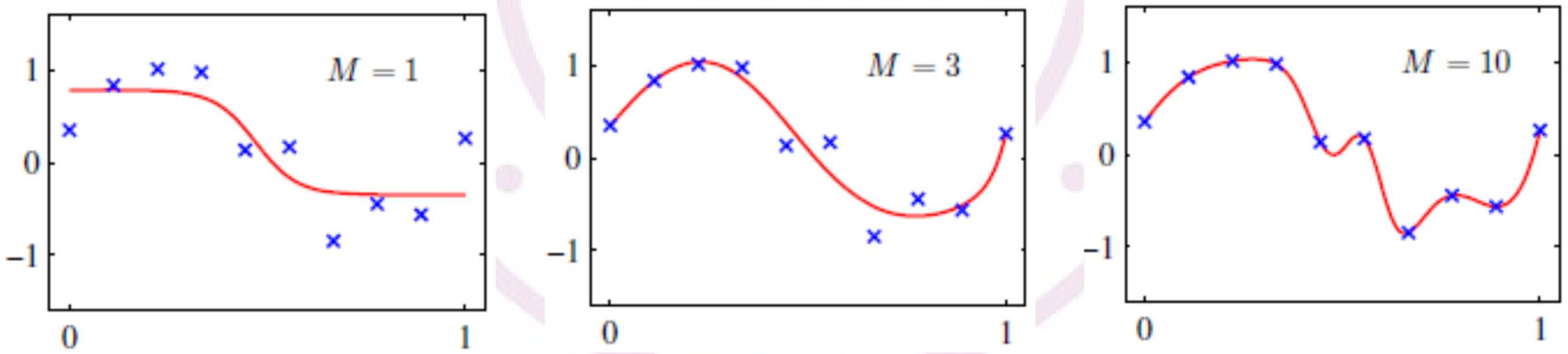
神经网络(2)

南开大学网络空间安全学院 计算机学院

正则化

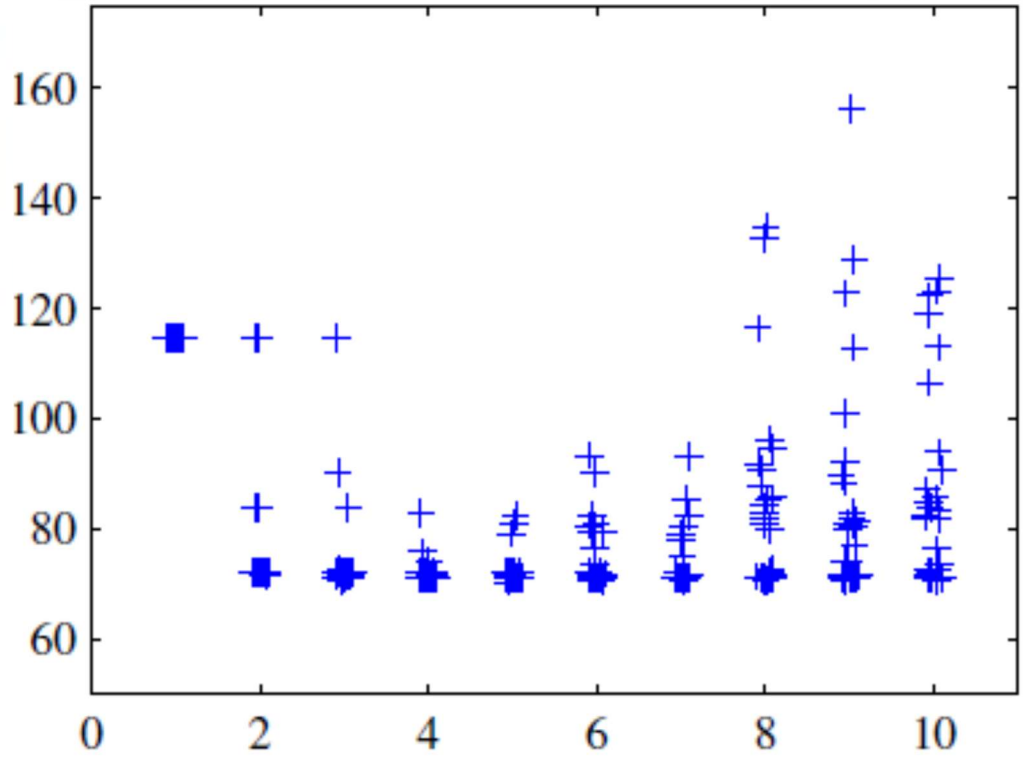
- 网络结构
  - 输入单元和输出单元数目由数据集的维度决定
  - 隐含单元数目 M 是可调的自由参数，由最好泛化性能决定。
    - 泛化性能(generalization performance)：欠拟合(under-fitting)和过拟合(over-fitting)之间的平衡

- 示例：正弦曲线回归问题中不同 M 取值的作用



正则化

- 由于误差函数中局部最小值的存在，泛化误差与 M 之间不存在简单的函数关系。
  - 纵坐标：多项式数据集测试集平方和误差
  - 横坐标：隐含单元数目
  - 结果统计意义：每个隐含单元数目以随机初始化参数重复30次



正则化

- 方法：选择相对大的 M 值，然后通过给误差函数添加正则项来控制模型复杂度。
  - 二次正则项(regularization term)，正则项也称为权重衰减(weight decay)

$$\tilde{E}(\mathbf{w}) = E(\mathbf{w}) + \frac{\lambda}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}$$

  - 模型有效复杂度由正则化系数 λ 决定
  - 正则项是权值矢量 w 的零均值高斯先验分布的负对数。

一致高斯先验概率

- 简单的权值衰减方法存在局限性
- 例：多层感知器网络（两层权值，线性输出单元）完成输入变量集合$\{x_i\}$到输出变量集合$\{y_k\}$的映射。
  - 第一个隐含层中隐含单元的激活值

$$z_j = h\left(\sum_i w_{ji} x_i + w_{j0}\right)$$

  - 输出单元的激活值

$$y_k = \sum_j w_{kj} z_j + w_{k0}$$

  - 假设，对输入数据进行线性变换

$$x_i \to \tilde{x}_i = a x_i + b$$

一致高斯先验概率

- 通过对从输入到隐含单元的权值和偏差做相应的线性变换，可以让网络完成同样的映射

$$w_{ji} \to \tilde{w}_{ji} = \frac{1}{a} w_{ji}$$

$$w_{j0} \to \tilde{w}_{j0} = w_{j0} - \frac{b}{a} \sum_i w_{ji}$$

- 相似地，输出变量的线性变换

$$y_k \to \tilde{y}_k = c y_k + d$$

- 第二层单元权值和偏差的线性变换

$$w_{kj} \to \tilde{w}_{kj} = c w_{kj}$$

$$w_{k0} \to \tilde{w}_{k0} = c w_{k0} + d$$

一致高斯先验概率

- 分别使用原始数据和线性变换数据训练得到两个网络
  - 一致性要求两个等价网络，差异只是权值的线性变换。
  - 任何正则化方法都应该满足这个性质，否则其将对解有所偏爱。
  - 但，权值衰减方法“平等地”对待权值和偏差，不满足这个性质。
- 对线性变换具有不变性的正则化公式
  - 对权值缩放和偏差平移具有不变性

$$\frac{\lambda_1}{2}\sum_{w\in\mathcal{W}_1} w^2 + \frac{\lambda_2}{2}\sum_{w\in\mathcal{W}_2} w^2$$

其中 $\mathcal{W}_1, \mathcal{W}_2$ 分别表示第一层和第二层权值集合，不包括偏差。且在正则化参数如下缩放时权值变换不改变正则项

$$\lambda_1 \to a^{1/2}\lambda_1,\ \lambda_2 \to c^{-1/2}\lambda_2$$

一致高斯先验概率

- 不变性正则化对应的先验概率

$$p\left(\mathbf{w}\middle|\alpha_1,\alpha_2\right)\propto\exp\left(-\frac{\alpha_1}{2}\sum_{w\in\mathcal{W}_1}w^2-\frac{\alpha_2}{2}\sum_{w\in\mathcal{W}_2}w^2\right)$$

注：因为偏差参数无约束，故先验概率是非归一化的；通常对偏差使用分开的具有独立超参数的先验概率。

- 将权值任意分组到$\mathcal{W}_k$的先验概率

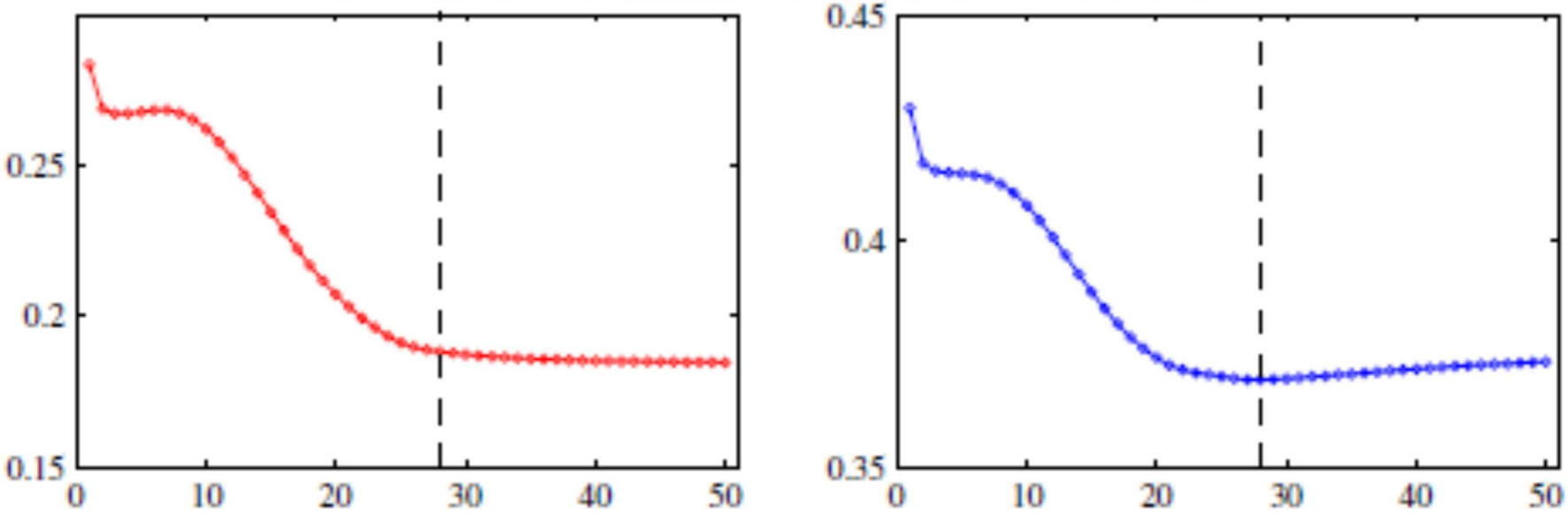
$$p\left(\mathbf{w}\right)\propto\exp\left(-\frac{1}{2}\sum_k\alpha_k\left\|\mathbf{w}\right\|_k^2\right)$$

其中

$$\left\|\mathbf{w}\right\|_k^2=\sum_{j\in\mathcal{W}_k}w_j^2$$

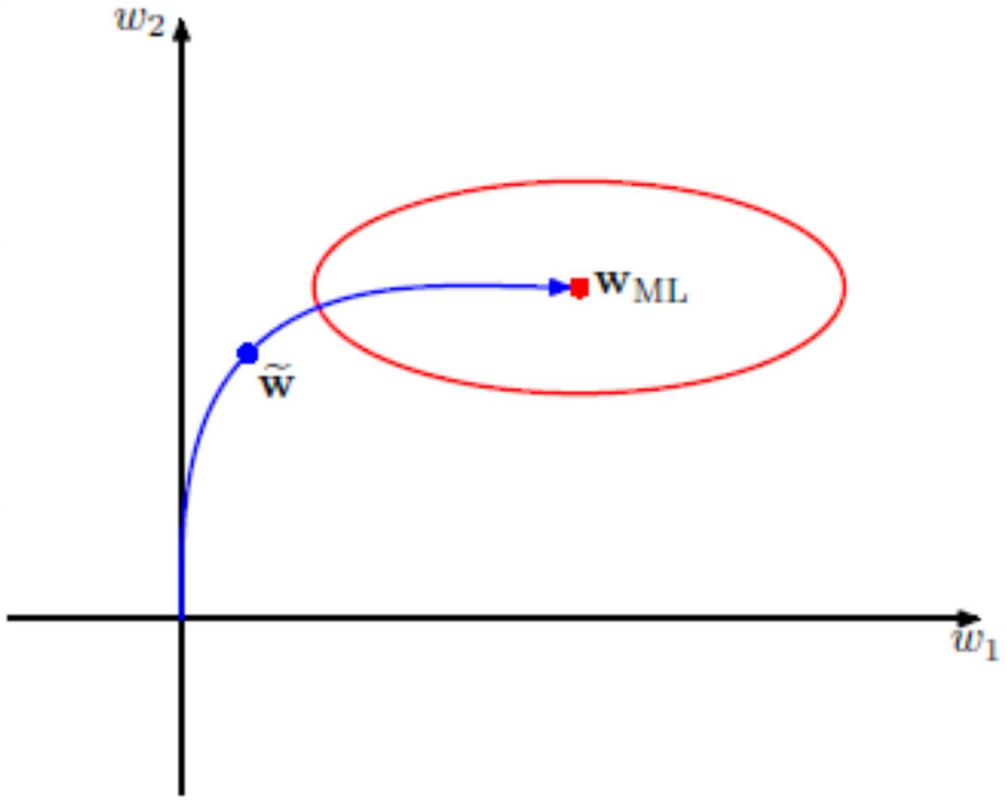
提前停止

- 提前停止(early stopping)：控制网络有效复杂度的方法
- 网络训练优化算法
  - 误差是迭代次数的非增函数
  - 对验证集，误差一般先降后增，转折点是过拟合(over-fit)的标志；
  - 训练过程应该在验证误差最小时停止



提前停止

- 当使用二次误差函数时，提前停止表现出与使用简单权值衰减正则化相似的行为。
  - 在没有权值衰减的训练过程中，权值将沿着局部负梯度矢量方向移动，通过大致对应 $\tilde{\mathbf{w}}$ 的点，最终到达误差函数最小值点 $\mathbf{w}_{\mathrm{ML}}$；
  - 提前在 $\tilde{\mathbf{w}}$ 点停止，则相似于权值衰减
  - $\tau\eta$ 扮演着 λ 倒数的角色



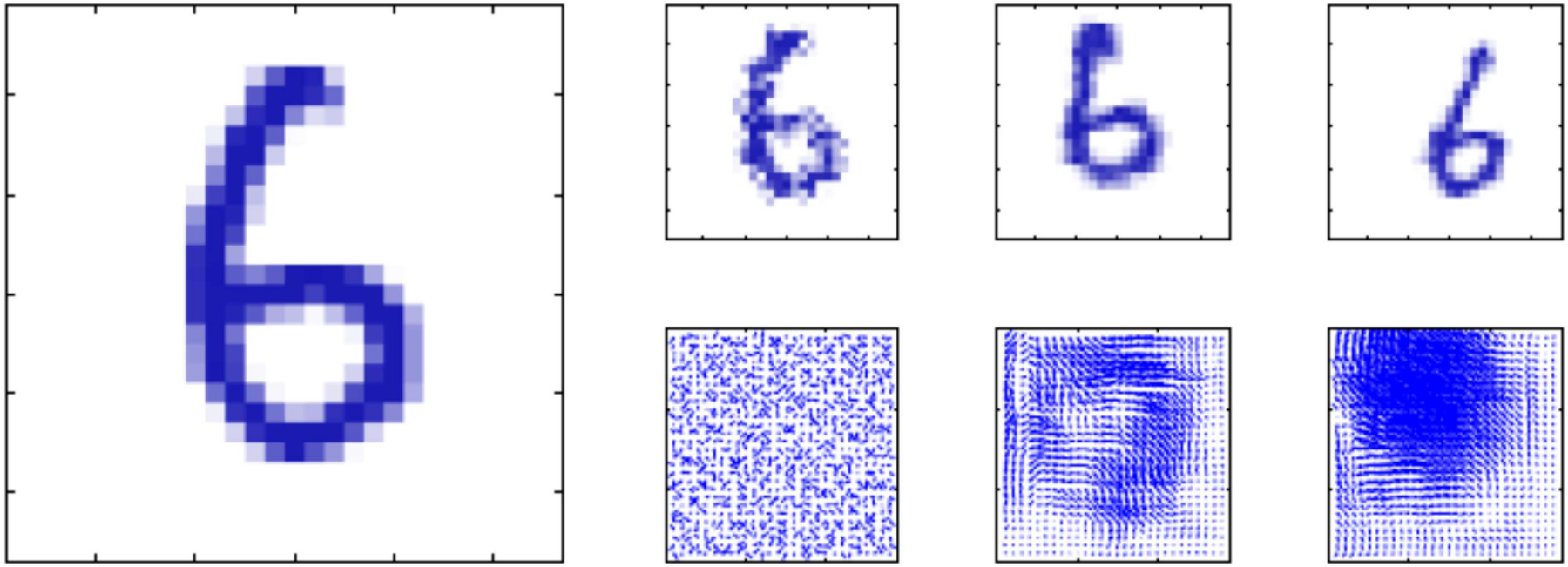
不变性

- 在对输入变量进行变换的情况下，预测结果应该是不变的。
  - 在物体分类问题中，将物体在图像中进行平移和缩放变换，物体的类别标签不变。

- 自适应模型获得不变性的几种方法
  - 根据所需不变性获得训练样本的变换副本，添加到训练集；
  - 通过包含正则项的误差函数，惩罚由于输入变换所带来的输出变化，称为切线传播(tangent propagation)；
  - 在预处理中，抽取具有变换不变性的特征；
  - 将不变性构建到神经网络结构中，如局部感受野和共享权值。

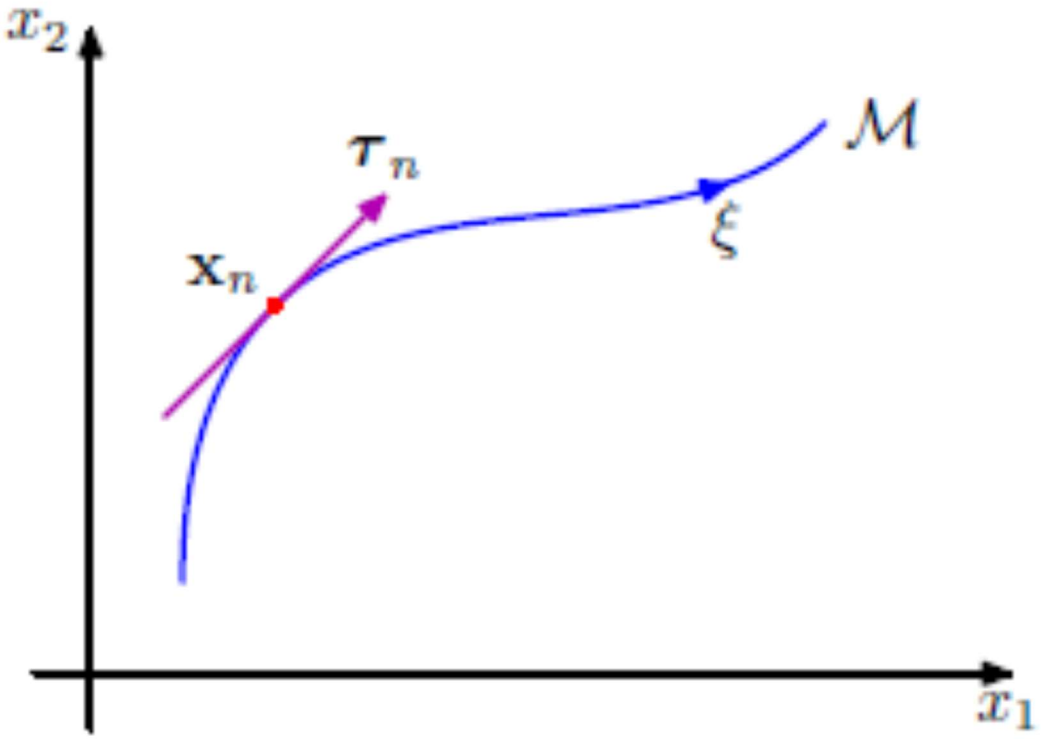
不变性

- 训练样本的变换不变性副本
  - 随机采样位移矢量 $\Delta x, \Delta y \in (0,1)$ ， 高斯卷积平滑(宽度： 0.01, 30, 60)



切线传播

- 切线传播(tangent propagation)：使用正则化来鼓励模型对输入变换保持不变的技术。
- 假设变换决定于单个参数 ξ（如旋转角度），输入矢量 $\mathbf{x}_n$ 扫过子空间 $\mathcal{M}$ 是一维的，且被 ξ 参数化。



切线传播

- 这个变换作用在 $\mathbf{x}_n$ 上得到的矢量表示为 $\mathbf{s}(\mathbf{x}_n, \xi)$，有 $\mathbf{s}(\mathbf{x}, 0) = \mathbf{x}$；
- 曲线 $\mathcal{M}$ 的切线由方向导数 $\tau = \partial \mathbf{s} / \partial \xi$ 给出，且点 $\mathbf{x}_n$ 处的切矢量为

$$\tau_n = \left. \frac{\partial \mathbf{s}(\mathbf{x}_n, \xi)}{\partial \xi} \right|_{\xi=0}$$

- 在输入矢量变换下，一般网络的输出矢量将改变，输出对参数 ξ 的导数为

$$\left. \frac{\partial y_k}{\partial \xi} \right|_{\xi=0} = \sum_{i=1}^{D} \frac{\partial y_k}{\partial x_i} \left. \frac{\partial x_i}{\partial \xi} \right|_{\xi=0} = \sum_{i=1}^{D} J_{ki} \tau_i$$

- 误差函数

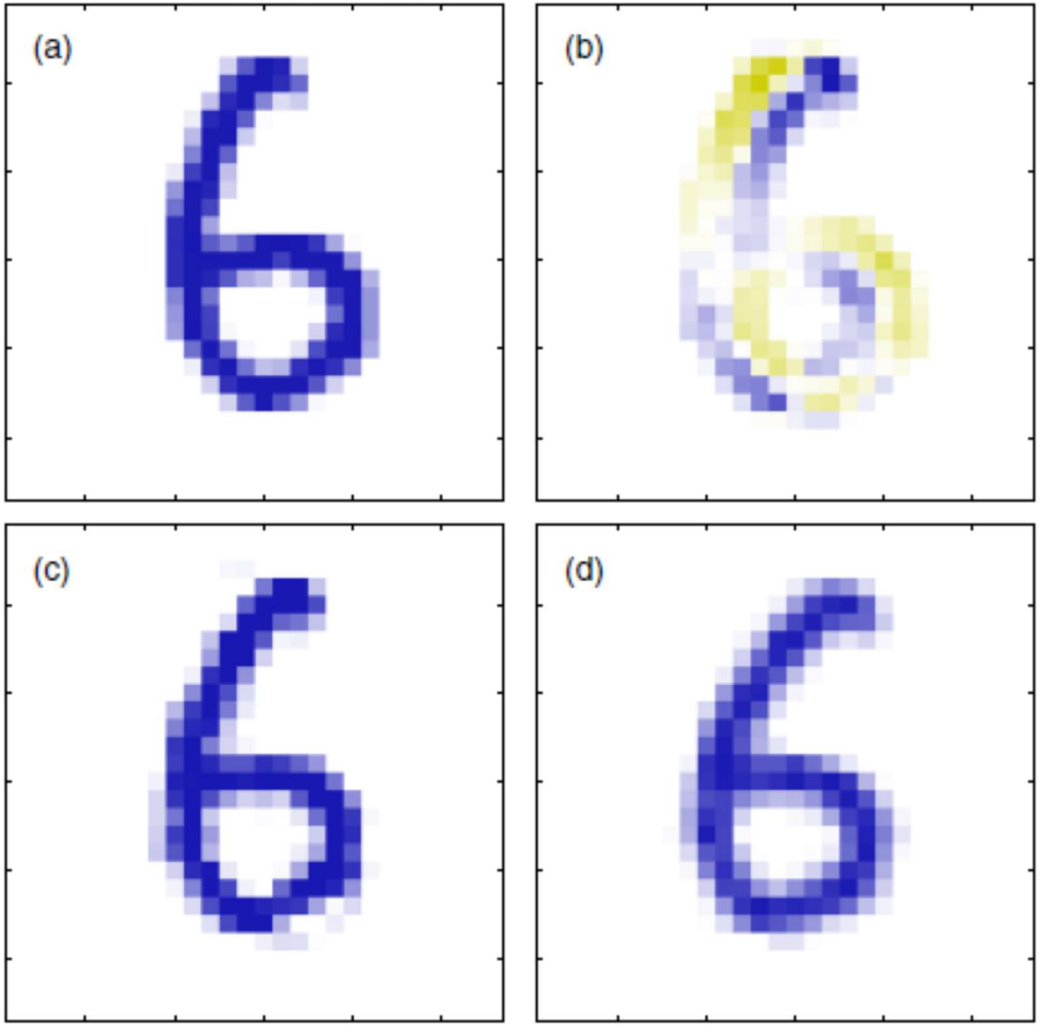
$$\tilde{E} = E + \lambda \Omega$$

其中 λ 是正则化系数，且

$$\Omega = \frac{1}{2} \sum_n \sum_k \left(\left. \frac{\partial y_{nk}}{\partial \xi} \right|_{\xi=0} \right)^2 = \frac{1}{2} \sum_n \sum_k \left(\sum_{i=1}^{D} J_{nki} \tau_{ni} \right)^2$$

切线传播

- 在实际应用中，切矢量 τ_n 可以使用有限差分近似。
- 如果变换由 L 个参数决定，那么流形 $\mathcal{M}$ 是 L 维的，对应的正则化项由求和项形式给出，每个对应一个变换。
- 例：
  - 原始手写数字图像
  - 微小顺时针旋转的切矢量
  - $\mathbf{x}+\varepsilon\tau,\ \varepsilon=15°$
  - 真实旋转图像



变换数据训练

- 单个参数 ξ 控制的变换函数 $\mathbf{s}(\mathbf{x},\xi)$，有 $\mathbf{s}(\mathbf{x},0)=\mathbf{x}$。
- 无变换输入（无限数据集）的误差函数

$$E=\frac{1}{2}\int\int\left\{y(\mathbf{x})-t\right\}^2 p(t|\mathbf{x})p(\mathbf{x})d\mathbf{x}dt$$

- 被变换扰动的无穷个数据点副本，变换参数 ξ 的分布为 $p(\xi)$，扩展数据集的误差函数为

$$\tilde{E}=\frac{1}{2}\int\int\int\left\{y(\mathbf{s}(\mathbf{x},\xi))-t\right\}^2 p(t|\mathbf{x})p(\mathbf{x})p(\xi)d\mathbf{x}dtd\xi$$

- 假设分布 $p(\xi)$ 是零均值小方差的，只需考虑原始输入样本的较小变换。

变换数据训练

- 变换函数的 Taylor 展开

$$\mathbf{s}(\mathbf{x},\xi)=\mathbf{s}(\mathbf{x},0)+\xi\frac{\partial}{\partial\xi}\mathbf{s}(\mathbf{x},\xi)\Big|_{\xi=0}+\frac{\xi^2}{2}\frac{\partial^2}{\partial\xi^2}\mathbf{s}(\mathbf{x},\xi)\Big|_{\xi=0}+O(\xi^3)$$

$$=\mathbf{x}+\xi\tau+\frac{1}{2}\xi^2\tau'+O(\xi^3)$$

其中 τ' 表示 $\mathbf{s}(\mathbf{x},\xi)$ 对 ξ 的二阶导数在 $\xi=0$ 的取值。

- 扩展模型的输出函数

$$y(\mathbf{s}(\mathbf{x},\xi))=y(\mathbf{x})+\xi\tau^{\mathrm{T}}\nabla y(\mathbf{x})$$

$$+\frac{\xi^2}{2}\left[(\tau')^{\mathrm{T}}\nabla y(\mathbf{x})+\tau^{\mathrm{T}}\nabla\nabla y(\mathbf{x})\tau\right]+O(\xi^3)$$

变换数据训练

- 扩展模型的误差函数

$$\tilde{E}=\frac{1}{2}\int\int\{y(\mathbf{x})-t\}^2\,p(t|\mathbf{x})p(\mathbf{x})d\mathbf{x}dt$$
$$+\mathbb{E}[\xi]\int\int\{y(\mathbf{x})-t\}\tau^{\mathrm{T}}\nabla y(\mathbf{x})p(t|\mathbf{x})p(\mathbf{x})d\mathbf{x}dt$$
$$+\mathbb{E}\left[\xi^2\right]\int\int\left[\{y(\mathbf{x})-t\}\frac{1}{2}\left\{(\tau')^{\mathrm{T}}\nabla y(\mathbf{x})+\tau^{\mathrm{T}}\nabla\nabla y(\mathbf{x})\tau\right\}\right.$$
$$\left.+\left(\tau^{\mathrm{T}}\nabla y(\mathbf{x})\right)^2\right]p(t|\mathbf{x})p(\mathbf{x})d\mathbf{x}dt+O\left(\xi^3\right)$$

因为变换分布零均值，故 $\mathbb{E}[\xi]=0$，也令 $\mathbb{E}\left[\xi^2\right]=\lambda$。

变换数据训练

- 忽略 $O\left(\xi^3\right)$，平均误差函数为

$$\tilde{E} = E + \lambda\Omega$$

其中 E 是原始平方和误差项，正则项 Ω 为

$$\Omega = \int \left[\left\{ y(\mathbf{x}) - \mathbb{E}\left[t \middle| \mathbf{x} \right] \right\} \frac{1}{2} \left\{ (\tau')^{\mathrm{T}} \nabla y(\mathbf{x}) + \tau^{\mathrm{T}} \nabla \nabla y(\mathbf{x}) \tau \right\} + \left(\tau^{\mathrm{T}} \nabla y(\mathbf{x}) \right)^2 \right] p(\mathbf{x}) d\mathbf{x}$$

上式已经完成对 t 的积分。

变换数据训练

- 正则化误差等于未正则化平方和误差加上 $O(\xi)$ 阶的项
- 最小化总误差的网络函数为

$$y(\mathbf{x}) = \mathbb{E}\left[t|\mathbf{x}\right] + O(\xi)$$

- 为了得到与 ξ 相同的阶，正则化的第一项消失，得到切线传播正则项的等价项

$$\Omega = \frac{1}{2}\int \left(\tau^{\mathrm{T}} \nabla y(\mathbf{x})\right)^2 p(\mathbf{x}) d\mathbf{x}$$

- 考虑将输入变换简化为添加随机噪声，即 $\mathbf{x} \to \mathbf{x} + \xi$ ，则正则项为

$$\Omega = \frac{1}{2}\int \left\|\nabla y(\mathbf{x})\right\|^2 p(\mathbf{x}) d\mathbf{x}$$

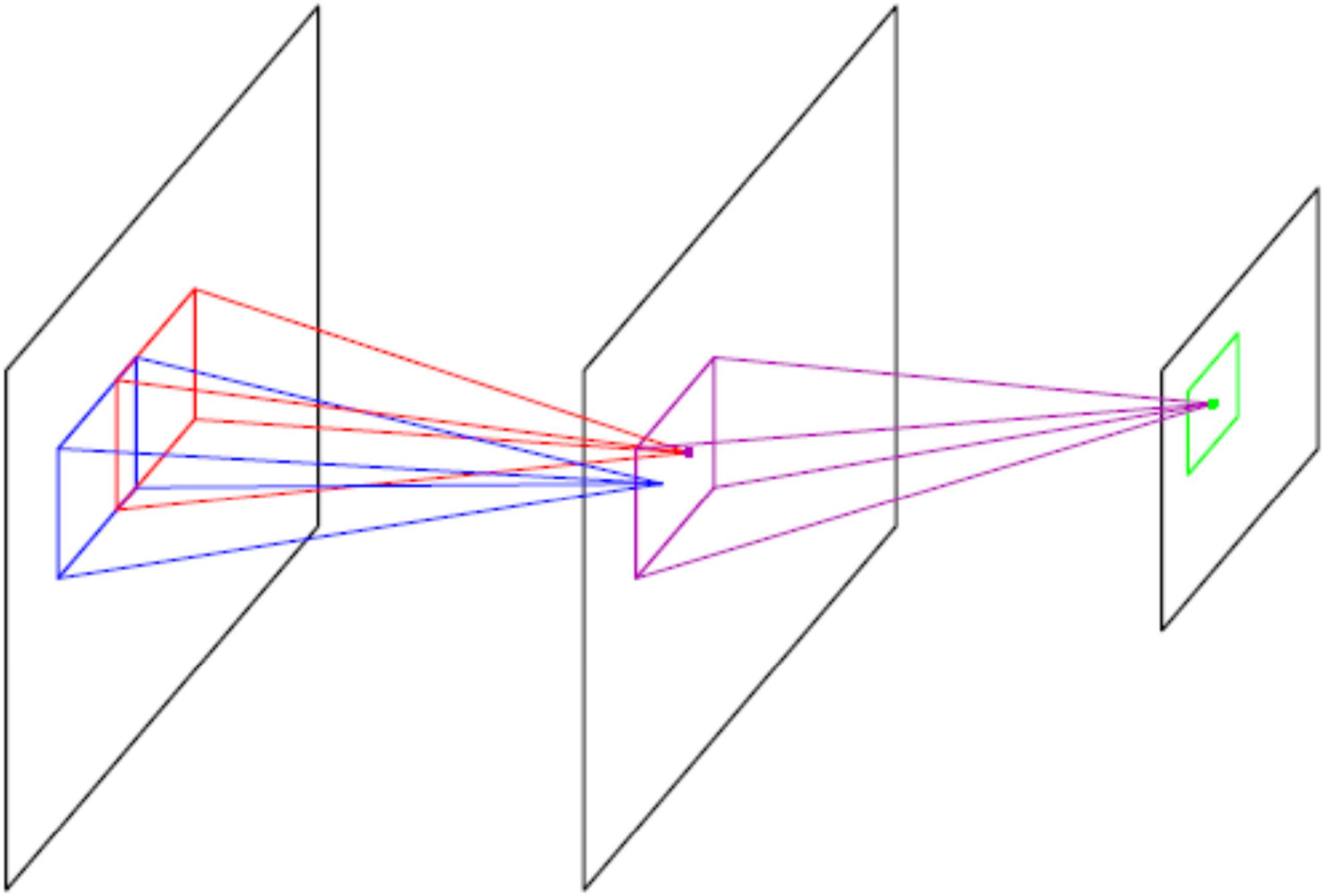
卷积网络

- 将对输入变换的不变性构建到神经网络结构中，这是卷积神经网络(convolutional neural network)的基础。

- 使用足够大规模的手写数字图像训练集，原理上全连接网络可以通过样本学到适当的不变性，获得好结果。
  - 忽略了相邻像素比远离像素具有更强的相关性
  - 计算机视觉方法通过局部图像特征来利用这种邻域相关性

- 卷积神经网络利用局部图像特征的三个机制
  - 局部感受野(local receptive fields)
  - 权值共享(weight sharing)
  - 子采样(subsampling)

卷积神经网络



卷积神经网络

- 卷积层单元构成一个平面，称为特征图(feature map)
  - 特征图的所有单元检测输入图像中不同位置上的相同模式
  - 由于共享权值，这些单元的激活值等价于图像像素与权值参数构成核函数的卷积。
- 卷积单元的输出形成网络子采样层的输入，如：每个子采样单元以特征图的 2x2 单元区域作为输入，计算这些输入的均值。
- 实际网络架构中，可以存在几对卷积和子采样层。
  - 空间分辨率的逐渐降低由越来越多的特征来补偿。
- 网络最后一层往往是全连接自适应层，多类分类问题输出一般使用 softmax 非线性函数。
- 最小化误差函数的网络训练通过误差函数梯度的反向传播完成
  - 与普通反向传播不同之处在于保证共享权值约束

软权值共享

- 软权值共享(soft weight sharing)：通过鼓励将某些组中权值约束为相似数值的正则化方法。
  - 学习过程决定：权值分组，每组权值均值，每组权值传播
- 鼓励权值分组可以看作高斯混合分布
  - 高斯分布成分的均值和方差与混合系数由学习过程确定
  - 简单权值衰减正则项可看作权值高斯先验分布的负对数
- 概率密度

$$p(\mathbf{w}) = \prod_i p(w_i)$$

其中

$$p(w_i) = \sum_{j=1}^{M} \pi_j \mathcal{N}\left(w_i \middle| \mu_j, \sigma_j^2\right)$$

π_j 是混和系数。

软权值共享

- 取负对数操作，获得正则项

$$\Omega(\mathbf{w}) = -\sum_i \ln\left(\sum_{j=1}^{M} \pi_j \mathcal{N}\left(w_i \middle| \mu_j, \sigma_j^2\right)\right)$$

- 总误差函数

$$\tilde{E}(\mathbf{w}) = E(\mathbf{w}) + \lambda\Omega(\mathbf{w})$$

其中 λ 是正则化系数。

- 最小化误差函数
  - 调整权值$\{w_i\}$和混合模型的参数$\{\pi_j, \mu_j, \sigma_j\}$
  - 如果权值为常数，则可以使用 EM 算法确定混合模型参数
  - 一般使用标准优化算法同时优化权值和混合模型参数

软权值共享

- 将$\{\pi_j\}$看作先验概率，根据 Bayes 定理引入对应的后验概率

$$\gamma_j(w)=\frac{\pi_j\mathcal{N}\left(w\middle|\mu_j,\sigma_j^2\right)}{\sum_k\pi_k\mathcal{N}\left(w\middle|\mu_k,\sigma_k^2\right)}$$

- 总误差函数对权值的导数

$$\frac{\partial\tilde{E}}{\partial w_i}=\frac{\partial E}{\partial w_i}+\lambda\sum_j\gamma_j(w_i)\frac{(w_i-\mu_j)}{\sigma_j^2}$$

  - 正则项的作用是将每个权值拉向第 j 个高斯均值，力度正比于权值所属组的高斯后验概率。

- 对高斯均值的导数

$$\frac{\partial\tilde{E}}{\partial\mu_j}=\lambda\sum_i\gamma_j(w_i)\frac{(\mu_i-w_j)}{\sigma_j^2}$$

软权值共享

- 对方差的导数

$$\frac{\partial \tilde{E}}{\partial \sigma_j} = \lambda \sum_i \gamma_j(w_i)\left(\frac{1}{\sigma_j} - \frac{(w_i - \mu_j)^2}{\sigma_j^3}\right)$$

  - 将 σ_j 推向对应均值 μ_j 周围权值平方偏差的加权平均，而加权系数仍然是成分 j 的后验概率。

- 在实现中，定义新变量 η_j 满足

$$\sigma_j^2 = \exp(\eta_j)$$

针对新变量最小化，保证参数σ_j为正，且阻碍一个或多个走向 0 的病态解（退化到一个权值参数值）。

软权值共享

- 对混合参数π_j 的导数，需要考虑约束（看作先验概率）

$$\sum_j \pi_j = 1,\ 0 \le \pi_i \le 1$$

  - 混合参数可以通过一组辅助变量的 softmax 函数来表达

$$\pi_j = \frac{\exp(\eta_j)}{\sum_{k=1}^{M} \exp(\eta_k)}$$

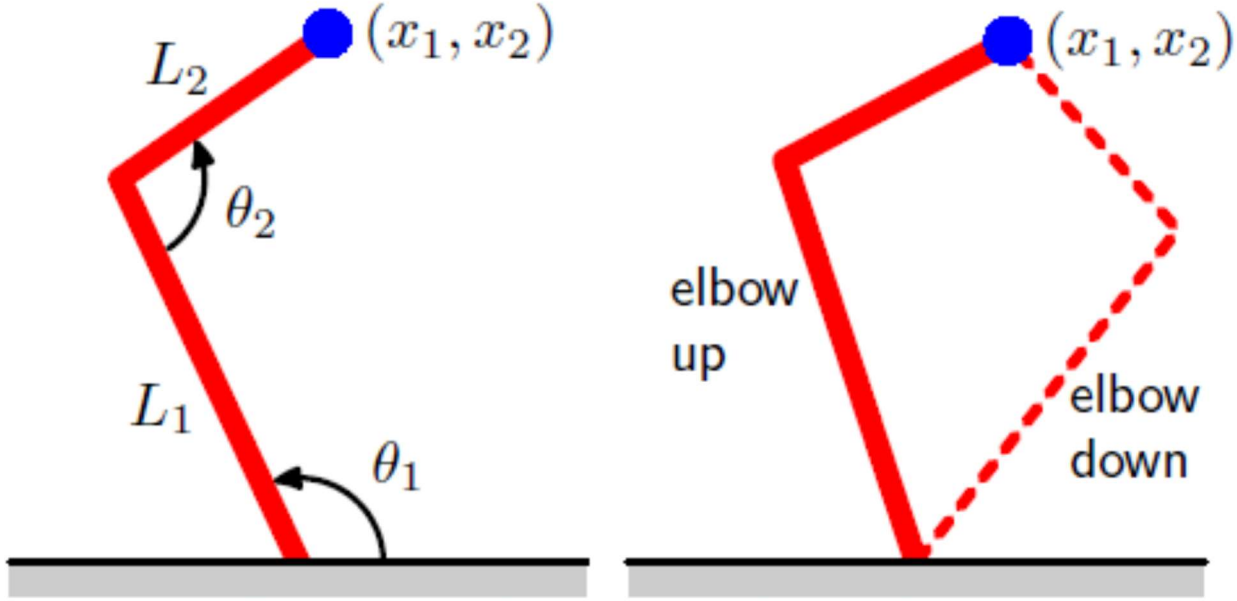
- 正则化误差函数对$\{\eta_j\}$ 的导数为

$$\frac{\partial \tilde{E}}{\partial \eta_j} = \sum_i \{\pi_j - \gamma_j(w_i)\}$$

π_j 推向成分 j 的平均后验概率。

混合密度网络(Mixture Density Networks)

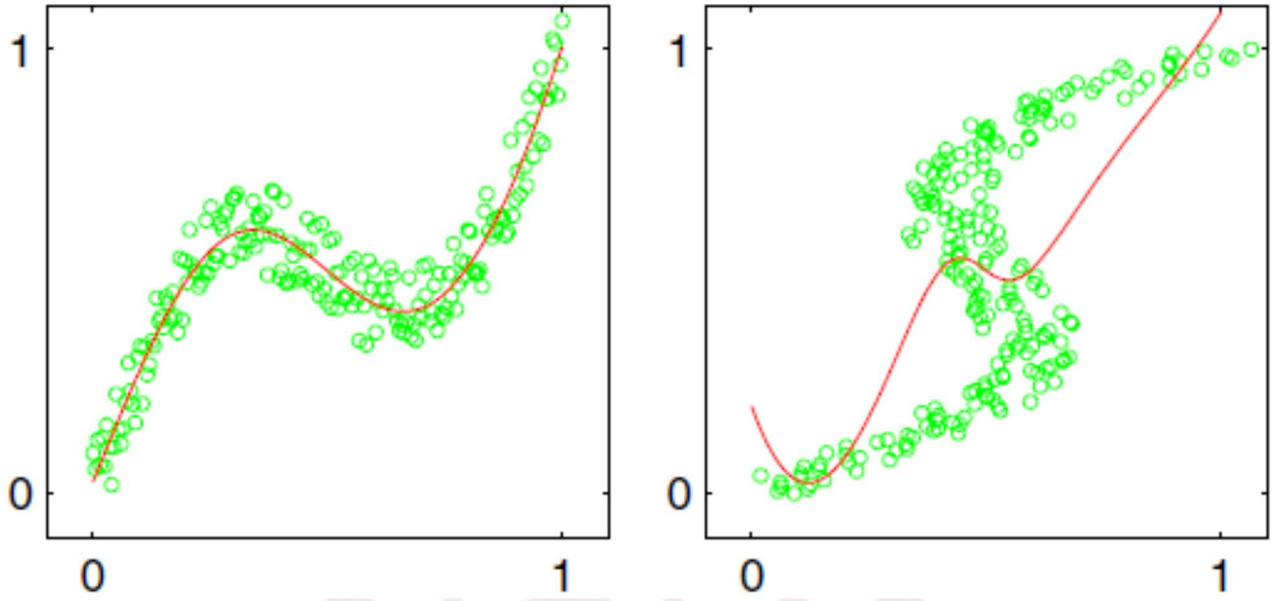
- 当数据的分布是多峰时，使用高斯分布假设将导致非常差的预测结果。
- 机械臂末端定位问题
  - 正向运动学和逆向运动学



  - 观察：如果正向问题(forward problem)包含多对一映射，那么反问题(inverse problem)将具有多个解。

混合密度网络

- 例：
  - 对均匀分布在区间 $(0,1)$ 中变量 x 进行采样得到数据集 $\{x_n\}$
  - 目标值 t_n 通过公式 $x_n + 0.3\sin(2\pi x_n)$ 和叠加均匀分布于 $(-0.1, 0.1)$ 的噪声获得
  - 反问题： 数据点集不变，交换 x 和 t 的角色。



正向问题　　反问题

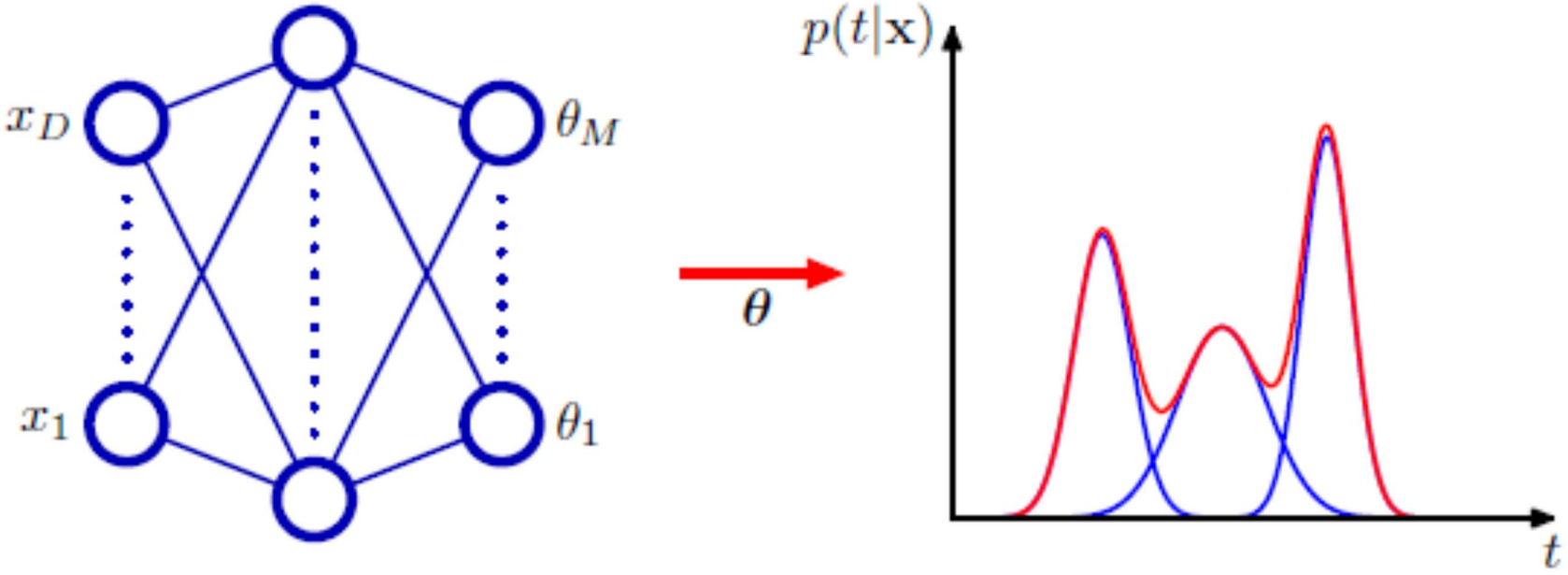
  - 绿色点是数据样本；红色曲线是两层网络最小化平方和误差函数的解

混合密度网络

- 高斯分布组成模型：异方差(heteroscedastic)模型

$$p(\mathbf{t}|\mathbf{x}) = \sum_{k=1}^{K} \pi_k(\mathbf{x}) \mathcal{N}\left(\mathbf{t}\middle|\mu_k(\mathbf{x}), \sigma_k^2(\mathbf{x})\right)$$

  - 数据噪声的方差是输入矢量的函数
  - 分布函数也可以选择其它分布，如：目标变量是二值时采用伯努利分布
- 混合模型的各种参数由传统网络的输出决定



混合密度网络

- 混合密度网络
  - 两层神经网络，隐含单元是 tanh 激活函数
  - 如果混合模型有 L 个组成成分，目标变量有 K 个组成，那么网络将有 L 个由混合系数 $\pi_k(\mathbf{x})$ 决定的输出单元净输入 a_k^{π}；K 个由核宽度$\sigma_k(\mathbf{x})$决定的输出 a_k^{σ}； L x K 个核均值 $\mu_k(\mathbf{x})$ 的组成 $\mu_{kj}(\mathbf{x})$ 决定的输出 a_{kj}^{μ} 。
  - 网络输出总数为 (K + 2)L，而通常网络是 K 个输出。
- 混合系数必须满足约束

$$\sum_{k=1}^{K}\pi_k(\mathbf{x})=1,\ 0\le\pi_k(\mathbf{x})\le 1$$

使用 softmax 实现

$$\pi_k(\mathbf{x})=\frac{\exp\left(a_k^{\pi}\right)}{\sum_{l=1}^{K}\exp\left(a_l^{\pi}\right)}$$

混合密度网络

- 方差必须满足约束$\sigma_k^2(\mathbf{x}) \geq 0$，表示为

$$\sigma_k(\mathbf{x}) = \exp(a_k^\sigma)$$

- 均值 $\mu_k(\mathbf{x})$ 是实数分量，直接使用网络输出净输入表示

$$\mu_{kj}(\mathbf{x}) = a_{kj}^\mu$$

- 误差函数

$$E(\mathbf{w}) = -\sum_{n=1}^{N} \ln\left\{\sum_{k=1}^{K} \pi_k(\mathbf{x}_n, \mathbf{w}) \mathcal{N}\left(\mathbf{t}_n \middle| \mu_k(\mathbf{x}_n, \mathbf{w}), \sigma_k^2(\mathbf{x}_n, \mathbf{w})\right)\right\}$$

- 最小化误差函数，需要计算误差函数对权值 w 的导数
  - 计算某个输入矢量的导数，然后对所有输入项求和得到对 E 的导数

混合密度网络

- 混合系数 $\pi_k(\mathbf{x})$ 看作 x 的先验概率，引入对应后验概率

$$\gamma_k\left(\mathbf{t}\middle|\mathbf{x}\right)=\frac{\pi_k\mathcal{N}_{nk}}{\sum_{l=1}^{K}\pi_l\mathcal{N}_{nl}}$$

其中 $\mathcal{N}_{nk}$ 表示 $\mathcal{N}(\mathbf{t}_n\mid\mu_k(\mathbf{x}_n),\sigma_k^2(\mathbf{x}_n))$。

- 计算误差函数对各个网络输出的导数

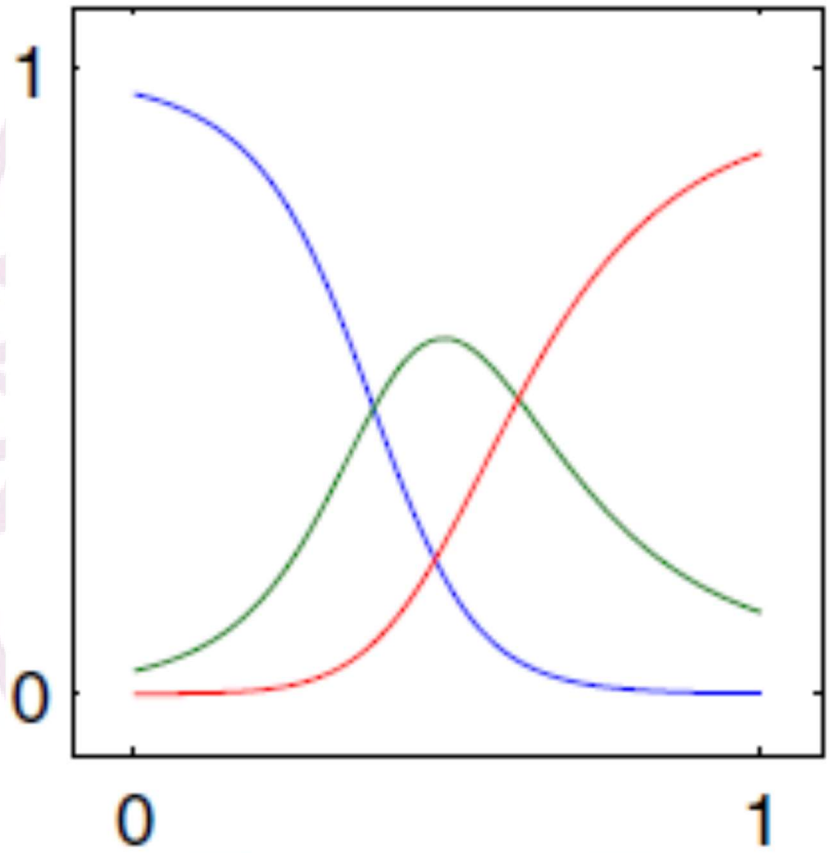
$$\frac{\partial E_n}{\partial a_k^{\pi}}=\pi_k-\gamma_k$$

$$\frac{\partial E_n}{\partial a_{kl}^{\mu}}=\gamma_k\left\{\frac{\mu_{kl}-t_l}{\sigma_k^2}\right\}$$

$$\frac{\partial E_n}{\partial a_k^{\sigma}}=-\gamma_k\left\{\frac{\|\mathbf{t}-\mu_k\|^2}{\sigma_k^3}-\frac{1}{\sigma_k}\right\}$$

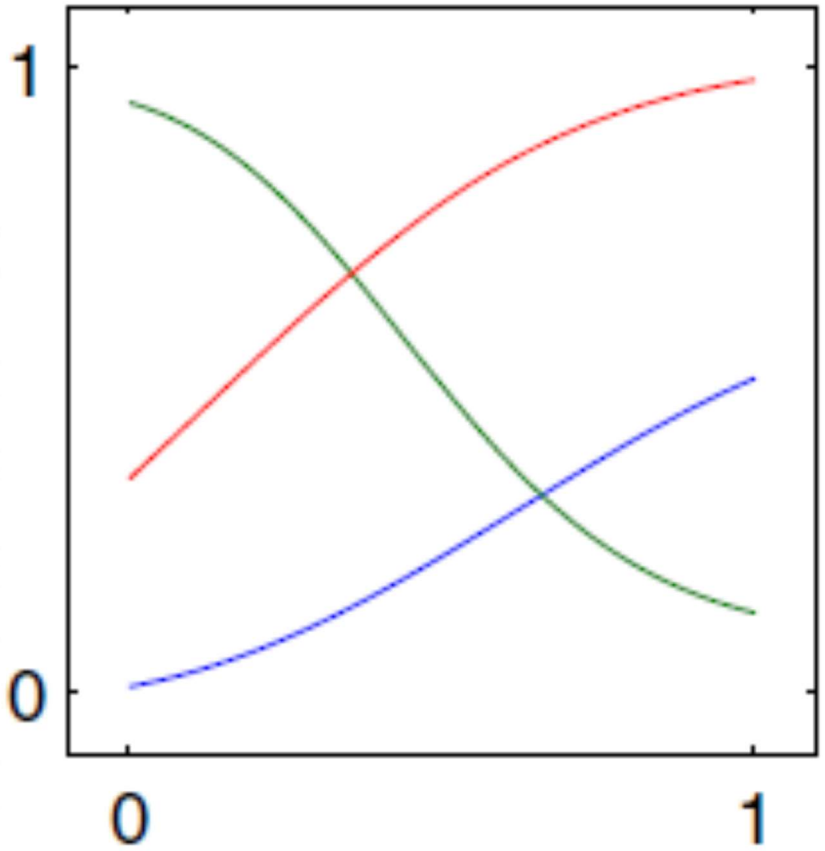
混合密度网络

- 例：
  - 三个核函数的混合系数 $\pi_k(x)$



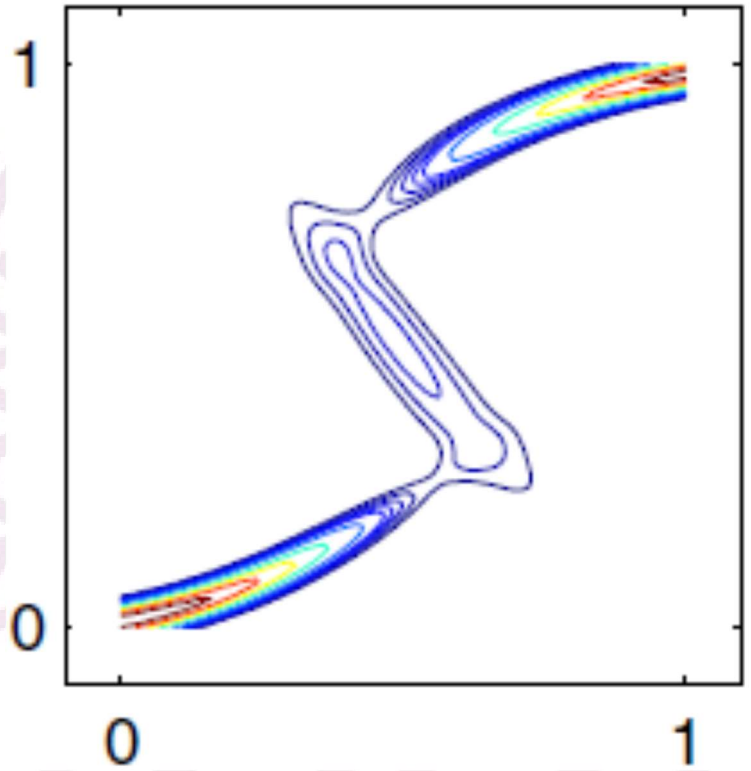
混合密度网络

- 与混合参数颜色相对应的均值 $\mu_k(x)$



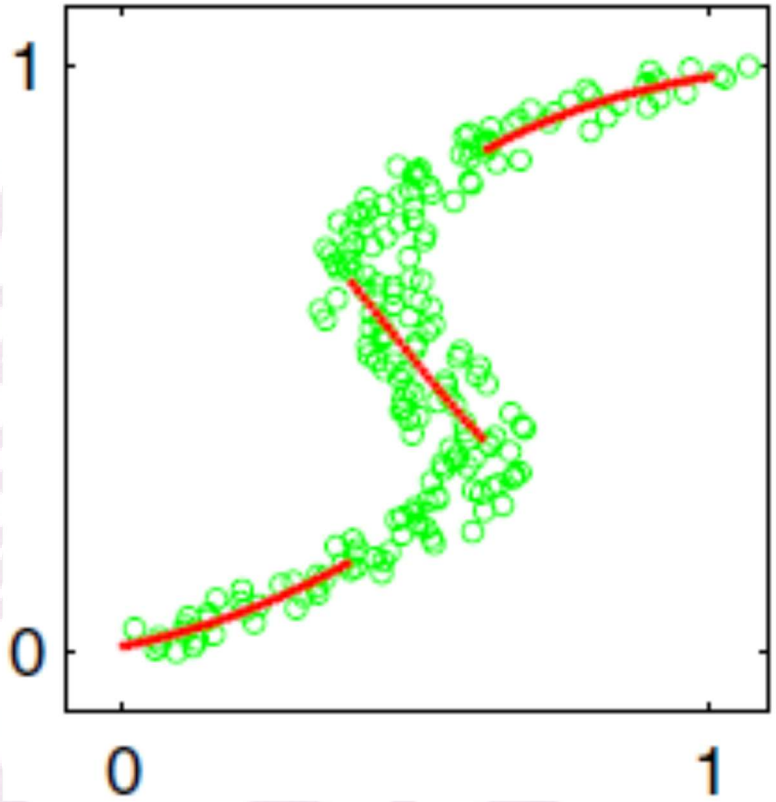
混合密度网络

- 相同混合密度网络，目标数据对应条件概率密度函数 $p(t \mid x)$ 的轮廓



混合密度网络

- 用红点表示的条件密度函数的结果



混合密度网络

- 一旦混合密度模型训练完毕，对输入矢量的任意给定数值，都能够预测目标数据的条件密度函数。
  - 条件密度函数是数据生成器的完整表示
- 由预测的条件密度函数，可以计算许多不同的兴趣值
  - 均值：目标数据条件均值

$$\mathbb{E}\left[\mathbf{t}|\mathbf{x}\right]=\int \mathbf{t}p\left(\mathbf{t}|\mathbf{x}\right)d\mathbf{t}=\sum_{k=1}^{K}\pi_k\left(\mathbf{x}\right)\mu_k\left(\mathbf{x}\right)$$

  - 密度函数的方差

$$\mathbf{s}^2\left(\mathbf{x}\right)=\mathbb{E}\left[\left\|\mathbf{t}-\mathbb{E}\left[\mathbf{t}|\mathbf{x}\right]\right\|^2|\mathbf{x}\right]$$

$$=\sum_{k=1}^{K}\pi_k\left(\mathbf{x}\right)\left\{\sigma_k^2\left(\mathbf{x}\right)+\left\|\mu_k\left(\mathbf{x}\right)-\sum_{l=1}^{K}\pi_l\left(\mathbf{x}\right)\mu_l\left(\mathbf{x}\right)\right\|^2\right\}$$

贝叶斯神经网络

- 已经看到
  - 使用最大似然方法确定网络参数（权值和偏差）
  - 正则化最大似然可解释为MAP方法，正则项看作参数先验概率的对数。
- 多层网络对参数值的高度非线性关系导致不能找到准确的贝叶斯方法
  - 后验概率的对数是非凸的，对应误差函数中存在多个局部最小值。
- 基于 Laplace 近似法
  - 通过以真实后验概率众数为中心的高斯分布来近似后验概率分布
  - 假设该高斯分布协方差很小，故网络函数在其后验概率显著非零的参数空间区域内，相对于参数的协方差近似为线性的。
- 后面的工作基于上述两个近似

后验参数分布

- 问题：从输入矢量 x 预测单个连续目标变量 t
  - 假设条件分布 $p(t \mid \mathbf{x})$ 是高斯分布，均值为网络模型输出 $y(\mathbf{x}, \mathbf{w})$，精度（方差倒数）为 β，则

$$p\left(t \middle| \mathbf{x}, \mathbf{w}, \beta\right) = \mathcal{N}\left(t \middle| y\left(\mathbf{x}, \mathbf{w}\right), \beta^{-1}\right)$$

  - 权值 w 的先验分布

$$p\left(\mathbf{w} \middle| \alpha\right) = \mathcal{N}\left(\mathbf{w} \middle| \mathbf{0}, \alpha^{-1}\mathbf{I}\right)$$

  - 独立同分布的 N 个观测量 $\mathbf{x}_1, \ldots, \mathbf{x}_N$，和对应目标值 $\mathcal{D} = \{t_1, \ldots, t_N\}$，似然函数为

$$p\left(\mathcal{D} \middle| \mathbf{w}, \beta\right) = \prod_{n=1}^{N} \mathcal{N}\left(t_n \middle| y\left(\mathbf{x}_n, \mathbf{w}\right), \beta^{-1}\right)$$

  - 后验概率分布（非高斯的）

$$p\left(\mathbf{w} \middle| \mathcal{D}, \alpha, \beta\right) \propto p\left(\mathbf{w} \middle| \alpha\right) p\left(\mathcal{D} \middle| \mathbf{w}, \beta\right)$$

后验参数分布

- 最大化对数后验概率以发现后验概率的局部最大值
  - 对应于正则化平方和误差函数

$$\ln p\left(\mathbf{w}\middle|\mathcal{D}\right)=-\frac{\alpha}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}-\frac{\beta}{2}\sum_{n=1}^{N}\left\{y\left(\mathbf{x}_n,\mathbf{w}\right)-t_n\right\}^2+\text{const}$$

  - 假设α 和 β 不变，通过标准非线性优化方法，如共轭梯度，可以得到后验概率最大值，记作$\mathbf{w}_{\text{MAP}}$。
- 计算负对数后验概率的二阶偏导数矩阵

$$\mathbf{A}=-\nabla\nabla\ln p\left(\mathbf{w}\middle|\mathcal{D},\alpha,\beta\right)=\alpha\mathbf{I}+\beta\mathbf{H}$$

其中：H 是 Hessian 矩阵，由平方和误差函数对权矢量分量的二阶偏导数组成。

后验参数分布

- 后验概率的高斯近似

$$q(\mathbf{w}|\mathcal{D}) = \mathcal{N}\left(\mathbf{w}|\mathbf{w}_{\mathrm{MAP}}, \mathbf{A}^{-1}\right)$$

- 对后验概率分布边缘积分得到预测分布

$$p(t|\mathbf{x},\mathcal{D}) = \int p(t|\mathbf{x},\mathbf{w})q(\mathbf{w}|\mathcal{D})d\mathbf{w}$$

  - 由于函数 $y(\mathbf{x},\mathbf{w})$ 是非线性的，故后验概率高斯近似难于解析处理。

- 假设后验概率与 w 的特征尺度相比具有较小的方差，这样可以使用 Taylor 级数将网络函数在 $\mathbf{w}_{\mathrm{MAP}}$ 周围展开且只保留线性项，有

$$y(\mathbf{x},\mathbf{w}) \simeq y(\mathbf{x},\mathbf{w}_{\mathrm{MAP}}) + \mathbf{g}^{\mathrm{T}}(\mathbf{w}-\mathbf{w}_{\mathrm{MAP}})$$

其中，定义

$$\mathbf{g} = \nabla_{\mathbf{w}} y(\mathbf{x},\mathbf{w})\Big|_{\mathbf{w}=\mathbf{w}_{\mathrm{MAP}}}$$

后验参数分布

- 条件分布

$$p\left(t\left|\mathbf{x},\mathbf{w},\beta\right.\right)\simeq\mathcal{N}\left(t\left|y\left(\mathbf{x},\mathbf{w}_{\mathrm{MAP}}\right)+\mathbf{g}^{\mathrm{T}}\left(\mathbf{w}-\mathbf{w}_{\mathrm{MAP}}\right),\beta^{-1}\right.\right)$$

- 预测分布

$$p\left(t\left|\mathbf{x},\mathcal{D},\alpha,\beta\right.\right)=\mathcal{N}\left(t\left|y\left(\mathbf{x},\mathbf{w}_{\mathrm{MAP}}\right),\sigma^{2}\left(\mathbf{x}\right)\right.\right)$$

其中：

$$\sigma^{2}\left(\mathbf{x}\right)=\beta^{-1}+\mathbf{g}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g}$$

  - 第一项是目标变量的固有噪声，第二项与 x 有关，表示由于模型参数的不确定性导致插值的不确定性。

超参数优化

- 前面假设超参数 α 和 β 是不变且已知的，超参数边缘似然

$$p\left(\mathcal{D}\middle|\alpha,\beta\right)=\int p\left(\mathcal{D}\middle|\mathbf{w},\beta\right)p\left(\mathbf{w}\middle|\alpha\right)d\mathbf{w}$$

- 利用 Laplace 近似，并取对数

$$\ln p\left(\mathcal{D}\middle|\alpha,\beta\right)\simeq -E\left(\mathbf{w}_{\mathrm{MAP}}\right)-\frac{1}{2}\ln\left|\mathbf{A}\right|+\frac{W}{2}\ln\alpha+\frac{N}{2}\ln\beta-\frac{N}{2}\ln\left(2\pi\right)$$

其中：W是参数总数，正则误差函数为

$$E\left(\mathbf{w}_{\mathrm{MAP}}\right)=\frac{\beta}{2}\sum_{n=1}^{N}\left\{y\left(\mathbf{x}_n,\mathbf{w}_{\mathrm{MAP}}\right)-t_n\right\}^2+\frac{\alpha}{2}\mathbf{w}_{\mathrm{MAP}}^{\mathrm{T}}\mathbf{w}_{\mathrm{MAP}}$$

- 使用最大化对数似然 $\ln p(\mathcal{D}\,|\,\alpha,\beta)$ 来对 α 和 β 做点估计，有

$$\alpha=\frac{\gamma}{\mathbf{w}_{\mathrm{MAP}}^{\mathrm{T}}\mathbf{w}_{\mathrm{MAP}}}$$

超参数优化

其中，γ 表示参数的有效数目，定义为

$$\gamma = \sum_{i=1}^{W} \frac{\lambda_i}{\alpha + \lambda_i}$$

和

$$\frac{1}{\beta} = \frac{1}{N - \gamma} \sum_{n=1}^{N} \left\{ y\left(\mathbf{x}_n, \mathbf{w}_{\mathrm{MAP}}\right) - t_n^2 \right\}$$

用于分类的贝叶斯神经网络

- 前面利用 Laplace 近似给出神经网络回归模型的贝叶斯方法
- 考虑网络单个 logistic sigmoid 输出对应两类问题，对数似然函数为

$$\ln p\left(\mathcal{D}\middle|\mathbf{w}\right)=\sum_{n=1}^{N}\left\{t_n \ln y_n+\left(1-t_n\right)\ln\left(1-y_n\right)\right\}$$

其中 $t_n \in \{0,1\}$ 是目标值，且 $y_n \equiv y(\mathbf{x}_n,\mathbf{w})$。

- 最小化正则化误差函数

$$E\left(\mathbf{w}\right)=-\ln p\left(\mathcal{D}\middle|\mathbf{w}\right)+\frac{\alpha}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}$$

得到最大后验概率解 $\mathbf{w}_{\mathrm{MAP}}$。

- 计算负对数似然函数的二阶偏导数组成的Hessian矩阵
  - 可以使用前面介绍计算 Hessian 的各种方法，如：外积近似方法

用于分类的贝叶斯神经网络

- 为了优化超参数 α，最大化边缘似然

$$\ln p\left(\mathcal{D}\middle|\alpha\right) \simeq -E\left(\mathbf{w}_{\mathrm{MAP}}\right) - \frac{1}{2}\ln\left|\mathbf{A}\right| + \frac{W}{2}\ln\alpha + \mathrm{const}$$

其中，正则化误差函数定义为

$$E\left(\mathbf{w}_{\mathrm{MAP}}\right) = -\sum_{n=1}^{N}\left\{t_n \ln y_n + \left(1-t_n\right)\ln\left(1-y_n\right)\right\} + \frac{\alpha}{2}\mathbf{w}_{\mathrm{MAP}}^{\mathrm{T}}\mathbf{w}_{\mathrm{MAP}}$$

上式中 $y_n \equiv y\left(\mathbf{x}_n, \mathbf{w}_{\mathrm{MAP}}\right)$。

- 最大化边缘似然得到 α 的重新估计公式

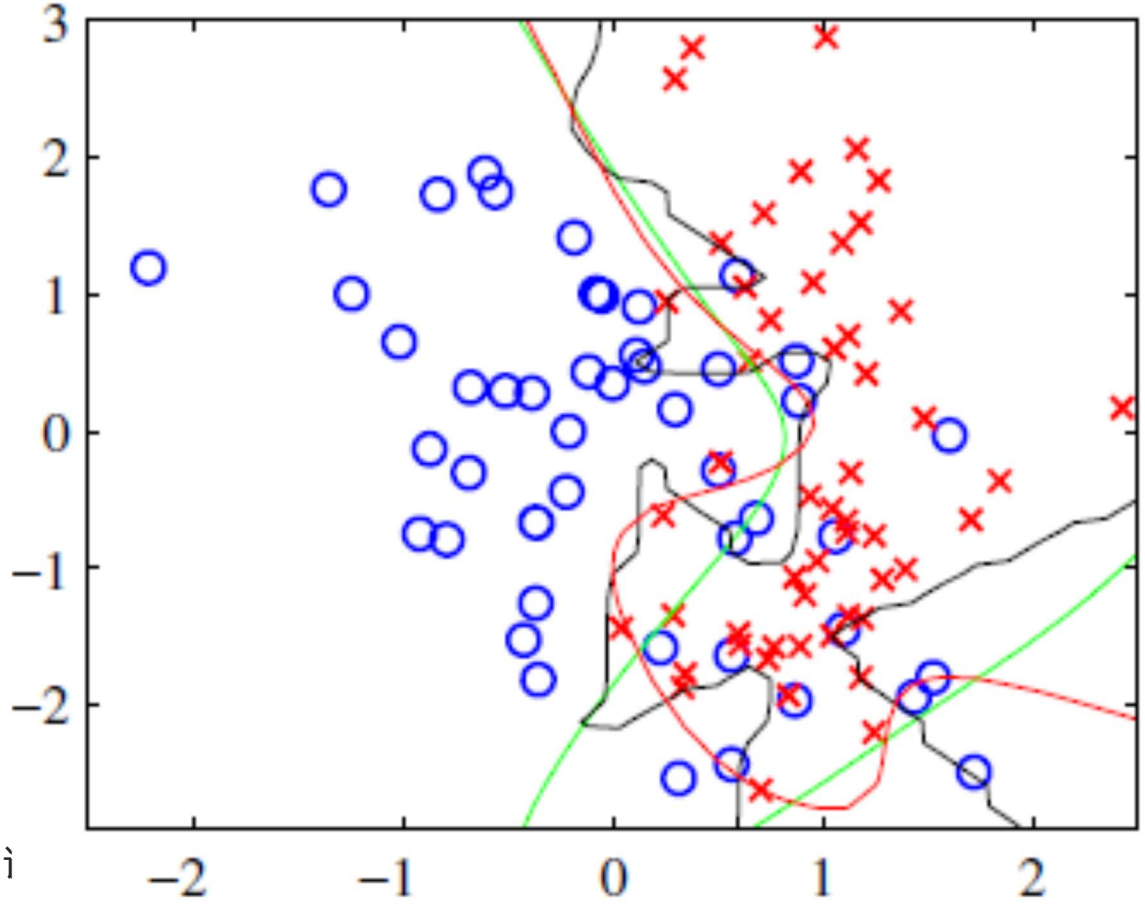
$$\alpha = \frac{\gamma}{\mathbf{w}_{\mathrm{MAP}}^{\mathrm{T}}\mathbf{w}_{\mathrm{MAP}}}$$

用于分类的贝叶斯神经网络

- 同样，得到预测分布的积分仍难于求解，简单近似解

$$p\left(t\middle|\mathbf{x},\mathcal{D}\right)\simeq p\left(t\middle|\mathbf{x},\mathbf{w}_{\mathrm{MAP}}\right)$$

- 图示：绿色曲线是最优决策边界，黑色曲线是两层网络最大似然结果，红色曲线是使用贝叶斯方法优化正则项的结果。



用于分类的贝叶斯神经网络

- 为了改进结果，对输出单元净输入进行线性近似

$$a(\mathbf{x},\mathbf{w}) \simeq a_{\mathrm{MAP}}(\mathbf{x}) + \mathbf{b}^{\mathrm{T}}(\mathbf{w}-\mathbf{w}_{\mathrm{MAP}})$$

其中 $a_{\mathrm{MAP}}(\mathbf{x}) = a(\mathbf{x},\mathbf{w}_{\mathrm{MAP}})$，矢量 $\mathbf{b} \equiv \nabla a(\mathbf{x},\mathbf{w}_{\mathrm{MAP}})$ 可以通过反向传播得到。

- 权值 w 的后验概率高斯近似，模型是 w 的线性函数，输出单元净输入值的分布为

$$p(a|\mathbf{x},\mathcal{D}) = \int \delta\left(a - a_{\mathrm{MAP}}(\mathbf{x}) - \mathbf{b}^{\mathrm{T}}(\mathbf{x})(\mathbf{w}-\mathbf{w}_{\mathrm{MAP}})\right) q(\mathbf{w}|\mathcal{D})\, d\mathbf{w}$$

其中 $q(\mathbf{w}|\mathcal{D})$ 是后验概率分布的高斯近似。

- 这个分布是均值为 $a_{\mathrm{MAP}} \equiv a(\mathbf{x},\mathbf{w}_{\mathrm{MAP}})$ 的高斯分布，其方差为

$$\sigma_a^2(\mathbf{x}) = \mathbf{b}^{\mathrm{T}}(\mathbf{x})\mathbf{A}^{-1}\mathbf{b}(\mathbf{x})$$

用于分类的贝叶斯神经网络

- 预测分布

$$p\left(t=1\middle|\mathbf{x},\mathcal{D}\right)=\int\sigma\left(a\right)p\left(a\middle|\mathbf{x},\mathcal{D}\right)da$$

- 由于难于求解高斯函数和 logistic sigmoid函数的卷积，故得到近似公式

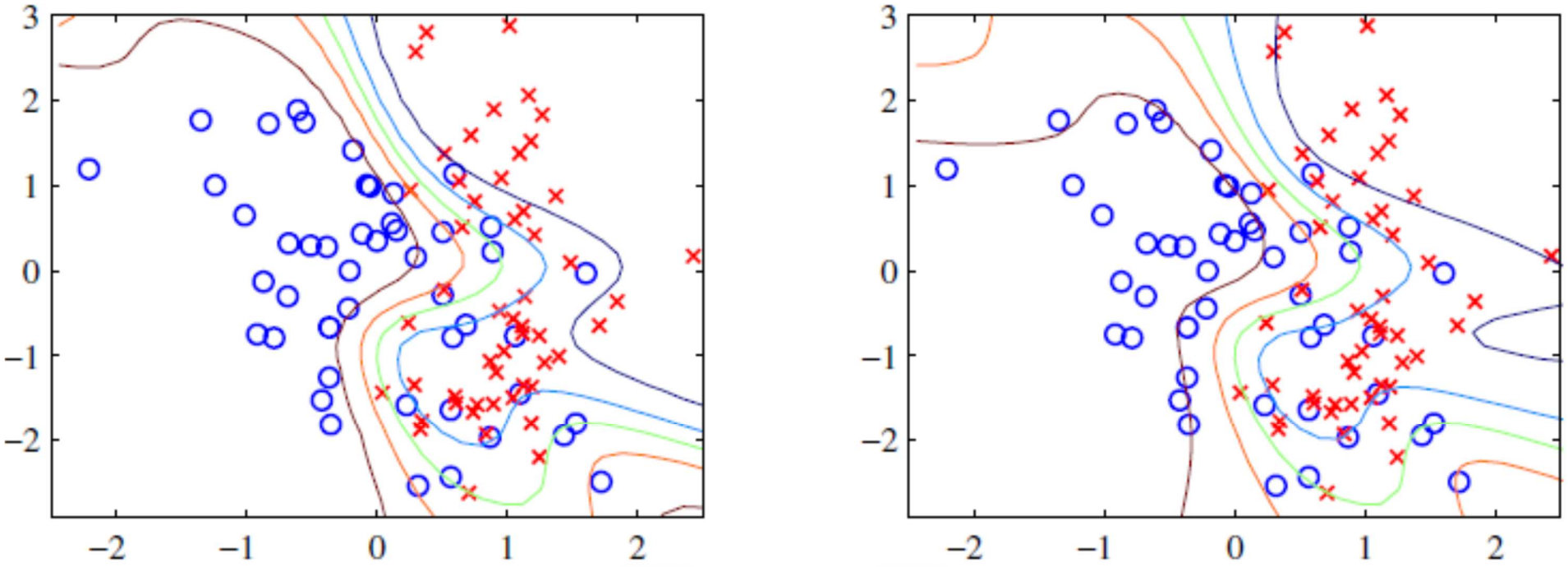
$$p\left(t=1\middle|\mathbf{x},\mathcal{D}\right)=\sigma\left(\kappa\left(\sigma_a^2\right)\mathbf{b}^{\mathrm{T}}\mathbf{w}_{\mathrm{MAP}}\right)$$

其中

$$\kappa\left(\sigma^2\right)=\left(1+\pi\sigma^2/8\right)^{-1/2}$$

用于分类的贝叶斯神经网络

- 图示：贝叶斯网络的 Laplace 近似
  - 8 个 tanh 激活函数的隐含单元，单个 logistic-sigmoid 输出单元
  - 权参数使用共轭梯度求解，超参数α 使用贝叶斯方法优化。
  - 左边为简单近似的结果，右边为改进近似的结果（五条曲线分别对应着y = 0.1, 0.3, 0.5, 0.7, 0.9的结果）



  - 右边结果使轮廓展开，决策不那么可信，输入点的后验概率向0.5靠近。

诚信 创新 实践