# Sector 28 批量归一化 (Batch Normalization)

|  |
| --- |
|  |
| 问题： |

1. 反向传播时，越深层的网络的梯度越大，即深层更新更快、收敛更快
2. 每当深层训练好后，浅层还未训练好，那么浅层一更新，已经训练好的深层又需要更新

## 如何解决：Batch Normalization

1. 批量归一化对每个批次的方差和均值进行规范化
2. 通过学习参数gamma和beta，对规范化后的方差和均值进行调整

|  |
| --- |
|  |

1. 对于FC层，批量归一化作用在特征维上，eg:对于3个组4维样本(3,4)，批量归一化是对所有样本的同一个特征进行归一化
2. 对于卷积层，作用在通道维，eg:对于(3,100,224,224)，将每个通道上对应位置的像素叠加在一起成为一个样本，即一个样本的长度为100，样本总数为3\*224\*224个，对每个这样的样本进行归一化

## 总结

|  |
| --- |
|  |

# Sector 29 ResNet

## 问题：

1. 加入更多的层总是能提升精度吗

## 解决：

1. 将原始输入加到块的输出上，Output = f(X)+X。X可以看作浅层网络训练得到的模型，因此在深层训练中，仍可以继续根据浅层网络的输出进行训练
2. ResNet由多个Residual块组成

|  |
| --- |
|  |
|  |

# Sector 37 微调Fine-tune / 迁移学习

## 问题:

1. 标注一个数据集很贵
2. 一个网络通常由特征提取层和softmax回归(线性分类器)组成
3. 使用其他数据集时，特征提取部分需要调整的可能不多，但由于数据集的标注不同，线性分类器可能需要改变
4. 底层的特征简单，更加通用

## 解决：

1. 预训练一个相同架构的网络，初始化时将训练得到的权重用于新模型的训练
2. 分类层可以随机初始化
3. 可以使用更强正则化的目标数据集作为源数据集(更小的学习率、更少的数据迭代)
4. 预训练使用的源数据集更加复杂时微调效果更好
5. 数据集小，容易overfitting时，可以固定一些层的参数，不参与更新

|  |
| --- |
|  |