# PyTorch 核心功能：

开源框架，支持张量计算、GPU 加速、自动梯度计算（gradient），适合研究（vs TensorFlow 更适合产品）。

# 张量操作：

创建：列表 / NumPy 转换、torch.zeros/ones；

维度：squeeze（压缩）、unsqueeze（扩展）；

运算：转置、拼接（concatenate）、加减乘除、次方、求和。

# 神经网络流程：

数据准备→定义模型（如nn.Linear）→损失函数（MSE / 交叉熵）→优化器（SGD）；

训练：前向传播→计算损失→backward()反向传播→优化器更新参数；

GPU 加速：torch.cuda.is\_available()检测，张量移至 GPU 运算。

# 进阶张量应用：

多维矩阵运算、复杂维度变换，调试技巧（如梯度查看x.grad）。

# 模型实战：

激活函数：ReLU、Sigmoid 增强非线性能力；

模型保存 / 加载：torch.save(model.state\_dict(), path)，torch.load恢复；

前向传播示例：自定义forward方法实现多层网络。

# 数据处理：

自定义Dataset类读取数据，DataLoader加载，训练时shuffle=True。

# 模型实现：

线性模型nn.Linear(input, output)，MSE 损失，SGD 优化器；

# 训练循环：

清零梯度→前向计算→损失计算→反向传播→参数更新。

# 验证与提交：

验证时关闭梯度（with torch.no\_grad()），测试结果保存为 CSV 文件。