Machine Learning Key Concepts

机器学习的目标
(Goal of Machine Learning)

提供的相关数据通过一次或连续多次的**训练**得到。智能体采用**统计学习方法**，通过正确的**概率分布**，来**预测**最有可能成功的动作。

> 学习：在外部刺激下记住大部分以往的**经验**。
>
> 智能体（agent）：从环境中接收信息的软件实体，选择达到特定目标的最佳行动并观察其结果；
>
> 动作（action）：值或决策。

自适应系统可以**从经验中学习**，并改变系统的行为以最大限度地达到特定目标。机器学习是这一系列技术的总称，通过实现自适应算法进行预测，并根据其**共同的特征**自动组织输入数据。

机器学习分类
(Classes of machine learing methods)

监督学习：主要功能是提供**误差的精确度量**（直接与**输出值**相比）。基于训练集，可以修正模型参数以减少**全局损失**函数。监督学习的目标是训练一个系统，使得该系统能够预测以前从未见过的样本。因此，需让模型具备**泛化**能力，以避免**过拟合**问题。

有时，除了预测实际的类别，最好是确定其**概率分布**，即通过离散概率分布来更好地描述实际输出。

常见的监督学习应用：

- 基于回归的预测分析或分类
- 垃圾邮件检测
- 模式检测
- 自然语言处理
- 情绪分析
- 自动图像分类
- 自动序列处理（例如音乐或语言）

> 过拟合：模型虽能正确拟合用于训练的样本，但对其他样本的预测误差却很大。
>
> **回归**和**分类**：基于连续的输出值则称为回归；基于离散量表示的结果（称为类别），则该过程被成为分类。
>
> 离散概率分布描述的应用：例如识别手写字母，不确定性可能产生多种可能结果。这时使 26 个字母表示的连续值**归一化**，使它们的总和为 1。
>
> 分类的本质：大多数算法尝试通过施加不同的条件来找到最佳的**分割超平面**。在分类过程中，目标是相同的，即减少**错误分类**的数量并增加对**噪声**的鲁棒性。

无监督学习：当需要对一组数据根据其相似度（或**距离**）进行分组（**聚类**）时，需要采用无监督学习方法。对于模糊的分类，好的聚类方法应该考虑**异常值**的存在并对它们进行处理，以增加内部一致性和**聚类之间的距离**。

增加内部一致性：意味着选择使局部密度最大化的分类。

异常值：是指一组测定值中与**平均值**的偏差超过**两倍标准差**的测定值，与平均值的偏差超过三倍标准差的测定值，称为高度异常的异常值。

常见的监督学习应用：

- 对象分割（例如用户、产品、电影、歌曲等）
- 相似性检测
- 自动标记

强化学习：当无法提供实际的监督数据时，强化学习使用基于环境提供的反馈来进行学习。此时反馈得到的更多是定性的信息，并不能确定其误差的精确度量。这种反馈通常被称为**奖励**（reward）（有时候，负面的反馈被定义为惩罚）。

行为顺序权衡：最理想的行为顺序应该能够得到最高的即时和累积奖励，做出最好的决策。一个动作可能是不完美的，但就整体策略而言，**累积奖励**最大化才是最重要的。

深度学习：

- 常见的深度学习应用：
- 图像分类
- 实时视觉跟踪
- 自动驾驶
- 物流优化
- 生物信息
- 语音识别

总结：

三个主要的学习策略是监督、无监督和强化学习。监督学习假设存在一个能提供**错误的精确度量**的教师，可以将正确输出与实际的输出进行比较，实现参数的校正。无监督学习没有教师，一切都直接从数据中学习。无监督学习算法尝试找出**一组数据的共同特征**，以便能够将新样本与正确的聚类相关联。根据一些**已知的特征**，监督学习通过将所有对象自动分类到特定类别中进行实例的分类，而无监督学习的常见应用是对具有后续标记或处理的实例进行自动分组。强化学习与监督学习类似，但它只接受关于其行为质量的**环境反馈**。虽然强化学习中**不能精确知道**什么是错以及错误的大小，但接收到的信息能帮助决定是继续采取策略还是选择另外的策略。

机器学习问题管道
(Pipeline of machine learning problems)

机器学习流程包括问题定义、评价指标选择、特征工程和处理过拟合等阶段。以下是机器学习较**规范化**的流程：

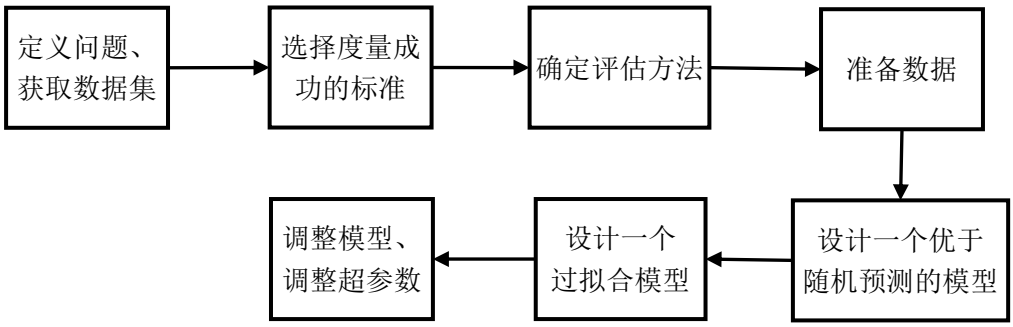


图 1-1 机器学习流程

（1）问题定义和数据集获取

定义问题时需要对输入数据和即将预测的对象（输出）非常清晰。接着，分析问题类型，确定是否为二分类问题、多分类问题，还是标量回归等问题？确定问题的类型有助于后面对模型结构、损失函数等具体技术的选择。

（2）选择度量成功的标准

在模型训练的过程中，我们可能需要实时地观察模型的训练情况，以控制某些因素的变化。有了这个衡量标准，我们才能使模型在训练时往正确的方向收敛，而不会像无头苍蝇，晕头转向！这个衡量标准可以是**模型准确度（accuracy）**，**精确率（precision）**和**召回率（recall）**等。作为对模型结果成功与否的评判标准，这个衡量标准可以指导我们选择**损失函数（Loss Function）**和**优化方法（Optimizer）**等。

（3）确定评估方法

三种常见的评估方法（evaluation protocols）：（1）维持一个验证集不变，这通常在数据充足的情况下使用；（2）k-折交叉验证，数据量较少情况下；（2）**迭代的 k-折交叉验证**，数据量较少而要求高精度模型评价时使用。

（4）准备数据

首先，应该把数据转化成一种机器学习模型能够输入的格式。假设模型是一个深度神经网络，则数据应该被格式化为张量。这些张量所取的值通常应该被缩放成较小的值，例如 Xception 和 InceptionV3 要求输入值在[-1, 1]范围内。较常见的是缩放至[0, 1]范围内。如果不同的特征在不同的范围取值，那么数据应该被**归一化**。而对于小数据问题，可能做一些特征工程。一旦你的输入数据和目标数据的张量准备好，你就可以开始训练模型了。

（5）设计一个优于**随机预测**的模型

假设我们的输出和输入存在某些关联，即输出能够由输入提供的信息来推测，而且我们有足够的数据用于训练，那么我们才可能建立有效的分类器模型。搭建模型有 3 个需要关注的点：①选择 last-activation；②选择损失函数；③选择优化器并设置学习率。下表列出的内容对于选择常见机器学习问题 last-activation 和损失函数具有指导意义。

表 1-1 机器学习问题类型及激活函数与损失函数

Problem type	Last-layer activation	Loss function
Binary classification	sigmoid	binary_crossentropy
Multi-class, single-label classification	softmax	categorical_crossentropy
Multi-class, multi-label classification	sigmoid	binary_crossentropy
Regression to arbitrary values	None	mse
Regression to values between 0 and 1	sigmoid	mse or binary_crossentropy

（6）设计一个过拟合模型

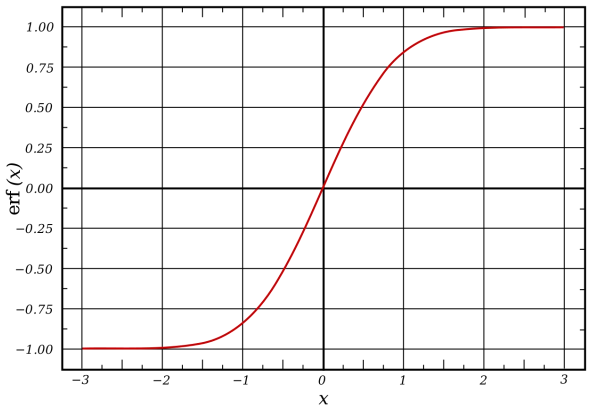
当创建完一个具有预测能力的模型之后，我们需要做的就是对模型进行改进，使模型足够强大。例如对于神经网络模型，可以添加更多的卷积层、使各个层更巨大，或者增加训练的 epochs 数量等。在训练的时候，我们需要时刻监测训练损失值和验证损失值。当模型在验证集上的表现开始下滑时，通常就意味着模型已经过拟合了。

（7）调整模型并调整超参数

这一步主要是为了获得理想的模型，即模型既不欠拟合也不过拟合。因为上一步我们得到的模型是过拟合的，所以这一步主要解决的问题是过拟合问题。常见的调整模型的方法有：添加 Dropout 层、移除某些层、增加 L1/L2 正则化、尝试不同的超参数寻找最优配置，以及迭代性地选择特征（尝试新的 filter，去除不具信息性的 filter）。

Sigmoid Function：sigmod 函数是一个具有“S”形状的曲线的数学函数。通常 sigmoid 函数指的是如图所示逻辑函数的特例，它由以下公式定义：

$$S(x)=\frac{1}{1+e^{-x}}=\frac{e^{x}}{e^{x}+1}$$



Softmax：或称归一化指数函数，是逻辑函数的一种推广。它能将一个含任意实数的 K 维向量 z “压缩”到另一个 K 维实向量 $\sigma(z)$ 中，使得每一个元素的范围都在 $(0,1)$ 之间，并且所有元素的和为 1。该函数的形式通常按下面的式子给出：

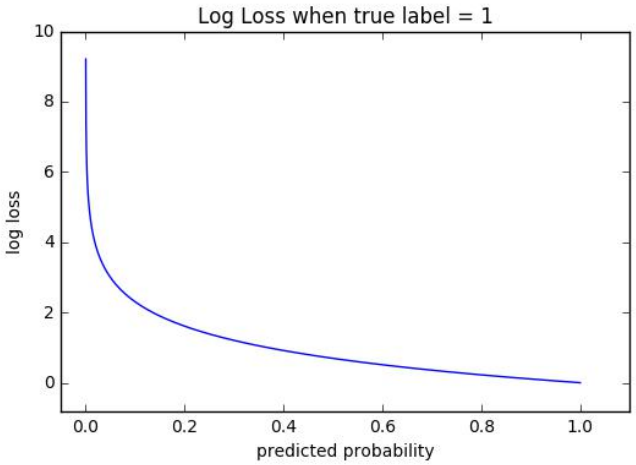
$$\sigma(z)_j=\frac{e^{z_j}}{\sum_{k=1}^{K}e^{z_k}}\quad \text{for } j=1,\ldots,K.$$

Softmax 函数实际上是有限项离散概率分布的梯度对数归一化。因此，Softmax 函数在包括多项逻辑回归，多项线性判别分析，朴素贝叶斯分类器和人工神经网络等的多种基于概率的**多分类问题**方法中都有着广泛应用。

sigmoid,softmax,binary/categorical cross-entropy 之间的联系：sigmoid 和 softmax 是神经网络**输出层**使用的激活函数，分别用于两类判别和多类判别;binary cross-entropy 和 categorical cross-entropy 是对应的**损失函数**。

MSE （Mean Squared Error）：均方误差。

Cross Entropy Loss Function：交叉熵损失，或称对数损失（Log loss），亦被称为逻辑回归损失（Logistic regression loss），衡量**分类模型**的性能，分类模型的输出是介于 0 和 1 之间的**概率值**。交叉熵损失随着预测概率偏离实际标签而增加。因此，若实际观测值为 1 时，预测为 0.012 的概率将是不理想的，会导致高损失值。一个完美的模型具有对数损失值 0。



Binary cross-entropy：在二分类中(M=2)，交叉熵可由下式计算：

$$-\left(y\log\left(p\right)+\left(1-y\right)\log\left(1-p\right)\right)$$

Categorical cross-entropy：在多分类中（M>2）,我们根据观察结果计算每个分类标签的单独损失，并求和：

$$-\sum_{c=1}^{M}y_{o,c}\log\left(p_{o,c}\right)$$

- M - 分类的数量
- y - 二进制指示器（0 或 1），若分类标签 c 是观测 o 的正确分类
- p - 预测观测 o 为类别 c 概率

机器学习的重要元素
Elements of machine learning

参考资源
Reference

http://wiki.fast.ai/index.php/Log_Loss

https://ml-cheatsheet.readthedocs.io/en/latest/loss_functions.html