# P19

1. **优化失败现象**：训练中损失不再下降，甚至初始阶段损失就无法降低，梯度下降因梯度为零停止参数更新。
2. **临界点类型**：梯度为零的 “临界点” 包括局部最小值和鞍点。鞍点是梯度为零但非极值的点，形状类似马鞍。
3. **临界点判断**：通过海森矩阵（含二次微分）的特征值判断：
   * 所有特征值为正→局部最小值；
   * 所有特征值为负→局部最大值；
   * 特征值有正有负→鞍点。
4. **鞍点的应对**：若卡在鞍点，可沿负特征值对应的特征向量方向更新参数，逃离鞍点以降低损失。
5. **常见性特点**：高维参数空间中，鞍点远多于局部最小值，多数训练停止源于鞍点，极少出现所有特征值均为正的临界点。

# P20

1. **批次（Batch）**
   * 概念：将数据分成若干批次（Mini Batch），每次用一批数据计算损失、梯度并更新参数，遍历所有批次为一个 Epoch。每个 Epoch 前会重新划分批次（Shuffle），确保批次组成不同。
   * 不同 Batch Size 对比：Full Batch（全量数据）更新少但稳定，Batch Size=1 时更新频繁但梯度嘈杂；因 GPU 并行计算，一定范围内（如 1-1000）不同 Batch Size 的单批计算时间接近，但小 Batch 跑完一个 Epoch 耗时更长。
   * 对训练的影响：小 Batch 的嘈杂梯度有助于跳出局部最小值和鞍点，训练及测试效果更优；大 Batch 易导致优化问题，训练和验证结果较差。Batch Size 是需调整的超参数，相关研究尝试结合大 Batch 的效率与小 Batch 的效果。
2. **动量（Momentum）**
   * 概念：模拟物理惯性，参数更新方向结合当前梯度反方向与前一步更新方向，减少震荡，助力跳出局部最小值和鞍点。
   * 原理：更新量为过去梯度的加权和，初始时前步更新量为 0，后续逐步累积历史方向，增强优化连续性。

# P21

1. **训练停滞的常见原因**：Loss 不再下降未必是因到达临界点（梯度为零），更多是参数在误差曲面谷壁震荡，此时梯度仍较大。
2. **普通梯度下降的局限**：即便在简单凸误差曲面中，学习率过大易导致参数震荡，过小则难以逼近最小值，效果欠佳。
3. **学习率定制思路**：为不同参数动态调整学习率 —— 梯度小（平缓方向）用大学习率，梯度大（陡峭方向）用小学习率。
4. **关键优化方法**：
   * Adagrad：通过梯度均方根（σ）调整学习率（η/σ），适配不同梯度大小，但难动态响应同一参数的梯度变化。
   * RMS Prop：σ 的更新融合历史与当前梯度（α 调节权重），可动态调整学习率，快速应对梯度变化。
   * Adam：结合 RMS Prop 与动量，是常用优化策略，框架默认参数效果较好。
5. **学习率调度**：
   * 学习率衰减：随训练推进减小 η，缓解震荡，使参数平稳靠近最小值。
   * Warm Up：学习率先增后减，用于 BERT 等模型，可能因初始统计不精准，先以小学习率收集信息再调整。
6. **现代优化器构成**：通常包含动量（考虑梯度方向）、梯度均方根（考虑梯度大小）及学习率调度，变种差异主要体现在这些部分的设计。

# P22

1. **分类的类别表示**：将分类当作回归处理时，直接给类别编号（如 1、2、3）会错误暗示类别间的远近关系，更合理的方式是用 One-hot 向量（如 3 类对应 100、010、001），确保类别间距离均等。
2. **网络输出与 Soft-max**：分类任务中，网络需输出与目标类别维度匹配的向量，再通过 Soft-max 函数将输出归一化到 0-1 之间（且和为 1），同时放大数值差距，便于与 One-hot 目标向量计算距离。
3. **损失函数对比**：计算预测与目标的距离可采用 MSE（平方和），但更常用 Cross-entropy（通过对数和求和）。从优化角度，Cross-entropy 在损失大的区域梯度更明显，避免 MSE 可能导致的训练停滞，且在 PyTorch 中常与 Soft-max 绑定使用，是分类任务的优选。