课程简介

1. 什么是学习?

“学习是系统通过经验提升性能的过程。”

--- Herbert Simon

卡内基·梅隆大学

图灵奖(1975)

人工智能，人类认知心理学

诺贝尔经济学奖(1978)

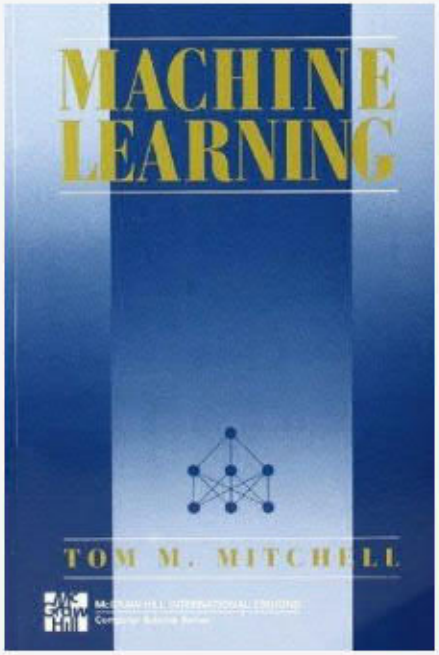
经济组织内的决策过程

课程简介

由Tom Mitchell 给出的更加数学化的定义

- Ability for machines to learn without being *explicitly* programmed

"A computer program is said to learn from experience *E* with respect to some class of tasks *T* and performance measure *P* if its performance at tasks in *T*, as measured by *P*, improves with experience *E*." --- Mitchell, T. (1997). Machine Learning. McGraw Hill. p. 2.



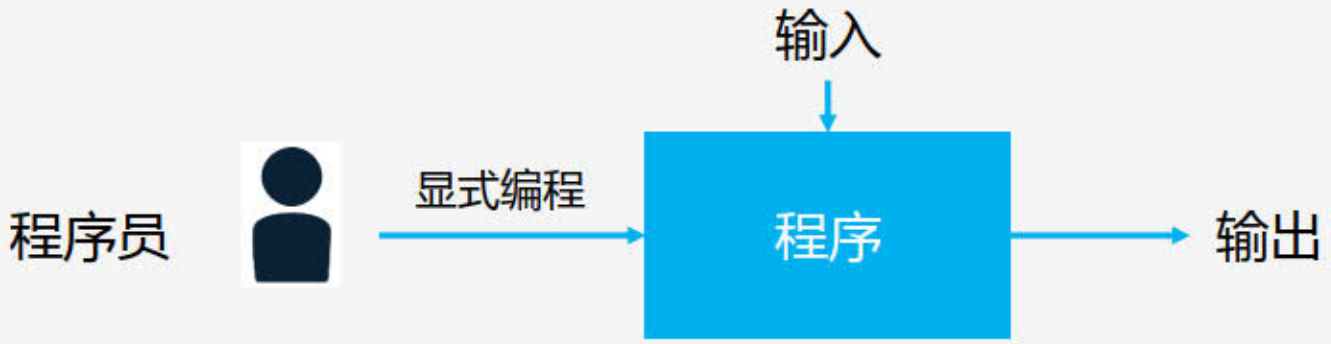
课程简介

由Tom Mitchell 给出的更加数学化的定义

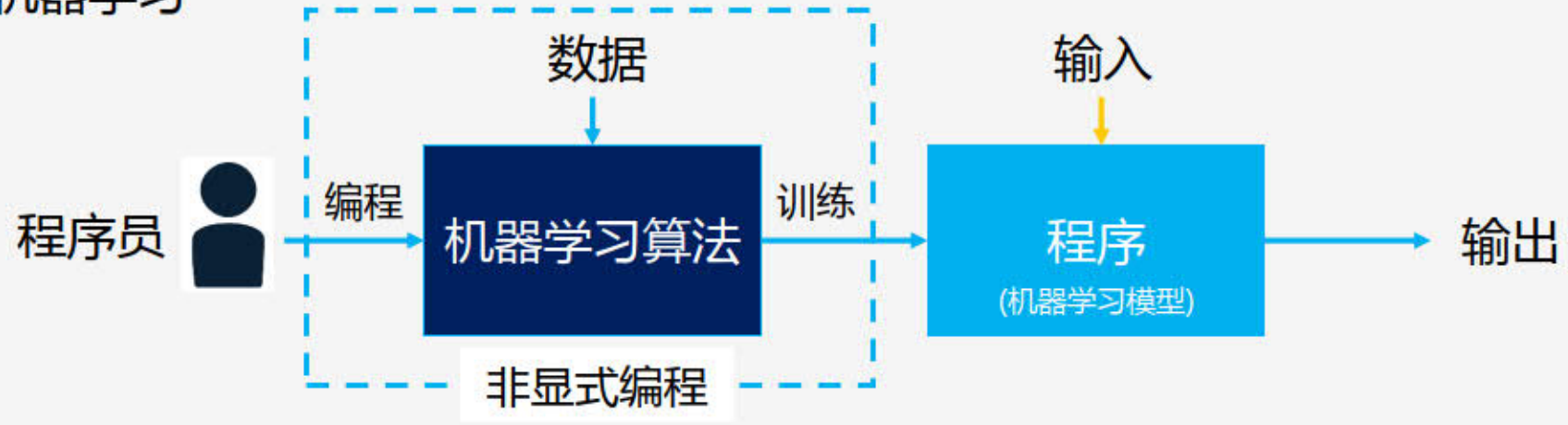
- 机器学习是一门研究学习算法的学科，这些算法能够：
  - 在某些任务 *T* 上
  - 通过经验 *E*
  - 提升性能 *P*
  - 非显式编程
- 一个学习任务可以由三元组 *<T,P,E>* 明确定义
  - 为何不能用知识、经验或者专业技能来训练机器呢？
  - 人类总能够解释他们的专业技能吗？

课程简介

- 传统编程



- 机器学习



课程简介

机器学习在什么情况下具有优势?

应用情形:

- 模型基于大量数据
  - 例子: Google 网络搜索, 垃圾邮件识别
- 输出必须是个性化的
  - 例子: 新闻/物品/广告推荐
- 人类不能解释专业知识
  - 例子: 语音/人脸识别, 异常行为检测
- 人类的专业知识不存在
  - 例子: 在火星上导航

课程简介

两种机器学习类型

- 预测
  - 根据数据预测所需的输出（监督学习）
  - 生成数据实例（无监督学习）
- 决策
  - 在动态环境中采取行动（强化学习）
    - 转变到新的状态
    - 获得即时奖励
    - 随着时间的推移最大化累积奖励

3. 机器学习基本思想

机器学习类型：

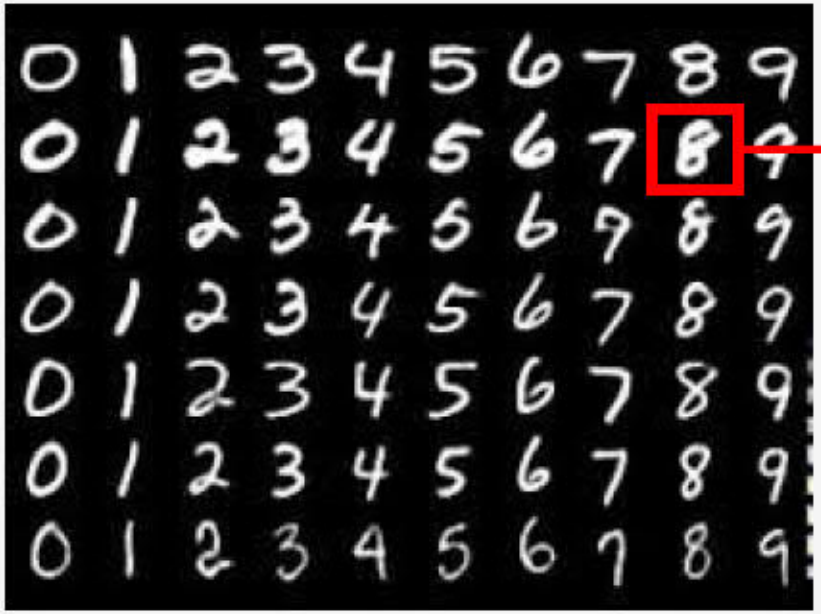
- 监督学习
  - 给定数据和标签，预测所需的输出
- 无监督学习
  - 分析和利用隐式数据模式/结构
- 强化学习
  - 学习在动态环境中动作执行的决策，并获得尽可能多的奖励值

课程简介

任务（T）：

- 给定一个手写数字图片集合 $x \in [0,255]^{28\times28}$ 和其对应的标签 y∈ [0,9] 找到一个映射函数

$$f: x \rightarrow y$$



经验（E）：

- 一个用于训练的标注好0~9标签的图片集

性能（P）：

- 这个映射函数在未标注标签的测试集上的识别准确率

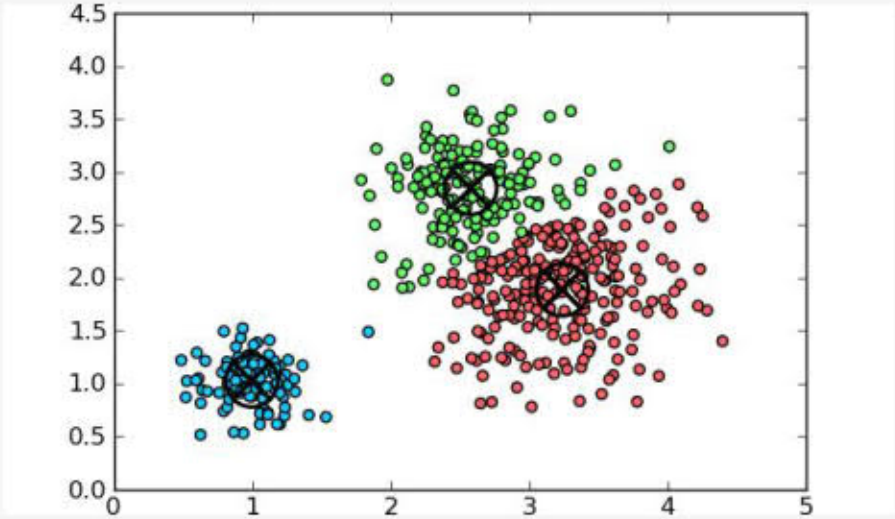
“监督学习（Supervised Learning）”

https://www.npmjs.com/package/mnist

课程简介

任务（T）：

- 如何将一组文档“聚类”（Cluster）成 k 个组，使得属性“相似”的文档出现在同一组中？



经验（E）：

- 一个用于训练的无标签的文档集合

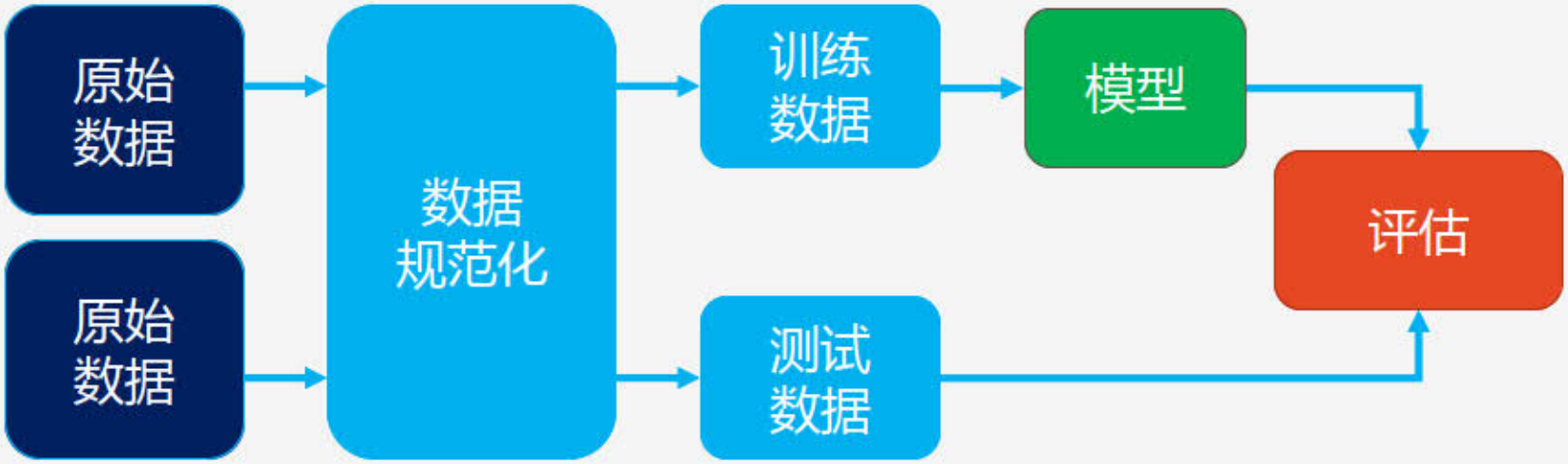
性能（P）：

- 所有文档的属性坐标到聚类中心的平均距离

“无监督学习（Unsupervised Learning）”

课程简介

机器学习过程：



- 基本假设：在训练和测试数据中存在相同的模式（pattern）

课程简介

监督学习：

定义

- 给定带标签的训练数据集：$D = \{(x_i, y_i)\}_{i=1,2,...N}$，其中$x_i$为特征数据，$y_i$为其对应的标签，让机器学习一个从特征数据映射到标签的函数映射

$$y_i \simeq f_\theta(x_i)$$

- 函数集 $\{f_\theta(\cdot)\}$ 被称为假设空间
- 学习的过程即为参数 θ的更新

如何学习?

- 更新参数以使预测结果接近真实的标签
  - 学习目标是什么?
  - 如何更新参数?

课程简介

监督学习：

学习目标

- 使预测结果接近真实的标签

$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(y_i, f_\theta(x_i))$$

- 损失函数$\mathcal{L}(y_i, f_\theta(x_i))$用来衡量标签和预测结果之间的误差
- 损失函数的定义取决于数据和任务
- 最常见的损失函数：平方误差（squared loss）

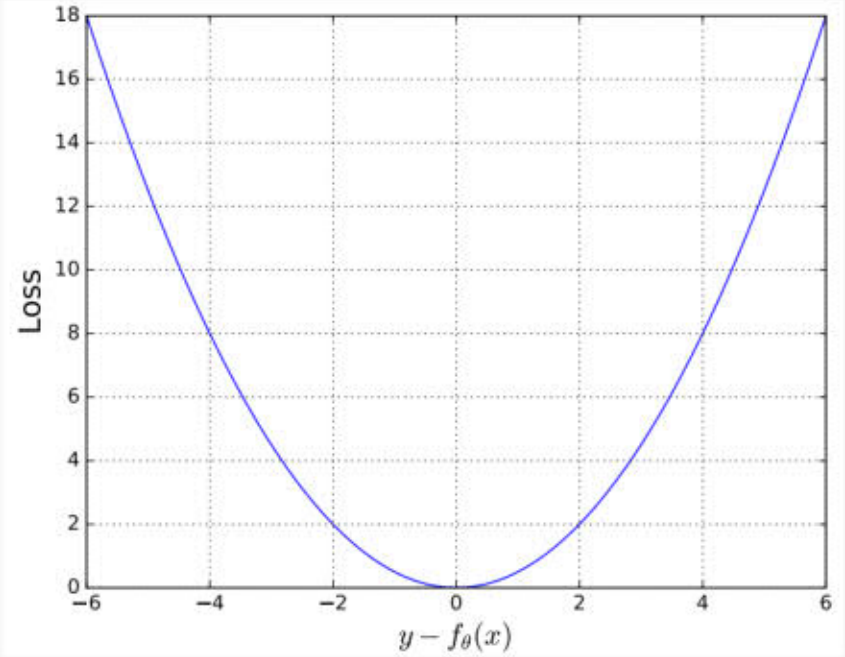
$$\mathcal{L}\big(y_i, f_\theta(x_i)\big) = \frac{1}{2}\big(y_i - f_\theta(x_i)\big)^2$$

课程简介

监督学习：

平方误差

$$\mathcal{L}(y_i, f_\theta(x_i)) = \frac{1}{2}(y_i - f_\theta(x_i))^2$$

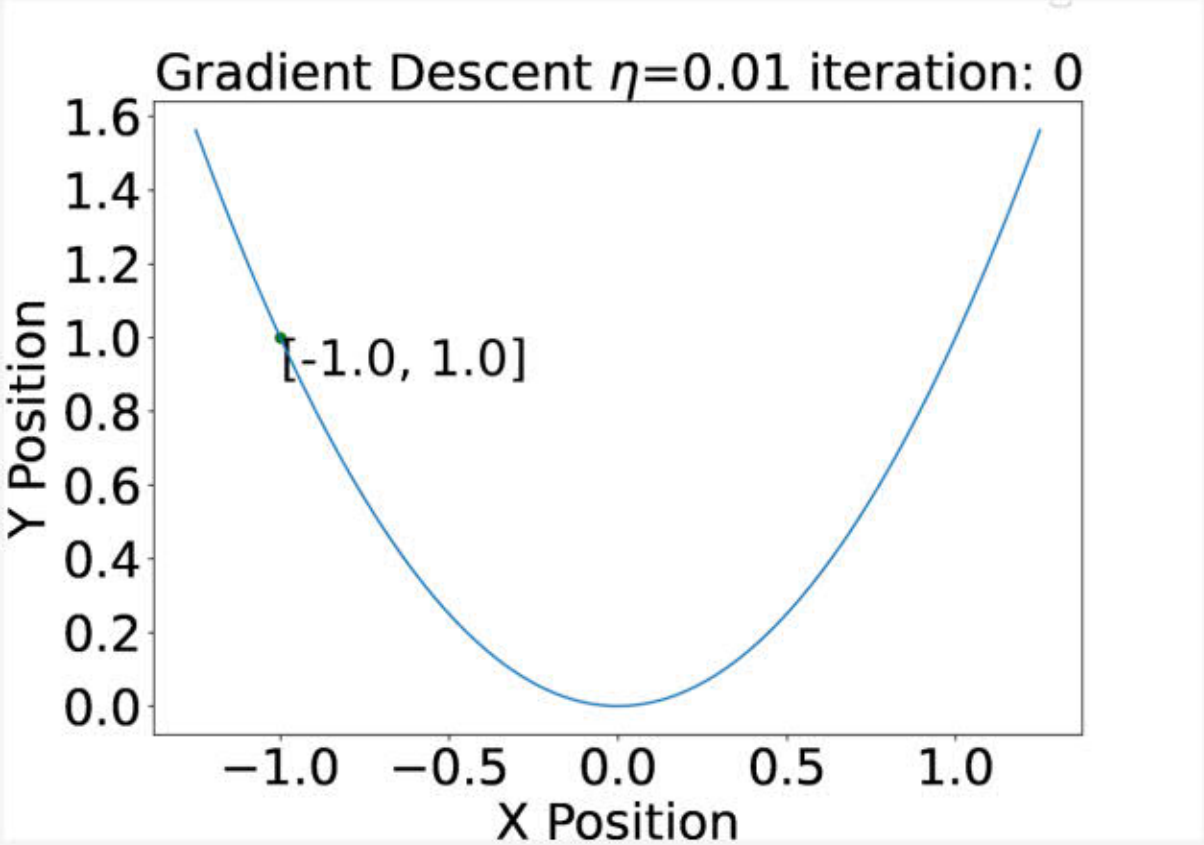


- 距离越远，得到的惩罚更多
- 容忍小距离（误差）
  - 观察噪声等
  - 泛化性

课程简介

梯度学习方法:

$\mathcal{L}(\theta)$



$$\theta_{new} \leftarrow \theta_{old} - \eta \frac{\partial \mathcal{L}(\theta)}{\partial \theta}$$

第一章：模型选择

1.1 欠拟合与过拟合

1.2 正则化

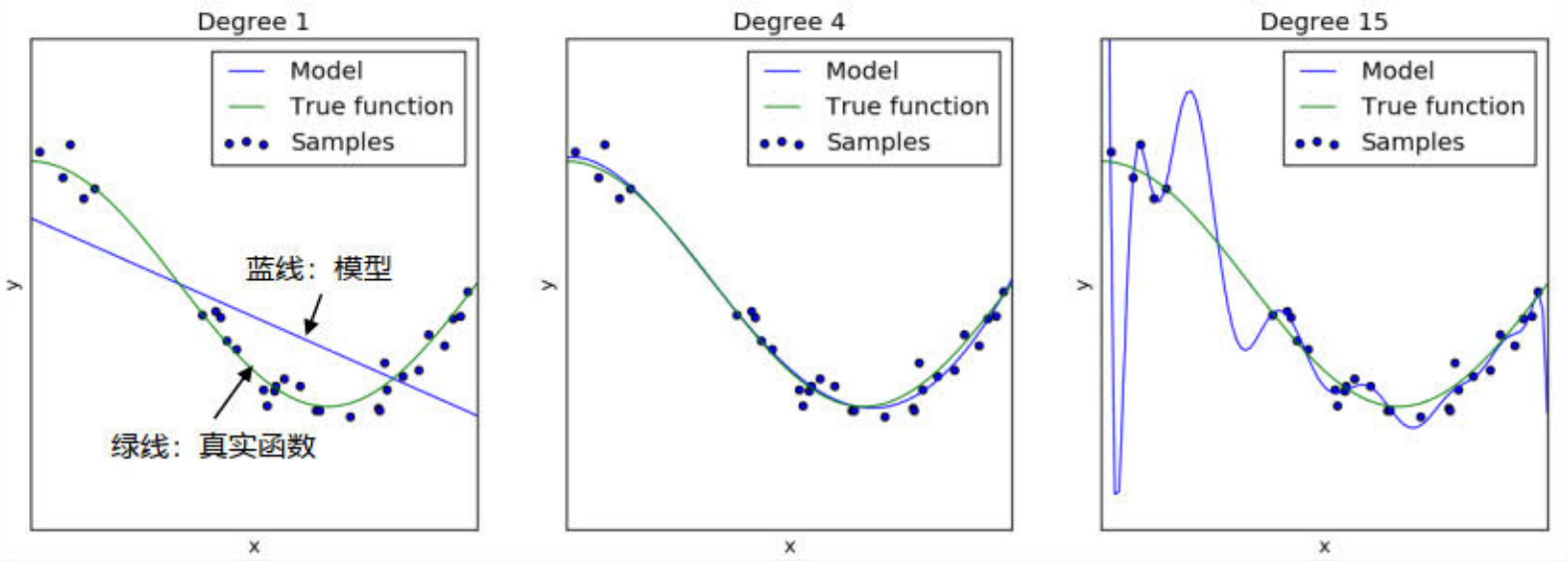
1.3 奥卡姆剃刀原则

1.4 交叉验证

1.5 模型泛化性

1.1 欠拟合与过拟合

- 下面哪个模型是最好的?



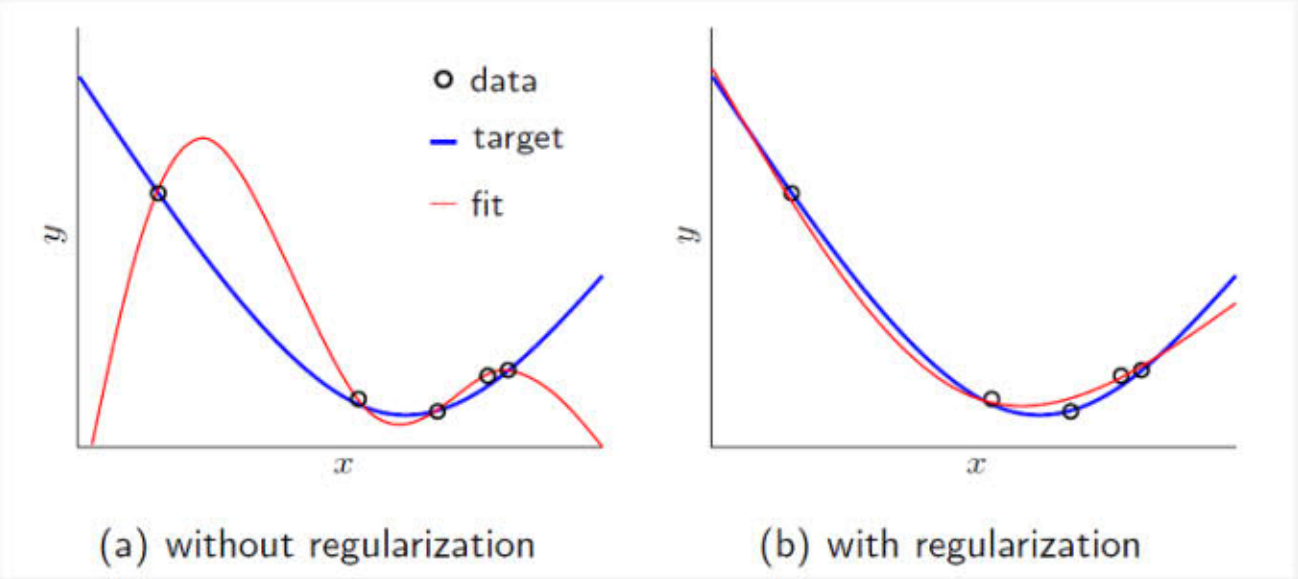
线性模型：欠拟合　　四阶模型：合适　　十五阶模型：过拟合

- 当统计模型或机器学习算法无法捕捉数据的基础变化趋势时,就会出现**欠拟合**。
- 当统计模型把随机误差和噪声也考虑进去而不仅仅是考虑数据的基础关联时，就会出现**过拟合**。

1.2 正则化

- 添加参数的惩罚项，防止模型对数据过拟合

$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}\left(y_i, f_\theta(x_i)\right) + \lambda \Omega(\theta)$$



(a) without regularization　　(b) with regularization

1.2 正则化

经典正则化方法

- L2正则化（岭回归Ridge）

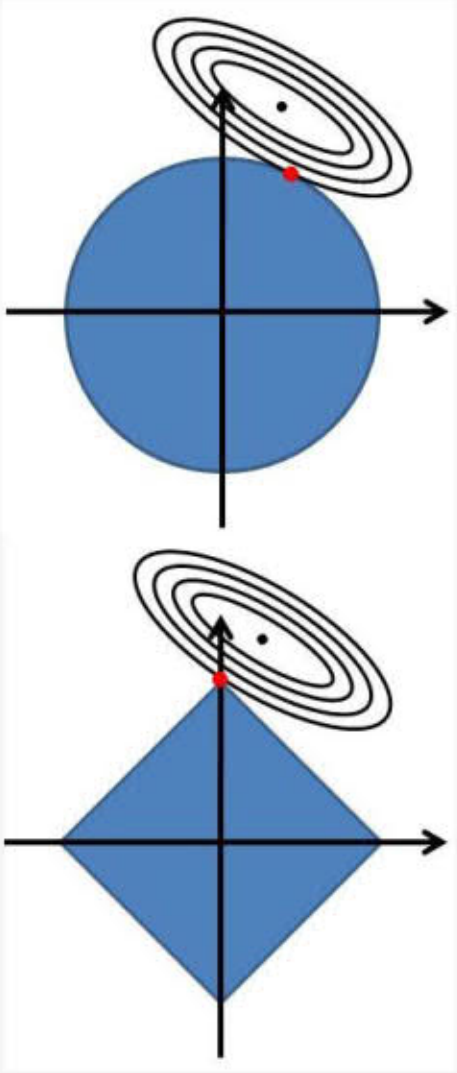
$$\Omega(\theta) = \|\theta\|_2^2 = \sum_{m=1}^{M} \theta_m^2$$

$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}\left(y_i, f_\theta(x_i)\right) + \lambda \|\theta\|_2^2$$

- L1正则化（拉索回归LASSO）

$$\Omega(\theta) = \|\theta\|_1 = \sum_{m=1}^{M} |\theta_m|$$

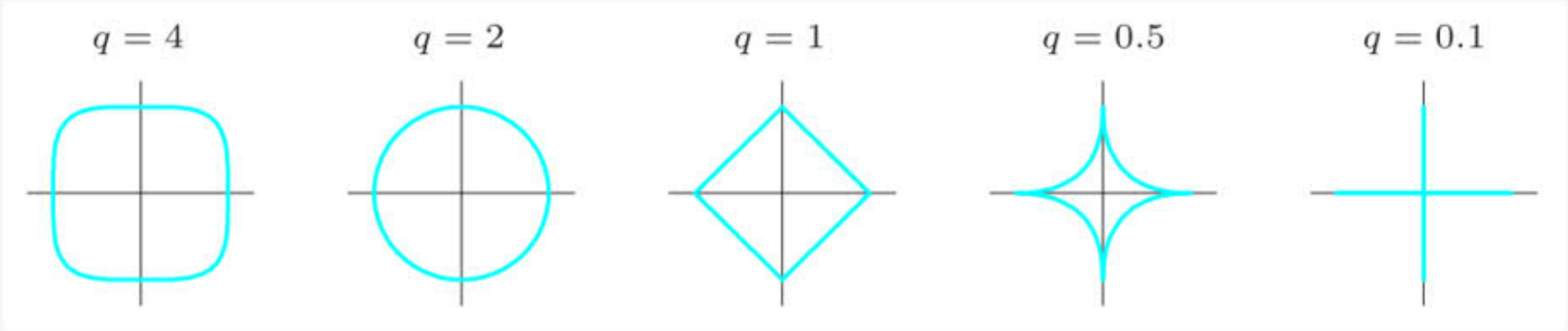
$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}\left(y_i, f_\theta(x_i)\right) + \lambda \|\theta\|_1$$



1.2 正则化

经典正则化方法

- 常值$\sum_j |\theta_j|^q$的数值分布图



Ridge　　LASSO

- 当$q \leq 1$ 的时候，模型进行稀疏性学习
- 很少会用$q > 2$来进行正则化
- 99%的情形下都取$q = 1$ 或 2

1.3 奥卡姆剃刀原则

- 有多个假设模型时，我们应该选择假设条件最少的建模方法。

- 函数集$\{f_\theta(\cdot)\}$被称作假设空间

$$\min_{\theta} \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}\left(y_i, f_\theta(x_i)\right) + \lambda\Omega(\theta)$$

原始损失　　基于假设的罚值

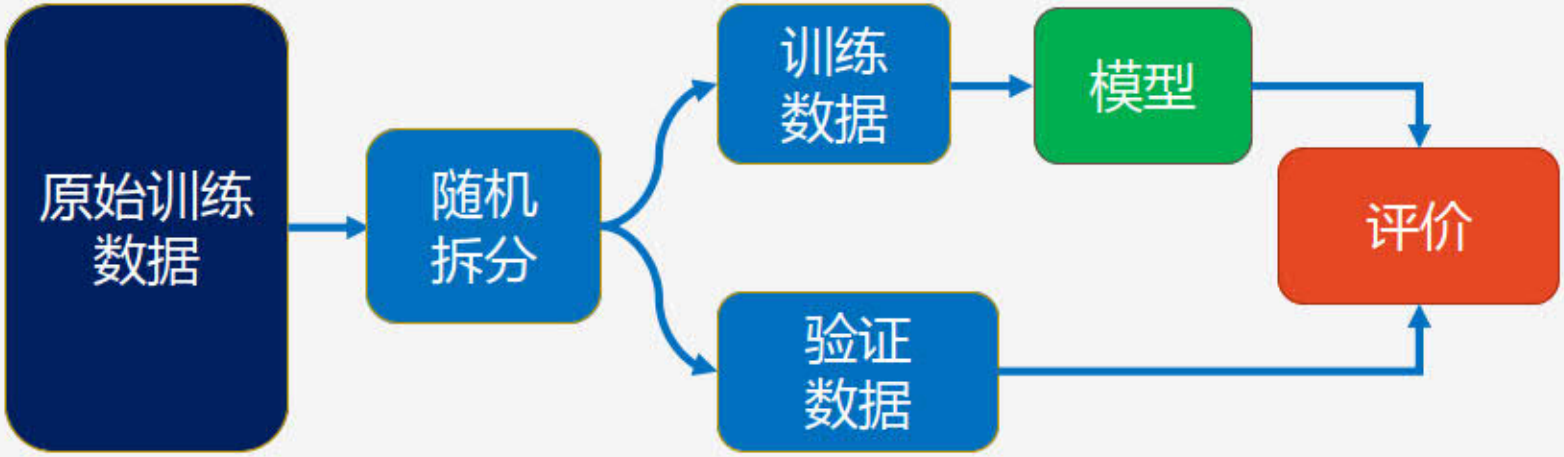
1.3 奥卡姆剃刀原则

模型选择

$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}\left(y_i, f_\theta(x_i)\right) + \lambda \|\theta\|_2^2$$

- 一个机器学习的解决方案的模型包含参数θ和超参数λ
- 超参数
  - 定义模型的更高层次的概念，如复杂性或学习能力。
  - 在标准模型训练过程中**无法直接从数据中学习**，需要预先定义。
  - 可以通过不同的参数设置、训练不同的模型，以及选择最好的测试结果来进行超参数选择
- 模型选择（或超参数优化）关注如何选择最佳超参数

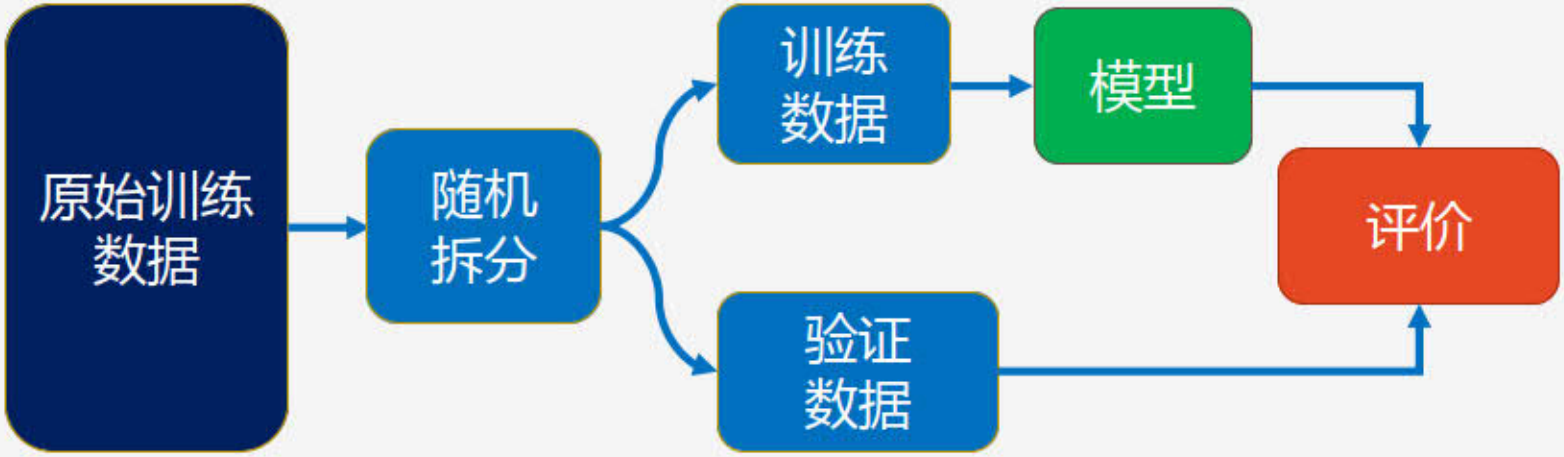
1.4 交叉验证



K-折交叉验证

1. 设置超参数
2. 将原始训练数据随机拆分为K份
3. 重复K次:
   - 若当前为第i次重复（i=1,...,K），选择第i份数据作为验证数据集，其余K-1份作为训练数据集
   - 对训练数据进行建模,并在验证数据上对其进行评估,从而获得评估分数
4. 对K个评估分数取平均作为模型性能

1.4 交叉验证



- 选择了“好的”超参数后，对整个训练数据进行模型训练，然后用测试数据对模型进行测试。

1.5 模型泛化性

泛化能力

- 泛化能力指的是模型对未观测数据的预测能力
  - 可以通过泛化误差来评估，定义如下：

$$R(f) = \mathbb{E}\left[\mathcal{L}\left(Y, f(X)\right)\right] = \int_{X\times Y} \mathcal{L}\left(y, f(x)\right) p(x,y)\,dxdy$$

  - $p(x,y)$是潜在的（可能是未知的）联合数据分布
- 在训练数据集上对泛化能力的经验估计为：

$$\hat{R}(f) = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}\left(y_i, f(x_i)\right)$$

1.5 模型泛化性

泛化误差

- 有限假设集 $\mathcal{F} = \{f_1, f_2, \cdots, f_d\}$
- 泛化误差约束定理：

对任意函数$f \in \mathcal{F}$，以不小于$1 - \delta$的概率满足下式：

$$R(f) \leq \hat{R}(f) + \epsilon(d, N, \delta)$$

其中，

$$\epsilon(d, N, \delta) = \sqrt{\frac{1}{2N}\left(\log d + \log\frac{1}{\delta}\right)}$$

- N：训练实例个数
- d：假设集的函数个数