

# 极客大学机器学习训练营

## Python/R 中的数据预处理和可视化

王然

众微科技 AI Lab 负责人

二〇二一年一月六日

- 1 内容简介
- 2 NumPy 和 Jax
- 3 Pandas 和 dplyr
- 4 Matplotlib 和 TensorBoard
- 5 总结和预习

- 1 内容简介
- 2 NumPy 和 Jax
- 3 Pandas 和 dplyr
- 4 Matplotlib 和 TensorBoard
- 5 总结和预习

本章主要内容是讲解常见的数据处理方法，大部分内容在 Jupyter Notebook 当中。本章的目的是为后面的学习做准备。

- ▶ NumPy: 矩阵 (Tensor) 处理工具
- ▶ Jax: 打了鸡血的 NumPy
- ▶ Pandas: 一般数据处理工具
- ▶ dplyr: 良好的探索性数据分析工具
- ▶ Matplotlib: 探索性数据分析的画图工具
- ▶ TensorBoard: 记录模型训练过程的工具

- ▶ NumPy 当中的 Einsum 和 Broadcast 转换
- ▶ Jax 函数式编程的习惯

- 1 内容简介
- 2 NumPy 和 Jax
- 3 Pandas 和 dplyr
- 4 Matplotlib 和 TensorBoard
- 5 总结和预习

详细内容见 Jupyter Notebook



- ▶ 核心加速原理: JIT+Jaxpr+Async Dispatch+XLA
- ▶ 核心写作限制: Tracable 和 Pure 函数

- ▶ JIT = Just-in-time Compilation
- ▶ 第一次运行时进行编译：充分利用各种 runtime information
- ▶ 可以指定重新编译的对象 (`static_argnums`)，解决 trace 的问题 (trace 问题和 Jax Primitives 在后面介绍)

- ▶ 还记得上次说的异步图的事情么？Jax 可以自动实现
- ▶ 原因：函数没有负效应
- ▶ 问题：需要改写一些常见用法，见 Jupyter Notebook

- ▶ XLA = Accelerated Linear Algebra. 详见[XLA 官方文档](#)
- ▶ 保证了 Jax 可以在各种 device (CPU/GPU/TPU) 上运行
- ▶ 特殊情况下, 如果觉得效率不够高, 可以自定义 XLA 算子, 一般情况下不需要

- 1 内容简介
- 2 NumPy 和 Jax
- 3 Pandas 和 dplyr
- 4 Matplotlib 和 TensorBoard
- 5 总结和预习

见 Jupyter Notebook

- 1 内容简介
- 2 NumPy 和 Jax
- 3 Pandas 和 dplyr
- 4 Matplotlib 和 TensorBoard
- 5 总结和预习

见 Jupyter Notebook



- 1 内容简介
- 2 NumPy 和 Jax
- 3 Pandas 和 dplyr
- 4 Matplotlib 和 TensorBoard
- 5 总结和预习

- ▶ 本章大部分内容在需要用的时候速查即可
- ▶ Einsum 和 Broadcast 如果不会没有关系，我们将会 DL 章节重新讲解
- ▶ 重点：玩一玩 Jax，很好玩的
  - ▶ 尝试把一些 numpy 的操作改为 Jax
  - ▶ 对于已经有 PyTorch 基础的童鞋，可以改一些简单的网络，可以对比一下 Jax 和 NumPy 或 PyTorch 在 CPU/GPU 上的表现
  - ▶ 注意：PyTorch 的写法跟 NumPy 几乎一样，所以即使不会 PyTorch 也不用担心

- ▶ 下一章对于数学基础较为缺乏的同学来说，会有一定困难
- ▶ 预习内容是两本统计学的书（钉钉群文件里可以下载），**不要担心看不懂，我们会讲全部内容**
- ▶ 可以找一些看数学的感觉，比如说做一些第一章的题