Edge AI 学习笔记

————2025090905005 陈旭彬

一、基础概念

1.人工智能与 机器学习与深度学习

【基础概念】

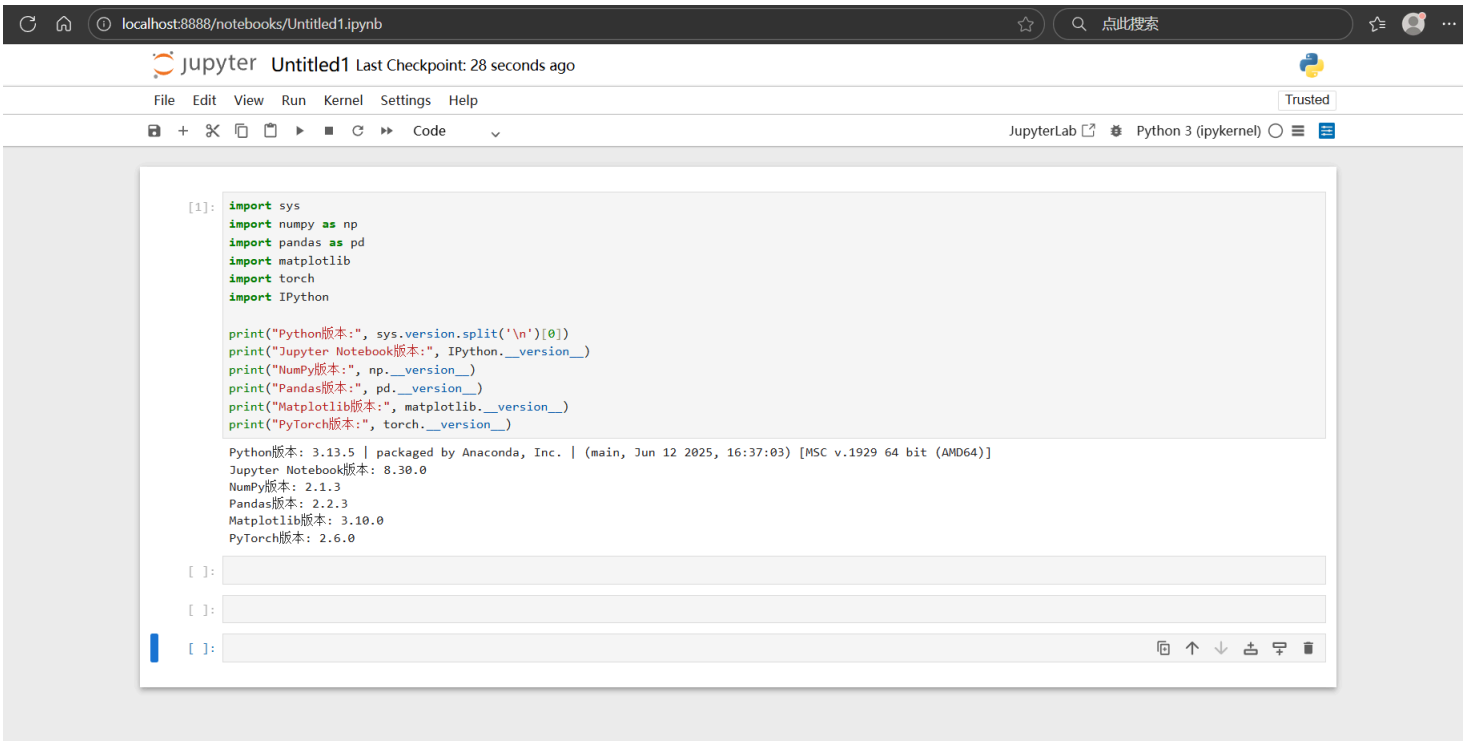
- ***人工智能***：AI是一个庞大的**科学体系**,人们基于机器学习、神经网络等工具，通过不断训练与迭代，以期达到机器像人类一样进行**思考与决策**（即人工智能）的目标，实现**智能行为的自动化**。形象点说，人工智能就像拥有超强大脑的人造智能体。
- ***机器学习***：机器学习是实现人工智能的一种**工具**。它指的是通过数据“**投喂**”，建立并逐渐完善**识别、思考、决策**的**模型**，使机器能够实现（或者说趋于实现）人类的智能行为。
- ***深度学习***：了解深度学习之前需要先了解*人工神经网络*，人工神经网络可以理解为以互联的计算单元为基础，通过分层结构建立的一个万能函数。而深度学习就是一种基于深度神经网络的*机器学习方法*，能够从海量的数据和知识中学习并**创建关联**，从而使机器具有人类一般的认知能力。（缺点是其可解释性差，让人难以理解从输入到输出的识别、思考、决策过程）

【区别】

- ***人工智能***：AI对“数据投喂”的依赖性没有那么强，其实现主要依赖于预设的**规则**或者**模型**，不过规则或模型的建立需要机器学习和深度学习，因此可以说**机器学习和深度学习都是实现人工智能的手段**。
- ***机器学习***：同样是通过投喂数据来实现输入到输出的非线性转换，机器学习需要人类工程师对数据进行**人工地**转换（提取特征、贴标签），并且由于人工设计的信息表意有限，机器学习的性能也有极限，不过ML的可解释性也因此增强。
- ***深度学习***：区别于机器学习，深度学习在接收输入的数据时会调用深度神经网络进行特征提取，实现更加**自动且复杂**的学习。

2.嵌入式AI发展方向

区别于云计算AI,嵌入式AI通过更加轻量化的模型与专用指令集等，实现**实时性**、**高可靠/安全性**、**高能效**（即从感知到行动与决策更加**准确、快速、安全、高效**）。不过嵌入式AI目前也存在单点故障维护困难、受物理环境影响大等缺点，因此目前来看嵌入式AI的发展仍需要通过端云协同以为万物赋能，不过当应用需求的碎片化、算力能效比等核心问题得到突破时，嵌入式AI有望**让智能从云端下沉至生产、生活物品**，达到“去中心化”，**让智能设备更自主、无缝地与生活衔接**的目标。



二、数学回归机器

1.监督学习和无监督学习 的*基本概念*:

> - 监督学习：机器根据**已标注的数据集**来建立、更新模型，最终**建立**起一套完善的映射规则（**函数**），达到只接收无标注的数据就能输出预测值的效果。
> - 无监督学习：机器根据**无标注的数据集**，自主建立模型以**探索**数据中隐含的**内在结构**、**模式**或**关系**。

2.监督学习和无监督学习的*异同*:

	监督学习 与 无监督学习
异	*监督学习*接收的数据是***已有标注特征***的，目标是建立一个可以根据无标签数据给出某种规则下对应的输出的***预测模型***； *无监督学习*接收的数据***无标注特征***，目标是自主发掘***数据中的结构与关系***。
同	二者都依赖***数据驱动***，且有相同的***算法基础***和***处理流程***（如数据预处理、特征工程）

3.（部分）监督学习的算法:

> - 【线性回归】:
>
>> - *基本概念*：线性回归是**监督学习**中的一种经典算法，它根据特征与目标之间存在的**线性关系**（或可近似为线性关系的关系）**拟合出线性模型**，用以解决数据的**回归问题**（即对连续数值的预测）。
>> - *原理*：此算法首先假设一个**线性模型**，然后用最小二乘法等**数学操作**以及梯度下降等**优化算法**拟合出一个相对准确、损失函数最小的**线性函数**。
>
> - 【逻辑回归】:
>
>> - *基本概念*：逻辑回归也是监督学习中的一种经典算法，用于解决数据的**判断或分类**问题。它将输入数据在线性模型中的输出通过函数映射到**概率区间**，以实现对特征输入的类型判断。

> - *原理*：先用线性拟合得出输入对应的输出，再用**Sigmoid函数**将输出转化为离散的概率值，最终同样利用**优化算法**得出拟合的**曲线型函数**。

- 【支持向量机（SVM）】：

> - *基本概念*：支持向量机是监督学习中一种强大的算法，可用于解决**回归**、**分类**问题。在SVM中，常使用将数据从原始特征空间映射到高维特征空间，以实现找出一个可将数据线性区分开来的决策边界的目标。
> - *原理*：利用**核技巧**将数据从原始特征空间**映射**到**更高维**的空间，最后在新空间里找到一个比新空间低一个维度的决策边界（**超平面**）将数据点变得线性可分。

4.无监督学习中的K-means聚类算法:

- 【 *基本概念* 】：这是**无监督学习**中的一种经典算法，用于处理**聚类**任务。它的运行目标是将一组**无标签的数据**自动聚集成K个不同的组，使得同组内的数据尽可能相似，而不同组的数据尽可能不同（可理解为“物以类聚”），而每个组的中心则由改组所有数据的均值（means）来决定。
- 【 *原理* 】：首先**随机**选择K个数据作为初始**数据组中心**，然后将每个数据分配到最相似的组中心，再通过**迭代**重新计算组中心（也就是优化聚类，使每组之间相似度更高，而不同组之间差异更大），当组中心不再变化时即输出聚类结果。

三、训练流程

1.训练集、验证集与测试集：

	训练集	验证集	测试集
基本概念	是训练模型时供机器学习和拟合函数的模型参数。	是训练数据中独立出来的一部分样本数据，在机器的参数学习阶段不会参与参数更新。	是在整个模型开发、训练完之后进行最终的、一次性的性能评估的数据集，且该数据集不同于训练集和验证集中的任何数据
作用	让模型对已有标签的数据进行特征提取以*建立*、*训练*模型。	在训练中实时*评估模型*，并为模型*选择算法*、*调整参数*，以*优化模型*。	用于*最终评估*模型对未知数据的泛化能力（可理解为性能）。

2.数据预处理:

> - **清洗**：**去除错误或不合规的数据**。如果没有清洗，异常数据会使模型产生严重的识别、思考、决策偏差，甚至报错以致无法学习。
> - **缺失值处理**：处理数据中的缺失部分（常见手段有删除、回归预测、分类预测等），保证**数据集完整性**，*避免模型因数据缺失而无法运行*。
> - **标准化**：在*特征分布未知或存在异常*时，将数据按其“标准差”进行缩放，使其符合均值为0、标准差为1的标准正态分布。目的是**消除数据本身量级不统一导致的权重不合理**。
> - **归一化**：在*数据边界已知*时，将所有数据按比例映射到同一个范围，以**解决数据的权重失衡**。（受异常值影响大，分布形状很容易被异常值破坏）
> - **特征工程**：通过改造或创造新的特征，提升**特征**对于目标问题的**代表性**和**表现力**，放大其有效权重以帮助模型更准确地抓住特征之间存在的关系，**提升预测的效能和准确率**。

3.基本概念的理解与阐述：

- **【过拟合、欠拟合】**：

> - 过拟合：模型过于**复杂**，在训练时对**噪声、异常值**也进行高精度的拟合，从而**缺失了对数据普遍规律的捕捉**，致使模型对新数据的**泛化能力差**。因此过拟合的模型在验证数据时表现很差。

- 欠拟合：模型过于**简单**，致使**预测值与特征误差过大**，无法捕捉到数据规律，也因此无法反映数据的真实普遍规律。

- **【样本、特征、标签】**：

  - 样本：数据集中的每一个**数据点**。（由一个特征向量及其对应标签组成）
  - 特征：用于描述每一个**样本的属性**，是模型进行预测的依据。
  - 标签：模型的**目标变量**（只存在于监督学习）。

- **【模型容量、泛化能力】**：

  - 模型容量：模型容量也称为表达能力，指的是模型**学习数据中的规律**以及**拟合函数**的**能力**。模型容量决定了模型能解决的问题的复杂度范围，模型容量过小时易出现欠拟合，过大则容易出现过拟合，因此模型容量也不是越大越好，需要根据**具体问题具体选择**。
  - 泛化能力：泛化能力指的是模型在处理从未见过的**全新数据上**的表现能力，它是**评估模型**的一个重要**指标**，直接决定了模型的使用价值和可靠性。

- **【超参数、参数】**：

  - 超参数：是模型外部的数据，在**训练开始前**就由开发者**手动设定**的配置选项，用于**模型优化**。（常见的超参数有学习率、正则化方法、训练代数等）
  - 参数：是模型从数据集中**学习得到**的，用于确定函数形式的未知的固定常量，**不可人为手动直接更改**。

- 模型评估：一个出色的模型应该具有**泛化能力强大**、**稳定性强**、**可解释性强**、**高能效**、**与应用场景高度契合**的特点。

四、深度学习

- Neural Network

  - **神经网络**由**输入层**、**隐藏层**、**输出层**组成，其中隐藏层中有一层及以上的层数。（神经网络中一层就是一组以相同/相似的特征作输入的神经元）。
    连接权重位于**相邻两层**之间，即输入层和隐藏层，隐藏层和输出层之间才有连接权重（连接权重表示**连接强度**，用于量化神经元的贡献程度和方向）。
    偏置只存在于隐藏层、输出层及每层的连接中，是独立于前一层输入的**固有参数**，是塑造决策平面的关键组成部分。神经元处理输入数据时，*权重和偏置都会参与运算*。

    - 【神经网络的**前向传播**】：输入信号经过神经网络这个复杂的函数，通过不断运算得出预测结果。作用是**得到预测结果**以及**拟合损失函数**。
    - 【神经网络的**反向传播**】：将损失函数的梯度**从输出层往输入层**方向一层层回传，从而更新每层的权重参数。作用是*指出每个参数对误差的影响权重，然后***优化模型**、**减小误差**。

- MLP:多层感知机是*神经网络的一种特定类型*，结构同样包含**输入层、隐藏层和输出层**，特点是**相邻两层的神经元之间互相全连接**。作用是找出数据集之间**隐含的、复杂的非线性关系**。

- 梯度下降算法：
  【基本原理】：该算法**沿着函数的梯度（导数）方向**不断**更新自变量**，使得函数的取值不断趋近最小值，直至达到**全局最优解**或局部最优解。
  【在优化神经网络参数中的作用】：用于求解损失函数的最小值，以提高神经网络的学习效率，使神经网络能够用更少的迭代此数、更稳定、准确地得到更好的预测结果。
  【基本步骤】：
  1.初始化学习参数
  2.前向传播、计算损失
  3.反向传播，（根据链式法则）计算损失函数对每一个参数的梯度（偏导数）

4.沿着负梯度方向，根据学习率和梯度更新学习参数
5.循环执行234直到损失函数的梯度不再下降（或达到预定迭代次数）。
【变体】：

- 批量梯度下降（**BGD**）：
  使用*全体数据*计算梯度，稳定准确，但是速度缓慢且内存需求高。
- 随机梯度下降（**SGD**）：
  使用*单个数据*计算梯度，收敛快、内存需求小，但是震荡严重、噪声大。
- 小批量梯度下降（**MBGD**）：
  使用*一小批样本*计算梯度，是BGD和SGD的折中方案，收敛较稳定，准确性也较好，是较为常用的一种方法。
- .....

- 常见激活函数:

  - ***Sigmoid***：常见于将数据集分为两个互斥的类别（**二分类**）问题的**输出层**。
  - ***Tanh***：常见于**输出包含正负特征**的**循环神经网络**的**隐藏层**。
  - ***ReLU***：是目前大部分神经网络的**隐藏层**的**默认激活函数**。
  - ***Leaky ReLU***：**输入始终为负值**时，**ReLU**激活函数会出现**神经元死亡**（梯度永远为0导致参数无法更新）的问题，这时候常用Leaky ReLU**替代ReLU**。
  - ***Softmax***：常见于**多分类问题**的**输出层**。

- **【损失函数与优化算法】**：

  - 损失函数：损失函数是用于**量化模型误差**的函数。常见类型有*绝对值损失函数*、*平方损失函数*、*Huber loss函数*、*交叉熵函数*。损失函数在模型训练中起到***指导模型拟合函数***、***衡量模型泛化能力***的作用。
  - 优化算法：优化算法是用于**寻找函数极值**的一种数学工具。常见类型有*梯度下降算法*、*自适应学习率算法*等。优化算法在模型训练中常用于***实现参数更新***、***寻找损失函数最小值***以达到全局最优解。

- **【梯度消失与梯度爆炸】**：解释什么是梯度消失和梯度爆炸现象、它们对神经网络训练的影响和产生原因//

  - 梯度消失：在反向传播过程中，**梯度急速变小**至**趋近于0**的情况称为梯度消失。它会导致**参数无法有效更新**。产生原因可能是*初始参数权重过小*；*层梯度小于1时，神经网络层数过多*；*激活函数存在过饱和区*，当输入过大或过小时，梯度值不断趋近于0。
  - 梯度爆炸：和梯度消失相反，在反向传播过程中，**梯度急速变大**至**趋近于无穷大**的情况称为梯度爆炸。它会导致**参数更新极其不稳定**甚至报错，导致**模型无法收敛**、训练失败。产生原因可能是*初始参数权重过大，层梯度大于1时，神经网络的层数过多*

- 正则化技术：正则化技术主要用于解决**过拟合**问题。常用的正则化手段有**L1正则化**和**L2正则化**，二者均用于**处理异常值的权重**。

tip: L1正则化会将不重要的特征权重直接压缩为0，从而实现特征的选择；L2正则化会让所有权重都均匀变小，从而限制异常特征对模型的影响

五、炼丹

- **心得体会**：有种触碰到AI模型训练内部的感觉，非常奇妙。遇到的问题是第一次在jupyter上运行100个迭代时遇到了jupyter notebook的内存限制。
- **简单理解**：跑出来的这些彩色图片是AI模型的训练集，而那些函数图像是模型处理训练集时通过提取特征，建立的输入和预测输出之间的函数模型。