1、 生物视觉感知系统的三个主要部分：外周脑、初级视皮层、纹外皮层。其中，锥状体细胞和杆状体细胞主要分布在外周脑，有更高级的视觉感知系统的区域是纹外皮层。同心圆感受野有感受野中心光兴奋而周边光抑制、感受野中心光抑制而周边光兴奋两个亚型。

V_1 $V_2, V_3, \cdots$

视网膜 HW1

2、 知觉的特性为恒常性。知觉是模式识别，模式识别的早期理论有模板匹配理论、原型理论、特征分析理论，近期理论有视觉计算理论、特征整合理论、成分识别理论。注意有哪些特点：指向性、集中性。

3、 记忆系统的三个子系统为感觉/瞬时记忆（SM）、短时记忆（STM）、长时记忆（LTM）。其中感觉记忆的主要编码格式为图像。画出三个记忆子系统与控制系统间的关系图。

定义 6*6 方阵 H：$H_{ij} = y_i y_j K(X_i, X_j)$，以及全由-1 构成的 6 维列向量 $\boldsymbol{f} = \begin{bmatrix}-1\\-1\\-1\\-1\\-1\\-1\end{bmatrix}$ 该问题可以简写为求二次型的最值：

$$\underset{\alpha}{\text{maximize}}\left(-\boldsymbol{f}^T\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}^T H\boldsymbol{\alpha}\right)$$

或者：

$$\underset{\alpha}{\text{minimize}}\left(\frac{1}{2}\boldsymbol{\alpha}^T H\boldsymbol{\alpha} + \boldsymbol{f}^T\boldsymbol{\alpha}\right)$$

带入样本值 X_i 以及标签 $y_i = \pm 1$，计算出 $H =$

4	1	1	0	-1	-1
1	4	0	-1	-9	-1
1	0	4	-1	-1	-9
0	-1	-1	4	1	1
-1	-9	-1	1	25	9
-1	-1	-9	1	9	25

该问题的约束条件为：

$$\begin{cases} 0 \le \alpha_i \le C \\ \sum_{i=1}^{6} \alpha_i y_i = 0 \end{cases}$$

SVM

考虑以下样本：

负样本：$X_1 = (1,0)^T, X_2 = (0,1)^T, X_3 = (0,-1)^T$

正样本：$X_4 = (-1,0)^T, X_5 = (0,2)^T, X_6 = (0,-2)^T$

(1)使用如下的非线性变换，将 6 个样本 $X = (x_1, x_2)$ 变换为 $Z = (z_1, z_2)$，并画图比较 6 个样本在 X 空间与 Z 空间中的分布情况

$$z_1 = x_1^2 - 2x_1 + 3, z_2 = x_1^2 - 2x_2 - 3$$

$Z_1 = (1,-2)^T, Z_2 = (4,-5)^T, Z_3 = (4,-1)^T$

$Z_4 = (5,-2)^T, Z_5 = (7,-7)^T, Z_6 = (7,1)^T$

(2) 通过计算求解或者观察图像，写出 Z 空间中的支持向量，并写出 X 空间中的决策函数

观察法得到支持向量：$(4,-5)^T, (4,-1)^T, (5,-2)^T$

Z空间分类界面为 $z_1 = 4.5$

X空间分类界面为：$z_1 = x_2^2 - 2x_1 + 3 = 4.5$，即 $x_2^2 - 2x_1 - 1.5 = 0$

HW2 SVM

(3) 若采用二阶多项式核函数 $K(X_i, X_j) = (1 + X_i^T X_j)^c$，写出支持向量机 Lagrange 对偶问题的目标函数及约束条件

$$\max \Sigma\alpha_i - \frac{1}{2}\Sigma_i\Sigma_j\alpha_i\alpha_j y_i y_j x_i^T x_j,$$

$$s.t. \alpha_i \ge 0, \Sigma_i \alpha_i y_i = 0$$

HW2

1.多层感知机网络-BP 算法

考虑某分类任务中，有参数如下的多层感知机结构。其中隐藏层使用 **Sigmoid** 激活函数，Loss 函数使用输出层向量 $\mathbf{p} = (p_1, p_2)^T$ 与 one-hot 标签向量 $\mathbf{y} = (y_1, y_2)^T$ 的二元交叉熵损失函数，即

HW3

BP

$$\text{Loss} = -\frac{1}{2}\sum_{i=1}^{2} y_i \ln(p_i) + (1 - y_i)\ln(1 - p_i)$$

使用 SGD 随机梯度下降法更新参数，每批只使用一个样本。当前迭代使用的样本是 $\{0.2, 0.5, 货车\}$，即输入 $(x_1, x_2)^T$ 为 $(0.2, 0.5)^T$，标签 $(y_1, y_2)^T$ 为 $(0, 1)^T$。

图 1. MLP 网络架构图

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

图 2. MLP 网络流程图

$$W_{Layer1} = \begin{bmatrix} 1, & 2 \\ 0.5, & 1.2 \\ -0.5, & 1.0 \end{bmatrix}, W_{Layer2} = \begin{bmatrix} -0.5, & 1.0, & 0.5 \\ 1.0, & -0.5, & 0.2 \end{bmatrix}$$

(矩阵形式的网络模型参数)

(1) 计算前向传播过程，给出各个神经元（包括隐藏层和输出层）和损失函数的计算结果；

① $z = W_{Layer1}x = \begin{bmatrix} 1.0 & 2.0 \\ 0.5 & 1.2 \\ -0.5 & 1.0 \end{bmatrix}\begin{bmatrix} 0.2 \\ 0.5 \end{bmatrix} = \begin{bmatrix} 1.2 \\ 0.7 \\ 0.4 \end{bmatrix}$ ② $a = \sigma(z) = \begin{bmatrix} 0.7685 \\ 0.6682 \\ 0.5987 \end{bmatrix}$

③ $output = W_{Layer2}a = \begin{bmatrix} -0.5 & 1.0 & 0.5 \\ 1.0 & -0.5 & 0.2 \end{bmatrix}\begin{bmatrix} 0.7685 \\ 0.6682 \\ 0.5987 \end{bmatrix} = \begin{bmatrix} 0.5833 \\ 0.5542 \end{bmatrix}$

④ $\text{loss} = -\frac{1}{2}\sum_{i=1}^{2} y_i \ln(p_i) + (1 - y_i)\ln(1 - p_i)$

$= -\frac{1}{2}[(\ln(1 - 0.5833)) + (\ln(0.5542)] \quad = 0.7328$

BP

(2) 设学习率为 0.1，计算一次参数更新后的 W_{Layer1} 和 W_{Layer2}。

反向传播

(适当保留有效位数进行计算即可)。

首先计算Loss函数对输出层p1,p2的偏导数

$$\text{Loss} = -\frac{1}{2}\sum_{i=1}^{2} y_i \ln(p_i) + (1 - y_i)\ln(1 - p_i)$$

$$\frac{\partial L}{\partial p_i} = -\frac{1}{2}\left[\frac{y_i}{p_i} - \frac{1 - y_i}{1 - p_i}\right]$$

所以 $\frac{\partial L}{\partial p_1} = -\frac{1}{2}\left[\frac{0}{0.5833} - \frac{1-0}{1-0.5833}\right] = 1.1999$;

$\frac{\partial L}{\partial p_2} = -\frac{1}{2}\left[\frac{1}{0.5542} - \frac{1-1}{1-0.5542}\right] = -0.9022$

根据链式求导法则，计算Loss对矩阵 W_{layer2} 的偏导数 $\nabla^L_{W_{layer2}}$ （利用 $\frac{\partial L}{\partial p_i}$）

$a = \sigma(z)$

$$\frac{\partial L}{\partial W_{layer2}(1,1)} = \frac{\partial L}{\partial p_1}\frac{\partial p_1}{\partial W_{layer2}(1,1)} = 1.1999 * 0.7685 = 0.9221$$

$$\frac{\partial L}{\partial W_{layer2}(1,2)} = \frac{\partial L}{\partial p_1}\frac{\partial p_1}{\partial W_{layer2}(1,2)} = 1.1999 * 0.6682 = 0.8018$$

$$\frac{\partial L}{\partial W_{layer2}(1,3)} = \frac{\partial L}{\partial p_1}\frac{\partial p_1}{\partial W_{layer2}(1,3)} = 1.1999 * 0.5987 = 0.7184$$

$$\frac{\partial L}{\partial W_{layer2}(2,1)} = \frac{\partial L}{\partial p_2}\frac{\partial p_2}{\partial W_{layer2}(2,1)} = -0.9022 * 0.7685 = -0.6933$$

$$\frac{\partial L}{\partial W_{layer2}(2,2)} = \frac{\partial L}{\partial p_2}\frac{\partial p_2}{\partial W_{layer2}(2,2)} = -0.9022 * 0.6682 = -0.6029$$

$$\frac{\partial L}{\partial W_{layer2}(2,3)} = \frac{\partial L}{\partial p_2}\frac{\partial p_2}{\partial W_{layer2}(2,3)} = -0.9022 * 0.5987 = -0.5401$$

$$W_{layer2} - 0.1 * \nabla_{W_{layer2}} L = \begin{bmatrix} -0.5 & 1.0 & 0.5 \\ 1.0 & -0.5 & 0.2 \end{bmatrix} - \begin{bmatrix} 0.0922 & 0.0801 & 0.0718 \\ -0.0693 & -0.0602 & 0.0540 \end{bmatrix}$$

(W_2 更新)

BP

$$= \begin{bmatrix} -0.5922, & 0.9198, & 0.4282 \\ 1.0693, & -0.4397, & 0.2540 \end{bmatrix}$$

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

$S'(x) = S(x)(1 - S(x))$ 链式求导法则，先计算求解 $\frac{\partial L}{\partial z_i}$

Sigmoid导数

$$\frac{\partial L}{\partial z_1} = \left(\frac{\partial L}{\partial p_1}\frac{\partial p_1}{\partial a_1} + \frac{\partial L}{\partial p_2}\frac{\partial p_2}{\partial a_1}\right)\frac{\partial a_1}{\partial z_1} =$$

$$(1.1999 * (-0.5) + (-0.9022) * 1) * \sigma'(z_1) = -0.2672$$

$$\frac{\partial L}{\partial z_2} = \left(\frac{\partial L}{\partial p_1}\frac{\partial p_1}{\partial a_2} + \frac{\partial L}{\partial p_2}\frac{\partial p_2}{\partial a_2}\right)\frac{\partial a_2}{\partial z_2} =$$

$$(1.1999 * 1 + (-0.9022) * (-0.5)) * \sigma'(z_2) = 0.3660$$

$$\frac{\partial L}{\partial z_3} = \left(\frac{\partial L}{\partial p_1}\frac{\partial p_1}{\partial a_3} + \frac{\partial L}{\partial p_2}\frac{\partial p_2}{\partial a_3}\right)\frac{\partial a_3}{\partial z_3} =$$

$$(1.1999 * 0.5 + (-0.9022) * 0.2) * \sigma'(z_3) = 0.1008$$

利用 $\frac{\partial L}{\partial z_i}$，求解 $\nabla^L_{W_{layer1}}$ $\{x_1, x_2\}^T$

$$\frac{\partial L}{\partial W_{layer1}(1,1)} = \frac{\partial L}{\partial z_1}\frac{\partial z_1}{\partial W_{layer1}(1,1)} = -0.2672 * 0.2 = -0.0534$$

$$\frac{\partial L}{\partial W_{layer1}(2,1)} = \frac{\partial L}{\partial z_1}\frac{\partial z_1}{\partial W_{layer1}(2,1)} = -0.2672 * 0.5 = -0.1336$$

$$\frac{\partial L}{\partial W_{layer1}(1,2)} = \frac{\partial L}{\partial z_2}\frac{\partial z_2}{\partial W_{layer1}(1,2)} = 0.3660 * 0.2 = 0.0732$$

$$\frac{\partial L}{\partial W_{layer1}(2,2)} = \frac{\partial L}{\partial z_2}\frac{\partial z_2}{\partial W_{layer1}(2,2)} = 0.3660 * 0.5 = 0.1830$$

$$\frac{\partial L}{\partial W_{layer1}(3,1)} = \frac{\partial L}{\partial z_3}\frac{\partial z_3}{\partial W_{layer1}(3,1)} = 0.1008 * 0.2 = 0.0202$$

$$\frac{\partial L}{\partial W_{layer1}(3,2)} = \frac{\partial L}{\partial z_3}\frac{\partial z_3}{\partial W_{layer1}(3,2)} = 0.1008 * 0.5 = 0.0504$$

$$W_{layer1} - 0.1 * \nabla_{W_{layer1}} L = \begin{bmatrix} 1.0053, & 2.0134 \\ 0.4927, & 1.1817 \\ -0.5020, & 0.9950 \end{bmatrix}$$

(W_1 更新)

1、 贝叶斯决策 贝叶斯

HW5

（1） 对于二维平面中的分类问题，假设：

先验

$$P(x|\omega_1) \sim N\left(\begin{pmatrix}1\\2\end{pmatrix}, I\right), \quad P(x|\omega_2) \sim N\left(\begin{pmatrix}3\\1\end{pmatrix}, I\right)$$

μ, σ^2 Σ_1 Σ_2

且有 $P(\omega_1) = P(\omega_2) = 0.5$，请计算其贝叶斯决策边界。

符合情况一：$\Sigma_i = \sigma^2 I$

$$g_1(x) = -\frac{(x_1 - 1)^2 + (x_2 - 2)^2}{2} + \ln 0.5, g_2(x) = -\frac{(x_1 - 3)^2 + (x_2 - 1)^2}{2} + \ln 0.5$$

贝叶斯决策边界满足 $g_1(x) - g_2(x) = 0$

解得：$2x_1 - x_2 - \frac{5}{2} = 0$

（2） 若（1）中类别的概率不变，先验概率为 $P(\omega_1) = \frac{2}{3}$，$P(\omega_2) = \frac{1}{3}$，请问贝叶斯决策边界会发生什么变化？

符合情况一：$\Sigma_i = \sigma^2 I$

$$g_1(x) = -\frac{(x_1 - 1)^2 + (x_2 - 2)^2}{2} + \ln\frac{2}{3}, g_2(x) = -\frac{(x_1 - 3)^2 + (x_2 - 1)^2}{2} + \ln\frac{1}{3}$$

贝叶斯决策边界满足 $g_1(x) - g_2(x) = 0$

解得：$2x_1 - x_2 - \frac{5}{2} - \ln 2 = 0$

（3） 若（1）中的先验概率不变，假设：

$$P(x|\omega_1) \sim N\left(0, \begin{pmatrix}1 & 0\\1 & 2\end{pmatrix}\right), \quad P(x|\omega_2) \sim N\left(\begin{pmatrix}1\\2\end{pmatrix}, I\right)$$

Σ_1 Σ_2

请计算贝叶斯决策边界。

满足情况三：$\Sigma_i \ne \Sigma_j$

$$g_i(x) = -\frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1}(x - \mu_i) - \frac{1}{2}\ln|\Sigma_i| + \ln p(w_i)$$

$$\Sigma_1^{-1} = \begin{pmatrix}1 & 0\\-0.5 & 0.5\end{pmatrix}, \Sigma_2^{-1} = I$$

$$g_1(x) = -\frac{1}{2}(x_1 \quad x_2)\begin{pmatrix}1 & 0\\-\frac{1}{2} & \frac{1}{2}\end{pmatrix}\begin{pmatrix}x_1\\x_2\end{pmatrix} - \frac{1}{2}\ln 2 + \ln\frac{1}{2} = -\frac{1}{2}x_1^2 + \frac{1}{4}x_1x_2 - \frac{x_2^2}{4} - \frac{3\ln 2}{2}$$

$$g_2(x) = -\frac{1}{2}(x_1 - 1 \quad x_2 - 2)\begin{pmatrix}1 & 0\\0 & 1\end{pmatrix}\begin{pmatrix}x_1 - 1\\x_2 - 2\end{pmatrix} - \frac{1}{2}\ln 1 + \ln\frac{1}{2} = -\frac{x_1^2}{2} - \frac{x_2^2}{2} + x_1 + 2x_2 - \frac{5}{2} - \ln 2$$

贝叶斯决策边界满足 $g_1(x) - g_2(x) = 0$

解得：$\frac{x_1x_2}{4} + \frac{x_2^2}{4} - x_1 - 2x_2 + \frac{5}{2} - \frac{\ln 2}{2} = 0$

正态分布条件下的贝叶斯决策

- 假设类条件概率密度函数为正态分布

$$p(\boldsymbol{x}|\omega_i) = \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma_i|^{\frac{1}{2}}}\exp\{-\frac{1}{2}(x - \mu_i)^T\Sigma_i^{-1}(x - \mu_i)\}$$

- 判别函数定义为：

$$G_i(x) = p(\boldsymbol{x}|\omega_i)p(\omega_i) = \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma_i|^{\frac{1}{2}}}\exp\{-\frac{1}{2}(x - \mu_i)^T\Sigma_i^{-1}(x - \mu_i)\} \cdot p(\omega_i)$$

贝叶斯

- 取对数后判别函数为：

$$g_i(x) = -\frac{1}{2}(x - \mu_i)^T\Sigma_i^{-1}(x - \mu_i) - \frac{d}{2}\ln(2\pi) - \frac{1}{2}\ln|\Sigma_i| + \ln p(\omega_i)$$

假设各类先验概率相等，协方差矩阵的三种情况

(1) $\Sigma_i = \sigma^2 I$ 最小欧氏距离分类器

(2) $\Sigma_i = \Sigma$ 最小马氏距离分类器

线性

马氏距离 Mahalanobis distance

$$d^2 = (x - \mu_i)^T\Sigma_i^{-1}(x - \mu_i)$$

(3) $\Sigma_i \ne \Sigma_j \quad i \ne j$，二次判别函数

神经元

2、 隐含马尔可夫模型

使用隐含马尔可夫模型对某分类器的分类进行建模。假设该分类器对人脸表情的图像进行分类，考虑该分类器有两种预测状态：判断正确（A）、判断错误（B）。有隐含状态 S 决定着该分类器的分类结果，S 包含人脸图像的三个类别：高兴（S_1）、平静（S_2）、生气（S_3）。

假设待预测的人脸表情的状态转移矩阵为：

HMM

$$T = \begin{array}{c} \\ S_1 \\ S_2 \\ S_3 \end{array}\begin{array}{c} \begin{array}{ccc} S_1 & S_2 & S_3 \end{array} \\ \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix} \end{array}$$

图像类别的初始分布为 $[0.2, 0.4, 0.4]^T$ π

该分类器的预测矩阵为：

$$P = \begin{array}{c} \\ S_1 \\ S_2 \\ S_3 \end{array}\begin{array}{c} \begin{array}{cc} A & B \end{array} \\ \begin{bmatrix} 0.5 & 0.5 \\ 0.4 & 0.6 \\ 0.3 & 0.7 \end{bmatrix} \end{array}$$

现有预测结果为 $O = ABA$ $t=3$

(1) 请使用前向变量法求解 $P(O|\lambda)$

$$\alpha_1 = \begin{bmatrix} 0.2 \\ 0.4 \\ 0.4 \end{bmatrix} * \begin{bmatrix} 0.5 \\ 0.4 \\ 0.3 \end{bmatrix} = \begin{bmatrix} 0.1 \\ 0.16 \\ 0.12 \end{bmatrix}$$

$t=1$

$t=2$

$$\begin{bmatrix} 0.1 \\ 0.16 \\ 0.12 \end{bmatrix} * \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix} = \begin{bmatrix} 0.05 & 0.02 & 0.03 \\ 0.048 & 0.08 & 0.032 \\ 0.024 & 0.036 & 0.06 \end{bmatrix}$$

$$\alpha_2 = \begin{bmatrix} 0.122 \\ 0.136 \\ 0.122 \end{bmatrix} * \begin{bmatrix} 0.5 \\ 0.6 \\ 0.7 \end{bmatrix} = \begin{bmatrix} 0.061 \\ 0.0816 \\ 0.0854 \end{bmatrix}$$

$$\begin{bmatrix} 0.061 \\ 0.0816 \\ 0.0854 \end{bmatrix} * \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix} = \begin{bmatrix} 0.0305 & 0.0122 & 0.0183 \\ 0.02448 & 0.0408 & 0.01632 \\ 0.01708 & 0.02562 & 0.0427 \end{bmatrix}$$

$$\alpha_3 = \begin{bmatrix} 0.07206 \\ 0.07862 \\ 0.07732 \end{bmatrix} * \begin{bmatrix} 0.5 \\ 0.4 \\ 0.3 \end{bmatrix} = \begin{bmatrix} 0.03603 \\ 0.031448 \\ 0.023196 \end{bmatrix}$$

$t=3$

$$P(O|\lambda) = 0.03603 + 0.031448 + 0.023196 = 0.090674$$

(2) 请使用 viterbi 算法求解最可能的隐含序列状态

HMM

$$t=1, \quad \delta_1 = \begin{bmatrix} 0.2 \\ 0.4 \\ 0.4 \end{bmatrix} * \begin{bmatrix} 0.5 \\ 0.4 \\ 0.3 \end{bmatrix} = \begin{bmatrix} 0.1 \\ 0.16 \\ 0.12 \end{bmatrix}, \varphi_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$t=2, \quad \begin{bmatrix} 0.1 \\ 0.16 \\ 0.12 \end{bmatrix} * \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix} = \begin{bmatrix} 0.05 & 0.02 & 0.03 \\ 0.048 & 0.08 & 0.032 \\ 0.024 & 0.036 & 0.06 \end{bmatrix}$$

$$\delta_2 = \begin{bmatrix} 0.05 \\ 0.08 \\ 0.06 \end{bmatrix} * \begin{bmatrix} 0.5 \\ 0.6 \\ 0.7 \end{bmatrix} = \begin{bmatrix} 0.025 \\ 0.048 \\ 0.042 \end{bmatrix}, \varphi_2 = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

$$t=3, \quad \begin{bmatrix} 0.025 \\ 0.048 \\ 0.042 \end{bmatrix} * \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix} = \begin{bmatrix} 0.0125 & 0.005 & 0.0075 \\ 0.0144 & 0.024 & 0.0096 \\ 0.0084 & 0.0126 & 0.021 \end{bmatrix}$$

$$\delta_3 = \begin{bmatrix} 0.0144 \\ 0.024 \\ 0.021 \end{bmatrix} * \begin{bmatrix} 0.5 \\ 0.4 \\ 0.3 \end{bmatrix} = \begin{bmatrix} 0.0072 \\ 0.0096 \\ 0.0063 \end{bmatrix}, \varphi_3 = \begin{bmatrix} 2 \\ 2 \\ 3 \end{bmatrix}$$

$$q_3 = S_2, q_2 = \varphi_3(S_2) = S_2, q_1 = \varphi_2(S_2) = S_2$$

最可能的隐含序列状态为 $S_2S_2S_2$

激活函数

a) 阈值函数：$\varphi(v)=\begin{cases}1 & v\ge 0\\ 0 & v<0\end{cases}$

b) 分段线性函数：

$$\varphi(v)=\begin{cases}1 & v\ge 1\\ \frac{1}{2}(1+v) & -1<v<1\\ 0 & v\le -1\end{cases}$$

c) sigmoid函数：

$$\varphi(v)=\frac{1}{1+\exp(-\beta v)}$$

d) 双曲正切函数：

$$\varphi(v)=\tanh(\beta v)=\frac{e^{\beta v}-e^{-\beta v}}{e^{\beta v}+e^{-\beta v}}$$

e) ReLU：

$$y=f(z)=\max(0,z)$$

f) LeakyReLU:

$$y=f(z)=\begin{cases}z & z\ge 0\\ \alpha z & z<0\end{cases}$$

α 为小常数

g) PReLU:

α 为通过学习得到的数值

h) Softmax

- 适用于多分类任务

$$\mathrm{softmax}(x)_i=q_i=\frac{\exp(x_i)}{\sum_{j=1}^{n}\exp(x_j)}$$

Softmax的一个作用是输出一个最大值

其他很多函数也可以完成选取最大值，并归一化的功能，但是为什么现在神经网络中普遍采用Softmax作为回归分类函数呢？

选Softmax，主要因为Softmax使用了指数，这样可以让大的值更大，让小的更小，增加了区分对比度，学习效率更高。第二个是因为softmax是连续可导的，消除了拐点，这个特性在机器学习的梯度下降法等地方非常必要

主成分分析 PCA

➢ 输入：n个p维的样本$\mathbf{x}_1,\mathbf{x}_2,\cdots,\mathbf{x}_n$

➢ 目标：寻找最优(方差保留)的k个投影方向并进行特征变换降维

线性判别分析 LDA

寻找一个投影方向(线性变换/组合)，将高维问题降到一维问题，要求变换后的一维数据:同类样本尽可能聚集在一起，不同类的样本尽可能地远。

混淆矩阵

真实类别	预测类别	
	正例	反例
正例	TP(真正例)	FN(假反例)
反例	FP(假正例)	TN(真反例)

召回率Recall、精度Precision

$$\mathrm{Recall}=\frac{TP}{TP+FN},\ \mathrm{Precison}=\frac{TP}{TP+FP}$$

真阳性率(True Positive Rate)、假阳性率(False Positive Rate)

$$\mathrm{TPR}=\frac{TP}{TP+FN},\ \mathrm{FPR}=\frac{FP}{TN+FP}$$

错误率(error rate)与准确率(accuracy):

准确率: 1-错误率 = $1-a/N$

卷积

输入数据 (input volume)：$N\times C_{in}\times H_{in}\times W_{in}$

- 输入数据 batch size: N

卷积

卷积层超参数(conv filter)：

- 卷积核大小kernel size：$K_H\times K_W$
- 滤波器数目 number of filters：C_{out}
- 步长stride: S
- 延拓padding: P

输出数据 (output volume): $N\times C_{out}\times H_{out}\times W_{out}$

- $W_{out}=(W_{in}-K_W+2P)/S+1$
- $H_{out}=(H_{in}-K_H+2P)/S+1$

参数量(不考虑偏置)： $K_W\cdot K_H\cdot C_{in}\cdot C_{out}$

参数量(考虑偏置)： $(K_W\cdot K_H\cdot C_{in}+1)\cdot C_{out}$

贝叶斯定理：

贝叶斯

◆ 两事件的贝叶斯定理为：

$$P(A_i|B)=\frac{P(A_i)P(B|A_i)}{P(B)},i=1,2$$

◆ 模式识别中的贝叶斯定理表示：

$$P(\omega_i|x)=\frac{P(\omega_i)P(x|\omega_i)}{P(x)},i=1,2$$

先验概率 似然度

后验概率

$$posterior=\frac{prior\times likelihood}{evidence}$$

假设某种病毒检测方法，感染者检测阳性率为80%，未感染者的检测阳性率为5%;在某高风险地区参加检测的人群中有2%是真实病人;假设某人检测结果为阳性，试问这个人感染病毒的概率是多少?

解答：记ω_1为感染者，ω_2为未感染者，则可获取先验概率

$$P(\omega_1)=0.02,\ P(\omega_2)=0.98$$

记x是病毒检测阳性，则 $P(x|\omega_1)=0.8,\ P(x|\omega_2)=0.05$

代入得 $P(\omega_1|x)=\frac{P(\omega_1)P(x|\omega_1)}{P(\omega_1)P(x|\omega_1)+P(\omega_2)P(x|\omega_2)}$

$$=\frac{0.02\times 0.8}{0.02\times 0.8+0.98\times 0.05}=0.246$$

人的大脑中，细胞大致分为两类:神经元(neurons)和神经胶质细胞(glial cells)。

- 神经元的功能是接受和传送信息。
- 神经元与神经元，通过突触建立的联系，构成了极端复杂的信息传递与加工的神经回路（nerve circuit）。
- 神经回路是脑内信息处理的基本单位。
- 神经元内的神经冲动，主要通过电传导；神经元之间的传导，主要通过化学传导。

V1区（初级视皮层）视神经细胞主要有3种感受野：

1. 同心圆感受野(视网膜细胞感受野)：分2种亚型：中心光兴奋周边光抑制；中心光抑制周边光兴奋
2. 简单感受野（简单细胞，皮层简单感受野）
3. 复杂感受野（复杂细胞，皮层复杂感受野）

知觉的定义：知觉是将感觉信息组成有意义的对象，在已贮存的知觉知识经验的参与下，理解当前刺激的意义。对这种刺激意义的理解(获得)就是当前刺激和已贮存的知识经验相互作用的结果。

(1) 知觉是模式识别；(2) 知觉的信息加工过程：自上而下加工(概念驱动加工)和自下而上加工(数据驱动加工)的结合；(3) 知觉从哪里开始：从整体开始；(4) 知觉的产生：有直接及间接。

根据知觉时起主导作用的感官特性，知觉可分为：视知觉、听知觉、触知觉、嗅知觉、味知觉。

根据人脑所认识的事物特性，知觉可分为：空间知觉、时间知觉和运动知觉。

注意的定义：注意是人的心理活动对一定对象的指向和集中。

注意的功能：信号检测，选择性注意，分配性注意。

注意的特征：注意的选择性，注意的持续性，注意的转移。

记忆是在头脑中积累和保存个体经验的心理过程。记忆是人脑对外界输入的信息进行编码、存储和提取的过程。

特征变换: 主成分分析

PCA

➢ **示例：** 计算PCA投影方向，并样本特征维数压缩成一维

ω_1：$X_1=[2,2]^T$，$X_2=[2,3]^T$，$X_3=[3,3]^T$

ω_2：$X_4=[-2,-2]^T$，$X_5=[-2,-3]^T$，$X_6=[-3,-3]^T$

求解： 第一步：计算协方差矩阵 (此处零均值)

$$E\{XX^T\}=\frac{1}{6}\sum_{j=1}^{6}X_jX_j^T=\begin{bmatrix}5.7 & 6.3\\ 6.3 & 7.3\end{bmatrix}$$

第二步：计算协方差矩阵的特征值，并根据大小排序。λ_1=12.85，λ_2=0.15。

选择λ_1=12.85所对应的特征向量作为投影向量

$$U=\mathbf{u}_1=[0.66,0.75]^T$$

第三步：利用U对样本集中每个样本进行PCA变换。得到：

$X_1^*=2.82$，$X_2^*=3,57$，$X_3^*=4.23$

$X_4^*=-2.82$，$X_5^*=-3,57$，$X_6^*=-4.23$

过拟合是指在模型训练中，模型过分关注于训练数据集的细节和噪音特征，导致模型在训练数据集上表现很好，但在测试数据集上表现很差的现象。过拟合的模型泛化能力弱。欠拟合是指在模型训练中，模型对数据的特征关注不够，没有能够较好地学习到数据的规律和复杂度，导致模型的表现在训练数据和测试数据集上都很差的现象。

深度学习中防止过拟合的方法：L1、L2正则化；使用dropout层；batchNormalize；LayerNormalize；降低学习率/减少迭代次数；增加数据量；池化。