# 资料读取：

用torch.utils.data.Dataset和DataLoader读取并批量处理数据，自定义Dataset需重写初始化、获取项、长度方法。

# Tensor 操作：

高维数组结构，支持数值运算（加减乘除、对数等）和维度操作（转置、合并等），可移至 GPU 加速运算。

# 梯度计算：

通过required\_grad和backward()自动计算梯度，用于神经网络优化。

# 模型定义：

继承nn.Module，用nn.Linear等层和非线性激活函数（如 ReLU）构建网络，重写forward方法定义运算流程。

# 训练流程：

定义损失函数（如均方误差）和优化器（如 SGD），通过迭代训练（归零梯度→反向传播→参数更新）优化模型，测试时关闭梯度计算。

# 模型管理：

用torch.save和torch.load实现模型保存与加载。

# 应用场景：

语音处理、自然语言处理、计算机视觉等领域，学界和业界应用广泛。

# 红楼 one 题目：

通过前五天数据预测第五天新冠测试阳性率，属回归问题，示例代码由红楼 one 改写，作业以 homo one 为准。

# 数据处理：

CSV 文件含 118 列（117 特征 + 1 标签），用 pandas 读取后分为输入 x（2699×117）和标签 y（2699），未划分验证集，可参考红楼 one 代码。

# 数据集与加载器：

定义数据集需实现init、getitem、len方法，数据加载器设 batch\_size，训练时 shuffle 设为 True。

# 模型构建：

以三层线性模型为例，输入维度 117，输出维度 1。

# 损失与优化器：

回归问题选 MSE 损失函数，优化器可用 SGD。

# 训练循环：

迭代约 3000 次，流程为前向传播、计算损失、反向传播、更新参数、重置梯度。

PyTorch 文档：查文档可了解函数输入输出、参数含义，参数分位置参数和关键字参数，部分有默认值。

# 常见错误：

设备不一致时将数据移至模型所在设备；张量形状不匹配用transpose调整；显存不足减小 batch\_size；类型不匹配（如交叉熵需标签为 long 型）时转换类型。