个病人患了恶性癌症，而我们错误预测了他没有患恶性癌症，这会容易导致他因耽搁治疗而去世。所以我们用正确预测出患恶性癌症的数量/患恶性癌症的总数量来作为评估指标，即召回率recall\_rate。得出训练误差（召回率）为0.8322147651006712，泛化误差（召回率）为0.9206349206349206，容易得出分界直线方程，绘图如下：

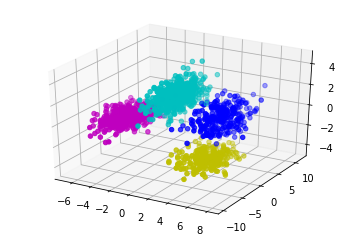


下一步用原数据集作分类训练，得出降维前的训练误差（召回率）为0.9194630872483222，泛化误差（召回率）为0.9365079365079365，显然比降维后的召回率小，即降维后的模型效果比降维前好。

任务三 多分类

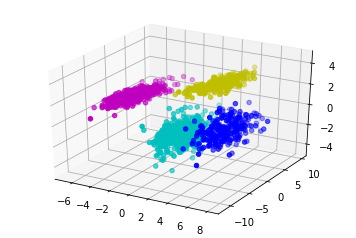
读取数据后，按类别将数据排序依次为A,B,C,D:

用PCA算法降维，前3个主成分的比例为0.40964276、0.28260179、0.10937106，总比列约为0.7，所以我们降成3维，可视化图像如下：



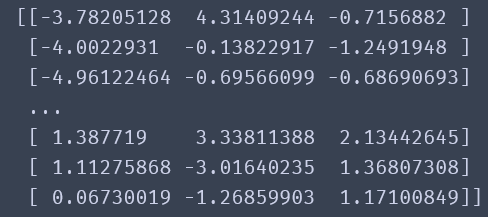
数据明显分为了4簇。

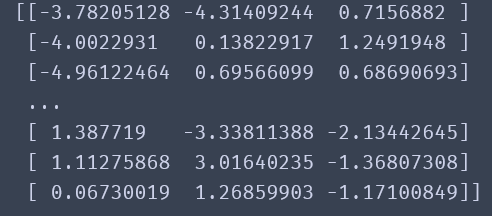
再用我自己实现的PCA算法my\_pca降维后的图像为：



可以看出此图是上图以z轴对称出来的

再来看看两种方法降维后的矩阵：





可以看出我my\_pca降维后的矩阵除了第二列的符号与PCA的相反外，其余均相同，而造成这个的原因是降维时选取的单位特征向量的符号不同。

导入LogisticRegression进行多分类，定义error\_rate错误率为评价指标，即预测错误的样本数量/总样本数量，导入train\_test\_split进行数据集划分，随机抽取70%的样本作为训练集，30%作为测试集，得出训练误差为0.00285714285714285，泛化误差为0.0022222222222222222。