第一部分：简述题

1. 简述 PCA 的原理、学习模型和算法步骤。
2. 简述 LAD 的原理和学习模型。
3. 作为一类非线性降维方法，简述流形学习的基本思想。
4. 根据特征选择与分类器的结合程度，简述特征提取的主要方法，指出各类方法的特点。
5. 简述最优特征提取的基本思想。

1. 主成分分析 (PCA)：

原理 (根据 PPT 和向老师讲的，有三条思路)：

(1) 寻找一组方差较大的方向，将原始数据（样本）在该方向进行投影。即将数据在新坐标系下进行表示，保留少数在方差最大方向上的投影，达到数据变换、尽可能地保留原始数据信息和降维的目的。经过优化问题的求解之后发现投影方向是数据的协方差矩阵的特征向量方向。

(2) 最大化投影之后的误差，使得样本点在这个超平面上的投影能够尽可能地分开，对投影设置后的样本计算协方差，使其方差最大 (协方差的迹最大)，优化的函数本质上和 (1) 的是接近的 (假定样本被零均值化了，这样不影响其协方差矩阵)。经过优化问题的求解之后发现投影方向是数据的协方差矩阵的特征向量方向。

(3) 对投影变换之后的值进行重构，希望能够重构回原来的数值，要求重构回来的数值和原来的数值之间的误差最小 (误差定义为平方和误差)，由此构造优化函数 (假定样本被零均值化了，这样不影响其协方差矩阵)，最终的优化函数的构造和 (2) 近似。

学习模型：

尽管有三种不同的思路，但是其具体的计算都是类似的，这里主要以 (1) 为例进行说明：

设要投影的方向为 w，原来的数值 x_i 以及其均值 $\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$，以及线性变换之后的值 $y_i = w^T x_i$ 以及其均值 $\overline{y} = w^T \overline{x}$，注意这里的投影方向是一个一维的向量，计算出的 y_i 只是一个数字，计算 y_i 的方差为:

$$var = \frac{1}{n}\sum_{i=1}^{n}(w^T x_i - w^T \overline{x})^2$$

要得到变化之后最大方差，所以说只要计算：

$$max \frac{1}{n}\sum_{i=1}^{n}(w^T x_i - w^T \overline{x})^2$$

$$s.t. w^T w = 1$$

通过化简可以计算出来：

$$var = \frac{1}{n}\sum_{i=1}^{n}(w^T x_i - w^T \overline{x})^2 = w^T C w$$

其中，C 是数据的协方差矩阵：

$$C = \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})(x_i - \overline{x})^T$$

对于这个优化问题，拉格朗日乘子法：

$$obj = w^T C w - \lambda(w^T w - 1)$$

求解得到：

$$Cw = \lambda w$$

将解带入原式得到：

$$w^T Cw = w^T \lambda w = \lambda w^T w = \lambda$$

从中可以看出，λ 和 w 是数据的协方差矩阵 C 对应的最大的特征值对应的特征向量。所以说变换矩阵 W 是选择数据的协方差矩阵 C 尽可能大的 m 个特征值 λ_i 和其对应的特征向量 w_i 构成 $W = [w_1, w_2, \cdots, w_m]$。

算法步骤：

step1: 计算数据的协方差矩阵 $C = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})^T$

step2: 根据先关的要求 (比如说选择的特征值的和比上所有的特征值的和大于等于 95%) 选择协方差矩阵 C 的前 m 个最大的特征值 $\lambda_1 > \lambda_2 > \cdots > \lambda_m$ 和其对应的特征向量 $w_1, w_2, \cdots, w_m$，构成线性变换的向量 $W = [w_1, w_2, \cdots, w_m]$.

step3: 计算线性变换之后的降维的数据 $y_i = W^T x_i$，注意这里的 y_i 是一个向量.

2.LDA

原理：

寻找一组线性变换，使得变换之后的两个类别的中心尽可能的远，而每一个类的类内的协方差尽可能的小。

学习模型：

寻找投影的最佳方向 w，是的满足两个类的数据投影之后，两个类的中心尽可能的远，两个类的类内协方差尽可能的小，为了表示类内协方差部分，定义类内散度矩阵 S_w，定义两个类的中心，定义类间散度矩阵 S_b，分别为如下所示：

$$S_w = \Sigma_0 + \Sigma_1 = \Sigma_{x \in X_0}(x - \mu_0)(x - \mu_0)^T + \Sigma_{x \in X_1}(x - \mu_1)(x - \mu_1)^T$$

$$S_b = (\mu_0 - \mu_1)(\mu_0 - \mu_1)^T$$

所以定义目标函数为：

$$J = \frac{w^T S_b w}{w^T S_w w}$$

拉格朗日乘子法求解可得到：

$$S_w^{-1} S_b w = \lambda w$$

其中 w 是矩阵 $S_w^{-1} S_b$ 的特征向量。进一步的使用构造性求解方法可得：

$$w = S_w^{-1}(\mu_0 - \mu_1)$$

3. 流形学习

非线性降维方法，可以使用欧氏距离来描述局部区域，但是在全局上欧氏距离不成立。降维之后的数据依然保持着降维之前的数据的一些关系。高维空间相似的数据点，映射到低维空间距离也是相似的。

PPT 中提到的三种流形学习算法：

LLE：高维空间中的样本的线性重构关系在低维空间中均得以保持。

Isomap：对于数据集，通过最近邻等方式构造一个数据图，然后计算任意两个点之间的最短路径 (即测地距离)，对于所有任意两个点对，期望在低维空间中保持其测地距离。

LE：给定数据集，通过最近邻方式构造数据图，在每一个局部区域，计算点与点之间的亲和度 (相似度)，期望点对的亲和度在低维空间中也可以得到保持。

4. 特征选择与分类器的结合程度：

(1) 过滤式特征选择方法：特征选择和学习是独立的，主要是装袋法 (背包问题)：单独特征选择法，顺序前进特征选择法，顺序后退特征选择发，增 l 减 r 特征选择法，启发式选择方法 relief 方法。

过滤式方法先对数据集进行特征选择，然后再训练学习器，特征选择过程与学习单独进行，特征选择评价盘踞简介反应分类性能。特征选择与后续学习器无关，由于是启发式特征选择方法，无法获得最优子集，与包裹式特征选择方法相比，计算量降低了许多。

在深度学习之前的绝大多数模型，都是特征选择然后特征提取再加上分类器进行从处理。课上讲的 PCA 和 LDA 方法其实也可以算，这一类方法可以有效的将高维复杂数据进行降维，方便后面的分类器输入，但是对于高纬度的数据，比如说图像，特征提取就比较困难，会出现维数爆炸的现象。

(2) 包裹式特征选择方法：特征选择依赖学习，特征选择与分类性能相结合，特征评价判据为分类器的性能。对给定的分类方法，选择最有利与分类性能的特征子集。包裹式特征选择方法要求分类器能够处理高维特征向量，在特征维数很高、样本个数较少的时候，分类器依然可以去的较好的效果。

主要方法有：直观方法，直接给定特征子集，然后训练分类器模型，尝试不同的特征组合，寻找最优的组合，需要的计算量大。递归策略，利用所有的特征进行分类器训练，然后考察各个特征在分类器中的贡献，逐步剔除贡献小的特征，方法包括递归支持向量机，支持向量机递归特征剔除，Adaboost。

(3) 嵌入式特征选择方法：特征选择和学习同时进行，方法包括：基于 L_1 范数的特征选择，将分类器学习与特征选择融为一体，分类器训练过程自动完成了特征选择。

现在的深度学习都是端到端的模型，提取特征的同时学习模型参数，效率更高，也更加准确，但是训练需要的数据集更大，小样本上收敛性不好

5. 最优特征提取

最优特征选择：

最优方法之一：穷举法，但是需要遍历所有子集，复杂度过高

最优方法之二：分支定界方法：将所有可能的特征组合以树的形式进行表示，然后采用分支定界方法对树进行搜索，是的所搜过程尽早达到最优解，从而减低算法的复杂度。

第二部分：编程题

编程实现 1：PCA+KNN：即首先 PCA 进行降维，然后采用最近邻分类器（1 近邻分类器）作为分类器进行分类。

编程实现 2：LDA+KNN，即首先 LDA 进行降维，然后采用最近邻分类器（1 近邻分类器）作为分类器进行分类。

任务：采用 80%作样本作训练集，20%样本做测试集，报告降至不同维数时的分类性能。

(a) 以测试集的准确率为准，即分类正确的数量比上测试集的总数，由于程序中是 20% 的样本作为测试集，80% 的样本作为测试集，这里数据集总共 40 个类别，每一类别有 10 个样本，所以总共 400 个样本。所以这里每一类别挑选 2 个样本作为测试集，其他的作为训练集，素有测试集总共是 80 个样本，训练集总共是 320 个样本。

由于对于 PCA 来说，降维的维度要小于样本数量，对于 LDA 来说，根据老师上课讲的要证明的那个式子 ($rank(S_b) < d - 1$) 可以推导出来，LDA 降维的结果必须小于类别数，所以这里对于 ORLdata 的数据来说，PCA 选择以 5 为间隔，从 5 到 395 维的降维，但是对于 LDA 来说，类别数只有 40 个，从 1 开

始以 1 为间隔，这里只降维到 39 维，最后的降维的结果如下图 1 所示，其中蓝线是 PCA 降维之后的情况，橙色 LDA 降维之后的情况，从中可以看出，对于维数比较大的数据集来说，保留的维数越高，保存的信息越多，分类的情况越好，但是随着维数上升到一定程度之后，随后的分类准确率不会再提升了：

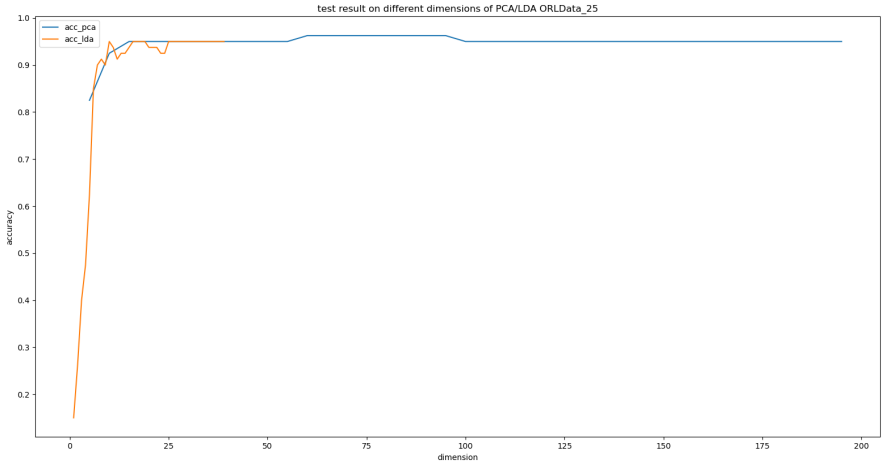

图 1: ORLData 数据集在不同降维情况下的准确率情况

(b) 以测试集的准确率为准，即分类正确的数量比上测试集的总数，由于程序中是 20% 的样本作为测试集，80% 的样本作为测试集，这里数据集总共是 4 个类别，选择其 20% 为数据集，20% 为测试集，同样的，对于 vehicle 来说，LDA 从 1 开始，1 为间隔，最高只能降维到 3 维，PCA 从 5 开始，5 为间隔，最高可以降维到了 15 维，为了对比，这里 PCA 专门加了 1 维的情况 (实际意义不大，这里是为了展示)，最后测试的结果如下图 2 所示。最后降维的结果如下图 2 所示，其中蓝线是 PCA 降维之后的情况，橙色 LDA 降维之后的情况，从中可以看出，对于数据维数比较低的数据集，保留的维数越高，保存的信息就越多，分类的准确率会随着维数呈现正相关。

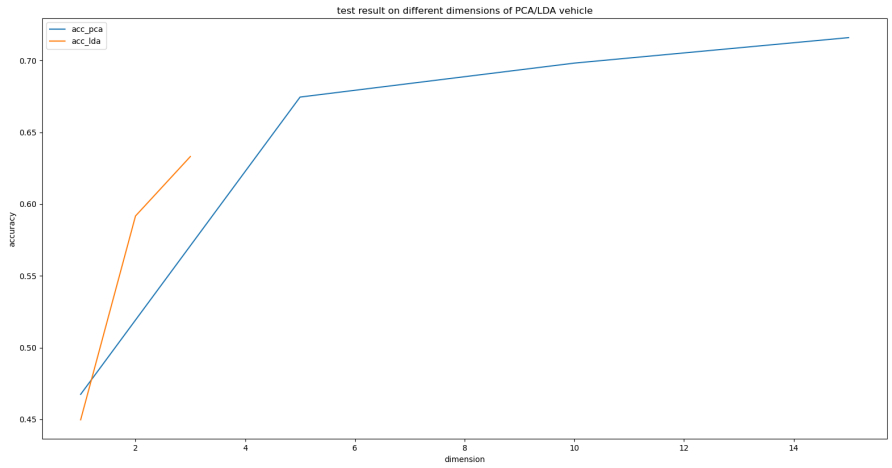

图 2: vehicle 数据集在不同降维情况下的准确率情况