# Optimization for deep learning

## P26

* **核心目标与符号**：目标是找到参数 θ，使训练数据的损失函数 L 最小，让模型预测 y (t) 接近真实值ŷ(t)。关键符号包括 θₜ（神经网络参数）、L（损失函数）、梯度（∇L）、动量（记录梯度历史信息）。
* **学习模式**：区分在线学习（每步仅处理单个样本）与离线学习（每步可处理全部数据），课程假设离线场景（可获取所有数据计算完整损失）。
* **经典优化器原理**：
  + SGD：从初始参数出发，沿梯度反方向（损失减小方向）逐步移动。
  + SGDM：引入动量，累加历史梯度信息，避免因单步梯度接近 0 而停滞。
  + Adagrad：学习率分母为历史梯度和，动态调整（梯度大则学习率小，反之则大）。
  + RMSProp：改进 Adagrad 的分母计算（指数加权平均），避免学习率过小。
  + Adam：结合 SGDM 的动量与 RMSProp 的自适应学习率，修正初始动量偏差。
* **优化器实践与对比**：
  + 知名模型中，BERT、Transformer 等用 Adam，YOLO、ResNet 等用 SGDM。
  + Adam 训练收敛快但验证集泛化波动大；SGDM 训练慢但更稳定，收敛效果好（可能与找到的最小值平坦度相关）。
* **优化器改进方法**：
  + SWATS：初期用 Adam 加速，后期换 SGDM 稳定收敛。
  + AMSGrad：通过记录最大历史梯度，缓解 Adam 中有效梯度影响弱的问题。
  + AdaBound：对学习率设上下界，解决梯度过大时学习率过小问题（人工设定界限，削弱自适应性）。
  + SGD 改进：通过周期性调整学习率（如 Cyclical、One-cycle 策略）加速收敛。

## P27

* **优化器特性与组合**：稳定型算法收敛效果好，Adam 计算快，实际中常组合使用；针对 Adam 的不稳定性，可通过限制学习率范围改进，对 SGDM 则可通过调整学习率策略（如周期性调度）加速收敛。
* **Adam 的改进与 warmup**：训练初期梯度分布混乱、分母估计不准，导致 Adam 不稳定，需引入 warmup（如学习率先增后减），改善初期波动；通过估计梯度方差（rt）动态调整步长，初期走小步保证稳定。
* **算法切换逻辑**：RM 算法因初期 rt 估计不准，先使用 SGDM，待估计可靠后切换为 Adam；SWATS 则为追求更好收敛，从 Adam 切换到 SGDM，两者切换原因本质不同。
* **Lookahead 算法**：作为优化器包装器，可与各类优化器结合，通过 “快权重” 探索与 “慢权重” 调整（取中间插值），避免进入崎岖参数区域，提升模型泛化能力。
* **进阶优化算法**：Nesterov 加速梯度（NG）对 SGDM 改进，通过预测下一步移动调整更新，避免陷入较差参数点；Nadam 将 NG 思想应用于 Adam，仅修改动量项超前部署。
* **正则化处理**：AdamW 等方法将 L2 正则化与梯度计算分离（最后单独减正则项），效果优于传统融入梯度的方式，在 BERT 等模型中广泛应用。
* **辅助优化技巧**：通过 shuffle、dropout、梯度噪声增加随机性，促进探索；采用 warmup、课程学习（先易后难训练数据）、微调预训练模型等策略提升优化效果。
* **总结与选择**：无万能优化器，SGDM 稳定但慢，Adam 快但可能泛化差；NLP、语音生成等常用 Adam，检测、分割等常用 SGDM，需结合任务测试调整，训练问题多源于数据、架构而非优化器。