

2. 过拟合(正则化, Dropout)

1. 过拟合与解决办法

过拟合: 机器学习算法过度的拟合了训练数据, 导致在新的测试样本上的泛化误差比较小。

解决办法:

15. 如何防止过拟合?

过拟合造成的原因有两点: ① 数据太少
② 模型太复杂

针对上述两点有 ① 数据增强 ② 调优模型

数据增强: a. smote 算法 b. mixup 算法 c. GAN
过采样, 欠采样,
代价敏感学习: 为不同类别样本分配不同权重

调优模型: a. Early stop
b. dropout (训练时剔除部分神经元, 测试时全部)
c. 正则化: L_1 : 稀疏: 实现特征选择
 L_2 : 正则参数: 降低数据偏移影响
d. 集成学习方法

2. L1, L2 正则化

2.1 原理

8. L_1 与 L_2 正则化的区别:

8.1 L_1 正则化是模型各参数的绝对值之和。

L_2 是模型各参数的平方之和。

特点: 稀疏性

8.2 L_1 会倾向于产生很多的特征, 而其他特征都是 0。

因为最优的参数值很大概率出现在坐标轴上, 这样就会导致某一维的值为 0, 产生稀疏特征矩阵。

L_2 会选择更多的特征, 这些特征都会接近于 0。

最优的参数值很小, 很少出现在坐标轴上, 因此每一维的参数都不会是 0。

当最小化 $\|w\|_2$ 时, 就会使每一项趋近于 0。

2.2 L1与L2的先验分布

L1正则先验分布是**Laplace分布**，L2正则先验分布是**Gaussian分布**。

解释：

在机器学习建模中，我们知道了 x 和 y 后，需要对参数 w 进行建模。那么后验概率表达式如下。

$$MAP = \log P(y|X, w)P(w) = \log P(y|X, w) + \log P(w)$$

可以看出后验概率函数为在似然函数的基础上增加了 $\log P(w)$ ， $P(w)$ 的意义就是对权重系数 w 的概率分布的先验假设，在收集到训练样本 X, y 后，则可根据 w 在 X, y 下的后验概率对 w 进行修正，从而做出对 w 更好的估计。

若假设的先验分布为0均值的高斯分布，即

$$w \sim N(0, \sigma^2)$$

则有

$$\begin{aligned}\log P(w) &= \log \prod_j P(w_j) = \\ \log \prod_j \left[\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(w_j)^2}{2\sigma^2}} \right] \\ &= -\frac{1}{2\sigma^2} \sum_j w_j^2 + C\end{aligned}$$

可以看到，在高斯分布下 $\log P(w)$ 的效果等价于在代价函数中增加L2正则项。

若假设的先验分布服从均值为0，参数为 a 的拉普拉斯分布，即

$$P(w_j) = \frac{1}{\sqrt{2a}} e^{-\frac{|w_j|}{a}}$$

则有

$$\begin{aligned}\log P(w) &= \log \prod_j P(w_j) = \\ \log \prod_j \left[\frac{1}{\sqrt{2a}\sigma} e^{-\frac{|w_j|}{a}} \right] \\ &= -\frac{1}{2a} \sum_j |w_j| + C\end{aligned}$$

可以看到，在拉普拉斯分布下 $\log P(w)$ 的效果等价在代价函数中增加L1正项。

3.Dropout详解

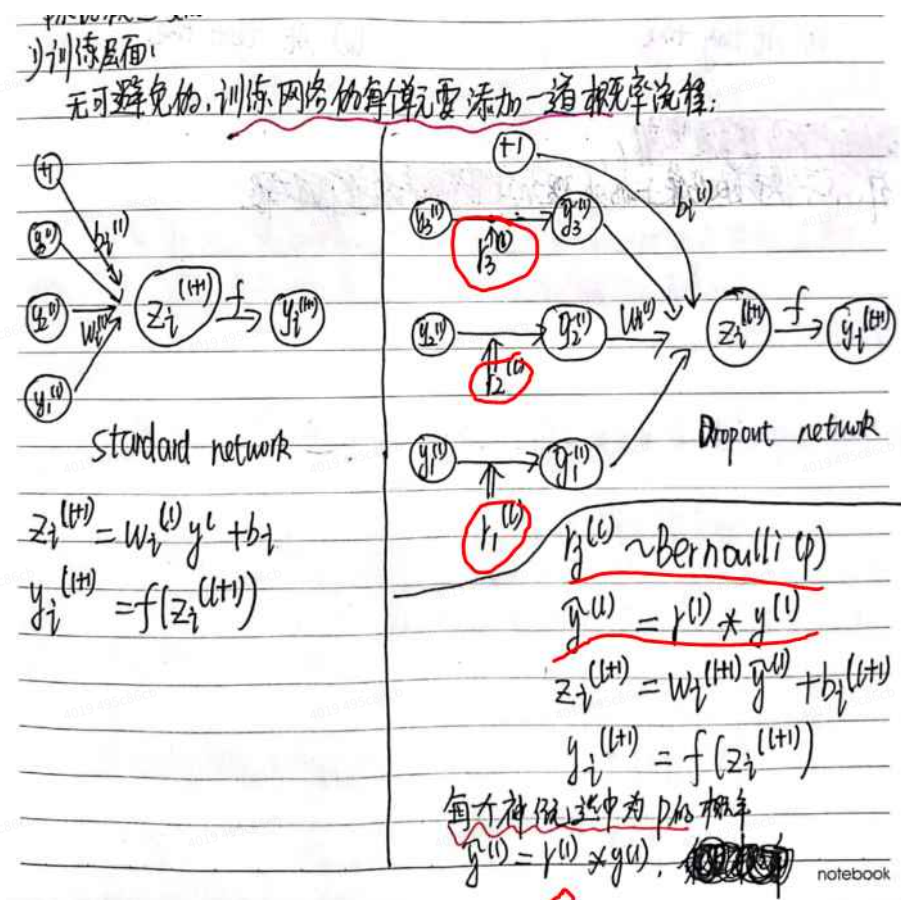
3.1原理&目标

做法：对于神经网络单元，按照一定的概率将其暂时从网络中丢弃。由于是随机放弃的，所以每一个 mini-batch 都在训练不同的网络。

目标：强迫神经元与其他随机挑选出来的神经元共同工作，达到好的效果。削弱神经元节点的联合适应性，增强泛化能力。

3.2 带来的变化

1) 训练上：



2) 测试时

