第五章 机器学习基础

1. 学习算法
   1. 任务T

通常机器学习任务定义为机器学习系统应该如何处理**样本**（example）。

样本是我们从某些机器学习系统处理的对象或事件中收集到的已经**量化的特征**的集合。

我们通常将样本表示成一个向量其中向量的每个元素是一个特征。列入一张图片的特征通常是这张图片的像素值。

常见的机器学习任务：

**分类**：计算机程序需要指定某些输入属于k类的哪一类。为了完成该任务，学习算法通常会返回一个函数。当时，模型将向量x所代表的输入分类到数字码y所代表的类别

**输入缺失分类**：当一些输入可能丢失时，学习算法必须学习一组函数，而不是单个分类函数。

**回归**：在这类任务中，计算机程序需要对给定的输入预测数值。

**转录**：这类任务中，机器学习系统观测一些相对非结构化表示的数据，并转录信息为离散的文本形式。

**机器翻译**：在机器翻译任务中，输入是一种语言的符号序列，计算机程序必须 将其转化成另一种语言的符号序列。

**结构化输出**：结构化输出任务的输出是向量或者其他包含多个值的数据结构， 并且构成输出的这些不同元素间具有重要关系。

**异常检测**：在这类任务中，计算机程序在一组事件或对象中筛选，并标记不正 常或非典型的个体。

**合成和采样**：在这类任务中，机器学习程序生成一些和训练数据相似的新样本。

**缺失值填补**：在这类任务中，机器学习算法给定一个新样本，中的某些元素确实，算法必须填补这些缺失值。

**去噪：**算法根据损坏后的样本预测干净的样本，或者更一般地预测条件概率分布 。

* 1. 性能度量p

对于诸如分类、缺失输入分类和转录任务，我们通常度量的**准确率**（accuracy），准确率是指该模型输出正确结果的样本比例。同样可通过**错误率**（error rate）得到相同信息，错误率是指该模型输出错误结果的样本比率。我们通常把错误率称为0-1损失的期望。

我们使用**测试集**（test set）数据来评估系统性能，将其与训练机器学习系统的训练集数据分开。

* 1. 经验E

机器学习算法大致分类为**无监督**（unsupervised）算法和**监督**（supervised）算法。

深度学习的大部分学习算法可以被理解为在整个**数据集**（dataset）上获取经验。

**无监督学习算法**（unsupervised learning algorithm）

训练含有很多特征的数据集， 然后学习出这个数据集上有用的结构性质。深度学习中通常要学习生成数据集的整个概率分布。

**监督学习算法**（supervised learning algorithm）

训练含有很多特征 数据集，不过数据集的样本都有一个**标签**（label）或**目标**（target）

注意：无监督学习和监督学习之间界限时模糊的原因如下：

概率的链式法则表明对于向量，联合分布可以分解成：

该分解意味着我们可以将其拆分成n个监督学习问题，来解决表面上的无监督学习问题。

另外我们求解监督学习问题时，也可以使用传统的无监督学习策略学习联合分布，然后推断

**强化学习**（reinforcement learning）算法会和环境进行交互，所以学习系统和它的训练过程会有反馈回路。

表示数据集的常用方法是**设计矩阵**（design matrix）设计矩阵的梅伊昂包括一个不同的样本，每一列对应不同的特征。

当每一个样本都能表示成向量，并且这些向量的维度相同，才能将一个数据集表示成设计矩阵。若出现不同类型的异构数据，可以将数据集表示成m个元素的集合。这种表示方式以为则样本向量和可以有不同的大小。

在监督学习时样本包含一个标签或目标和一组特征。通常在处理包含观测特征的设计矩阵X的数据集时，也会提供一个标签向量y，其中表示样本i的标签。有时标签可能不止一个数，例如训练语音模型转录整个句子，那么每个句子样本的标签时一个单词序列。

* 1. 示例：线性回归（linear regression）

线性回归解决回归问题。我们的目标时建立一个系统，将向量作为输入，预测标量作为输出。线性回归的输出时其输入的线性函数，令表示模型预测应取值，定义输出为

其中是**参数**（parameter）向量。

参数是控制系统行为的值。我们可以将看作是一组决定每个特征如何影响预测的**权重**（weight）。

定义任务T:通过输出从预测y。

定义性能度量——P：

假设我们有 m 个输入样本组成的设计矩阵，我们不用它来训练模型，而是评 估模型性能如何。我们也有每个样本对应的正确值 y 组成的回归目标向量。因为这 个数据集只是用来评估性能，我们称之为**测试集**（test set）。

度量性能的一种方法是计算模型在测试集上的**均方误差**（mean squared error， MSE）。

若使用表示模型在测试集上的预测值，那么均方误差表示为：

直观上，当时，会发现误差降为0。也可以知道：

当预测值和目标值之间的欧几里得距离增加时，误差也会增加。

为构建一个机器学习算法，我们可以通过观察训练集获得经验，减少以改进权重。一种直观的方式时最小化训练机上的均方误差，即。

最小化 ，我们可以求解其导数为0 的情况

而通过上述最后一个表达式给出解的系统方程被称为正规方程（normal equation）。

而附加额外参数（截距项b）的模型中：

因此从参数到预测的映射仍是一个线性函数，而从特征到预测的映射是一个仿射函数。截距项b通常被称为仿射变换的**偏置**（bias）参数。

1. 容量、过拟合和欠拟合
   1. 基础概念

在先前未观测到的输入上表现良好的能力被称为**泛化**（generalization）。

在训练机器学习模型时，我们可以使用某个训练集，在训练集上计算一些被称为**训练误差**（training error）的度量误差，目标是降低训练误差。

机器学习和优化不同的地方在于我们也希望**泛化误差**（generalization error）（也被称为**测试误差**（test error））很低。

通常我们度量模型在训练集中分出来的**测试集**（test set）样本上的性能来评估机器学习模型的泛化误差。

训练集和测试集数据通过数据集上被称为**数据生成过程**（data generation process）的概率分布生成。通常我们会做一系列被统称为**独立同分布假设**（i.i.d.assumption）的假设。这个假设含义是：每个数据集中的样本都是**彼此相互独立**的，并且训练集和测试集是**同分布**的，采样自相同的分布。这个假设使我们能够在单个样本的概率分布描述数据的生成过程，然后相同的分布可以用来生成每一个训练样本和每一个测试样本。我们将这个共享的潜在分布称为**数据生成分布**（data generating distribution）。

训练误差和测试误差之间的直接联系是，随机模型训练误差的期望和该模型测试误差的期望是一样的。

决定机器学习算法效果是否好的啷个因素：

降低训练误差

缩小训练误差和测试误差的差距

分别对应机器学习的两个主要挑战：**欠拟合**（underfitting）和**过拟合**（overfitting）欠拟合是指模型不能再训练集上获得足够低的误差。而过拟合是指训练误差和测试误差之间的差距过大。

统计学习理论提供了量化模型容量的不同方法，在这些中，最有名的是**Vapnik——chervonenkis维度**（VC）。VC维度量二元分类器的容量。VC维定义未该分类器能够分类的训练样本的最大数目，假设存在 m 个 不同 x 点的训练集，分类器可以任意地标记该 m 个不同的 x 点，VC维被定义为 m 的最大可能值。

为了考虑容量任意高的极端情况，我们介绍**非参数**(non-parametric)**模型**的概念。

有时，非参数模型仅是一些不能实际实现的理论抽象，然而可以设计一些使用的非参数模型，使得他们的复杂度和训练集大小有关。例如：**最邻近回归**（nearest neighbor regression）

**最邻近回归**（nearest neighbor regression）模型存储了训练集中所有的X和y。当需要为测试点x进行分类时，模型会查询训练集中离该点最近的点，并返回相关的回归目标。再最近向量不唯一的情况下，如果允许算法对所有离x最近的

关联的求平均那么该算法会在任意回归数据集上达到最小可能的训练误差。

我们也可以将参数学习算法嵌入另一个增加参数数目的算法来创建非参数学习算法。

理想模型假设我们能够预先知道生成数据的真实概率分布。从预先知道真实分布预测而出现的误差称为**贝叶斯误差**（Bayes error）。

* 1. 没有免费午餐定理

机器学习的**没有免费午餐定理**（no free lunch theorem）表明 (Wolpert, 1996)，在所有可能的数据生成分布上平均之后，每一个分类算法在未事先观测的点上都有相同的错误率。换言之，在某种意义上，没有一个机器学习算法总是比其他的要好。我们能够设想的最先进的算法和简单地将所有点归为同一类的简单算法有着相同的平均性能（在所有可能的任务上）。

幸运的是，这些结论仅在我们考虑所有可能的数据生成分布时才成立。在真实 世界应用中，如果我们对遇到的概率分布进行假设的话，那么我们可以设计在这些 分布上效果良好的学习算法。

这意味着机器学习研究的目标不是找一个通用学习算法或是绝对最好的学习算 法。反之，我们的目标是理解什么样的分布与人工智能获取经验的 ‘‘真实世界’’ 相 关，什么样的学习算法在我们关注的数据生成分布上效果最好。

* 1. 正则化

1. 超参数和验证集

大多数机器学习算法都有超参数，可以设置来控制算法行为。

有时一个选项被设为学习算法不用学习的超参数，是因为它太难优化，如果在训练集上学习超参数，这些超参数总是趋向于最大可能的模型容量，导致过拟合，为了解决这一问题，需要一个训练算法观测不到的**验证集**（validation set）。

用于学习参数的数据子集通常仍被称为训练集，尽管这会和整个训练过程用到的更大的数据集相混。用于挑选超参数的数据子集被称为**验证集**（validation set）。通常，80% 的训练数据用于训练，20% 用于验证。

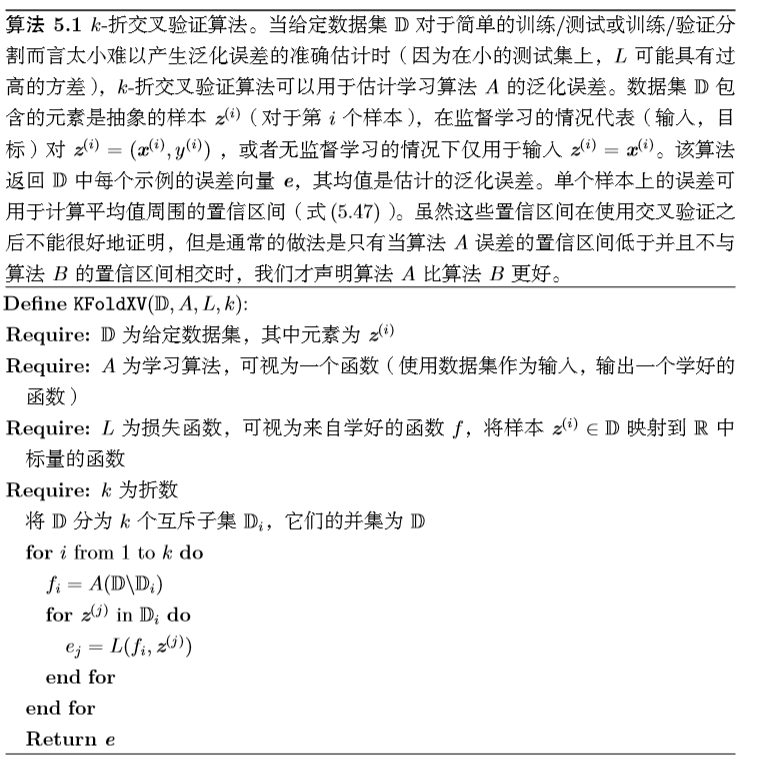
* 1. 交叉验证

在原始数据上随机采样或分理处不同的数据集重复训练和测试。

最常见的是**k-折交叉验证过程**。

将数据集分成k个不重合的子集，测试误差可以估计为k次计算后的平均测试误差。在第i次测试时，数据的第i个子集用于测试集，其他数据用于训练集。带来的一个问题是不存在平均误差方差的无偏估计，但通常可以使用近似来解决。

k-折交叉验证算法：



1. 估计、偏差和方差
2. 最大似然估计
3. 贝叶斯统计
4. 监督学习算法
5. 无监督学习算法
6. 随机梯度下降
7. 构建机器学习算法
8. 促使深度学习发展的挑战