一、判断题

1.单个神经元为多输入、单输出模型（T）

2.神经元激励函数的目的是实现线性变换（F）

3.BP 和离散型 Hopfield 神经网络权值都是由训练得到（F）

4.单层感知器几乎能实现任意线性可分问题的分类（T）

5.多层感知器几乎能够实现任意线性不可分问题的分类（T）

6.BP 神经网络是由误差反传算法而得名（T）

7.离散型 Hopfield 神经网络吸引子数量可以超过神经元数量（F）

8.离散型 Hopfield 神经网络用外积和设计权值时要求所记忆的模式正交(T）

9.离散型 Hopfield 神经网络只有异步工作模式（F）

10.单个神经元不能做任意简单的分类（F）

11.支持向量机的核心目的是为了寻找最优分类超平面（T）

12.支持向量位于两个支撑超平面之内（F）

13.支持向量的拉格朗日常数 α_i=0（F）

14.支持向量机核函数使解决高维数据分类问题变得复杂（F）

15.SMO 算法是一种支持向量机快速算法，其地位与 FFT 相当（T）

16.将原问题转化为对偶问题是为了降低解决问题的难度（T）

17.软阈值支持向量机能够解决任意线性不可分问题（F）

18.支持向量机分类正例与反例可用 1 与 0 表示（F）

19.高斯核函数是支持向量机常用的核函数之一（T）

20.支持向量机特别适合于小样本分类问题（T）

二、简答题

1. BP 神经网络输入与输出为何要求进行归一化？

输入特征是随意，身高用米 1.5，毫米 1500，单位差距太大，会造成系统权重差别较大（大数吃小数现象）。

输入：多少维向量多少个神经元

输出：多少种输出多少个神经元

输出：转换函数有范围，比如：SIG 函数是（0,1）。输出差距最大，有多少类选择多少神经元。（1,0,0），（0,1,0），（0,0,1）。转化函数（0,1）怎么输出 1000 呢？

2. BP 神经网络存在的主要问题？

1）BP 反传梯度消失现象，梯度是 0 了，梯度没了无法反传

2）局部极小，有多少权值多少个山谷，多次计算蒙到最小值。

3）收敛速度慢，全连接，深了梯度消失，浅了隐层神经元多。

3. 在训练 BP 神经网络时训练精度与泛化能力具有怎样的关系？

精度越高，看到的东西误差越大，过拟合。训练时理论上有一个最佳训练次数。训练目的为了泛化。

4. 离散型 Hopfield 神经网络记忆模式数量与吸引域的关系？

5 个神经元，随便初值都会收敛到 5，模式越多误差越小，神经元越多吸引域越大。模式越少越好，吸引域越大越好。

5. BP 神经网络中隐层的作用是什么？

单层解决不了非线性问题，隐层相当于升维，一个神经元一个分类面，

隐层作用（砍了多刀构成一个多边形）。空间升维后解决异或问题。

隐层变换点位置，将非线性可分问题转换为线性可分问题。

6. SMO 算法中为什么同时选择 α 对进行优化？

原问题多少个样本多少个优化问题，一次优化一个是对的，但变一个约束条件就不对了，带着公式求导得 0，比传统的快了多倍。优化问题变成带公式问题。违背约束条件最重的一对训练时步长最大。

7. SMO 算法收敛速度快的主要原因是什么？

不破坏条件情况下，不破坏公式，收敛速度快。

8. 简述支持向量机中支持向量与非支持向量的作用。

支撑超平面的点有用，其他点没有太大作用。

9.在选择神经网络的深度时，哪些参数需要考虑？

1 神经网络的类型(如 MLP,CNN)

2 输入数据

3 计算能力(硬件和软件能力决定)

4 学习速率

5 映射的输出函数

三、综合题

1. 考虑下面定义的分类问题

$$\left\{X_1=\begin{bmatrix}-1\\1\end{bmatrix}d_1=1\right\}\quad\left\{X_2=\begin{bmatrix}-1\\-1\end{bmatrix}d_2=1\right\}\quad\left\{X_3=\begin{bmatrix}0\\0\end{bmatrix}d_3=-1\right\}\quad\left\{X_4=\begin{bmatrix}1\\0\end{bmatrix}d_4=-1\right\}$$

1）用单神经元感知器能够求解这个问题吗？为什么？

2）设计该单神经元感知器解决分类问题，用全部4个输入向量验证求解结果。

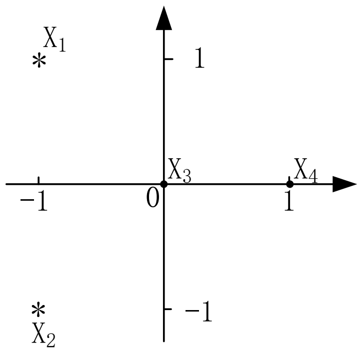
3）用求解结果对下面4个输入向量分类。

$$X_5=\begin{bmatrix}-2\\0\end{bmatrix}\quad X_6=\begin{bmatrix}1\\1\end{bmatrix}\quad X_7=\begin{bmatrix}0\\1\end{bmatrix}\quad X_8=\begin{bmatrix}-1\\-2\end{bmatrix}$$

4）上述输入向量中哪些向量的分类与权值和阈值无关？哪些向量的分类依赖于权值和阈值的选择？

（给定初值 $w_{11}=0.1, w_{12}=0.1$，给定函数为双极阈值函数，给定学习率 $\eta=0.4$，给定阈值 b=1）

答：1）X_1，X_2 与 X_3，X_4 两类样本之间线性可分，故可以用单神经元感知器解决这个问题。



2）经过两次推导即可学习好（$w_{11}=-1.5, w_{12}=0.1$）

3）可以采用模型直接识别出

2.数学方法证明下面问题对于两输入/单输出神经元感知器而言是不可解的。

$$\left\{X_1=\begin{bmatrix}-1\\1\end{bmatrix},d_1=1\right\}\quad\left\{X_2=\begin{bmatrix}-1\\-1\end{bmatrix},d_2=-1\right\}\quad\left\{X_3=\begin{bmatrix}1\\-1\end{bmatrix},d_3=1\right\}\quad\left\{X_4=\begin{bmatrix}1\\1\end{bmatrix},d_4=-1\right\}$$

异或问题求解

2. 数学方法证明:

$(x_1;x_2)$	O
(-1,1)	1
(-1,-1)	-1
(1,-1)	1
(1,1)	-1

相异为1，相同为-1

$W_{11}<0$

$\theta+W_1 < W_{12} < \theta - W_1$

对 W_{11}, W_{12}, θ 必须满足关系:

1) $-W_{11}+W_{12}-\theta>0 \rightarrow \theta<-W_{11}+W_{12}$ ~~$W_{12}>\theta+W_{11}$~~

2) $-W_{11}-W_{12}-\theta<0 \rightarrow \theta>-W_{11}-W_{12}$ ~~θ~~

$\theta>0$

3) $W_{11}-W_{12}-\theta>0 \rightarrow \theta<W_{11}-W_{12}$

4) $W_{11}+W_{12}-\theta<0 \rightarrow \theta>W_{11}+W_{12}$ ~~$W_{12}<\theta-W_{11}$~~

1)+3): $\theta<\theta$ ~~$<-(W_{11}+W_{12})$~~

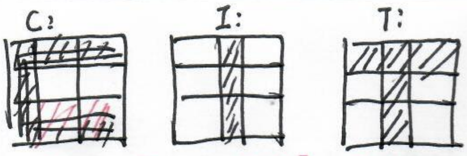
两者相矛盾，故单层感知器无法解决"异或"问题

2)+4): $\theta>\theta$ ~~$>W_{11}+W_{12}$~~

4.根据识别字母要求，设计一个三层 BP 神经网络，（输入层神经元，

隐层，输出神经元）

识别字母 C, I, T ：



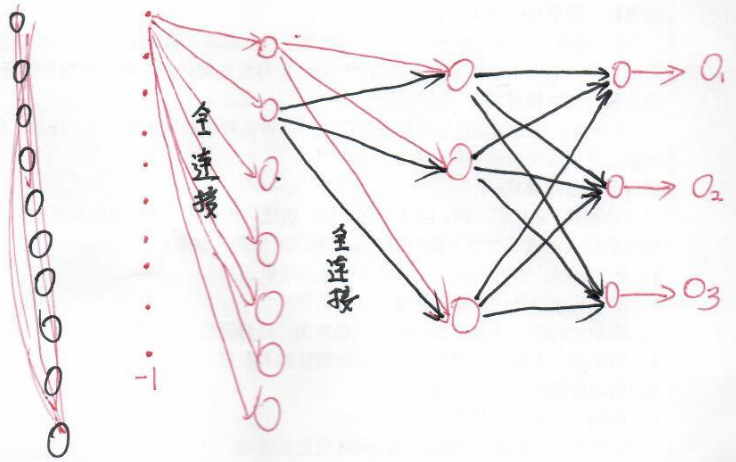
C:[111100111], I:[010010010], T:[111010010]

分析题目：① 输入为9宫格+1个固定偏置输入，故输入层结点数为10.

② 输出：输出为3个字母：C, I, T，神经网络输出~~范围~~为0或1. 故待识别3个符号的输出结点数为3！

C: 001 , I: 010 , T: 100

③ 设计该三层BP网络：



6、假设数据集 D 为：

	x_{i1}	x_{i2}	y_i
i=1	3	3	1
i=2	4	3	1
i=3	1	1	-1

试用对偶算法来计算该数据集的硬间隔支持向量机的决策边界。

解：

(1) 改写条件极值。原算法要求解的条件极值为：

$$\min_{\boldsymbol{w},b} \quad \frac{1}{2}||\boldsymbol{w}||^2$$
$$s.t. \quad y_i(\boldsymbol{w}\cdot\boldsymbol{x_i}+b)\geq 1, i=1,2,3$$

根据该条件极值，首先写出拉格朗日函数：

$$L=\frac{1}{2}||\boldsymbol{w}||^2+\sum_{i=1}\lambda_i[1-y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i+b)]$$

然后根据拉格朗日乘数法以及KKT 条件，从上述条件极值可以得到下面的方程组：

$$\begin{cases}\dfrac{\partial L}{\partial \boldsymbol{w}}=0, \dfrac{\partial L}{\partial b}=0\\ \lambda_i(1-y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i+b))=0\\ \lambda_i\geq 0, 1-y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i+b)\leq 0\end{cases}$$

据此可得：

$$\frac{\partial L}{\partial \boldsymbol{w}}=0\Longrightarrow \boldsymbol{w}=\sum_{i=1}^{n}\lambda_i y_i\boldsymbol{x}_i$$

又：

$$\frac{\partial L}{\partial b}=0+0+0+\sum_{i=1}^{n}\lambda_i y_i$$

所以可得：

$$\frac{\partial L}{\partial b}=0\Longrightarrow\sum_{i=1}^{n}\lambda_i y_i=0$$

综合下即有：

$$\begin{cases}\dfrac{\partial L}{\partial \boldsymbol{w}}=0 \Longrightarrow \boldsymbol{w}=\sum_{i=1}^{3}\lambda_i y_i\boldsymbol{x}_i\\ \dfrac{\partial L}{\partial b}=0 \Longrightarrow \sum_{i=1}^{n}\lambda_i y_i=0\end{cases}$$

（1）消去条件中的 $\sum_{i=1}^{n} \lambda_i y_i = 0$ 。根据数据集 D ，我们要求解的对偶算法的条件极值如下：

$$
\begin{aligned}
\min_{\lambda} \quad & \frac{1}{2}\sum_{i=1}^{3}\sum_{j=1}^{3}\lambda_i\lambda_j y_i y_j \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j - \sum_{i=1}^{3}\lambda_i \\
& = \frac{1}{2}\left(18\lambda_1^2 + 25\lambda_2^2 + 2\lambda_3^2 + 42\lambda_1\lambda_2 - 12\lambda_1\lambda_3 - 14\lambda_2\lambda_3\right) - \lambda_1 - \lambda_2 - \lambda_3 \\
\text{s.t.} \quad & \lambda_1 + \lambda_2 - \lambda_3 = 0 \\
& \lambda_i \geq 0, \quad i = 1,2,3
\end{aligned}
$$

由第一个条件可得 $\lambda_3 = \lambda_1 + \lambda_2$ ，将其代入要求最小值的目标函数，就得到了新的函数，记作：

$$
s(\lambda_1, \lambda_2) = 4\lambda_1^2 + \frac{13}{2}\lambda_2^2 + 10\lambda_1\lambda_2 - 2\lambda_1 - 2\lambda_2
$$

这个函数融合了第一个条件，所以要优化的条件极值可以改写为：

$$
\begin{aligned}
\min_{\lambda} \quad & s(\lambda_1, \lambda_2) = 4\lambda_1^2 + \frac{13}{2}\lambda_2^2 + 10\lambda_1\lambda_2 - 2\lambda_1 - 2\lambda_2 \\
\text{s.t.} \quad & \lambda_i \geq 0, \quad i = 1,2,3
\end{aligned}
$$

实际上这就消去了条件中的 $\sum_{i=1}^{n} \lambda_i y_i = 0$ 。

（2）通过数据集 D 找到支持向量。根据拉格朗日乘数法以及KKT 条件，从修改后的条件极值可以得到下面的方程组：

$$
\begin{cases}
\dfrac{\partial s}{\partial \lambda_1} = 8\lambda_1 + 10\lambda_2 - 2 = 0 \\
\dfrac{\partial s}{\partial \lambda_2} = 10\lambda_1 + 13\lambda_2 - 2 = 0 \\
\lambda_i \geq 0
\end{cases}
$$

根据前两个方程可以算出：

$$
\begin{cases}
\dfrac{\partial s}{\partial \lambda_1} = 8\lambda_1 + 10\lambda_2 - 2 = 0 \\
\dfrac{\partial s}{\partial \lambda_2} = 10\lambda_1 + 13\lambda_2 - 2 = 0
\end{cases}
\Longrightarrow \lambda_1 = \frac{3}{2}, \lambda_2 = -1
$$

因为：

$$
\lambda_2 = -1 \ngeq 0
$$

最小值只能在 λ_i 的边界处取得，即或者在 $\lambda_1 = 0$ 或者在 $\lambda_2 = 0$ 处取得。

$$\frac{\partial s}{\partial \lambda_2}\Big|_{\lambda_1=0} = 13\lambda_2 - 2 = 0 \implies \lambda_2 = \frac{2}{13}$$

同样的道理，$\lambda_2 = 0$ 也是一个平面，其上的最小值在下面的点处取得：

$$\frac{\partial s}{\partial \lambda_1}\Big|_{\lambda_2=0} = 8\lambda_1 - 2 = 0 \implies \lambda_1 = \frac{1}{4}$$

即分别在 $(\frac{1}{4}, 0)$ 和 $(0, \frac{2}{13})$ 处取得，比较两处的函数值：

$$s\left(\frac{1}{4}, 0\right) = -\frac{1}{4} < s\left(0, \frac{2}{13}\right) = -\frac{2}{13}$$

即条件极值在 $(\frac{1}{4}, 0)$ 处取得，因此有：

$$\lambda_1 = \frac{1}{4}, \quad \lambda_2 = 0, \quad \lambda_3 = \frac{1}{4}$$

（3）根据支持向量求出决策边界。根据对偶算法的结论 ，如果 $\lambda_i \neq 0$ 则说明该点为支持向量，据此可以得到所有支持向量的集合
$K = \{(\boldsymbol{x_k}, y_k)\}$ 。根据集合 K 可求出：

$$\hat{\boldsymbol{w}} = \sum_k \lambda_k y_k \boldsymbol{x}_k$$

再在支持向量中随便挑选一个点 $(\boldsymbol{x_j}, y_j) \in K$ ，可求出：

$$\hat{b} = y_j - \sum_k \lambda_k y_k \boldsymbol{x}_k^{\mathrm{T}} \boldsymbol{x}_j$$

$$\hat{\boldsymbol{w}} = y_1 \lambda_1 \boldsymbol{x}_1 + y_3 \lambda_3 \boldsymbol{x}_3 = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}$$

$$\hat{b} = y_1 - (y_1 \lambda_1 \boldsymbol{x}_1 \cdot \boldsymbol{x}_1 + y_3 \lambda_3 \boldsymbol{x}_3 \cdot \boldsymbol{x}_1) = -2$$

进而得到决策边界为：

$$h(\boldsymbol{x}) = \mathrm{sign}(\hat{\boldsymbol{w}} \cdot \boldsymbol{x} + \hat{b}) = \mathrm{sign}(\frac{1}{2}x_1 + \frac{1}{2}x_2 - 2)$$