机器学习与深度学习习题集答案-2

本文是机器学习和深度学习习题集答案的第 2 部分，也是《机器学习-原理、算法与应用》一书的配套产品。此习题集可用于高校的机器学习与深度学习教学，以及在职人员面试准备时使用。

第 8 章 线性判别分析

1.解释 LDA 的原理。

LDA 是有监督的降维算法，它将数据向量向最大化类间差异，最小化类内差异的方向投影。

2.推导两个类和二维时 LDA 的投影矩阵计算公式。

假设有 n 个样本，它们的特征向量为 $\mathbf{x}_i$，这些样本属于两个类。属于类 C_1 的样本集为 D_1，有 n_1 个样本；属于类 C_2 的样本集为 D_2，有 n_2 个样本。投影向量为 $\mathbf{w}$，所有向量对该向量做投影可以得到一个标量

$$y = \mathbf{w}^{\mathrm{T}} \mathbf{x}$$

投影运算产生 n 个标量，分属于与 C_1 和 C_2 相对应的两个集合 Y_1 和 Y_2。

类间差异用投影后两个类的中心的距离而定义

$$\left| \widetilde{m}_1 - \widetilde{m}_2 \right| = \left| \mathbf{w}^{\mathrm{T}} \left(\mathbf{m}_1 - \mathbf{m}_2 \right) \right|$$

其中投影之前每类样本的均值为

$$\mathbf{m}_i = \frac{1}{n_i} \sum_{\mathbf{x} \in D_i} \mathbf{x}$$

类内差异由每个类的类内散布之和而定义，类内散布定义为

$$\tilde{s}_i^2 = \sum_{y \in Y_i} \left(y - \widetilde{m}_i \right)^2$$

LDA 寻找投影方向的目标是使得类间差异与类内差异的比值最大化

$$L(\mathbf{w}) = \frac{\left(\widetilde{m}_1 - \widetilde{m}_2 \right)^2}{\tilde{s}_1^2 + \tilde{s}_2^2}$$

定义类内散布矩阵为

$$\mathbf{S}_i = \sum_{\mathbf{x}\in D_i} \left(\mathbf{x}-\mathbf{m}_i\right)\left(\mathbf{x}-\mathbf{m}_i\right)^{\mathrm{T}}$$

总类内散布矩阵为：

$$\mathbf{S}_W = \mathbf{S}_1 + \mathbf{S}_2$$

各个类的类内散布可以写成

$$\begin{aligned}\tilde{s}_i^2 &= \sum_{\mathbf{x}\in D_i} \left(\mathbf{w}^{\mathrm{T}}\mathbf{x}-\mathbf{w}^{\mathrm{T}}\mathbf{m}_i\right)^2 \\ &= \sum_{\mathbf{x}\in D_i} \mathbf{w}^{\mathrm{T}}\left(\mathbf{x}-\mathbf{m}_i\right)\left(\mathbf{x}-\mathbf{m}_i\right)^{\mathrm{T}}\mathbf{w} \\ &= \mathbf{w}^{\mathrm{T}}\mathbf{S}_i\mathbf{w}\end{aligned}$$

各类的散布之和可以写成

$$\tilde{s}_1^2 + \tilde{s}_2^2 = \mathbf{w}^{\mathrm{T}}\mathbf{S}_W\mathbf{w}$$

各类样本的均值之差可以写成

$$\left(\tilde{m}_1-\tilde{m}_2\right)^2 = \left(\mathbf{w}^{\mathrm{T}}\left(\mathbf{m}_1-\mathbf{m}_2\right)\right)^2 = \mathbf{w}^{\mathrm{T}}\left(\mathbf{m}_1-\mathbf{m}_2\right)\left(\mathbf{m}_1-\mathbf{m}_2\right)^{\mathrm{T}}\mathbf{w}$$

如果定义类间散布矩阵

$$\mathbf{S}_B = \left(\mathbf{m}_1-\mathbf{m}_2\right)\left(\mathbf{m}_1-\mathbf{m}_2\right)^{\mathrm{T}}$$

则类间差异可以写成

$$\left(\tilde{m}_1-\tilde{m}_2\right)^2 = \mathbf{w}^{\mathrm{T}}\mathbf{S}_B\mathbf{w}$$

要优化的目标函数可以写为

$$L\left(\mathbf{w}\right) = \frac{\mathbf{w}^{\mathrm{T}}\mathbf{S}_B\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{S}_W\mathbf{w}}$$

这个最优化问题的解不唯一，如果$\mathbf{w}^*$是最优解，将它乘上一个非零系数k之后，$k\mathbf{w}^*$还是最优解。可以加上一个约束条件消掉冗余，同时简化问题。为$\mathbf{w}$加上如下约束

$$\mathbf{w}^{\mathrm{T}}\mathbf{S}_W\mathbf{w} = 1$$

上面的最优化问题转化为带等式约束的极大值问题：

$$\begin{gathered}\max\ \mathbf{w}^{\mathrm{T}}\mathbf{S}_B\mathbf{w} \\ \mathbf{w}^{\mathrm{T}}\mathbf{S}_W\mathbf{w} = 1\end{gathered}$$

用拉格朗日乘数法求解。构造拉格朗日乘子函数：

$$L\left(\mathbf{w},\lambda\right) = \mathbf{w}^{\mathrm{T}}\mathbf{S}_B\mathbf{w} + \lambda\left(\mathbf{w}^{\mathrm{T}}\mathbf{S}_W\mathbf{w}-1\right)$$

对$\mathbf{w}$求梯度并令梯度为$\mathbf{0}$，可以得到：

$$\mathbf{S}_B\mathbf{w} + \lambda\mathbf{S}_W\mathbf{w} = \mathbf{0}$$

即

$$\mathbf{S}_B\mathbf{w} = \lambda \mathbf{S}_W\mathbf{w}$$

如果$\mathbf{S}_W$可逆，上式两边左乘$\mathbf{S}_W^{-1}$后可以得到

$$\mathbf{S}_W^{-1}\mathbf{S}_B\mathbf{w} = \lambda \mathbf{w}$$

即λ是矩阵$\mathbf{S}_W^{-1}\mathbf{S}_B$的特征值，$\mathbf{w}$为对应的特征向量。假设$\lambda$和$\mathbf{w}$是上面特征值问题的解，代入目标函数可以得到

$$\frac{\mathbf{w}^{\mathrm{T}}\mathbf{S}_B\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{S}_W\mathbf{w}} = \frac{\mathbf{w}^{\mathrm{T}}\left(\lambda \mathbf{S}_W\mathbf{w}\right)}{\mathbf{w}^{\mathrm{T}}\mathbf{S}_W\mathbf{w}} = \lambda$$

这里的目标是要让该比值最大化，因此最大的特征值λ及其对应的特征向量是最优解。

3.解释 LDA 降维算法的流程。

首先计算投影矩阵，流程为：

1.计算各个类的均值向量与总均值向量。

2.计算类间散布矩阵$\mathbf{S}_B$，类内散布矩阵$\mathbf{S}_W$。

3.计算矩阵乘法$\mathbf{S}_W^{-1}\mathbf{S}_B$。

4.对$\mathbf{S}_W^{-1}\mathbf{S}_B$进行特征值分解，得到特征值和特征向量。

5.对特征值从大到小排序，截取部分特征值和特征向量构成投影矩阵。

接下来进行降维，流程为：

1.将样本减掉均值向量。

2.左乘投影矩阵，得到降维后的向量。

4.解释 LDA 重构算法的流程。

1.输入向量左乘投影矩阵的转置矩阵。

2.加上均值向量，得到重构后的结果。

5.LDA 是有监督学习还是无监督学习？

LDA 利用了每个样本的类别标签，是有监督学习算法。

第 9 章 人工神经网络

1.神经网络为什么需要激活函数？

保证神经网络的映射是非线性的，如果不使用激活函数，无论神经网络有多少层，其所

表示的复合函数还是一个线性函数。

2.推导 sigmoid 函数的导数计算公式。

sigmoid 函数定义为

$$f(x)=\frac{1}{1+e^{-x}}$$

其导数为

$$\begin{aligned}f'(x)&=-\frac{1}{\left(1+e^{-x}\right)^{2}}\left(1+e^{-x}\right)'\\&=-\frac{1}{\left(1+e^{-x}\right)^{2}}\left(e^{-x}\right)'\\&=-\frac{1}{\left(1+e^{-x}\right)^{2}}\left(e^{-x}\right)(-x)'\\&=\frac{e^{-x}}{\left(1+e^{-x}\right)^{2}}\end{aligned}$$

而

$$\frac{e^{-x}}{\left(1+e^{-x}\right)^{2}}=\frac{1}{1+e^{-x}}\frac{e^{-x}}{1+e^{-x}}=\frac{1}{1+e^{-x}}\left(1-\frac{1}{1+e^{-x}}\right)$$

因此

$$f'(x)=f(x)\left(1-f(x)\right)$$

3.激活函数需要满足什么数学条件？

激活函数需要满足：

1.非线性。保证神经网络实现的映射是非线性的。

2.几乎处处可导。保证可以用梯度下降法等基于导数的算法进行训练。

3.单调递增或者递减。保证满足万能逼近定理的要求，且目标函数有较好的特性。

4.为什么激活函数只要求几乎处处可导而不需要在所有点处可导？

如果将激活函数的输入值看作连续型随机变量，如果激活函数几乎处处可导，则在训练时激活函数的输入值落在不可导点处的概率为 0。

5.什么是梯度消失问题，为什么会出现梯度消失问题？

在用反向传播算法计算误差项时每一层都要乘以本层激活函数的导数

$$\boldsymbol{\delta}^{(l)}=\left(\mathbf{W}^{(l+1)}\right)^{\mathrm{T}}\boldsymbol{\delta}^{(l+1)}\odot f'\left(\mathbf{u}^{(l)}\right)$$

如果激活函数导数的绝对值小于 1，多次连乘之后误差项很快会衰减到接近于 0，参数的梯度值由误差项计算得到，从而导致前面层的权重梯度接近于 0，参数无法有效的更新，

称为梯度消失问题。

6.如果特征向量中有类别型特征，使用神经网络时应该如何处理？
通常采用 one hot 编码，而不直接将类别编号整数值作为神经网络的输入。

7.对于多分类问题，神经网络的输出值应该如何设计？
类别标签通常采用 one hot 编码，输出层的神经元个数等于类别数。

8.神经网络参数的初始值如何设定？
一般用随机数进行初始化。

9.如果采用欧氏距离损失函数，推导输出层的梯度值。推导隐含层参数梯度的计算公式。
使用均方误差，则优化的目标为：

$$L(W)=\frac{1}{2m}\sum_{i=1}^{m}\left\|h(\mathbf{x}_i)-\mathbf{y}_i\right\|^2$$

下面对单个样本的损失进行推导。神经网络每一层的变换为

$$\mathbf{u}^{(l)}=\mathbf{W}^{(l)}\mathbf{x}^{(l-1)}+\mathbf{b}^{(l)}$$

$$\mathbf{x}^{(l)}=f\left(\mathbf{u}^{(l)}\right)$$

对单个样本$(\mathbf{x}_i,\mathbf{y}_i)$的损失函数为

$$L(W,\mathbf{x}_i,\mathbf{y}_i)=\frac{1}{2}\left\|h(\mathbf{x}_i)-\mathbf{y}_i\right\|^2$$

如果第l层是输出层，损失函数对输出层的临时变量的梯度为

$$\nabla_{\mathbf{u}^{(l)}}L=\left(\nabla_{\mathbf{x}^{(l)}}L\right)\odot f'\left(\mathbf{u}^{(l)}\right)=\left(\mathbf{x}^{(l)}-\mathbf{y}\right)\odot f'\left(\mathbf{u}^{(l)}\right)$$

损失函数对输出层权重的梯度为

$$\nabla_{\mathbf{W}^{(l)}}L=\left(\mathbf{x}^{(l)}-\mathbf{y}\right)\odot f'\left(\mathbf{u}^{(l)}\right)\left(\mathbf{x}^{(l-1)}\right)^{\mathrm{T}}$$

损失函数对偏置项的梯度为

$$\nabla_{\mathbf{b}^{(l)}}L=\left(\mathbf{x}^{(l)}-\mathbf{y}\right)\odot f'\left(\mathbf{u}^{(l)}\right)$$

如果第l层是隐含层，则有

$$\mathbf{u}^{(l+1)}=\mathbf{W}^{(l+1)}\mathbf{x}^{(l)}+\mathbf{b}^{(l+1)}=\mathbf{W}^{(l+1)}f\left(\mathbf{u}^{(l)}\right)+\mathbf{b}^{(l+1)}$$

假设梯度$\nabla_{\mathbf{u}^{(l+1)}}L$已经求出，有

$$\nabla_{\mathbf{u}^{(l)}}L=\left(\nabla_{\mathbf{x}^{(l)}}L\right)\odot f'\left(\mathbf{u}^{(l)}\right)=\left(\left(\mathbf{W}^{(l+1)}\right)^{\mathrm{T}}\nabla_{\mathbf{u}^{(l+1)}}L\right)\odot f'\left(\mathbf{u}^{(l)}\right)$$

通过 $\nabla_{\mathbf{u}^{(l+1)}}L$ 可以递推地计算出 $\nabla_{\mathbf{u}^{(l)}}L$，递推的终点是输出层，输出层的梯度值之前已经算出。根据 $\nabla_{\mathbf{u}^{(l)}}L$ 可以计算出 $\nabla_{\mathbf{W}^{(l)}}L$ 和 $\nabla_{\mathbf{b}^{(l)}}L$，因此可以计算出任意层权重与偏置的梯度值。

定义误差项为损失函数对临时变量 $\mathbf{u}$ 的梯度

$$\boldsymbol{\delta}^{(l)} = \nabla_{\mathbf{u}^{(l)}}L = \begin{cases} \left(\mathbf{x}^{(l)} - \mathbf{y}\right) \odot f'\left(\mathbf{u}^{(l)}\right) & l = n_l \\ \left(\mathbf{W}^{(l+1)}\right)^{\mathrm{T}} \left(\boldsymbol{\delta}^{(l+1)}\right) \odot f'\left(\mathbf{u}^{(l)}\right) & l \neq n_l \end{cases}$$

从输出层开始，利用上面的递推公式可以计算出每一层的误差项。根据每一层的误差项可以计算出损失函数对该层权重矩阵以及偏置项的梯度。对权重矩阵的梯度为

$$\nabla_{\mathbf{W}^{(l)}}L = \boldsymbol{\delta}^{(l)} \left(\mathbf{x}^{(l-1)}\right)^{\mathrm{T}}$$

对偏置项的梯度为

$$\nabla_{\mathbf{b}^{(l)}}L = \boldsymbol{\delta}^{(l)}$$

计算出损失函数对每一层参数的梯度值之后，可以用梯度下降法进行参数更新。

10.如果采用 softmax+交叉熵的方案，推导损失函数对 softmax 输入变量的梯度值。

softmax 变换为

$$y_i^* = \frac{\exp\left(x_i\right)}{\sum_{k=1}^{K} \exp\left(x_k\right)}$$

其中 $\mathbf{x}$ 是本层的输入向量，$\mathbf{y}^*$ 是概率估计向量，$\mathbf{y}$ 是样本的真实标签值。交叉熵损失函数定义为

$$L = -\mathbf{y}^{\mathrm{T}} \log \mathbf{y}^*$$

样本的类别标签中只有一个分量为 1，其他都是 0，这在第 11.4 节中已经介绍过。假设标签向量的第 j 个分量为 1，该函数的导数为：

$$\frac{\partial L}{\partial x_i} = -\frac{1}{y_j^*} \frac{\partial y_j^*}{\partial x_i}$$

下面分两种情况讨论。如果 $i = j$ 即 $y_i = 1$，有：

$$\frac{\partial L}{\partial x_i} = -\frac{1}{y_i^*} \times \frac{\exp(x_i)\sum_{k=1}^{K}\exp(x_k) - \exp(x_i)\exp(x_i)}{\left(\sum_{k=1}^{K}\exp(x_k)\right)^2}$$

$$= -\frac{\sum_{k=1}^{K}\exp(x_k)}{\exp(x_i)} \times \frac{\exp(x_i)\left(\sum_{k=1}^{K}\exp(x_k) - \exp(x_i)\right)}{\left(\sum_{k=1}^{K}\exp(x_k)\right)^2}$$

$$= -\frac{\sum_{j=k}^{K}\exp(x_k) - \exp(x_i)}{\sum_{k=1}^{K}\exp(x_k)} = -\left(1 - y_i^*\right) = y_i^* - y_i$$

否则有：

$$\frac{\partial L}{\partial x_i} = \frac{1}{y_j^*} \times \frac{\exp(x_j)\exp(x_i)}{\left(\sum_{k=1}^{K}\exp(x_k)\right)^2} = \frac{\sum_{k=1}^{K}\exp(x_k)}{\exp(x_j)} \times \frac{\exp(x_j)\exp(x_i)}{\left(\sum_{k=1}^{K}\exp(x_k)\right)^2} = \frac{\exp(x_i)}{\sum_{k=1}^{K}\exp(x_k)} = y_i^*$$

此时 $y_i = 0$。将两种情况合并起来写成向量形式为：

$$\nabla_{\mathbf{x}} L = \mathbf{y}^* - \mathbf{y}$$

11.解释动量项的原理。

动量项累积了之前的权重更新值，加上此项之后的参数更新公式为

$$W_{t+1} = W_t + V_{t+1}$$

其中 V_{t+1} 是动量项，计算公式为

$$V_{t+1} = -\alpha \nabla_W L(W_t) + \mu V_t$$

它是上一时刻的动量项与本次梯度值的加权平均值，其中 α 是学习率，μ 是动量项系数。如果按照时间 t 进行展开，则第 t 次迭代时使用了从 1 到 t 次迭代时的所有梯度值，且老的梯度值安 μ^t 的系数指数级衰减。动量项是为了加快梯度下降法的收敛，它使用历史信息对当前梯度值进行修正，以抵消在病态条件问题上的来回震荡。

12.列举神经网络的正则化技术。

典型的正则化技术包括：

L1，L2 或者谱正则化

Dropout

提前终止

13.推导 ReLU 函数导数计算公式。

ReLU 函数定义为

$$f(x)=\max(0,x)$$

它在 0 点处不可导，如果忽略该点，导数为

$$\mathrm{ReLU}'(x)=\begin{cases}1 & x>0\\ 0 & x\le 0\end{cases}$$

14.神经网络训练时的目标函数是否为凸函数？

一般情况下不是凸函数。因此面临局部极小值和鞍点问题。

第 10 章 支持向量机

1.推导线性可分时 SVM 的原问题：

$$\min \frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}$$

$$y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)\ge 1$$

假设训练样本集有 l 个样本，特征向量 x_i 是 n 维向量，类别标签 y_i 取值为+1 或者-1，分别对应正样本和负样本。支持向量机预测函数的超平面方程为

$$\mathrm{w}^{\mathrm{T}}\mathrm{x}+b=0$$

首先要保证每个样本都被正确分类。对于正样本有

$$\mathrm{w}^{\mathrm{T}}\mathrm{x}+b\ge 0$$

对于负样本有

$$\mathrm{w}^{\mathrm{T}}\mathrm{x}+b<0$$

由于正样本的的类别标签为+1，负样本的类别标签为-1，可以统一写成

$$y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)\ge 0$$

第二个要求是超平面离两类样本的距离要尽可能大。根据点到平面的距离公式，每个样本离分类超平面的距离为

$$d=\frac{\left|\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right|}{\|\mathrm{w}\|}$$

上面的超平面方程有冗余，将方程两边都乘以不等于 0 的常数，还是同一个超平面，利用这个特点可以简化求解的问题。对 w 和 b 加上如下约束

$$\min_{\mathrm{x}_i}\left|\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right|=1$$

可以消掉这个冗余。这样对分类超平面的约束变成

$$y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)\geq 1$$

分类超平面与两类样本之间的间隔为

$$\begin{aligned}
d(\mathrm{w},b)&=\min_{\mathrm{x}_i,y_i=-1}d\left(\mathrm{w},b;\mathrm{x}_i\right)+\min_{\mathrm{x}_i,y_i=1}d\left(\mathrm{w},b;\mathrm{x}_i\right)\\
&=\min_{\mathrm{x}_i,y_i=-1}\frac{\left|\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right|}{\|\mathrm{w}\|}+\min_{\mathrm{x}_i,y_i=1}\frac{\left|\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right|}{\|\mathrm{w}\|}\\
&=\frac{1}{\|\mathrm{w}\|}\left(\min_{\mathrm{x}_i,y_i=-1}\left|\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right|+\min_{\mathrm{x}_i,y_i=1}\left|\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right|\right)\\
&=\frac{2}{\|\mathrm{w}\|}
\end{aligned}$$

目标是使得这个间隔最大化，这等价于最小化下面的目标函数

$$\frac{1}{2}\|\mathrm{w}\|^2$$

加上前面定义的约束条件之后，求解的优化问题可以写成

$$\min\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}$$
$$y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)\geq 1$$

2.证明线性可分时 SVM 的原问题是凸优化问题且 Slater 条件成立

$$\min\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}$$
$$y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)\geq 1$$

目标函数的 Hessian 矩阵是 n 阶单位矩阵，是严格正定矩阵，因此目标函数是严格凸函数。可行域是由线性不等式围成的区域，是一个凸集。因此这个优化问题是一个凸优化问题。

由于假设数据是线性可分的，因此一定存在 w 和 b 使得不等式约束严格满足。如果 w 和 b 是一个可行解

$$y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)\geq 1$$

则 $2\mathrm{w}$ 和 $2b$ 也是可行解，且是严格可行的

$$y_i\left(2\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+2b\right)\geq 2>1$$

Slater 条件成立。

3.推导线性可分时 SVM 的对偶问题

$$\min_{\alpha}\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j \mathrm{x}_i^{\mathrm{T}}\mathrm{x}_j-\sum_{i=1}^{l}\alpha_i$$

$$\alpha_i\geq 0, i=1,...,l$$

$$\sum_{i=1}^{l}\alpha_i y_i=0$$

为原问题造拉格朗日函数

$$L\left(\mathrm{w},b,\alpha\right)=\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}-\sum_{i=1}^{l}\alpha_i\left(y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)-1\right)$$

约束条件为$\alpha_i\geq 0$。

先固定住拉格朗日乘子α，调整 w 和b，使得拉格朗日函数取极小值。把α看成常数，对 w 和b求偏导数并令它们为 0，得到如下方程组

$$\frac{\partial L}{\partial b}=0$$

$$\nabla_{\mathrm{w}}L=0$$

从而解得

$$\sum_{i=1}^{l}\alpha_i y_i=0$$

$$\mathrm{w}=\sum_{i=1}^{l}\alpha_i y_i \mathrm{x}_i$$

将上面两个解代入拉格朗日函数消掉 w 和b

$$
\begin{aligned}
&\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}-\sum_{i=1}^{l}\alpha_i\left(y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)-1\right)\\
&=\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}-\sum_{i=1}^{l}\left(\alpha_i y_i \mathrm{w}^{\mathrm{T}}\mathrm{x}_i+\alpha_i y_i b-\alpha_i\right)\\
&=\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}-\sum_{i=1}^{l}\alpha_i y_i \mathrm{w}^{\mathrm{T}}\mathrm{x}_i-\sum_{i=1}^{l}\alpha_i y_i b+\sum_{i=1}^{l}\alpha_i\\
&=\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}-\mathrm{w}^{\mathrm{T}}\sum_{i=1}^{l}\alpha_i y_i \mathrm{x}_i-b\sum_{i=1}^{l}\alpha_i y_i+\sum_{i=1}^{l}\alpha_i\\
&=\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}-\mathrm{w}^{\mathrm{T}}\mathrm{w}+\sum_{i=1}^{l}\alpha_i\\
&=-\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}+\sum_{i=1}^{l}\alpha_i\\
&=-\frac{1}{2}\left(\sum_{i=1}^{l}\alpha_i y_i \mathrm{x}_i\right)^{\mathrm{T}}\left(\sum_{j=1}^{l}\alpha_j y_j \mathrm{x}_j\right)+\sum_{i=1}^{l}\alpha_i
\end{aligned}
$$

接下来调整乘子变量α，使得目标函数取极大值

$$\max_{\alpha}-\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j \mathrm{x}_i^{\mathrm{T}}\mathrm{x}_j+\sum_{i=1}^{l}\alpha_i$$

这等价于最小化下面的函数

$$\min_{\alpha}\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j \mathrm{x}_i^{\mathrm{T}}\mathrm{x}_j-\sum_{i=1}^{l}\alpha_i$$

约束条件为

$$
\begin{aligned}
&\alpha_i \ge 0, i=1,...,l\\
&\sum_{i=1}^{l}\alpha_i y_i=0
\end{aligned}
$$

4.证明加入松弛变量和惩罚因子之后，SVM 的原问题是凸优化问题且 Slater 条件成立：

$$
\begin{aligned}
&\min\ \frac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}+C\sum_{i=1}^{l}\xi_i\\
&y_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i+b)\ge 1-\xi_i\\
&\xi_i\ge 0, i=1,...,l
\end{aligned}
$$

目标函数的前半部分是凸函数，后半部分是线性函数显然也是凸函数，两个凸函数的非负线性组合还是凸函数。上面优化问题的不等式约束都是线性约束，构成的可行域显然是凸集。因此该优化问题是凸优化问题。

如果令$\mathbf{w}=0$，$b=0$，$\xi_i=2$，这是一组可行解，且有

$$y_i(\mathbf{w}^{\mathrm{T}}\mathbf{x}_i+b)=0>1-\xi_i=1-2=-1$$

不等式条件严格满足，因此上述问题满足 Slater 条件。

5.推导线性不可分时 SVM 的对偶问题：

$$\min_{\alpha} \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j \mathrm{x}_i^{\mathrm{T}}\mathrm{x}_j - \sum_{i=1}^{l}\alpha_i$$

$$0 \le \alpha_i \le C$$

$$\sum_{j=1}^{l}\alpha_j y_j = 0$$

构造拉格朗日函数：

$$L(\mathrm{w},b,\alpha,\xi,\beta) = \frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w} + C\sum_{i=1}^{l}\xi_i - \sum_{i=1}^{l}\alpha_i\left(y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i + b\right) - 1 + \xi_i\right) - \sum_{i=1}^{l}\beta_i\xi_i$$

首先固定住乘子变量α和β，对w,b,ξ求偏导数并令它们为 0，得到如下方程组

$$\frac{\partial L}{\partial b} = 0$$

$$\nabla_{\xi} L = 0$$

$$\nabla_{\mathrm{w}} L = 0$$

解得

$$\sum_{i=1}^{l}\alpha_i y_i = 0$$

$$\alpha_i + \beta_i = C$$

$$\mathrm{w} = \sum_{i=1}^{l}\alpha_i y_i \mathrm{x}_i$$

将上面的解代入拉格朗日函数中，得到关于α和β的函数

$$L(\mathrm{w},b,\alpha,\xi,\beta)=\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}+C\sum_{i=1}^{l}\xi_i-\sum_{i=1}^{l}\alpha_i\left(y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)-1+\xi_i\right)-\sum_{i=1}^{l}\beta_i\xi_i$$
$$=\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}+C\sum_{i=1}^{l}\xi_i-\sum_{i=1}^{l}\beta_i\xi_i-\sum_{i=1}^{l}\alpha_i\xi_i-\sum_{i=1}^{l}\alpha_i\left(y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)-1\right)$$
$$=\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}+\sum_{i=1}^{l}\left(C-\alpha_i-\beta_i\right)\xi_i-\sum_{i=1}^{l}\left(\alpha_i y_i\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+\alpha_i y_i b-\alpha_i\right)$$
$$=\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}-\sum_{i=1}^{l}\alpha_i y_i\mathrm{w}^{\mathrm{T}}\mathrm{x}_i-\sum_{i=1}^{l}\alpha_i y_i b+\sum_{i=1}^{l}\alpha_i$$
$$=\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}-\mathrm{w}^{\mathrm{T}}\mathrm{w}+\sum_{i=1}^{l}\alpha_i$$
$$=-\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}+\sum_{i=1}^{l}\alpha_i$$
$$=-\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j\mathrm{x}_i^{\mathrm{T}}\mathrm{x}_j+\sum_{i=1}^{l}\alpha_i$$

接下来调整乘子变量求解如下最大化问题

$$\max_\alpha -\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j\mathrm{x}_i^{\mathrm{T}}\mathrm{x}_j+\sum_{i=1}^{l}\alpha_i$$

由于$\alpha_i+\beta_i=C$并且$\beta_i\geq 0$，因此有$\alpha_i\leq C$。这等价与如下最优化问题

$$\min_\alpha \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j\mathrm{x}_i^{\mathrm{T}}\mathrm{x}_j-\sum_{i=1}^{l}\alpha_i$$
$$0\leq\alpha_i\leq C$$
$$\sum_{j=1}^{l}\alpha_j y_j=0$$

6.证明线性不可分时 SVM 的对偶问题是凸优化问题：

$$\min_\alpha \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j\mathrm{x}_i^{\mathrm{T}}\mathrm{x}_j-\sum_{i=1}^{l}\alpha_i$$
$$0\leq\alpha_i\leq C$$
$$\sum_{j=1}^{l}\alpha_j y_j=0$$

为了简化表述，定义矩阵Q，其元素为

$$Q_{ij}=y_i y_j\mathrm{x}_i^{\mathrm{T}}\mathrm{x}_j$$

对偶问题可以写成矩阵和向量形式

$$
\begin{aligned}
&\min_{\alpha} \frac{1}{2}\alpha^{\mathrm{T}}\mathrm{Q}\alpha-\mathrm{e}^{\mathrm{T}}\alpha \\
&0 \le \alpha_i \le C \\
&\mathrm{y}^{\mathrm{T}}\alpha = 0
\end{aligned}
$$

其中 e 是分量全为 1 的向量，y 是样本的类别标签向量。Q 可以写成一个矩阵和其自身转置的乘积

$$\mathrm{Q}=\mathrm{X}^{\mathrm{T}}\mathrm{X}$$

矩阵 X 为所有样本的特征向量分别乘以该样本的标签值组成的矩阵：

$$\mathrm{X}=\left[y_1\mathrm{x}_1,...,y_l\mathrm{x}_l\right]$$

对于任意非 0 向量 x 有：

$$\mathrm{x}^{\mathrm{T}}\mathrm{Q}\mathrm{x}=\mathrm{x}^{\mathrm{T}}\left(\mathrm{X}^{\mathrm{T}}\mathrm{X}\right)\mathrm{x}=\left(\mathrm{X}\mathrm{x}\right)^{\mathrm{T}}\left(\mathrm{X}\mathrm{x}\right)\ge 0$$

因此矩阵 Q 半正定，它就是目标函数的 Hessian 矩阵，目标函数是凸函数。上面问题的等式和不等式约束条件都是线性的，可行域是凸集，故对偶问题也是凸优化问题。

7.用 KKT 条件证明 SVM 所有样本满足如下条件：

$$
\begin{aligned}
&\alpha_i = 0 \Rightarrow y_i(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b)\ge 1 \\
&0<\alpha_i<C \Rightarrow y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)=1 \\
&\alpha_i = C \Rightarrow y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)\le 1
\end{aligned}
$$

将 KKT 条件其应用于原问题，对于原问题中的两组不等式约束，必须满足

$$
\begin{aligned}
&\alpha_i\left(y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)-1+\xi_i\right)=0, i=1,...,l \\
&\beta_i\xi_i=0, i=1,...,l
\end{aligned}
$$

对于第一个方程，如果 $\alpha_i>0$，则必须有 $y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)-1+\xi_i=0$，即

$$y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)=1-\xi_i$$

而由于 $\xi_i\ge 0$，因此必定有

$$y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)\le 1$$

再看第二种情况。如果 $\alpha_i=0$，则对 $y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)-1+\xi_i$ 的值没有约束。由于有 $\alpha_i+\beta_i=C$ 的约束，因此 $\beta_i=C$；又因为 $\beta_i\xi_i=0$ 的限制，如果 $\beta_i>0$，则必须有 $\xi_i=0$。

由于原问题中有约束条件 $y_i(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b)\ge 1-\xi_i$，而 $\xi_i=0$，因此有

$$y_i(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b)\ge 1$$

对于 $\alpha_i>0$ 的情况，我们又可以细分为 $\alpha_i<C$ 和 $\alpha_i=C$ 。如果 $\alpha_i<C$ ，由于有 $\alpha_i+\beta_i=C$ 的约束，因此有 $\beta_i>0$ ，因为有 $\beta_i\xi_i=0$ 的约束，因此 $\xi_i=0$ ，不等式约束 $y_i(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b)\ge 1-\xi_i$ 变为 $y_i(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b)\ge 1$ 。由于 $0<\alpha_i<C$ 时既要满足 $y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)\le 1$ 又要满足 $y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)\ge 1$，因此有

$$y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)=1$$

将三种情况合并起来，在最优点处，所有的样本都必须要满足下面的条件

$$\begin{aligned}&\alpha_i=0\Rightarrow y_i(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b)\ge 1\\&0<\alpha_i<C\Rightarrow y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)=1\\&\alpha_i=C\Rightarrow y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)\le 1\end{aligned}$$

8.SVM 预测函数中的 b 值如何计算？

根据 KKT 条件，在最优解处有

$$\begin{aligned}&\alpha_i=0\Rightarrow y_i(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b)\ge 1\\&0<\alpha_i<C\Rightarrow y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)=1\\&\alpha_i=C\Rightarrow y_i\left(\mathrm{w}^{\mathrm{T}}\mathrm{x}_i+b\right)\le 1\end{aligned}$$

根据第二种情况可以计算出 b 的值。任意满足第二种情况的训练样本均可以计算出此值。

9.解释核函数的原理，列举常用的核函数。

如果样本线性不可分，可以对特征向量进行映射将它转化到更高维的空间，使得在该空间中线性可分。核映射 ϕ 将特征向量变换到更高维的空间

$$\mathbf{z}=\phi\left(\mathbf{x}\right)$$

在对偶问题中计算的是两个样本向量之间的内积，映射后的向量在对偶问题中为

$$\mathbf{z}_i^{\mathrm{T}}\mathbf{z}_j=\phi(\mathbf{x}_i)^{\mathrm{T}}\phi(\mathbf{x}_j)$$

直接计算这个映射效率太低，而且不容易构造映射函数。如果映射函数选取得当，存在

函数 K，使得下面等式成立

$$K\left(\mathbf{x}_i,\mathbf{x}_j\right)=\phi(\mathbf{x}_i)^{\mathrm{T}}\phi(\mathbf{x}_j)$$

这样只需先对向量用函数 K 进行计算，这等价于先对向量做核映射然后再做内积，这将能有效的简化问题的求解。

线性核：

$$K(\mathrm{x}_i,\mathrm{x}_j)=\mathrm{x}_i^{\mathrm{T}}\mathrm{x}_j$$

多项式核：

$$K(\mathrm{x}_i,\mathrm{x}_j)=(\gamma\mathrm{x}_i^{\mathrm{T}}\mathrm{x}_j+b)^d$$

径向基函数核/高斯核：

$$K(\mathrm{x}_i,\mathrm{x}_j)=\exp(-\gamma\left\|\mathrm{x}_i-\mathrm{x}_j\right\|^2)$$

sigmoid 核：

$$K(\mathrm{x}_i,\mathrm{x}_j)=\tanh(\gamma\mathrm{x}_i^{\mathrm{T}}\mathrm{x}_j+b)$$

10.什么样的函数可以作为核函数？

一个对称函数 $K\left(\mathrm{x},\mathrm{y}\right)$ 是核函数的条件是对任意的有限个样本的样本集，核矩阵半正定。核矩阵的元素是由样本集中任意两个样本的内积构造的一个数

$$K_{ij}=K\left(\mathrm{x}_i,\mathrm{x}_j\right)$$

11.解释 SMO 算法的原理。

SMO 算法是一种分治法，每次挑选出两个变量进行优化，这个子问题可以得到解析解，而一个带等式和不等式约束的二次函数极值问题。

12.SMO 算法如何挑选子问题的优化变量？

第一个变量的选择方法是在训练样本中选取违反 KKT 条件最严重的那个样本。首先遍历所有满足约束条件 $0<\alpha_i<C$ 的样本点，检查它们是否满足 KKT 条件。如果都满足 KKT 条件，则遍历整个训练样本集，判断它们是否满足 KKT 条件，直到找到一个违反 KKT 条件的变量 α_i。

找到这个变量之后，接下来寻找 α_j，选择的标准是使得 α_j 有足够大的变化，选择使得 $\left|E_i-E_j\right|$ 最大的 α_j。由于 α_i 已经确定，因此 E_i 已知。如果 $E_i>0$，则选择最小的 E_j；否

则选择最大的E_j。其中

$$E_i = u_i - y_i$$

以及

$$u_i = \sum_{j=1}^{l} y_j \alpha_j K\left(\mathrm{x}_i, \mathrm{x}_j\right) + b$$

13.证明 SMO 算法中子问题是凸优化问题。

两个变量的目标函数的 Hessian 为

$$\begin{bmatrix} Q_{ii} & Q_{ij} \\ Q_{ji} & Q_{jj} \end{bmatrix}$$

如果是线性核，这个矩阵也可以写成一个矩阵和它的转置的乘积形式

$$\begin{bmatrix} y_i \mathrm{x}_i^{\mathrm{T}} \\ y_i \mathrm{x}_i^{\mathrm{T}} \end{bmatrix} \left[y_i \mathrm{x}_i, y_i \mathrm{x}_j \right] = \mathrm{A}^{\mathrm{T}} \mathrm{A}$$

矩阵 A 为训练样本特征向量乘上类别标签形成的矩阵。显然这个 Hessian 矩阵是半正定的，因此必定有$\eta \geq 0$。如果是非线性核，因为核函数相当于对两个核映射之后的向量做内积，因此上面的结论同样成立。

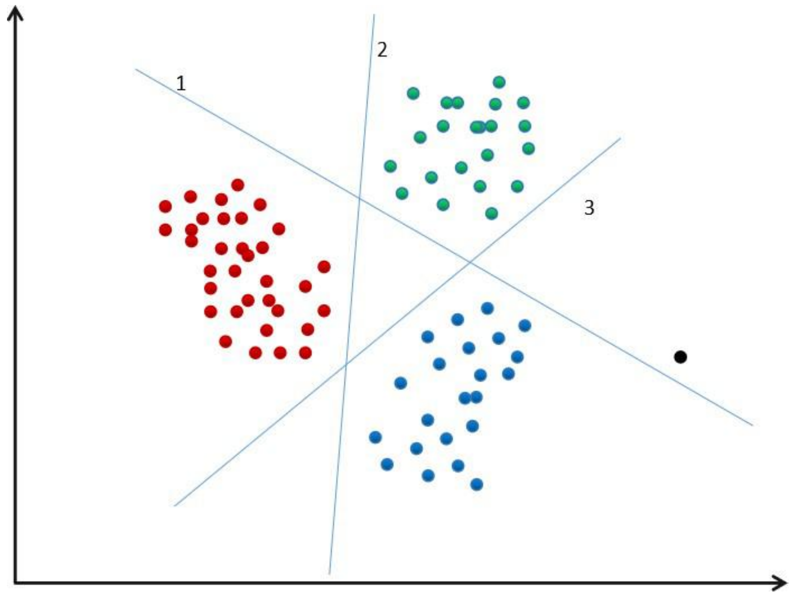
14.证明 SMO 算法能够收敛。

无论本次迭代时α_i和α_j的初始值是多少，通过上面的子问题求解算法得到是在可行域里的最小值，因此每次求解更新这两个变量的值之后，都能保证目标函数值小于或者等于初始值，即函数值下降，所以 SMO 算法能保证收敛。

15.SVM 如何解决多分类问题？

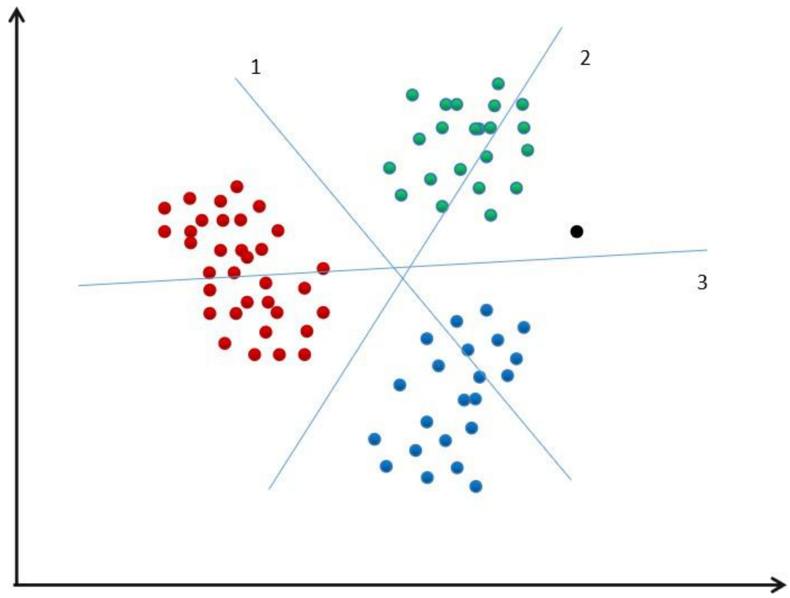
对于多分类问题，可以用二分类器的组合来解决，有以下几种方案：

1 对剩余方案。对于有k个类的分类问题，训练k个二分类器。训练时第个i分类器的正样本是第i类样本，负样本是除第i类之外其他类型的样本，这个分类器的作用是判断样本是否属于第i类。在进行分类时，对于待预测样本，用每个分类器计算输出值，取输出值最大那个作为预测结果。对于 3 个类的分类问题，这种方案如下图所示



其中黑色样本为待预测样本，三条线分别为此方案的 3 个分类器。

1 对 1 方案。如果有 k 个类，训练 C_k^2 个二分类器，即这些类两两组合。训练时将第 i 类作为正样本，其他各个类依次作为负样本，总共有 $k\left(k-1\right)/2$ 种组合。每个分类器的作用是判断样本是属于第 i 类还是第 j 类。对样本进行分类时采用投票的方法，依次用每个二分类器进行预测，如果判定为第 m 类，则 m 类的投票数加 1，得票最多的那个类作为最终的判定结果。对于 3 个类的分类问题，这种方案如下图所示



其中黑色样本为待预测样本，三条线分别为此方案的 3 个分类器。