媒认计算速通

1.损失与求导

- 二分类交叉熵（BCE）：

$$loss = -p\log q - (1-p)\log(1-q)$$

  p为类别真值，q为模型输出的属于正类的概率，例如0.5。

真值p	预测 q	BCE
1	1	0
1	0.5	log2
1	0	∞
0	1	∞
0	0.5	log2
0	0	0

  注：$\log$实际上是$\ln$，以自然对数为底。

- 交叉熵目标函数：

$$H(P,Q) = -\sum p(x_i)\log q(x_i)$$

  若使用one-hot编码，则类别真值的概率分布为$y=[0,\cdots,1,\cdots,0]$，则Softmax回归的交叉熵目标函数为：

$$L = -\sum_{j=1}^{C} y_j \log q_j = -\log q_i$$

  因为其他$y_j=0$，仅有真实标签$y_i=1$

【例】：手写数字识别任务中类别数C=10。训练过程中，输入一个数字9的图片样本，采用 one-hot编码表示样本类别真值(ground truth, GT)为：$y^* = (0,0,0,0,0,0,0,0,0,1)^T$
一般情况下，初始模型softmax输出接近均匀分布：

$$y = (0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1)^T$$

交叉熵 $loss = -\log(0.1) = -\log\left(\frac{1}{C}\right) = \log(C) = \log(10) = 2.3026$

在一次训练结束后，对于一个数字9的图片样本，模型softmax输出：

$$y = (0.02, 0, 0, 0, 0, 0, 0.03, 0, 0, 0.95)^T$$

交叉熵 $loss = -\log(0.95) = 0.0513$

Softmax GT

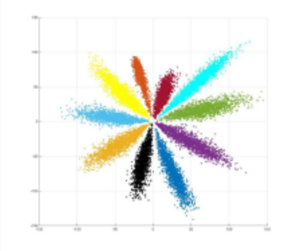
与二分类交叉熵的关系

- 当类别数C=2时，即为二分类交叉熵 (Binary Cross Entropy, BCE)

$$L=-\sum_{j=1}^{2} y_j \log q_j = -y_1 \log q_1 - y_2 \log q_2$$

$$\because y_2 = 1 - y_1 , q_2 = 1 - q_1$$

$$\therefore L = -y_1 \log q_1 - (1-y_1)\log(1-y_1)$$



- 矩阵求导

基本矩阵求导法则：

$$\mathbf{y}=W\mathbf{x}\rightarrow\frac{\partial\mathbf{y}}{\partial\mathbf{x}}=W^T$$
$$\mathbf{y}=\mathbf{x}W^T\rightarrow\frac{\partial\mathbf{y}}{\partial\mathbf{x}}=W$$
$$y=\mathbf{a}^T\mathbf{x}\mathbf{b}\rightarrow\frac{\partial y}{\partial\mathbf{x}}=\mathbf{a}\mathbf{b}^T$$

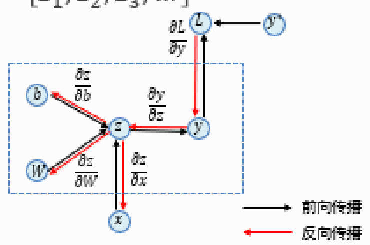
求导法则：

神经网络中的求导问题

- 如何求导：转化为标量对向量和矩阵的求导，利用$z=[z_1,z_2,z_3,...]$。

$z=W^Tx+b \quad y=\sigma(z) \quad L=(y-y^*)^2$

$$\frac{\partial L}{\partial W}=\frac{\partial L}{\partial z_1}\cdot\frac{\partial z_1}{\partial W}+\frac{\partial L}{\partial z_2}\cdot\frac{\partial z_2}{\partial W}+\cdots$$
$$\frac{\partial L}{\partial b}=\frac{\partial L}{\partial z_1}\cdot\frac{\partial z_1}{\partial b}+\frac{\partial L}{\partial z_2}\cdot\frac{\partial z_2}{\partial b}+\cdots$$
$$\frac{\partial L}{\partial x}=\frac{\partial L}{\partial z_1}\cdot\frac{\partial z_1}{\partial x}+\frac{\partial L}{\partial z_2}\cdot\frac{\partial z_2}{\partial x}+\cdots$$



- 要求哪些导
  - 终极目标：loss（标量）对模型各层网络参数（向量或矩阵）的导数：$\frac{\partial L}{\partial W}$和$\frac{\partial L}{\partial b}$
  - 中间过程：loss对模型输出的导数$\frac{\partial L}{\partial y}$、激活函数输出对激活函数输入的导数$\frac{\partial y_i}{\partial z_i}$、中间层输出对中间层输入的导数$\frac{\partial z_i}{\partial x}$、中间层输出对模型参数的导数$\frac{\partial z_i}{\partial W}$和$\frac{\partial z_i}{\partial b}$

24

其中：

- $\mathbf{y}=[y_1,\cdots,y_n]$，$\mathbf{z}=[z_1,\cdots,z_n]$，
- L为标量，$\frac{\partial L}{\partial\mathbf{y}}=[\frac{\partial L}{\partial y_1},\frac{\partial L}{\partial y_2},\cdots]$
- $\mathbf{y}=\sigma(\mathbf{z})$，向量对向量求导为：
  $\frac{\partial\mathbf{y}}{\partial\mathbf{z}}=[\frac{\partial y_1}{\partial z_1},\frac{\partial y_2}{\partial z_2},\cdots]=[y_1(1-y_1),y_2(1-y_2),\cdots]=\mathbf{y}*(1-\mathbf{y})$，交叉项都是0。
- $\frac{\partial L}{\partial\mathbf{z}}=[\frac{\partial L}{\partial y_1}\frac{\partial y_1}{\partial z_1},\frac{\partial L}{\partial y_2}\frac{\partial y_2}{\partial z_2},\cdots]=\frac{\partial L}{\partial\mathbf{y}}*\mathbf{y}*(1-\mathbf{y})$
- $\mathbf{z}=\mathbf{x}W^T+\mathbf{b}$

  前在前，后在后，系数都转置

  - $\frac{\partial L}{\partial\mathbf{x}}=\frac{\partial L}{\partial\mathbf{z}}\cdot W$

- $\frac{\partial L}{\partial W} = (\frac{\partial L}{\partial \mathbf{z}})^T \cdot \mathbf{x}$，其中：$\frac{\partial L}{\partial W^T} = \mathbf{x}^T \frac{\partial L}{\partial \mathbf{z}} \to \frac{\partial L}{\partial W} = (x^T \frac{\partial L}{\partial \mathbf{z}})^T = (\frac{\partial L}{\partial \mathbf{z}})^T \cdot \mathbf{x}$
- $\frac{\partial L}{\partial \mathbf{b}} = \frac{\partial L}{\partial \mathbf{z}}$

- eg1：

  1.5 标量 $y = \mathbf{a}^T W \mathbf{x}$，其中 $\mathbf{a} \in R^{n\times 1}, W \in R^{n\times n}, \mathbf{x} \in R^{n\times 1}$，现求标量 y 对向量 x 的偏导数 $\frac{\partial y}{\partial \mathbf{x}}$ 为：

  (A) $\mathbf{a}^T W$

  (B) $W\mathbf{a}^T$

  (C) $W\mathbf{a}$

  (D) $W^T\mathbf{a}$

  解析：令$D = a^T W$，则$y = Dx$，根据矩阵求导法则：$\frac{\partial y}{\partial x} = D^T = W^T a$。

  主要还是看维度：求导结果维度应与x相同

- eg2：

  2.1 设隐含层为 $\mathbf{z} = \mathbf{x}\mathbf{W}^T + \mathbf{b}$，其中 $\mathbf{x} \in R^{(1\times m)}$，$\mathbf{z} \in R^{(1\times n)}$，$\mathbf{W} \in R^{(n\times m)}$，$\mathbf{b} \in R^{(1\times n)}$ 均为已知，其激活函数如下：

  $$\mathbf{y} = \sigma(\mathbf{z}) = \frac{1}{1 + e^{-\mathbf{z}}}$$

  若训练过程中的目标函数为 L，且已知 L 对 y 的导数 $\frac{\partial L}{\partial \mathbf{y}} = [\frac{\partial L}{\partial y_1}, \frac{\partial L}{\partial y_2}, ..., \frac{\partial L}{\partial y_n}]$ 和 $\mathbf{y} = [y_1, y_2, ..., y_n]$ 的值。

2.1.1 请使用 y 表示出 $\frac{\partial \mathbf{y}}{\partial \mathbf{z}}$

解析：

激活函数 $\mathbf{y}=\sigma(\mathbf{z})$ 是逐点运算，因此只有对应角标的梯度 $\frac{\partial y_i}{\partial z_i}$ 需要计算，其他元素的梯度均为 0，根据求导法则可得，$\frac{\partial y_i}{\partial z_i}=y_i*(1-y_i)$。

严格来说，向量 $\mathbf{y}$ 对向量 $\mathbf{z}$ 的导数应该是
$[\frac{\partial y_1}{\partial \mathbf{z}},...,\frac{\partial y_n}{\partial \mathbf{z}}]=[y_1*(1-y_1),0,...,0,y_2*(1-y_2),0,...,0,...,0,...,0,y_n*(1-y_n)]$，
对于逐点运算的激活函数来说，这种表示方法并不实用，因此本题只需用 y_i 表示出 $\frac{\partial y_i}{\partial x_i}$ 即可。

评分标准：正确得出 $\frac{\partial y_i}{\partial x_i}$ 的表达式可得 3 分，满分 3 分。

2.1.2 请使用 y 和 $\frac{\partial L}{\partial y}$ 表示 $\frac{\partial L}{\partial \mathbf{x}}$，$\frac{\partial L}{\partial \mathbf{W}}$，$\frac{\partial L}{\partial \mathbf{b}}$。

提示:$\frac{\partial L}{\partial \mathbf{x}}$，$\frac{\partial L}{\partial \mathbf{W}}$，$\frac{\partial L}{\partial \mathbf{b}}$ 与 x,W,b 具有相同维度。

解析：

记矩阵 W 的每个行向量分别是 $\mathbf{w}_1,...,\mathbf{w}_n$，用 [;] 表示列向量，用 [,] 表示行向量，$\cdot$ 表示矩阵乘法，$*$ 表示元素乘法，$diag(\mathbf{x})$ 表示以 $x_1,...,x_n$ 为对角线值的对角矩阵。

$$
\begin{aligned}
\frac{\partial L}{\partial \mathbf{x}} &= \frac{\partial L}{\partial \mathbf{y}}\cdot[\frac{\partial y_1}{\partial z_1}*\frac{\partial z_1}{\partial \mathbf{x}};...;\frac{\partial y_n}{\partial z_n}*\frac{\partial z_n}{\partial \mathbf{x}}] \\
&= \frac{\partial L}{\partial \mathbf{y}}\cdot[\frac{\partial y_1}{\partial z_1}*\mathbf{w}_1;...;\frac{\partial y_n}{\partial z_n}*\mathbf{w}_n] \\
&= \frac{\partial L}{\partial \mathbf{y}}\cdot diag(\mathbf{y}*(1-\mathbf{y}))\cdot\mathbf{W} \\
&= (\frac{\partial L}{\partial \mathbf{y}}*\mathbf{y}*(1-\mathbf{y}))\cdot\mathbf{W}
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial L}{\partial \mathbf{W}} &= [\frac{\partial L}{\partial y_1}*\frac{\partial y_1}{\partial z_1}*\frac{\partial z_1}{\partial \mathbf{w}_1};...;\frac{\partial L}{\partial y_n}*\frac{\partial y_n}{\partial z_n}*\frac{\partial z_n}{\partial \mathbf{w}_n}] \\
&= [\frac{\partial L}{\partial y_1}*\frac{\partial y_1}{\partial z_1}*\mathbf{x};...;\frac{\partial L}{\partial y_n}*\frac{\partial y_n}{\partial z_n}*\mathbf{x}] \\
&= (\frac{\partial L}{\partial \mathbf{y}}*\mathbf{y}*(1-\mathbf{y}))^T\cdot\mathbf{x}
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial L}{\partial \mathbf{b}} &= [\frac{\partial L}{\partial y_1}*\frac{\partial y_1}{\partial z_1}*\frac{\partial z_1}{\partial b_1},...,\frac{\partial L}{\partial y_n}*\frac{\partial y_n}{\partial z_n}*\frac{\partial z_n}{\partial b_n}] \\
&= \frac{\partial L}{\partial \mathbf{y}}*\mathbf{y}*(1-\mathbf{y})
\end{aligned}
$$

- eg3:

3. 考虑一个输出神经元，采用 Sigmoid 函数作为激活函数 $p(z)=\frac{1}{1+e^{-z}}$ (在计算中取 $e\approx 2.7$)。考虑给定包含两个样本的数据集 $D=\{(X_i,y_i)\}, i=1,2$；神经元输入为 $X_i=(x_{0,i}=1,x_{1,i},x_{2,i})^T$；权值向量为 $\mathbf{w}=(w_0,w_1,w_2)^T$；$z_i=\mathbf{w}^T X_i$，神经元输出为 $p(z_i)$，样本对应的真值为 $y_i\in\{0,1\}$。在同一批次中输入两个样本，该批次的损失函数为：

$$L=-\sum_{i=1}^{2}\left[y_i\log(p(z_i))+(1-y_i)\log(1-p(z_i))\right]。$$

(1) 试推导损失函数 L 对权值向量 w_0,w_1,w_2 的导数，并写出学习率为 λ 的权值更新公式。

(2) 假设某轮迭代，权值 $\mathbf{w}=(w_0,w_1,w_2)^T=(0.5,0.5,1)^T$，神经元在同一批次中输入两个样本，分别为 $X_1=(1,2,-0.5)^T, y_1=1; X_2=(1,-2,-1.5)^T, y_2=0$，请给出两个样本在本次迭代前向计算过程中的神经元输出值。

(3) 假设学习率 $\lambda=0.1$，请在(2)基础上结合 BP 算法计算误差反向传播权值更新后的数值。

- (1)

  先考虑一个样本的损失函数的导数，再将所有样本的结果相加即可。对于一个样本而言，损失函数的相反数为：

  $$f(p)=y\log p+(1-y)\log(1-p)$$

  对p求导得：

  $$\frac{\partial f}{\partial p}=\frac{y}{p}-\frac{1-y}{1-p}$$

  p（激活函数）的表达式为：

  $$p(z)=\frac{1}{1+e^{-z}}$$

  因此p对z的导数为：

  $$\frac{\partial p}{\partial z}=p(1-p)$$

  因此由链式法则可得f对z的导数：

  $$\frac{\partial f}{\partial z}=\frac{\partial f}{\partial p}\frac{\partial p}{\partial z}=y(1-p)-(1-y)p=y-p$$

  继续将z的表达式展开：

  $$z=w^TX_i=w_0+w_1x_1+w_2x_2$$

  应用链式法则：

  $$\frac{\partial f}{\partial w_0}=\frac{\partial f}{\partial z}\frac{\partial z}{\partial w_0}=y-p$$
  $$\frac{\partial f}{\partial w_1}=\frac{\partial f}{\partial z}\frac{\partial z}{\partial w_1}=x_1(y-p)$$
  $$\frac{\partial f}{\partial w_2}=\frac{\partial f}{\partial z}\frac{\partial z}{\partial w_2}=x_2(y-p)$$

将每个样本的损失函数的导数累加，综上可得：

$$\frac{\partial L}{\partial w_0} = -\sum_{i=1}^{2}(y_i - p(z_i))$$
$$\frac{\partial L}{\partial w_1} = -\sum_{i=1}^{2}x_{1,i}(y_i - p(z_i))$$
$$\frac{\partial L}{\partial w_2} = -\sum_{i=1}^{2}x_{2,i}(y_i - p(z_i))$$

权值更新公式：

$$w_0 \leftarrow w_0 - \lambda\frac{\partial L}{\partial w_0}$$
$$w_1 \leftarrow w_1 - \lambda\frac{\partial L}{\partial w_1}$$
$$w_2 \leftarrow w_2 - \lambda\frac{\partial L}{\partial w_2}$$

- （2）

  $w = (0.5, 0.5, 1)^T$，$X_1 = (1, 2, -0.5)^T$，$X_2 = (1, -2, -1.5)$，$y_1 = 1, y_2 = 0$

  则前向计算过程为：

  对于样本1：

$$z_1 = w_0 + w_1 x_{1,1} + w_2 x_{1,2} = 0.5 + 0.5 \times 2 + 1 \times (-0.5) = 1$$
$$p(z_1) = \frac{1}{1+e^{-z_1}} = \frac{1}{1+e^{-1}} = 0.731$$

  也即神经元输出结果为0.731

  对于样本2：

$$z_2 = w_0 + w_1 x_{2,1} + w_2 x_{2,2} = 0.5 + 0.5 \times (-2) + 1 \times (-1.5) = -2$$
$$p(z_2) = \frac{1}{1+e^{-z_2}} = \frac{1}{1+e^{2}} = 0.119$$

  也即神经元的输出结果为0.119

- （3）

  梯度分别为：

$$\frac{\partial L}{\partial w_0} = -[(1-0.731) + (0-0.119)] = -0.150$$
$$\frac{\partial L}{\partial w_1} = -[2(1-0.731) + (-2)(0-0.119)] = -0.776$$
$$\frac{\partial L}{\partial w_2} = -[(-0.5)(1-0.731) + (-1.5)(0-0.119)] = -0.044$$

  更新后的参数：

$$w_0 = 0.5 - 0.1 \times (-0.150) = 0.515$$
$$w_1 = 0.5 - 0.1 \times (-0.776) = 0.578$$
$$w_2 = 1 - (-0.044) \times 0.1 = 1.004$$

- 卷积求导

  主要思路即为把卷积核大小的2D数据展开成1D列向量（img2col），然后将其排列（X_{col}），以便将卷积运算转化为矩阵运算。下面W_{re}表示把卷积核展开成列向量的结果，Y_{re}表示把输出特征图Y展开成行向量的结果。

  例题：

  首先进行常规前向计算：

2.1　已知某卷积层的输入为 X(该批量中样本数目为 1，输入样本通道数为 1)，采用一个卷积核 W，即卷积输出通道数为 1，卷积核尺寸为 2×2，卷积的步长为 1，无边界延拓，偏置量为 b：

$$X=\begin{bmatrix}-0.5 & 0.2 & 0.3\\ 0.6 & -0.4 & 0.1\\ 0.4 & 0.5 & -0.2\end{bmatrix},W=\begin{bmatrix}-0.2 & 0.1\\ 0.4 & -0.3\end{bmatrix},b=0.05$$

2.1.1　请计算卷积层的输出 Y。

解析：本题需要注意的是不要忘记加上偏置量 b.

$$X_{col}=\begin{bmatrix}-0.5 & 0.2 & 0.6 & -0.4\\ 0.2 & 0.3 & -0.4 & 0.1\\ 0.6 & -0.4 & 0.4 & 0.5\\ -0.4 & 0.1 & 0.5 & -0.2\end{bmatrix},W_{re}=\begin{bmatrix}-0.2\\ 0.1\\ 0.4\\ -0.3\end{bmatrix}$$

$$Y_{re}=W_{re}^{T}X_{col}+\boldsymbol{b}=\begin{bmatrix}0.53 & -0.15 & -0.1 & 0.4\end{bmatrix}\Rightarrow Y=\begin{bmatrix}0.53 & -0.15\\ -0.1 & 0.4\end{bmatrix}$$

  然后计算求导：

2.1.2 若训练过程中的目标函数为 L，且已知 $\frac{\partial L}{\partial Y}=\begin{bmatrix} 0.1 & -0.2 \\ 0.2 & 0.3 \end{bmatrix}$，请计算 $\frac{\partial L}{\partial X}$。

解析：

$$\frac{\partial L}{\partial Y_{re}}=\begin{bmatrix} 0.1 & -0.2 & 0.2 & 0.3 \end{bmatrix}, W_{re}=\begin{bmatrix} -0.2 \\ 0.1 \\ 0.4 \\ -0.3 \end{bmatrix}$$

$$\frac{\partial L}{\partial X_{col}}=W_{re}\cdot\frac{\partial L}{\partial Y_{re}}=\begin{bmatrix} -0.02 & 0.04 & -0.04 & -0.06 \\ 0.01 & -0.02 & 0.02 & 0.03 \\ 0.04 & -0.08 & 0.08 & 0.12 \\ -0.03 & 0.06 & -0.06 & -0.09 \end{bmatrix}$$

$$\xrightarrow{col2img}\frac{\partial L}{\partial X}=\begin{bmatrix} -0.02 & 0.01+0.04 & -0.02 \\ 0.04-0.04 & -0.03-0.08+0.02-0.06 & 0.06+0.03 \\ 0.08 & -0.06+0.12 & -0.09 \end{bmatrix}$$

$$=\begin{bmatrix} -0.02 & 0.05 & -0.02 \\ 0.0 & -0.15 & 0.09 \\ 0.08 & 0.06 & -0.09 \end{bmatrix}$$

上边的计算过程中，主要使用的公式为：

$$\frac{\partial L}{\partial X_{col}}=W_{re}\cdot\frac{\partial L}{\partial Y_{re}}$$

上式求出了损失函数关于X_{col}导数。

然后再进行col2img，也即把X_{col}中的每一列都还原为卷积核大小的2D矩阵，再将这些2D矩阵按照原先的2D数据形状排列好，然后融合其交界处的数值，得到原数据形状的2D矩阵（以该题为例）：

$$\begin{bmatrix} -0.02 & 0.01 \\ 0.04 & -0.03 \end{bmatrix}\begin{bmatrix} 0.04 & -0.02 \\ -0.08 & 0.06 \end{bmatrix}$$

$$\begin{bmatrix} -0.04 & 0.02 \\ 0.08 & -0.06 \end{bmatrix}\begin{bmatrix} -0.026 & 0.03 \\ 0.12 & -0.09 \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} -0.02 & 0.01+0.04 & -0.02 \\ 0.04-0.04 & -0.03-0.08+0.02-0.06 & 0.06+0.03 \\ 0.08 & -0.06+0.12 & -0.09 \end{bmatrix}$$

这就是$\frac{\partial L}{\partial X}$的最终结果。

2.参数量计算

- MLP参数量计算

  总结来看：每一层的参数量都是：输入特征维数*输出特征维数+输出特征维数。

  eg：

  采用具有一个隐含层的多层感知机完成三分类问题，隐含层节点数为4。输入一个批量的数据，数据矩阵的尺寸为10×1，其中10为样本数量，1为样本原始特征向量维数。。若网络中每一层神经元节点都使用偏置量，则模型的总参数量为：

  A 23

  仅考虑一条数据即可。第一层输入特征维数为1，输出特征维数为4，因此参数量：1*4+4=8；第二层输入特征维数为4，输出特征维数为3，因此参数量：4*3+3=15；总参数量：15+8=23。

  > 详细解析：
  >
  > 输入：
  >
  > $$x = [x_1]$$
  >
  > 此时样本特征维数为1。
  >
  > 第一层：
  >
  > $$W_1^T x + b_1 = \begin{bmatrix} w_{11} \\ w_{21} \\ w_{31} \\ w_{41} \end{bmatrix} [x_1] + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} = \begin{bmatrix} x'_1 \\ x'_2 \\ x'_3 \\ x'_4 \end{bmatrix}$$
  >
  > 此时样本特征维数为4，该层参数量为4+4=8。
  >
  > 第二层（输出层）：
  >
  > $$W_2^T x' + b_2 = \begin{bmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \end{bmatrix} \begin{bmatrix} x'_1 \\ x'_2 \\ x'_3 \\ x'_4 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} x''_1 \\ x''_2 \\ x''_3 \end{bmatrix}$$
  >
  > 此时样本特征维数为3（也即模型输出结果），该层参数量为4*3+3=15。
  >
  > 综上，总参数量为：8+15=23。

- 卷积参数量计算：

  $(K_W^2 \times C_{in} + 1) \times C_{out}$，其中$K_W$为卷积核边长，$C_{in}$为输入通道数，$C_{out}$为输出通道数。公式中+1是考虑到偏置量。

3.特征降维

- 两种分解
  - 特征值分解（对角化）：

$$X = U \Lambda U^{-1}$$

- SVD分解：

$$X = USV^T$$

其中U, V的列向量均为归一化向量，S为奇异值构成的对角阵。设$X_{m\times n}$，则维度：$U_{m\times m}$，$S_{m\times n}$，$V_{n\times n}$。计算方法：U为XX^T归一化特征向量构成的矩阵，V为X^TX归一化特征向量构成的矩阵，S为前两步求出的非0特征值（应该是一样的）开根号后从大到小排列构成的对角阵，且满足形状要求。

SVD求解实例

- 对于一个矩阵A:

$$\mathbf{A} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{pmatrix}$$

- 首先计算出A^TA和AA^T

$$\mathbf{A^TA} = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$$

$$\mathbf{AA^T} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

- 求出A^TA的特征值与特征向量

$$\lambda_1 = 3; v_1 = \begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix}; \lambda_2 = 1; v_2 = \begin{pmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix}$$

- 求出AA^T的特征值与特征向量

$$\lambda_1 = 3; u_1 = \begin{pmatrix} 1/\sqrt{6} \\ 2/\sqrt{6} \\ 1/\sqrt{6} \end{pmatrix}; \lambda_2 = 1; u_2 = \begin{pmatrix} 1/\sqrt{2} \\ 0 \\ -1/\sqrt{2} \end{pmatrix}; \lambda_3 = 0; u_3 = \begin{pmatrix} 1/\sqrt{3} \\ -1/\sqrt{3} \\ 1/\sqrt{3} \end{pmatrix}$$

- 利用$Av_i = \sigma_i u_i, i = 1, 2$求得奇异值，我们会发现求得的结果与$\sigma_i = \sqrt{\lambda_i}$的结果相同；

$$\begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix} = \sigma_1 \begin{pmatrix} 1/\sqrt{6} \\ 2/\sqrt{6} \\ 1/\sqrt{6} \end{pmatrix} \Rightarrow \sigma_1 = \sqrt{3}$$

$$\begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix} = \sigma_2 \begin{pmatrix} 1/\sqrt{2} \\ 0 \\ -1/\sqrt{2} \end{pmatrix} \Rightarrow \sigma_2 = 1$$

- 最终得到A的奇异值分解为

$$A = U\Sigma V^T = \begin{pmatrix} 1/\sqrt{6} & 1/\sqrt{2} & 1/\sqrt{3} \\ 2/\sqrt{6} & 0 & -1/\sqrt{3} \\ 1/\sqrt{6} & -1/\sqrt{2} & 1/\sqrt{3} \end{pmatrix} \begin{pmatrix} \sqrt{3} & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ -1/\sqrt{2} & 1/\sqrt{2} \end{pmatrix}$$

- PCA

基本步骤：假设输入向量$\mathbf{x}$的原始维度D，降维后维度d。首先算出协方差矩阵，然后计算其特征值和特征向量，按照特征值大小排序，取其前d大的**归一化**特征值对应的特征向量构成$D \times d$的矩阵W，此即为降维变换矩阵。然后再计算$W^T(x-\mu)$即可得到降维后向量（注意数据需减去均值）。

该方法不会考虑类别。

  - eg1：

➢ 给定两个模式类的样本分别为

ω_1： $\boldsymbol{X}_1=[2,2]^{\mathrm{T}}$， $\boldsymbol{X}_2=[2,3]^{\mathrm{T}}$， $\boldsymbol{X}_3=[3,3]^{\mathrm{T}}$

ω_2： $\boldsymbol{X}_4=[-2,-2]^{\mathrm{T}}$， $\boldsymbol{X}_5=[-2,-3]^{\mathrm{T}}$， $\boldsymbol{X}_6=[-3,-3]^{\mathrm{T}}$

要求利用PCA变换，把样本特征维数压缩成一维。

解：第一步：利用极大似然估计，计算协方差矩阵，其中，样本均值 $\mu=0$

$$E\{(X-\mu)(X-\mu)^T\}=\frac{1}{6}\Sigma_{j=1}^{6}(X_j-\mu)(X_j-\mu)^T=\begin{bmatrix}5.7 & 6.3\\ 6.3 & 7.3\end{bmatrix}$$

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。选择最大特征值所对应的特征向量作为投影向量：

$$\mathrm{W}=w_1=[0.66,0.75]^T$$

思路：

首先，计算所有样本均值：

$$\mu=\frac{1}{6}(X_1+X_2+X_3+X_4+X_5+X_6)=[0,0]^T$$

然后计算协方差矩阵。正常来讲协方差矩阵应为：

- 协方差矩阵为Σ $\quad \Sigma=\begin{bmatrix}\sigma_{11} & \sigma_{12} & \cdots & \sigma_{1d}\\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2d}\\ \vdots & \vdots & \ddots & \vdots\\ \sigma_{d1} & \sigma_{d2} & \cdots & \sigma_{dd}\end{bmatrix}$

其中σ_{ij}为随机变量x_i与x_j的协方差， $\sigma_{ij}=E\left[(x_i-\mu_i)(x_j-\mu_j)\right]$, $i,j=1,2,\cdots,d$

但此时只有有限个样本，因此使用**最大似然估计来计算协方差矩阵（详见"最大似然估计"部分）**：

$$\begin{aligned}\Sigma=E\{(X-\mu)(X-\mu)^T\}&=\frac{1}{6}\sum_{i=1}^{6}(X_i-\mu)(X_i-\mu)^T\\&=\frac{1}{6}\left\{\begin{bmatrix}2\\2\end{bmatrix}[2,2]+\begin{bmatrix}2\\3\end{bmatrix}[2,3]+\begin{bmatrix}3\\3\end{bmatrix}[3,3]+\begin{bmatrix}-2\\-2\end{bmatrix}[-2,-2]+\cdots\right\}\\&=\frac{1}{6}\begin{bmatrix}34 & 38\\38 & 44\end{bmatrix}\\&=\begin{bmatrix}5.7 & 6.3\\6.3 & 7.3\end{bmatrix}\end{aligned}$$

然后对协方差矩阵进行特征值分解：

$$|\Sigma-\lambda I|=0$$

解得：

$$\lambda_1=12.85,\ \ x_1=[0.66,0.75]^T$$
$$\lambda_2=0.15,\ \ x_2=[0.75,-0.66]^T$$

由于是降到1维，因此选取第1大的特征值λ_1对应的特征向量x_1，然后即可计算降维结果：

$$X_1'=x_1^T(X_1-\mu)=[0.66,0.75]\begin{bmatrix}2-0\\2-0\end{bmatrix}=2.82$$
$$X_2'=\cdots$$
$$\vdots$$

- eg2:

  某样本集合，其均值为$\mu=[0,1]^T$，样本协方差矩阵为C，且已知$CU=U\lambda$，

  $$其中U=\begin{bmatrix}-0.7 & 0.3\\ -0.5 & -0.4\end{bmatrix},\lambda=\begin{bmatrix}0.58 & 0\\ 0 & 15.82\end{bmatrix},$$

  请结合主成分分析 PCA 将某样本$x=[2,1]^T$变换至一维。

  由$CU=U\lambda$，可知协方差矩阵对应的特征值对角阵λ以及各特征值对应的特征向量矩阵U，可见最大的特征值为15.82，其对应的特征向量为$[0.3,-0.4]^T$，归一化后为$w=[0.6,-0.8]^T$，即得到降维变换矩阵。

  然后即可计算降维结果：

  $$x'=w^T(x-\mu)=[0.6,-0.8]\begin{bmatrix}2-0\\1-1\end{bmatrix}=1.2$$

- LDA（线性判别分析）

  基本步骤：首先根据：$S_{wi}=E\{(x-\mu_i)(x-\mu_i)^T\}$计算第$i$个类别的类内散度矩阵，其中$\mu_i$是第$i$个类别的均值向量。对所有类都计算完后，再根据每类的样本数进行加权求和，得到总的平均类内散度矩阵：$S_w=\sum_{i=1}^{C}P(w_i)S_{wi}$，其中$C$是类别总数。另外，再计算类间散度矩阵：$S_b=\sum_{i=1}^{C}P(w_i)(\mu_i-\mu)(\mu_i-\mu)^T$，其中$\mu$是全局总均值向量。然后求$S_w^{-1}S_b$的特征值和归一化特征向量，设要降维到$d$维，则取前$d$大的特征值的归一化特征向量拼成变换矩阵$W$，然后计算$W^Tx$即可对样本降维。

  eg1：

  给定两个模式类的样本分别为

  $\omega_1:\{(4,1),(2,4),(2,3),(3,6),(4,4)\}$

  $\omega_2:\{(9,10),(6,8),(9,5),(8,7),(10,8)\}$

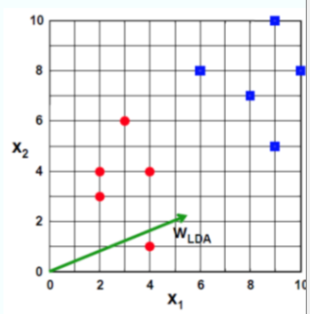
  利用LDA，把样本特征维数压缩成一维。

  解：$S_1=\begin{bmatrix}0.80 & -0.40\\ -0.40 & 2.64\end{bmatrix}$　$S_2=\begin{bmatrix}1.84 & -0.04\\ -0.04 & 2.64\end{bmatrix}$

  $\mu_1=[3.00\quad 3.60]$　$\mu_2=[8.40\quad 7.60]$　$\mu=[5.70\quad 5.60]$

  类内、类间散度矩阵为：

  $S_b=\begin{bmatrix}7.29 & 5.40\\ 5.40 & 4.00\end{bmatrix}$　$S_w=\begin{bmatrix}1.32 & -0.22\\ -0.22 & 2.64\end{bmatrix}$

  LDA通过广义特征值分解求解：

  $$S_w^{-1}S_bv=\lambda v\Rightarrow S_w^{-1}S_b-\lambda I=0\Rightarrow\begin{vmatrix}5.9462-\lambda & 4.4046\\ 2.5410 & 1.8822-\lambda\end{vmatrix}=0\Rightarrow\lambda=7.8284$$

  $$\begin{bmatrix}5.9462 & 4.4046\\ 2.5410 & 1.8822\end{bmatrix}\begin{bmatrix}v_1\\v_2\end{bmatrix}=7.8284\begin{bmatrix}v_1\\v_2\end{bmatrix}\quad\Rightarrow\quad\begin{bmatrix}v_1\\v_2\end{bmatrix}=\begin{bmatrix}0.9196\\0.3930\end{bmatrix}$$

  即投影方向为$w_{LDA}=\begin{bmatrix}0.9196\\0.3930\end{bmatrix}$　投影后样本为　$\omega_1:\{4.0714,3.4112,3.0182,5.1168,5.2504\}$

  $\omega_2:\{12.2064,8.6616,10.2414,10.1078,12.3400\}$　47

  对于第1个类别，其每个样本减去该类均值μ_1后的向量分别为：

  $$[1,-2.6]^T,[-1,0.4]^T,[-1,-0.6]^T,[0,2.4]^T,[1,0.4]^T$$

  然后计算该类的类内散度矩阵：

$$
\begin{aligned}
S_{w1} &= \frac{1}{5}\left\{\begin{bmatrix}1\\-2.6\end{bmatrix}[1,-2.6]+\begin{bmatrix}-1\\0.4\end{bmatrix}[-1,0.4]+\cdots\right\} \\
&= \frac{1}{5}\left\{\begin{bmatrix}1 & -2.6\\-2.6 & 6.76\end{bmatrix}+\cdots\right\} \\
&= \begin{bmatrix}0.8 & -0.4\\-0.4 & 2.64\end{bmatrix}
\end{aligned}
$$

同理求得第2类的类内散度矩阵S_{w2}

由于两个类别的样本数均为5，故$P(w_1)=P(w_2)=0.5$，将S_{w1}和S_{w2}加权平均后得总平均类内散度矩阵：

$$
S_w = \begin{bmatrix}1.32 & -0.22\\-0.22 & 2.64\end{bmatrix}
$$

然后求类间散度矩阵。由于第1类的均值$\mu_1=[3,3.6]^T$，第2类均值$\mu_2=[8.4,7.6]^T$，总均值$\mu=[5.7,5.6]^T$，二者减去总均值后分别为：$\mu_1'=[-2.7,-2]^T,\mu_2'=[2.7,2]^T$因此可得类间散度矩阵：

$$
\begin{aligned}
S_b &= \frac{1}{2}\begin{bmatrix}-2.7\\-2\end{bmatrix}[-2.7,-2]+\frac{1}{2}\begin{bmatrix}2.7\\2\end{bmatrix}[2,7,2] \\
&= \begin{bmatrix}7.29 & 5.4\\5.4 & 4\end{bmatrix}
\end{aligned}
$$

然后求$S_w^{-1}S_b$的特征值与归一化特征向量。由于要降到1维，因此取最大特征值对应的归一化特征向量：$w=[0.9196,0.3930]^T$。它即为降维矩阵，由此可得降维后的数据：

$$
x_1=[4,1]^T \to x_1' = w^T x_1 = \begin{bmatrix}0.9196\\0.3930\end{bmatrix}[4,1] = 4.0714
$$
$$
x_2 = \cdots
$$
$$
\vdots
$$

4.统计模式分类

- 正态分布中均值和协方差矩阵的最大似然估计：

$$
\boxed{\hat{\mu}=\hat{\theta}_1=\frac{1}{n}\sum_{k=1}^{n}x_k \qquad \hat{\Sigma}=\hat{\theta}_2=\frac{1}{n}\sum_{k=1}^{n}(x_k-\hat{\mu})(x_k-\hat{\mu})^T}
$$

也即给定一些样本后，计算其均值和协方差矩阵，非常常用。

最大似然的协方差矩阵估计是有偏的，协方差矩阵的无偏估计为：

$$
\boxed{\hat{\Sigma}_{unbiased}=\frac{1}{n-1}\sum_{k=1}^{n}(x_k-\hat{\mu})(x_k-\hat{\mu})^T}
$$

eg1：

3. 考虑如下从某正态分布获取的三个样本点：$x_1=[-3,3]^T, x_2=[-2,4]^T, x_3=[-1,-1]^T$，请利用最大似然估计获取正态分布的均值和协方差矩阵。

计算得均值为：

$$
\mu = \frac{1}{3}(x_1+x_2+x_3) = [-2,2]^T
$$

协方差矩阵:

$$
\begin{aligned}
\Sigma &= \frac{1}{3}\sum_{i=1}^{3}(x_i-\mu)(x_i-\mu)^T \\
&= \frac{1}{3}\left\{\begin{bmatrix}-1\\1\end{bmatrix}[-1,1]+\begin{bmatrix}0\\2\end{bmatrix}[0,2]+\begin{bmatrix}1\\-3\end{bmatrix}[1,-3]\right\} \\
&= \frac{1}{3}\begin{bmatrix}1 & -1\\-1 & 1\end{bmatrix}+\begin{bmatrix}0 & 0\\0 & 4\end{bmatrix}+\begin{bmatrix}1 & -3\\-3 & 9\end{bmatrix} \\
&= \frac{1}{3}\begin{bmatrix}2 & -4\\-4 & 14\end{bmatrix}
\end{aligned}
$$

- 贝叶斯分类器

  设某个未知类别的特征向量为x，共有C种可能的类别，w_i代表第i类。则后验概率为（贝叶斯公式）：

$$p(w_i|x)=\frac{p(w_i)p(x|w_i)}{p(x)}=\frac{p(w_i)p(x|w_i)}{\sum_j p(w_j)p(x|w_j)}$$

  w_i表示第i个类别，x表示一个未知类别的样本。$p(w_i|x)$为给定未知样本下判定其为w_i类的后验概率。

  贝叶斯决策:

  ➢ 贝叶斯决策：已知先验概率$P(\omega_i)$和类条件概率密度函数$P(x|\omega_i)$，计算后验概率$P(w_i|x)$对未知样本分类。选择后验概率最大的类别，可实现最小错误率的判决:

  - $\omega(x)=\arg\max p(\omega_i|x)=\arg\max p(x|\omega_i)p(\omega_i)=\arg\max g_i(x)$

  ➢ 判别函数:

  - 若$g_i(x)>g_j(x)$对一切$j\neq i$成立，则将x归入ω_i类

  ➢ 两类问题决策面方程:

  - 判别函数定义为$g(x)=g_1(x)-g_2(x)$，若$g(x)>0$判别为w_1
  - 决策面方程为$g(x)=0$

  也即，对于未知类别的样本x，选取后验概率$P(w_i|x)$最大的类别w_i（第i类），作为x的预测类。可实现最小错误率的判决。

  ➢ 正态分布：$p(\mathbf{x}|\omega_i)=\frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma_i|^{\frac{1}{2}}}\exp\{-\frac{1}{2}(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i)\}$

  ➢ 判别函数定义为：$G_i(x)=p(\mathbf{x}|\omega_i)p(\omega_i)=\frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma_i|^{\frac{1}{2}}}\exp\{-\frac{1}{2}(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i)\}\cdot p(\omega_i)$

  ➢ 取对数后判别函数为：$g_i(x)=-\frac{1}{2}(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i)-\frac{d}{2}\ln(2\pi)-\frac{1}{2}\ln|\Sigma_i|+\ln p(\omega_i)$

  - (计算题2.3)分类界面$g(x)=g_1(x)-g_2(x)$, $g(x)>0$为第一类,$g(x)<0$为第二类

  ➢ 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
  - （3）$\Sigma_i\neq\Sigma_j,\ i\neq j$ 二次判别函数

计算分类界面：分别对每个类别求出对数判别函数$g_i(x)$，然后分别两两相减即可求出每两个类别之间的判决平面。

eg1：

➤ 例：正态分布下有两类先验概率相等的样本集

$$D=\{(\boldsymbol{x}_n,y_n)\}_{n=1}^{N}=\left\{\begin{matrix}([0.5,1.5]^{\mathrm{T}},1),([1.5,1.5]^{\mathrm{T}},1),([0.5,0.5]^{\mathrm{T}},1),([1.5,0.5]^{\mathrm{T}},1),\\([1.5,0.5]^{\mathrm{T}},0),([2.5,0.5]^{\mathrm{T}},0),([1.5,-0.5]^{\mathrm{T}},0),([2.5,-0.5]^{\mathrm{T}},0)\end{matrix}\right\}$$

满足$\Sigma_1=\Sigma_2=\Sigma$。

$$\mu_1=\frac{1}{4}([0.5,1.5]^T+[1.5,1.5]^T+[0.5,0.5]^T+[1.5,0.5]^T)=[1,1]^T$$

$$\mu_2=\frac{1}{4}([1.5,0.5]^T+[2.5,0.5]^T+[1.5,-0.5]^T+[2.5,-0.5]^T)=[2,0]^T$$

$$\Sigma=\frac{1}{8}\left[\left(\begin{bmatrix}0.5\\1.5\end{bmatrix}-\begin{bmatrix}1\\1\end{bmatrix}\right)\left(\begin{bmatrix}0.5\\1.5\end{bmatrix}-\begin{bmatrix}1\\1\end{bmatrix}\right)^T+\cdots\right]=\begin{bmatrix}0.25&0\\0&0.25\end{bmatrix}$$

$$g(x)=g_1(x)-g_2(x)=-\frac{1}{2}(x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1)+\frac{1}{2}(x-\mu_2)^T\Sigma_2^{-1}(x-\mu_2)-\frac{1}{2}\ln\frac{|\Sigma_1|}{|\Sigma_2|}+\ln\frac{p(w_1)}{p(w_2)}$$

$$=(\mu_1-\mu_2)^T\Sigma^{-1}x-\frac{1}{2}(\mu_1^T\Sigma^{-1}\mu_1-\mu_2^T\Sigma^{-1}\mu_2)=-4x_1+4x_2+4$$

也即，先使用最大似然估计分别计算每个类别的均值μ_1,μ_2和协方差矩阵Σ（因为题目中给定了$\Sigma_1=\Sigma_2=\Sigma$，因此只需计算一个即可，是最小马氏距离分类器），然后代入公式求判别函数$g_1(x),g_2(x)$，最后用$g(x)=g_1(x)-g_2(x)$即为判决平面。

eg2：

设有三类正态分布的样本集，第一类均值为$\mu_1=[0,1]^T$，第二类均值为$\mu_2=[-1,0]^T$，第三类均值为$\mu_3=[1,-1]^T$。三类共享协方差矩阵，且出现的先验概率相等：

$$\Sigma_1=\Sigma_2=\Sigma_3=\Sigma=\begin{bmatrix}1.2&0.4\\0.4&1.8\end{bmatrix},\quad p(\omega_1)=p(\omega_2)=p(\omega_3)$$

(1) 结合正态分布条件下的贝叶斯决策，试证明上述分类界面为线性判别界面，且形式为 $g(\mathbf{x})=(\mu_i-\mu_j)^T\Sigma^{-1}\mathbf{x}-\frac{1}{2}(\mu_i^T\Sigma^{-1}\mu_i-\mu_j^T\Sigma^{-1}\mu_j)$。

(2) 计算上述两两分类的分界面。

(3) 根据上述贝叶斯分类器，对特征向量$\mathbf{x}=[0.2,1]^T$进行分类。

（1）

任取两类i,j，有后验概率：

$$p(w_i|x)=\frac{p(w_i)p(x|w_i)}{p(x)},p(w_j|x)=\frac{p(w_j)p(x|w_j)}{p(x)}$$

则分类界面应为这两个后验概率相等时，取对数得：

$$\log p(w_i|x)=\log p(w_j|x)$$

由于每个类别先验概率相等，也即$p(w_i)=p(w_j)$，因此代入贝叶斯公式可将上述等式转化为类别先验相等：

$$\log p(x|w_i)=\log p(x|w_j)$$

而又由样本的似然函数满足正态分布，且协方差矩阵一样，因此：

$$p(x|w_i)=\frac{1}{(2\pi)|\Sigma|^{1/2}}\exp[-(x-\mu_i)^T\Sigma^{-1}(x-\mu_i)]$$
$$p(x|w_j)=\frac{1}{(2\pi)|\Sigma|^{1/2}}\exp[-(x-\mu_j)^T\Sigma^{-1}(x-\mu_j)]$$

（也即在分布为w_i的条件下，观测到x的概率分布为正态分布）

代入得：

$$(x-\mu_i)^T\Sigma^{-1}(x-\mu_i)=(x-\mu_j)^T\Sigma^{-1}(x-\mu_j)$$

化简得：

$$-\mu_i\Sigma^{-1}x-x^T\Sigma^{-1}\mu_i+\mu_i^T\Sigma^{-1}\mu_i=-\mu_j\Sigma^{-1}x-x^T\Sigma^{-1}\mu_j+\mu_j^T\Sigma^{-1}\mu_j$$
$$-2\mu_i^T\Sigma^{-1}x+\mu_i^T\Sigma^{-1}\mu_i=-2\mu_j^T\Sigma^{-1}x+\mu_j^T\Sigma^{-1}\mu_j$$

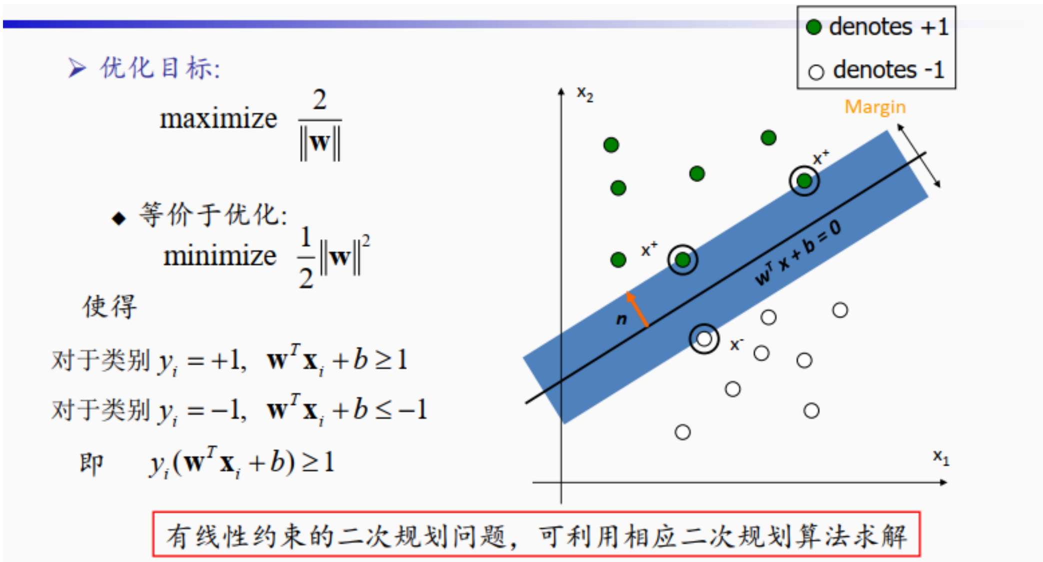
整理即可得到所求式。

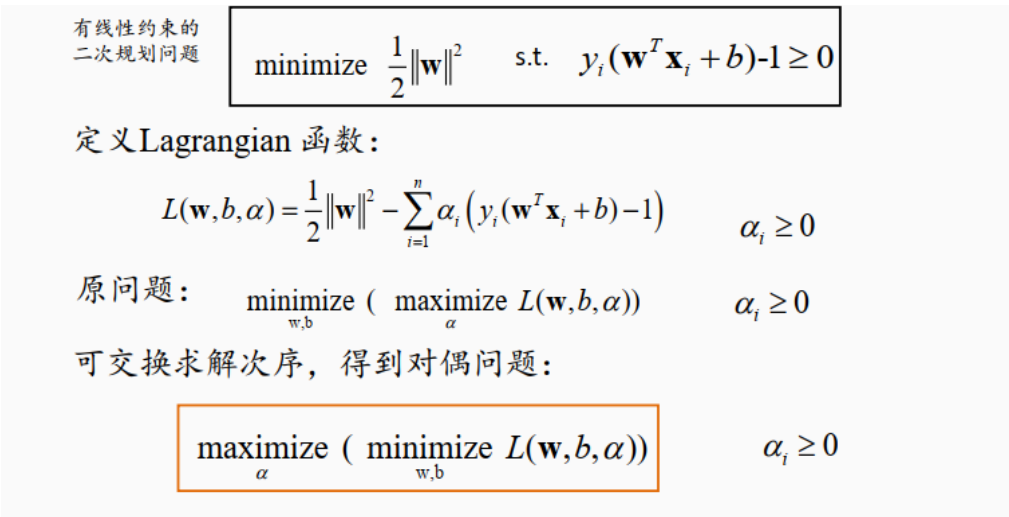
（2）（3）代入即可。

5.支持向量机

- 线性可分问题：

  设最优分类平面为$g(x)=w^T\mathbf{x}+b$，则基本优化问题：





其中y_i是样本$\mathbf{x}_i$的标签，为+1或-1

经过一番推导，待优化函数可总结为：

$$L=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j$$

目标为：

$$\max_{\alpha}(\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j)$$

且满足：

$$\alpha_i\geq 0,\quad \sum_{i=1}^{n}\alpha_i y_i=0$$

分别将上式对所有α_i求偏导，并让导数=0，即可求出所有α_i。若$\alpha_i>0$则代表$\mathbf{x}_i$是一个支持向量，记作$i\in SV$。

然后即可求解最佳分类平面的权值向量和偏置量：$g(\mathbf{x})=w^T\mathbf{x}+b$

最佳分类界面的权值向量：

$$w=\sum_{i\in SV}\alpha_i y_i \mathbf{x}_i$$

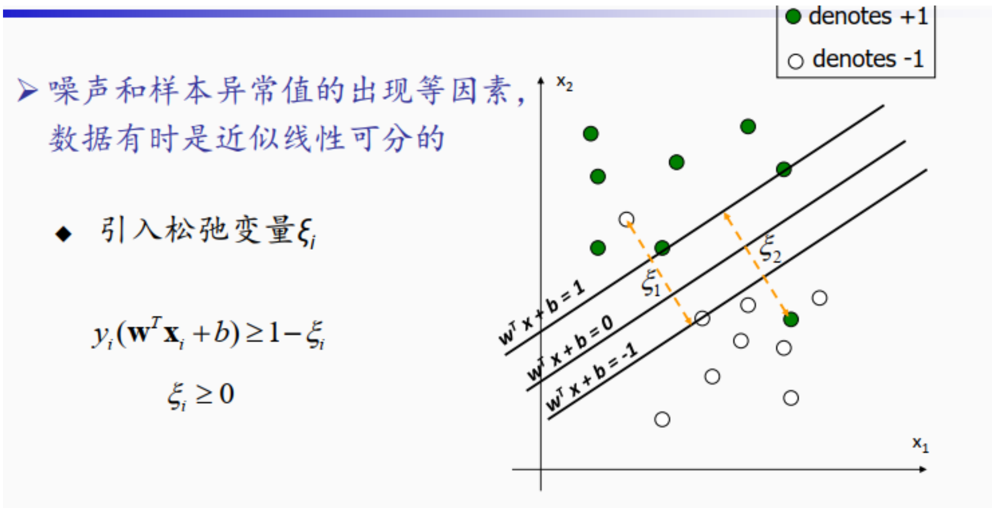
然后代入任意一个支持向量$(\mathbf{x}_i,y_i)$，即可解得偏置量b。

此时就完全得到了最佳分类界面。对于一个未知输入样本$\mathbf{x}$，分类决策函数$y=w^T\mathbf{x}+b$可写为：

$$y=\sum_{i=SV}\alpha_i y_i \mathbf{x}_i^T\mathbf{x}+b$$

若$y\geq 1$则判定为+1标签，若$y\leq -1$则判定为-1标签。

- 近似线性可分问题：



➢ 求解:

$$\text{minimize} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i$$

$$\text{使得} \quad y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i$$

$$\xi_i \geq 0$$

- 引入正则化参数 C 控制模型拟合能力
- 定义Lagrange函数

$$L(\mathbf{w}, b, \alpha, \gamma) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i\left(y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 + \xi_i\right) - \sum_{i=1}^{n}\gamma_i\xi_i$$

$$\text{s.t.} \quad \alpha_i \geq 0 \quad \gamma_i \geq 0$$

$$\boxed{\begin{aligned} &\text{minimize} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i(y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 + \xi_i) - \sum_{i=1}^{n}\gamma_i\xi_i \\ &\text{s.t.} \quad \alpha_i \geq 0 \quad \gamma_i \geq 0 \end{aligned}}$$

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^{n}\alpha_i y_i \mathbf{x}_i$$

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

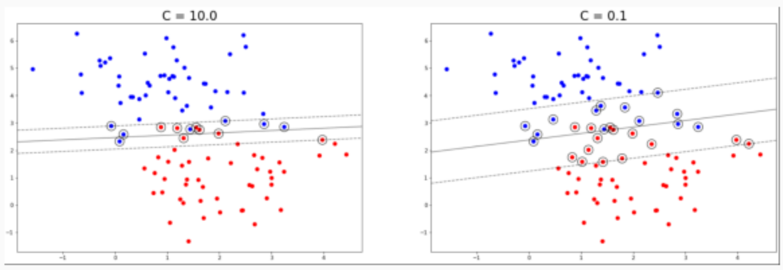
$$\frac{\partial L}{\partial \xi_i} = 0 \Rightarrow C - \alpha_i - \gamma_i = 0 \Rightarrow 0 \leq \alpha_i \leq C$$

➢ 利用SMO算法求解:

$$\text{maximize} \quad \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j$$

$$0 \leq \alpha_i \leq C; \quad \sum_{i=1}^{n}\alpha_i y_i = 0$$

- 正则化参数C允许模型有误差，用于控制允许部分样本位于间隔区间
  - C越大，即要求模型的误差越小，进入间隔区间的点越少，容易过拟合
  - C越小，即模型的误差越大，进入间隔区间的点越多，训练集上容易出现欠拟合

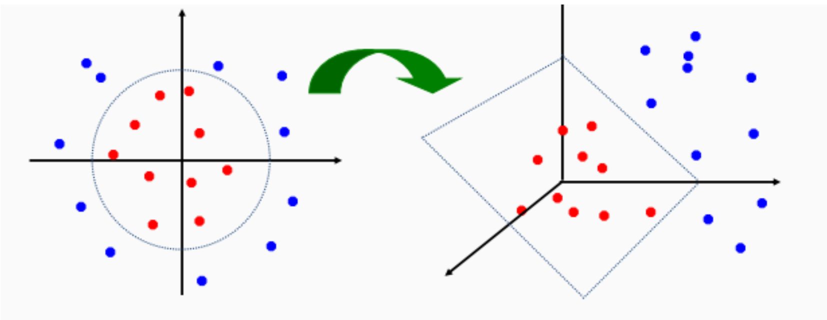


27

求解表达式是一样的，无非是要求支持向量的$0 \leq \alpha_i \leq C$。

- 线性不可分问题：

  将低维样本$\mathbf{x}_i$映射到高维空间：$\phi(\mathbf{x}_i)$，使之线性可分。



此时在高维空间中两个样本的内积：

$$\mathbf{x}_i^T\mathbf{x}_j \to \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$$

求映射ϕ不易，但由上边可知最后的判决函数（$y=\sum_{i=SV}\alpha_i y_i \mathbf{x}_i^T\mathbf{x}+b$）里只需求出两个样本的内积即可，因此定义核函数为高维空间中两个样本的内积：

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

这样只需使用核函数即可得到判决函数，不用具体求ϕ。

常用的核函数：

➢ 常用核函数：

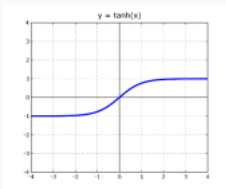
- Linear kernel: $K(\mathbf{x}_i,\mathbf{x}_j)=\mathbf{x}_i^T\mathbf{x}_j$
- Polynomial kernel: $K(\mathbf{x}_i,\mathbf{x}_j)=(1+\mathbf{x}_i^T\mathbf{x}_j)^p$
- Gaussian (Radial-Basis Function (RBF)) kernel:

$$K(\mathbf{x}_i,\mathbf{x}_j)=\exp\left(-\frac{\left\|\mathbf{x}_i-\mathbf{x}_j\right\|^2}{2\sigma^2}\right)$$

- Sigmoid型:

$$K(\mathbf{x}_i,\mathbf{x}_j)=\tanh\left(\beta_0\mathbf{x}_i^T\mathbf{x}_j+\beta_1\right)$$

$$y=\tanh(x)=\frac{e^x-e^{-x}}{e^x+e^{-x}}$$



在高维空间中的待优化函数即为将原先的$\mathbf{x}_i^T\mathbf{x}_j$换成$K(\mathbf{x}_i,\mathbf{x}_j)$：

$$L=\sum_{i=1}^n\alpha_i-\frac{1}{2}\sum_{i=1}^n\sum_{j=1}^n\alpha_i\alpha_j y_i y_j K(\mathbf{x}_i,\mathbf{x}_j)$$

在高维空间中求得的最优判决超平面的权重即为将原先的$\mathbf{x}_i$换成$\phi(\mathbf{x}_i)$，判决超平面方程即为将原先的$\mathbf{x}_i^T\mathbf{x}_j$换成$K(\mathbf{x}_i,\mathbf{x}_j)$：

$$w=\sum_{i\in SV}\alpha_i y_i\phi(\mathbf{x}_i)$$

$$y=\sum_{i\in SV}\alpha_i y_i K(\mathbf{x}_i,\mathbf{x}_j)+b$$

eg1：

例：利用非线性SVM求解XOR问题

输入向量x	二维向量	期望输出 y
x_1	$(-1,-1)^T$	-1
x_2	$(-1,1)^T$	1
x_3	$(1,-1)^T$	1
x_4	$(1,1)^T$	-1

1. 核函数

输入向量x	二维向量	期望输出 y
x_1	$(-1,-1)^T$	-1
x_2	$(-1,1)^T$	1
x_3	$(1,-1)^T$	1
x_4	$(1,1)^T$	-1

二维平面向量

坐标值标量

核函数 $K(x_i,x_j)=(1+x_i^Tx_j)^2$， $x_i=(x_{i1},x_{i2})^T$

$$K(x_i,x_j)=1+x_{i1}^2x_{j1}^2+2x_{i1}x_{i2}x_{j1}x_{j2}+x_{i2}^2x_{j2}^2+2x_{i1}\,x_{j1}+2x_{i2}x_{j2}$$

核函数的矩阵形式

$$K=\begin{bmatrix}K(x_1,x_1) & K(x_1,x_2) & K(x_1,x_3) & K(x_1,x_4)\\ K(x_2,x_1) & K(x_2,x_2) & K(x_2,x_3) & K(x_2,x_4)\\ K(x_3,x_1) & K(x_3,x_2) & K(x_3,x_3) & K(x_3,x_4)\\ K(x_4,x_1) & K(x_4,x_2) & K(x_4,x_3) & K(x_4,x_4)\end{bmatrix}$$

$$=\begin{bmatrix}9 & 1 & 1 & 1\\ 1 & 9 & 1 & 1\\ 1 & 1 & 9 & 1\\ 1 & 1 & 1 & 9\end{bmatrix}$$

33

这里选取了二次多项式核函数：$K(\mathbf{x}_i,\mathbf{x}_j)=(1+\mathbf{x}_i^T\mathbf{x}_j)^2$，将其展开可得用$\mathbf{x}_i=(x_{i1},x_{i2})$这些分量的表达式。将每组样本两两之间的核函数计算后可得矩阵K，方便下边使用。

K矩阵是半正定的

通过核函数的展开式，其实也可以求出来ϕ，将它分解成x_1、x_2分别的部分即可。

2. 基函数

输入向量x	二维向量	期望输出 y
x_1	$(-1,-1)^T$	-1
x_2	$(-1,1)^T$	1
x_3	$(1,-1)^T$	1
x_4	$(1,1)^T$	-1

在本例题中，根据核函数的展开式，可以得到基函数，也就是二维输入向量在六维空间中的映射

$$\phi(x_i)=\left[1,x_{i1}^2,\sqrt{2}x_{i1}x_{i2},x_{i2}^2,\sqrt{2}x_{i1},\sqrt{2}x_{i2}\right]^T$$

分别计算出每个样本映射到六维空间中的向量

$$\phi(x_1)=\left[1,1,\sqrt{2},1,-\sqrt{2},-\sqrt{2}\right]^T$$

$$\phi(x_2)=\left[1,1,-\sqrt{2},1,-\sqrt{2},\sqrt{2}\right]^T$$

$$\phi(x_3)=\left[1,1,-\sqrt{2},1,\sqrt{2},-\sqrt{2}\right]^T$$

$$\phi(x_4)=\left[1,1,\sqrt{2},1,\sqrt{2},\sqrt{2}\right]^T$$

这样可以具体求得每个样本在高维空间的映射向量。

然后求解使目标函数最大的α：

$$L=\sum_{i=1}^{4}\alpha_i-\frac{1}{2}\sum_{i=1}^{4}\sum_{j=1}^{4}\alpha_i\alpha_j y_i y_j K(\mathbf{x}_i,\mathbf{x}_j)$$

将其分别对α_i求导，列出：$\frac{\partial L}{\partial \alpha_1}=0,\ \frac{\partial L}{\partial \alpha_2}=0,\cdots$，整理得：

$$\begin{cases} 9\alpha_1-\alpha_2-\alpha_3+\alpha_4=1 \\ -\alpha_1+9\alpha_2+\alpha_3-\alpha_4=1 \\ -\alpha_1+\alpha_2+9\alpha_3-\alpha_4=1 \\ \alpha_1-\alpha_2-\alpha_3+9\alpha_4=1 \end{cases}$$

求解所有α_i：

求解得到拉格朗日系数的最优值为：$\alpha_1=\alpha_2=\alpha_3=\alpha_4=\frac{1}{8}$

说明此例中的四个样本都是支持向量

根据w的计算公式 $w=\sum_{i=1}^{N}\alpha_i y_i \phi(x_1)$，得到权值向量 $w=[0,0,-1/\sqrt{2},0,0,0]^T$

取一个样本数据，比如x_1，根据$y=w^T\phi(x_1)+\mathrm{b}$，

即 $-1=\frac{-1}{\sqrt{2}}\cdot\sqrt{2}+b$，求得偏置量$b=0$

然后即可按照公式求出高维空间最优判决超平面的权重和偏置。

根据最优分类超平面的定义 $g(x)=w^T\phi(x)+\mathrm{b}=0$

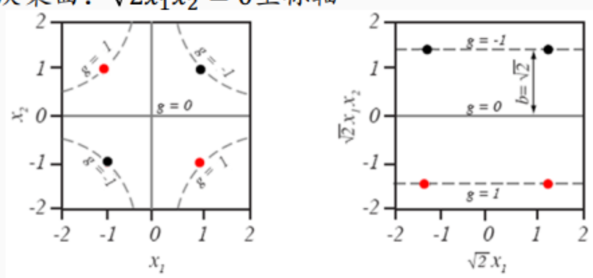
对于输入 $x=(x_1,x_2)^T$，　$\phi(x)=[1,x_1^2,\sqrt{2}x_1x_2,x_2^2,\sqrt{2}x_1,\sqrt{2}x_2]^T$

$w=[0,0,-1/\sqrt{2},0,0,0]^T$

计算得到决策面$g(x)=0$为：$-\sqrt{2}x_1x_2+0=0$，即$x_1x_2=0$

输入数据空间中的决策面：　$x_1=0$ 或 $x_2=0$ 两个坐标轴

高维空间中的决策面：$\sqrt{2}x_1x_2=0$坐标轴



原始数据空间非线性分类　　对应高维空间中的线性分类

最终再代入回判决函数$g(\mathbf{x})=w^T\phi(\mathbf{x})+b$，代入低维分类，即可得到用$\mathbf{x}=(x_1,x_2)$低维坐标分量表示的低维分类界面：$x_1x_2=0$，也即决策面体现在低维是坐标轴。

eg2：

4. 在特征空间 $D=(x_1, x_2)$中，给定 6 个训练样本如下：

数据	类别标签
$D_1=(1,1)^T$	1
$D_2=(2,1)^T$	1
$D_3=(2,0)^T$	1
$D_4=(1,2)^T$	-1
$D_5=(2,2)^T$	-1
$D_6=(1,-3)^T$	-1

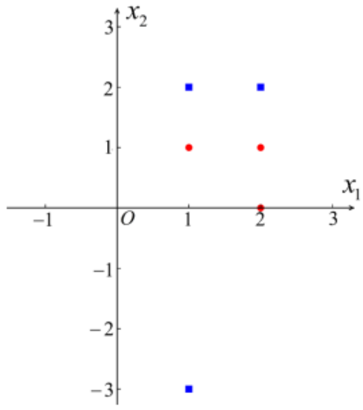


图 1

用如下的非线性变换，先将输入的样本 $D=(x_1,x_2)$ 变换为向量 $Z=(\phi_1(D),\phi_2(D))$，其中：

$\phi_1(D)=x_1^2+x_2^2$，$\phi_2(D)=x_1-x_2$。

(1). 写出非线性映射后的数据。

(2). 在映射后的空间中，画出映射后的数据，以及利用支持向量机设计的分类界面示意图。

(3). 在原始特征空间中，画出分类界面示意图。

（1）

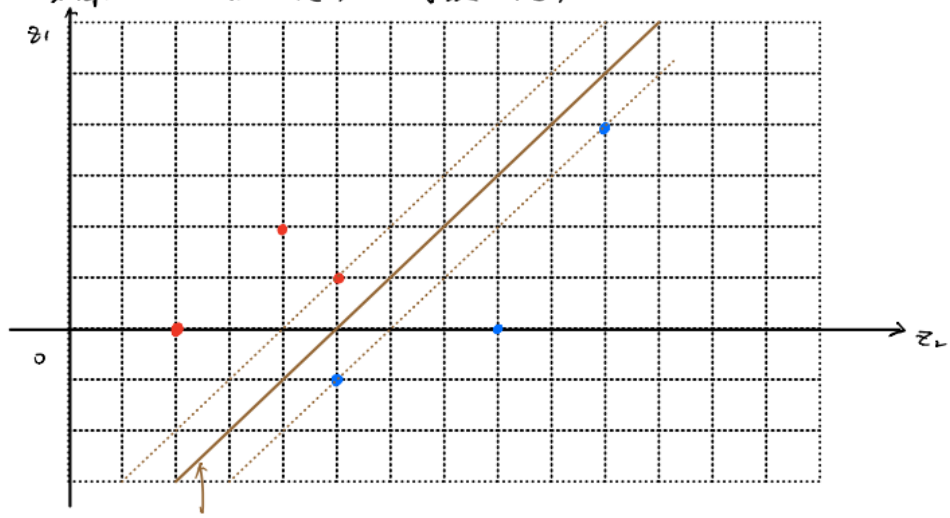
代入计算即可得到各点的变换结果：

4. Ans.

		类别
(1) $D_1=(1,1)^T \Rightarrow$	$Z=(2,0)$	1
$D_2=(2,1)^T \Rightarrow$	$Z=(5,1)$	1
$D_3=(2,0)^T \Rightarrow$	$Z=(4,2)$	1
$D_4=(1,2)^T \Rightarrow$	$Z=(5,-1)$	-1
$D_5=(2,2)^T \Rightarrow$	$Z=(8,0)$	-1
$D_6=(1,-3)^T \Rightarrow$	$Z=(10,+4)$	-1

（2）

(2) 如图，"●"代表+1类，"●"代表-1类.



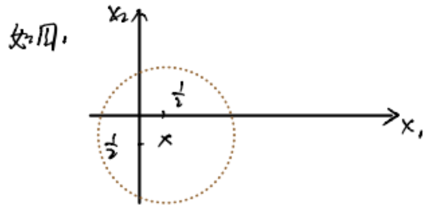
分界线：$z_2=z_1-5$

（3）

将高维空间中的线性分类界面按照变换公式还原成原空间的分界面，由于：$(z_1, z_2) = (x_1^2 + x_2^2, x_1 - x_2)$：

$$z_2 = z_1 - 5$$
$$\Downarrow$$
$$x_1 - x_2 = x_1^2 + x_2^2 - 5 \Rightarrow (x_1 - 1/2)^2 + (x_2 + 1/2)^2 = 11/2$$

可见原空间中的分类界面为一个圆：



6. 隐马尔可夫模型

设o_t是t时刻的观测值，q_t是t时刻的状态值，S_i表示第i种状态值。

模型用三元组λ＝（π，A，B）用来描述

参数	含义	实例
A	与时间无关的状态转移概率矩阵	类间转移概率
B	给定状态下，观察值概率分布	给定类别，特征向量分布
π	初始状态空间的概率分布	初始时选择类别的概率

- 评估问题

  给定模型$\lambda = (\pi, A, B)$，计算观测序列$O = \{o_1, o_2, \cdots, o_T\}$的概率$P(O|\lambda)$。

  设$b_i(o_j)$表示状态处于S_i时，观测到观测值o_j的概率，设共有N种状态$S_1, \cdots S_N$。a_{ij}为状态转移矩阵的矩阵元。

  - 前向算法：

    引入前向辅助变量：

    $$\alpha_t(i) = P(o_1, \cdots, o_t, q_t = S_i|\lambda)$$

    这里i代表第i个状态，$\alpha_t(i)$即为状态S_i在t时刻的前向辅助变量。

    初始化，分别求出第1个时刻每个状态的前向变量：

    $$\alpha_1(i) = \pi_i b_i(o_1), \quad 1 \le i \le N$$

    递归求得所有时刻的$\alpha_t(j)$：

    $$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(o_{t+1})$$

    算法终止时，求得$\alpha_T(1) \sim \alpha_T(N)$，则目标结果为：

    $$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

- 后向算法：

  引入后向辅助变量：

  $$\beta_t(i) = P(o_{t+1}, \cdots, o_T | q_t = S_i, \lambda)$$

  初始化，分别求出第1个时刻每个状态的后向变量：

  $$\beta_T(i) = 1,\ \ 1 \leq i \leq N$$

  递归：

  $$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)$$

  算法终止时，求得$\beta_1(1) \sim \beta_1(N)$，则目标结果为：

  $$P(O|\lambda) = \sum_{i=1}^{N} \beta_1(i) \pi_i b_i(o_1)$$

- 前向-后向算法：

  利用前向、后向变量计算

  $$\begin{aligned} & P(O, q_t = S_i \mid \lambda) \\ & = P(O_1, O_2, \cdots O_t, q_t = S_i, O_{t+1}, O_{t+2}, \cdots O_T \mid \lambda) \\ & = P(O_1, O_2, \cdots O_t, q_t = S_i \mid \lambda) P(O_{t+1}, O_{t+2}, \cdots O_T \mid q_t = S_i, \lambda) \\ & = \alpha_t(i) \beta_t(i) \end{aligned}$$

  $$P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_t(i) * \beta_t(i) \qquad 1 \leq t \leq T$$

- 解码问题

  给定模型$\lambda = (\pi, A, B)$和观测序列$O = \{o_1, \cdots, o_t\}$，求最可能的状态序列。

  Viterbi算法：使每一时刻状态序列出现相应观测值的可能达到最大。

  定义：

  $$\delta_t(i) = \max_{q} P(q_1, q_2, \cdots, q_t = i, o_1, \cdots, o_t | \lambda)$$

  Viterbi算法流程：

  初始化：

  $$\delta_1(i) = \pi_i b_i(o_1)$$
  $$\varphi_1(i) = 0$$

  递归：

  $$\delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] b_j(o_t)$$
  $$\varphi_t(j) = \arg\max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}]$$

其中$\delta_t(i)$相当于t时刻前向变量中的最大值，$\varphi_t(i)$相当于记录t时刻状态i对应的上一时刻的最优状态。

终止：

$$P^* = \max_{1 \le i \le N}[\delta_T(i)]$$
$$q_T^* = \arg\max_{1 \le i \le N}[\delta_T(i)]$$

其中q_T即为最后一个时刻的最优状态，从q_T开始回溯，即可依次得到每个时刻的最优状态：

$$q_t^* = \varphi_{t+1}(q_{t+1}^*)$$

eg1：

2.1 隐含马尔可夫模型的解码

某手机专卖店今年元旦新开业，每月上旬进货时，由专卖店经理决策，采用三种进货方案中的一种：高档手机 (H)，中档手机 (M)，低档手机 (L)。

当月市场行情假设分为畅销 (S_1) 和滞销 (S_2) 两种。畅销时，三种进货方案的概率分别为 0.4, 0.4, 0.2；滞销时，三种进货方案的概率分别为 0.2, 0.3, 0.5。

某月份市场行情为畅销，下一个月份为畅销和滞销的概率分别为 0.6 和 0.4；某月份市场行情为滞销，下一个月份为畅销和滞销的概率分别为 0.5 和 0.5。

开业第一个月市场行情为畅销和滞销的可能性均为 0.5。

(1) 如果我们采用隐含马尔可夫模型 (HMM) 对该专卖店进货环节建模，请写出 HMM 对应的参数 $\lambda = \{\pi, A, B\}$。由于 A 为状态转移概率矩阵，B 为给定状态下观察值的概率分布，π 为初始状态空间的概率分布，因此：

$$\pi = (0.5, 0.5)$$

$$A = \begin{bmatrix} 0.6 & 0.4 \\ 0.5 & 0.5 \end{bmatrix}$$

$$B = \begin{cases} S_1 : (0.4, 0.4, 0.2) \\ S_2 : (0.2, 0.3, 0.5) \end{cases}$$

(2) 在第一季度中，采购业务员执行的进货方案为“高档手机，中档手机，低档手机”，即观测序列为 H, M, L。请利用 Viterbi 算法推测前三个月的市场行情。

初始化：

$$\delta_1(1) = \pi_1 b_1(O_1) = 0.5 \times 0.4 = 0.2$$
$$\delta_1(2) = \pi_2 b_2(O_1) = 0.5 \times 0.2 = 0.1$$
$$\varphi_1(1) = 0$$
$$\varphi_1(2) = 0$$

递归：

$t = 2$ 时：

$$\delta_2(1) = \max_{1\le i\le 2}[\delta_1(i)a_{i1}]b_1(O_2) = 0.12 \times 0.4 = 0.048$$
$$\delta_2(2) = \max_{1\le i\le 2}[\delta_1(i)a_{i2}]b_2(O_2) = 0.08 \times 0.3 = 0.024$$
$$\varphi_2(1) = \arg\max_{1\le i\le 2}[\delta_1(i)a_{i1}] = 1$$
$$\varphi_2(2) = \arg\max_{1\le i\le 2}[\delta_1(i)a_{i2}] = 1$$

$t = 3$ 时：

$$\delta_3(1) = \max_{1\le i\le 2}[\delta_2(i)a_{i1}]b_1(O_3) = 0.0288 \times 0.2 = 0.00576$$
$$\delta_3(2) = \max_{1\le i\le 2}[\delta_2(i)a_{i2}]b_2(O_3) = 0.0192 \times 0.5 = 0.0096$$
$$\varphi_3(1) = \arg\max_{1\le i\le 2}[\delta_2(i)a_{i1}] = 1$$
$$\varphi_3(2) = \arg\max_{1\le i\le 2}[\delta_2(i)a_{i2}] = 1$$

终止：

$$P^* = \max_{1 \le i \le 2} [\delta_3(i)] = 0.0096$$

$$q_3^* = \arg \max_{1 \le i \le 2} [\delta_3(i)] = 2$$

回溯：

$$q_2^* = \varphi_3(q_3^*) = \varphi_3(2) = 1$$

$$q_1^* = \varphi_2(q_2^*) = \varphi_2(1) = 1$$

综上，最可能的序列为：S_1, S_1, S_2

7. 序列建模

- RNN

  设输入的数据为一个离散的序列：$x_0, x_1, \cdots, x_{t-1}, x_t, x_{t+1}, \cdots$。$x_t$表示$t$时刻输入模型的样本。$h_t$表示t时刻的状态变量（或者叫隐藏状态），其在0时刻初始化为h_0。

  t时刻输入的不仅是x_t，还有上一时刻的输出状态h_{t-1}。它们共同输入，并输入本时刻的输出状态h_t与本时刻输出结果y_t。其中h_t会翻回头与x_{t+1}输入模型，来预测下一次的输出。

  通过以下公式来对状态h进行更新：

$$h_t = f(h_{t-1}, x_t)$$

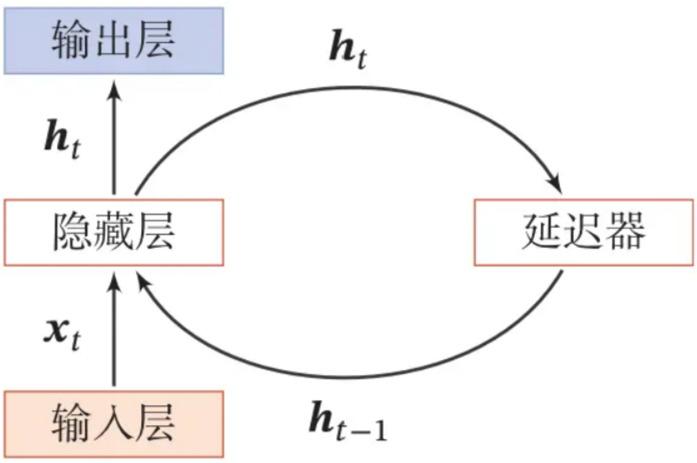
  具体而言，更新与输出公式如下：

$$h_t = \sigma\left(W^{(hh)} h_{t-1} + W^{(hx)} x_t\right)$$

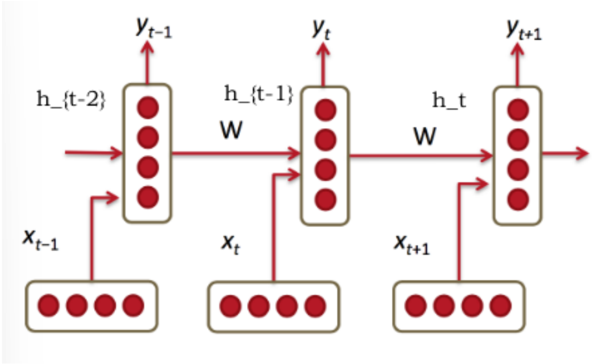
$$y_t = Softmax\left(W^{(S)} h_t\right)$$

  其中三个W均为可学习权重参数（若为inference模式，则从始至终它们都不变）。

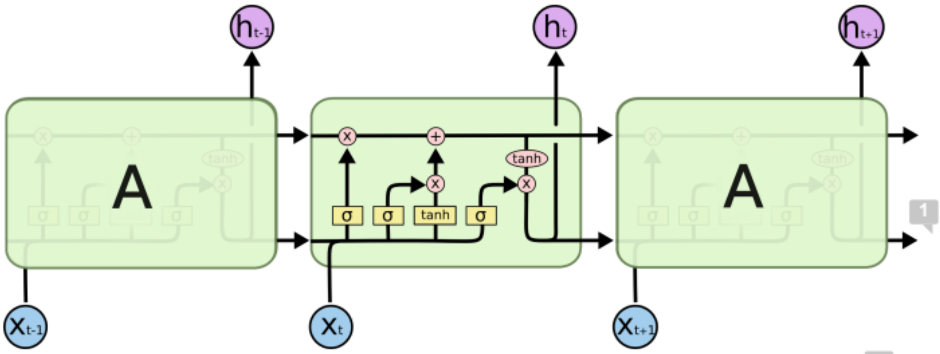
  直接来看：



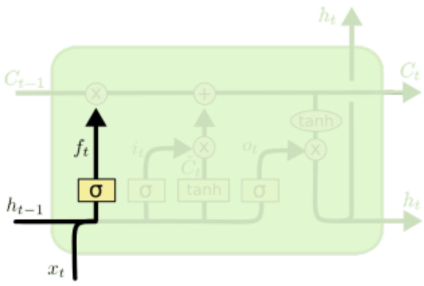
  按时序从左到右展开来看：



- LSTM



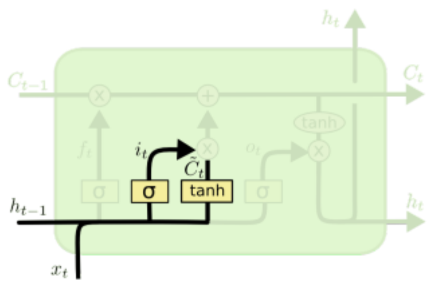
  - 第一步：决定上一步状态有多少比例被允许通过（**Forget Gate：** f_t ）

    $$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$

    这里生成的f_t即为控制上一步状态流动的控制信号。

  - 第二步：决定本时刻要储存的cell state（**Input Gate：** i_t ）

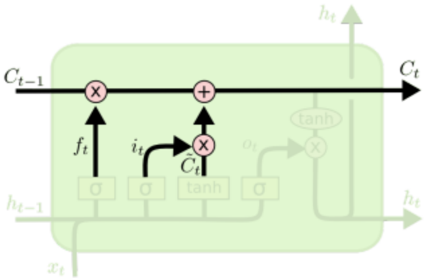
    

    $$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$
    $$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

    其中，σ是"input gate layer"，它生成一个控制信号（i_t）来决定本步状态有多少比例被保存。$\tanh$则产生本时刻的cell state（$\tilde{C}_t$）

  - 第三步：更新cell state

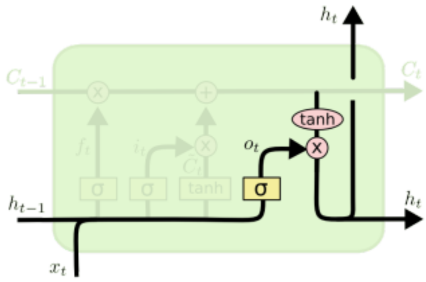
    将上一步的cell state和本步的cell state分别乘以它们对应的控制信号，然后相加即可组成本步输出的新cell state：



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

- 第四步：产生输出（**Output Gate：** o_t ）

  输出h_t是cell state（C_t）和本时刻输入与hidden state（h_{t-1}）的综合结果：



$$o_t = \sigma\left(W_o\ [h_{t-1}, x_t]\ +\ b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

  其中，o_t是一个gate产生的控制信号，其决定cell state在多大程度上参与本时刻的输出。