5.1 试述将线性函数 $f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}$ 用作神经元激活函数的缺陷.

理想的激活函数是阶跃函数，它将输入值映射为输出值"0"或"1"，"0"对应于神经元兴奋，"1"对应于抑制。但阶跃函数具有不连续、不光滑等不太好的性质，因此实际常用 sigmoid 函数作为激活函数，它把可能在较大范围内变化的输入值挤压到 (0,1) 输出值范围内，且在 0 附近变化很快，但线性函数斜率不变，不能很好得拟合阶跃函数，且无论结构多复杂的神经网络都会退化成一个线性回归.

• 讨论 $\frac{\exp(x_i)}{\sum_{j=1}^{C}\exp(x_j)}$ 和 $\log\sum_{j=1}^{C}\exp(x_j)$ 的数值溢出问题

$$h(x) = \frac{\exp(x_i)}{\sum_{j=1}^{C}\exp(x_j)}$$

上溢： $\exp(x_i)$ 过大， $\exp(x_i) \to +\infty$

下溢： $\sum_{j=1}^{C}\exp(x_j)$ 过小，在分母上 $\sum_{j=1}^{C}\exp(x_j) \to 0$

解决： 令 $max = \max\{x_1, \cdots, x_C\}$， $x_i = x_i - max$

$$\text{则 } h'(x) = \frac{\exp(x_i - max)}{\sum_{j=1}^{C}\exp(x_j - max)} = \frac{\exp(x_i)/\exp(max)}{\sum_{j=1}^{C}\exp(x_j)/\exp(max)} = \frac{\exp(x_i)}{\sum_{j=1}^{C}\exp(x_j)} = h(x)$$

且此时 $\exp(x_i) \le \exp(max - x_i) = \exp(0) = 1$，不会导致上溢.

$\sum_{j=1}^{C}\exp(x_j) \geq \exp(max - max) = \exp(0) = 1$，不会导致下溢

$$g(x) = \log\sum_{j=1}^{C}\exp(x_j)$$

上溢： $\sum_{j=1}^{C}\exp(x_j)$ 过大， $\sum_{j=1}^{C}\exp(x_j) \to +\infty$, $g(x) \to +\infty$

下溢： $\sum_{j=1}^{C}\exp(x_j)$ 过小， $\sum_{j=1}^{C}\exp(x_j) \to 0$, $g(x) \to -\infty$

解决： $\forall j \in [1, C]$，令 $max = \max\{x_1, \cdots, x_C\}$， $x_j = x_j - max$

$$\text{则 } g(x) = \log\sum_{j=1}^{C}\exp(x_j - max) + max = \log\sum_{j=1}^{C}\exp(x_j - max) + \log\exp\{max\}$$

$$= \log\sum_{j=1}^{C}\exp(x_j - max + max) = \log\sum_{j=1}^{C}\exp(x_j) = g(x)$$

且此时 $\sum_{j=1}^{C}\exp(x_j - max) \le \sum_{j=1}^{C}\exp(max - max) = C$，不会导致上溢.

$\sum_{j=1}^{C}\exp(x_j - max) \geq \exp(max - max) = \exp(0) = 1$，不会导致下溢

• 计算 $\frac{\exp(x_i)}{\sum_{j=1}^{C}\exp(x_j)}$ 和 $\log\frac{\exp(x_i)}{\sum_{j=1}^{C}\exp(x_j)}$ 关于向量 $\boldsymbol{x}=[x_1,\cdots,x_C]$ 的梯度

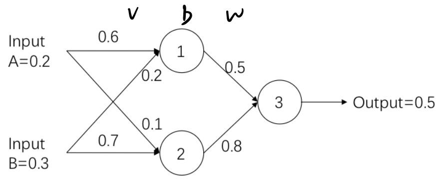
设 $f(x) = \frac{\exp(x_i)}{\sum_{j=1}^{C}\exp(x_j)}$, $g(x) = \log\frac{\exp(x_i)}{\sum_{j=1}^{C}\exp(x_j)}$

$$\frac{\partial f(x)}{\partial x_k} = \begin{cases} -\frac{\exp(x_i)\cdot\exp(x_k)}{\left(\sum_{j=1}^{C}\exp(x_j)\right)^2}, & k\neq i \\ \frac{\exp(x_i)\sum_{j=1}^{C}\exp(x_j)-\exp^2(x_i)}{\left(\sum_{j=1}^{C}\exp(x_j)\right)^2}, & k=i \end{cases}$$

$$\frac{\partial g(x)}{\partial x_k} = \begin{cases} -\frac{\sum_{j=1}^{C}\exp(x_j)}{\exp(x_i)}\cdot\frac{\exp(x_i)\cdot\exp(x_k)}{\left(\sum_{j=1}^{C}\exp(x_j)\right)^2} = -\frac{\exp(x_k)}{\sum_{j=1}^{C}\exp(x_j)}, & k\neq i \\ \frac{\sum_{j=1}^{C}\exp(x_j)}{\exp(x_i)}\cdot\frac{\exp(x_i)\sum_{j=1}^{C}\exp(x_j)-\exp^2(x_i)}{\left(\sum_{j=1}^{C}\exp(x_j)\right)^2} = \frac{\sum_{j=1}^{C}\exp(x_j)-\exp(x_i)}{\sum_{j=1}^{C}\exp(x_j)} = 1-\frac{\exp(x_i)}{\sum_{j=1}^{C}\exp(x_j)}, & k=i \end{cases}$$

• 考虑如下简单网络，假设激活函数为ReLU，用平方损失$\frac{1}{2}(y-\hat{y})^2$计算误差，请用BP算法更新一次所有参数（学习率为1），给出更新后的参数值（给出详细计算过程），并计算给定输入值x=(0.2,0.3)时初始时和更新后的输出值，检查参数更新是否降低了平方损失值.

v b w



初始:

单元1: $\alpha_1 = 0.2\times0.6+0.3\times0.2 = 0.18$ $b_1 = relU(0.18) = \max\{0, 0.18\} = 0.18$

单元2: $\alpha_2 = 0.2\times0.1+0.3\times0.7 = 0.23$ $b_2 = relU(0.23) = \max\{0, 0.23\} = 0.23$

单元3: $\beta = 0.18\times0.5+0.23\times0.8 = 0.274$ $\hat{y} = relU(0.274) = \max\{0, 0.274\} = 0.274$

$E = \frac{1}{2}(y-\hat{y})^2 = \frac{1}{2}\times(0.5-0.274)^2 = 0.025538$

更新:

$\frac{\partial E}{\partial v_{A1}} = \frac{\partial E}{\partial \hat{y}}\cdot\frac{\partial \hat{y}}{\partial \beta}\cdot\frac{\partial \beta}{\partial b_1}\cdot\frac{\partial b_1}{\partial \alpha_1}\cdot\frac{\partial \alpha_1}{\partial v_{A1}} = (\hat{y}-y)\cdot1\cdot W_{13}\cdot1\cdot x_A = (0.274-0.5)\times1\times0.5\times1\times0.2 = -0.0226$

$\frac{\partial E}{\partial v_{A2}} = \frac{\partial E}{\partial \hat{y}}\cdot\frac{\partial \hat{y}}{\partial \beta}\cdot\frac{\partial \beta}{\partial b_2}\cdot\frac{\partial b_1}{\partial \alpha_2}\cdot\frac{\partial \alpha_2}{\partial v_{A2}} = (\hat{y}-y)\cdot1\cdot W_{23}\cdot1\cdot x_A = (0.274-0.5)\times1\times0.8\times1\times0.2 = -0.03616$

$\frac{\partial E}{\partial v_{B1}} = \frac{\partial E}{\partial \hat{y}}\cdot\frac{\partial \hat{y}}{\partial \beta}\cdot\frac{\partial \beta}{\partial b_1}\cdot\frac{\partial b_1}{\partial \alpha_1}\cdot\frac{\partial \alpha_1}{\partial v_{B1}} = (\hat{y}-y)\cdot1\cdot W_{13}\cdot1\cdot x_B = (0.274-0.5)\times1\times0.5\times1\times0.3 = -0.0339$

$\frac{\partial E}{\partial v_{B2}} = \frac{\partial E}{\partial \hat{y}}\cdot\frac{\partial \hat{y}}{\partial \beta}\cdot\frac{\partial \beta}{\partial b_2}\cdot\frac{\partial b_1}{\partial \alpha_2}\cdot\frac{\partial \alpha_2}{\partial v_{B2}} = (\hat{y}-y)\cdot1\cdot W_{23}\cdot1\cdot x_B = (0.274-0.5)\times1\times0.8\times1\times0.3 = -0.05424$

$V_{A1} = V_{A1} + \Delta V = V_{A1} - \eta\frac{\partial E}{\partial V_{A1}} = 0.6 + 1\times(0.0226) = 0.6226$

$V_{A2} = V_{A2} + \Delta V = V_{A2} - \eta\frac{\partial E}{\partial V_{A2}} = 0.1 + 1\times(0.03616) = 0.13616$

$V_{B1} = V_{B1} + \Delta V = V_{B1} - \eta\frac{\partial E}{\partial V_{B1}} = 0.2 + 1\times(0.0339) = 0.2339$

$V_{B2} = V_{B2} + \Delta V = V_{B2} - \eta\frac{\partial E}{\partial V_{B2}} = 0.7 + 1\times(0.05424) = 0.75424$

$$\frac{\partial E}{\partial w_{13}} = \frac{\partial E}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial \beta} \cdot \frac{\partial \beta}{\partial w_{13}} = (\hat{y} - y) \cdot 1 \cdot b_1 = (0.274 - 0.5) \times 1 \times 0.18 = -0.04068$$

$$\frac{\partial E}{\partial w_{23}} = \frac{\partial E}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial \beta} \cdot \frac{\partial \beta}{\partial w_{23}} = (\hat{y} - y) \cdot 1 \cdot b_2 = (0.274 - 0.5) \times 1 \times 0.23 = -0.05198$$

$$w_{13} = w_{13} + \Delta w = w_{13} - \eta \frac{\partial E}{\partial w_{13}} = 0.5 + 1 \times 0.04068 = 0.54068$$

$$w_{23} = w_{23} + \Delta w = w_{23} - \eta \frac{\partial E}{\partial w_{23}} = 0.8 + 1 \times 0.05198 = 0.85198$$

更新后：$\alpha_1' = 0.2 \times 0.6226 + 0.3 \times 0.2339 = 0.19469$　　$b_1' = reLU(0.19469) = 0.19469$

$\alpha_2' = 0.2 \times 0.1361b + 0.3 \times 0.75424 = 0.253504$　　$b_2' = reLU(0.253504) = 0.253504$

$\beta' = 0.19469 \times 0.54068 + 0.253504 \times 0.85198 = 0.321245$

$E' = \frac{1}{2}(y - \hat{y})^2 = \frac{1}{2}(0.5 - 0.321245)^2 = 0.015977 < 0.025538 = E$

降低了平方损失误差.

6.4 试讨论线性判别分析与线性核支持向量机在何种条件下等价.

线性判别分析（LDA）的思想是将样例投影到一条直线上，使得同类样例的投影点尽可能接近，异类样例的投影点尽可能远离，也可以解决多分类问题，仅在线性可分数据上取得较好结果.

线性核支持向量机（KLDA）是通过核化对线性判别分析进行非线性扩展，解决二分类问题.

当处理二分类问题，且两类样本线性可分时，二者等价.

6.6 试析 SVM 对噪声敏感的原因.

SVM 是要最大化支持向量间的距离，支持向量在决策中占比很大，若噪音样本出现在支持向量集中，会对决策造成影响

6.9 试使用核技巧推广对率回归, 产生“核对率回归”.

对率回归：

$$\ln \frac{p(y=1 \mid x)}{p(y=0 \mid x)} = w^T x + b$$

$$y = \frac{1}{1+e^{-(w^T x+b)}}$$

$$l(\beta) = \sum_{i=1}^{m} \left(-y_i \beta^T \hat{x}_i + \ln(1+e^{\beta^T \hat{x}_i})\right) \qquad \beta = (w; b) \qquad \hat{x} = (x; 1)$$

假设可通过某种映射 $\phi: \mathcal{X} \to F$ 将样本映射到一个特征空间 F，然后在 F 中执行对率回归，以求得 $h(x) = w^T \phi(x)$

由表示定理，$h(x)$ 可写为 $h(x) = \sum_{i=1}^{m} \alpha_i k(x, x_i)$ ，$w = \sum_{i=1}^{m} \alpha_i \phi(x_i)$

可将 $w^T x + b$ 改写为 $\sum_{i=1}^{m} \alpha_i \cdot \phi(x_i) \cdot \phi(x) + b = \sum_{i=1}^{m} \alpha_i k(x, x_i) + b$

令 $K \in R^{m \times m}$ 为核函数 k 所对应的核矩阵，$(K)_{ij} = k(x_i, x_j)$

令 $\beta^* = (\alpha; b)$，$x_i^* = (K_{\cdot i}; 1)$，其中 $K_{\cdot i}$ 表示核矩阵 K 的第 i 列

则 $L(\beta^*) = \sum_{i=1}^{m} \left(-y_i \beta^{*T} x_i^* + \ln(1+e^{\beta^{*T} x_i^*}\right)$

支持向量回归的对偶问题如下，

$$\max_{\alpha,\hat{\alpha}} g(\boldsymbol{\alpha},\hat{\boldsymbol{\alpha}}) = -\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}(\alpha_i-\hat{\alpha}_i)(\alpha_j-\hat{\alpha}_j)\kappa(\boldsymbol{x}_i,\boldsymbol{x}_j)+\sum_{i=1}^{m}\left(y_i(\hat{\alpha}_i-\alpha_i)-\epsilon(\hat{\alpha}_i+\alpha_i)\right)$$

$$s.t.\ C \geqslant \boldsymbol{\alpha},\hat{\boldsymbol{\alpha}} \geqslant 0 \text{ and } \textstyle\sum_{i=1}^{m}(\alpha_i-\hat{\alpha}_i)=0$$

请将该问题转化为类似于如下标准型的形式（$\boldsymbol{u},\boldsymbol{v},\boldsymbol{K}$均已知），

$$\max_{\alpha} g(\boldsymbol{\alpha}) = \boldsymbol{\alpha}^{\top}\boldsymbol{v}-\frac{1}{2}\boldsymbol{\alpha}^{\top}\boldsymbol{K}\boldsymbol{\alpha}$$

$$s.t.\ C \geqslant \boldsymbol{\alpha} \geqslant 0 \text{ and } \boldsymbol{\alpha}^{\top}\boldsymbol{u}=0$$

例如在软间隔SVM中$\boldsymbol{v}=\boldsymbol{1},\boldsymbol{u}=\boldsymbol{y},\boldsymbol{K}[i,j]=y_iy_j\kappa(\boldsymbol{x}_i,\boldsymbol{x}_j)$.

令 $\alpha = \begin{pmatrix}\alpha_1\\ \alpha_2\\ \vdots\\ \alpha_m\end{pmatrix}$ $\quad y=\begin{pmatrix}y_1\\ y_2\\ \vdots\\ y_m\end{pmatrix}$ $\quad K_{ij}=k(x_i,x_j)$ $\quad \epsilon^*=\begin{pmatrix}\epsilon\\ \epsilon\\ \vdots\\ \epsilon\end{pmatrix}$

令 $\alpha^*=\begin{pmatrix}\alpha\\ \hat{\alpha}\end{pmatrix}$ $\quad v=\begin{pmatrix}-y-\epsilon^*\\ y-\epsilon^*\end{pmatrix}$ $\quad K^*=\begin{pmatrix}K & -K\\ -K & K\end{pmatrix}$

则 $\sum_{i=1}^{m}\left(y_i(\hat{\alpha}_i-\alpha_i)-\epsilon(\hat{\alpha}_i+\alpha_i)\right)$

$= \sum_{i=1}^{m}\left(\alpha_i(-y_i-\epsilon)+\hat{\alpha}_i(y_i-\epsilon)\right) = \alpha^{T}(-y-\epsilon^*)+\hat{\alpha}^{T}(y-\epsilon^*)$

$= \alpha^{*T}v$

$-\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}(\alpha_i-\hat{\alpha}_i)(\alpha_j-\hat{\alpha}_j)k(x_i,x_j)$

$= -\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\left(\alpha_i k(x_i,x_j)\alpha_j-\alpha_i k(x_i,x_j)\hat{\alpha}_j-\hat{\alpha}_i k(x_i,x_j)\alpha_j+\hat{\alpha}_i k(x_i,x_j)\hat{\alpha}_j\right)$

$= -\frac{1}{2}\left(\alpha^T K\alpha-\alpha^T K\hat{\alpha}-\hat{\alpha}^T K\alpha+\hat{\alpha}^T K\hat{\alpha}\right)$

$= -\frac{1}{2}\alpha^{*T}K^*\alpha^*$

其中 $\alpha_i^*=\alpha_i$ 或 $\hat{\alpha}_i$，$\therefore\ 0\le\alpha^*\le C$

令 $u=\begin{pmatrix}1_{m\times1}\\ -1_{m\times1}\end{pmatrix}$，则有 $\alpha^{*T}u=\begin{pmatrix}\alpha^T & \hat{\alpha}^T\end{pmatrix}\begin{pmatrix}1_{m\times1}\\ -1_{m\times1}\end{pmatrix}=\sum_{i=1}^{m}(\alpha_i-\hat{\alpha}_i)=0$