《机器学习基础》

崔志勇

交通科学与工程学院

2024年5月

基础

为什么选择机器学习?

- 搜索引擎（例如：谷歌）
- 推荐系统（例如：哔哩哔哩）
- 自动翻译（例如：ChatGPT）
- 语音理解（例如：Siri）
- 游戏对战（例如：AlphaGo）
- 自动驾驶汽车（例如：Waymo、百度）
- 个性化医学
- 各个科学领域的进展：遗传学、天文学、化学、神经学、物理学等。

人工智能

机器学习

神经网络（NN）

- 卷积神经网络
- 递归神经网络
- 全连接神经网络

机器学习是什么?

基于**经验**（例如**样本**）X 学习执行任务的过程，**目标**是**最小化误差** $\mathcal{E}$。

例如，尽可能准确地识别图像中的人。

通常，我们希望学习一个**函数**（**模型**）f，它具有一些**模型参数**θ，能够产生正确的**输出** y。

$$f_\theta(X) = y$$

$$\underset{\theta}{\operatorname{argmin}}\ \mathcal{E}(f_\theta(X))$$

通常，这种学习过程是更大系统的一部分，该系统能够提供数据X正确的形式。数据需要被收集、清洗、标准化，并检查是否存在数据偏差。

机器学习的框架

将预测函数应用于图像的特征表示，以获得所需的输出:

f(

) = “Stop”

f() = “No U-Turn”

f(

) = “Speed Limit 60”

Slide credit: L. Lazebnik

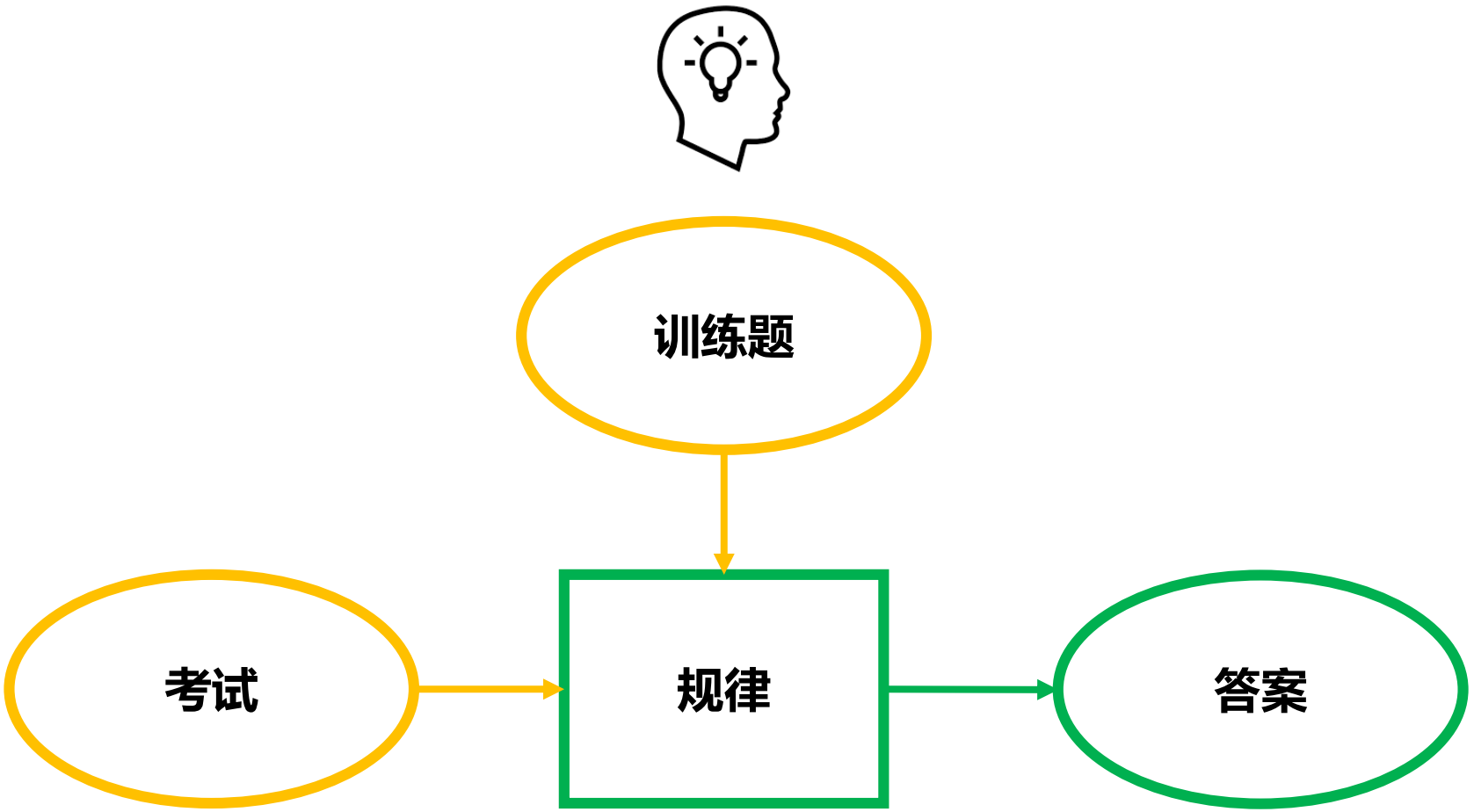
机器学习的类型

- **监督学习**：从带标签的数据（X，y）（ground truth）中学习一个模型f。
  - 给定新的输入X，预测正确的输出y。
  - 例如，给定恒星和星系的示例，识别天空中的新物体。
- **无监督学习**：探索数据（X）的结构，提取有意义的信息。
  - 根据输入X，找出哪些是特殊的、相似的、异常的等。
- **半监督学习**：从（少量）带标签和（大量）无标签的示例中学习一个模型。
  - 未标记的示例提供了关于可能出现的新示例的信息。
- **强化学习**：开发一个代理程序，根据与环境的交互改善其性能。
- 注意：实际的机器学习系统可以将许多类型结合在一个系统中。

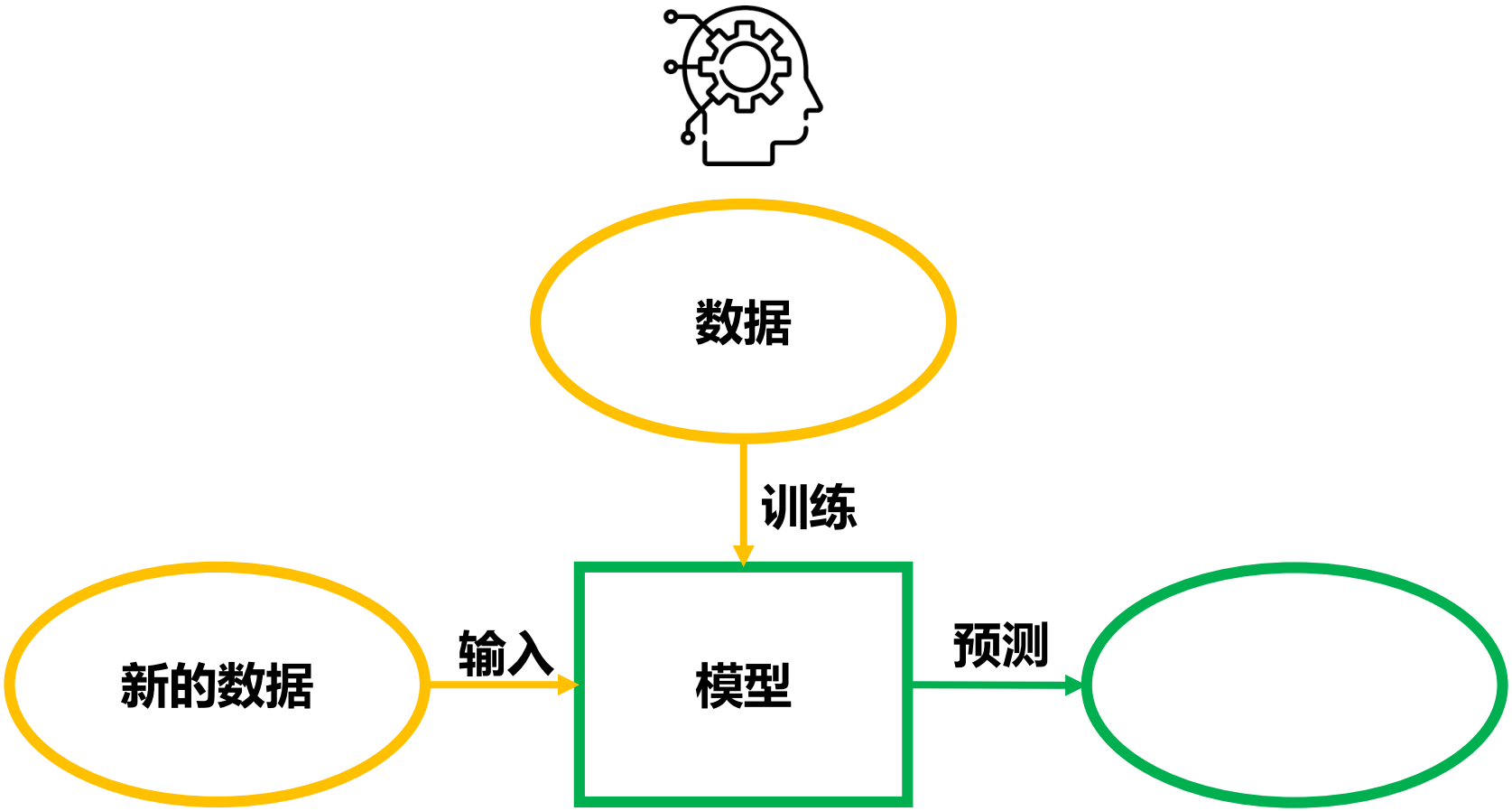
Machine Learning Problems

	Supervised Learning	*Unsupervised Learning*
Discrete	classification or categorization	clustering
Continuous	regression	dimensionality reduction

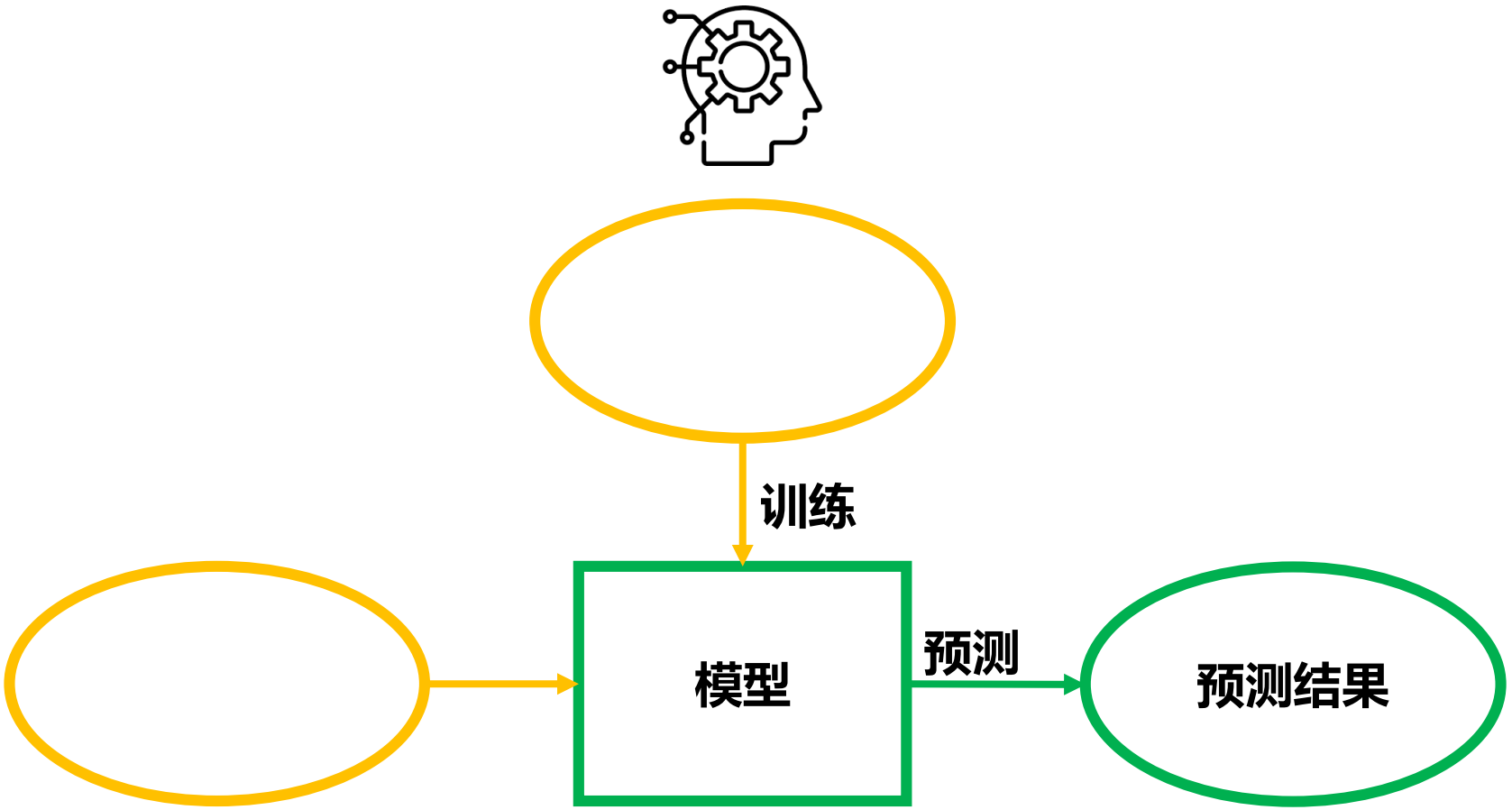
监督学习(Supervised Learning)



监督学习(Supervised Learning)



监督学习(Supervised Learning)



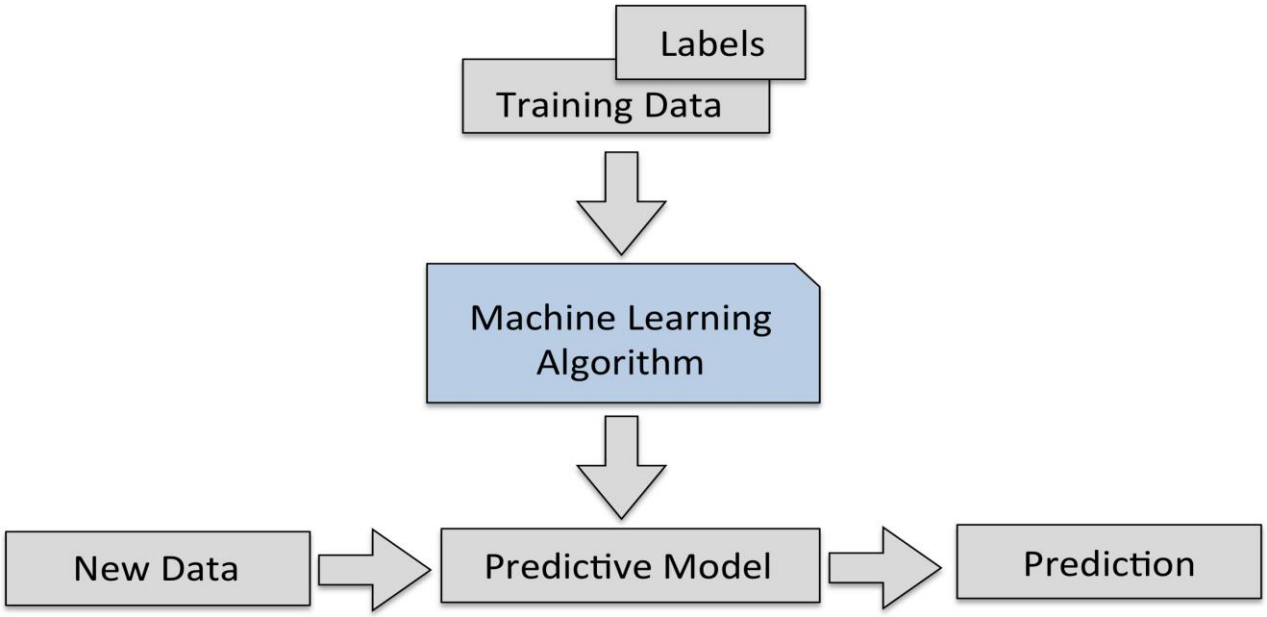
监督学习(Supervised Learning)

从带标签的训练数据中学习模型，然后进行预测。

监督学习：我们知道正确/期望的结果（标签）的情况。

监督学习的**子类型**包括**分类**（预测类别）和**回归**（预测数值）。

大多数我们将看到的监督算法都可以同时进行分类和回归。



机器学习框架

$$y = f(\boldsymbol{x})$$

输入　　预测函数　　图像特征

训练：给定一个带标签的训练集$\{(\mathbf{x}_1,y_1), \ldots, (\mathbf{x}_N,y_N)\}$，通过最小化训练集上的预测误差来估计预测函数$f$。

测试：将f应用于从未见过的测试示例$\boldsymbol{x}$，并输出预测值$y = f(\boldsymbol{x})$。

Slide credit: L. Lazebnik

学习步骤

训练

训练的图像数据

训练标签

图像特征

学习得到的模型

预测

图像特征

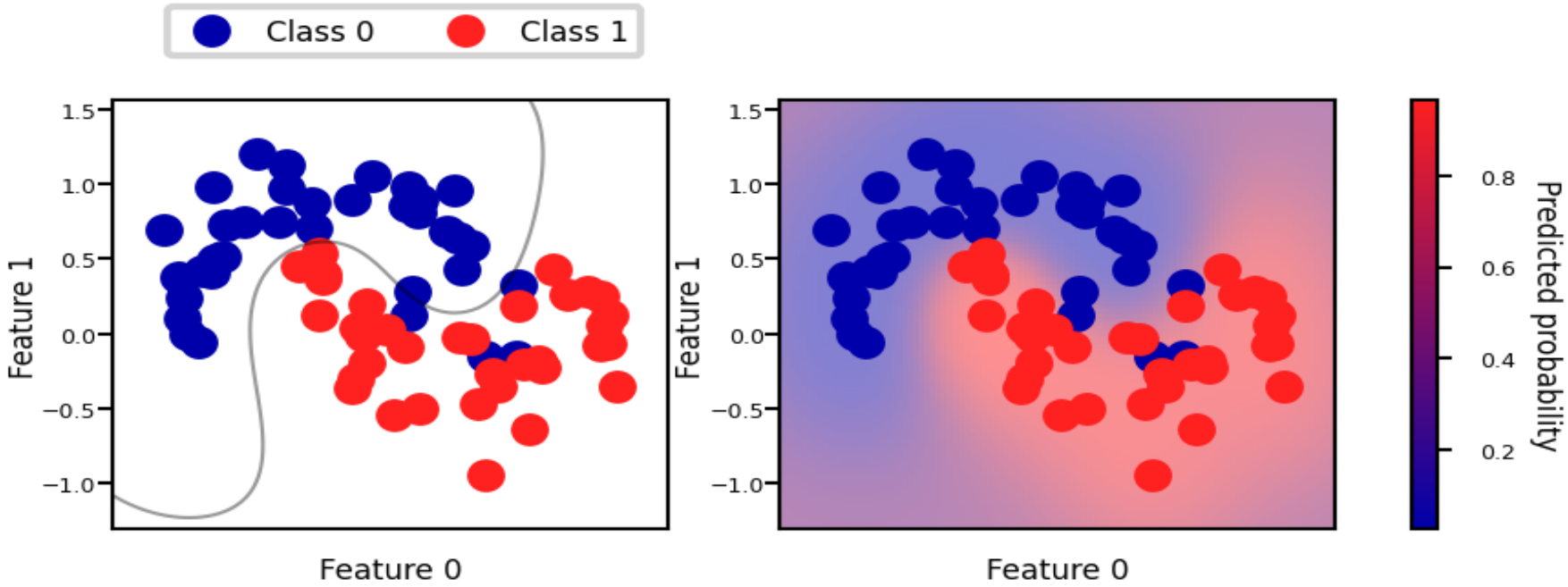
预测

测试图像数据

Slide credit: D. Hoiem and L. Lazebnik

分类

- **预测**一个类别标签（**分类**），离散且无序。
  - 可以是**二分类**（例如垃圾邮件/非垃圾邮件）或**多类别**（例如字母识别）。许多分类器可以针对每个类别返回置信度。
- 模型的预测结果是产生一个将类别分开的决策边界。



示例：花卉分类

- 对鸢尾花的类型进行分类（山鸢尾、变色鸢尾或维吉尼亚鸢尾），你该如何做呢?

变色鸢尾

山鸢尾

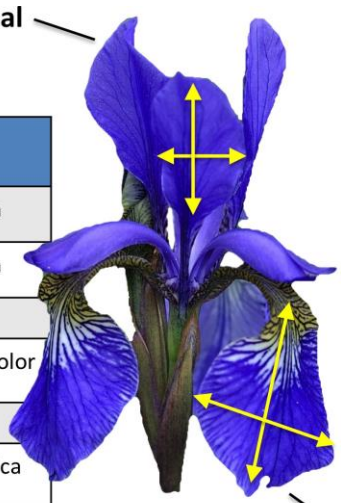
维吉尼亚鸢尾

表示：输入特征和标签

- 我们可以拍照并将它们（像素值）用作输入（->深度学习）。
- 我们可以手动定义一些输入特征（变量），例如叶子的长度和宽度。
- 每个“示例”都是一个（可能是高维的）空间中的一个点。

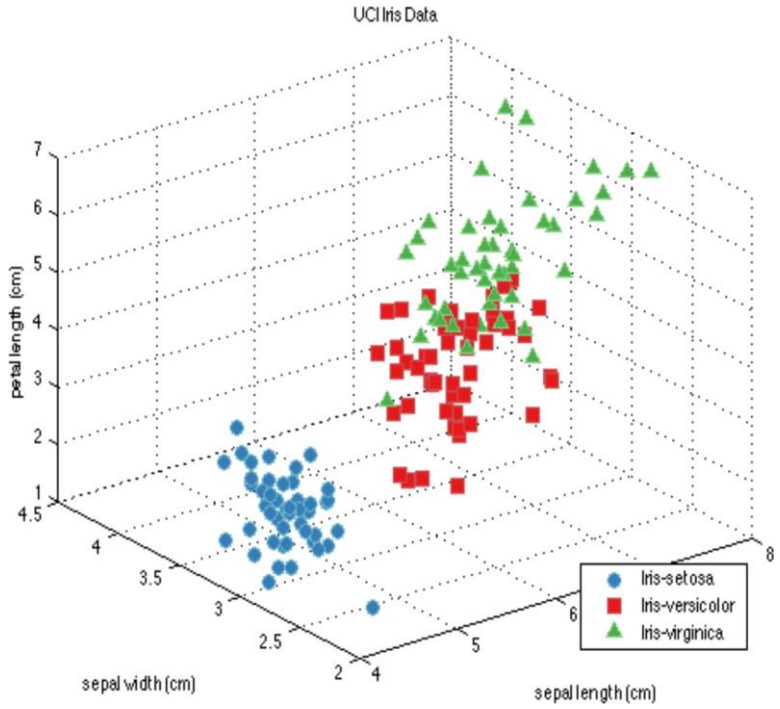
Samples
(instances, observations)

	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
...					
50	6.4	3.5	4.5	1.2	Versicolor
...					
150	5.9	3.0	5.0	1.8	Virginica



Features
(attributes, measurements, dimensions)

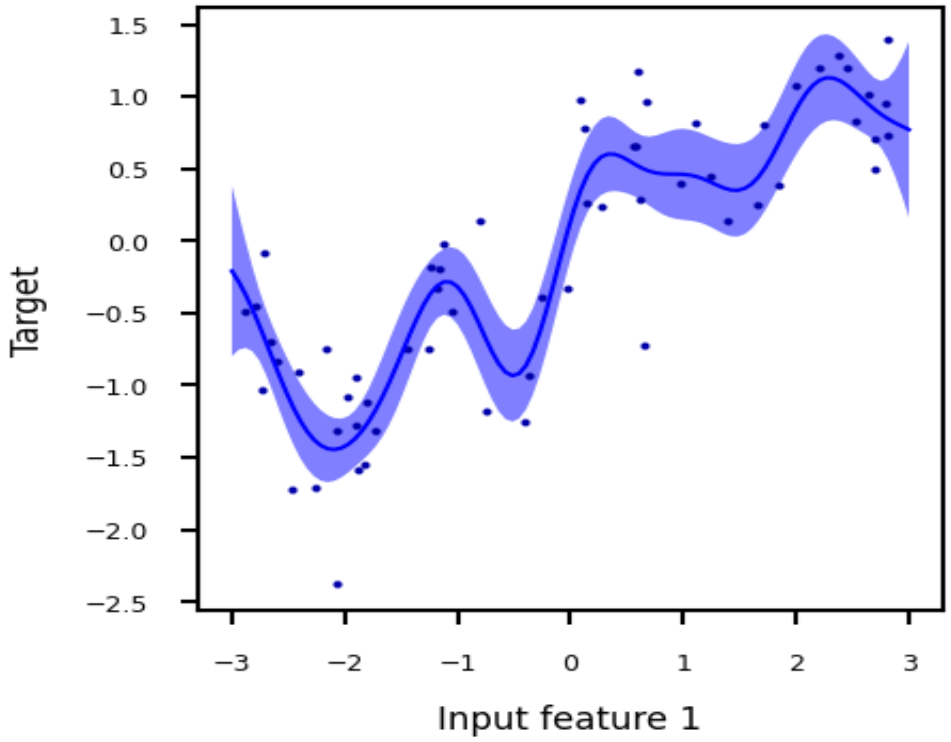
Class labels
(targets)



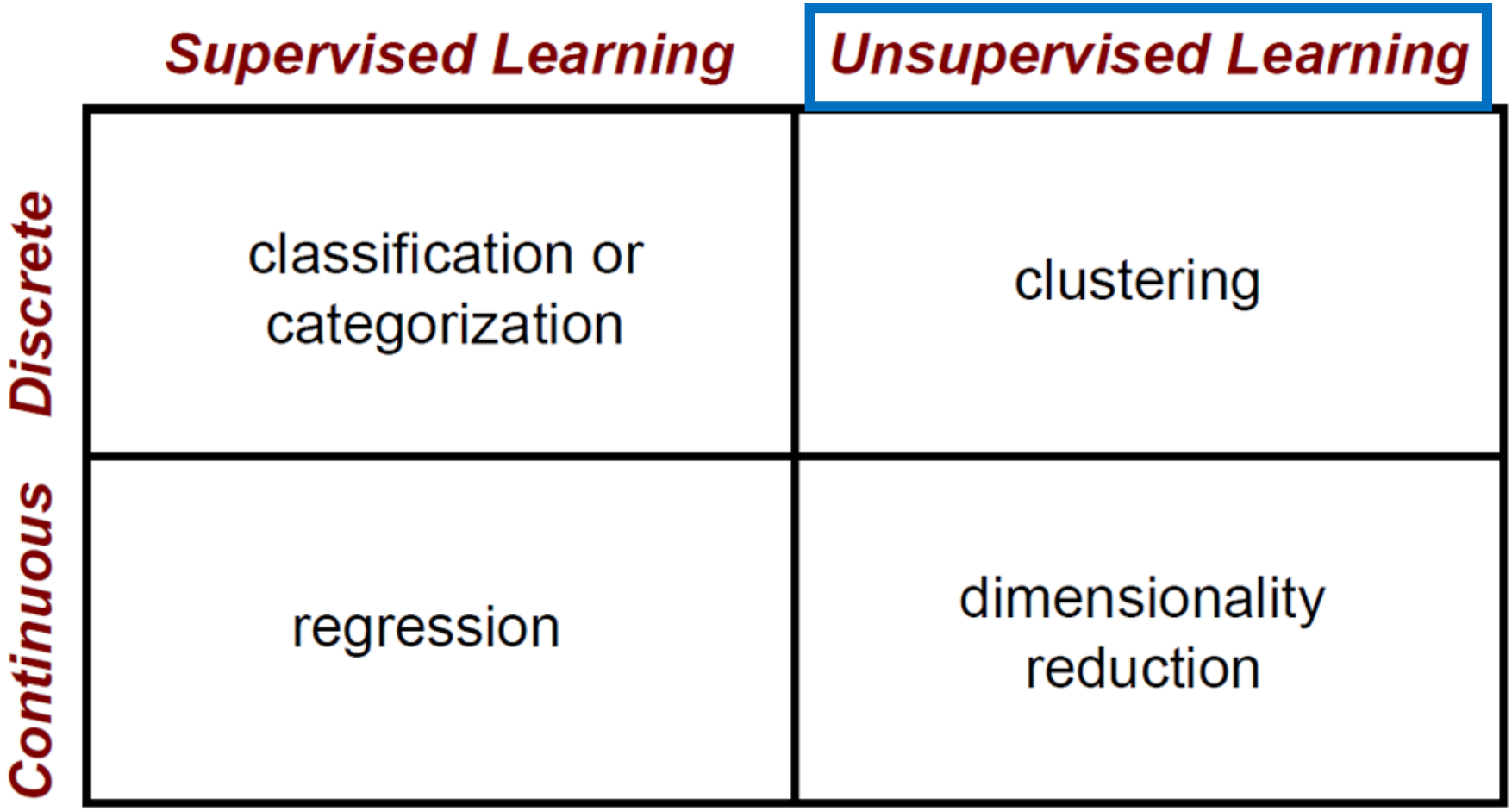
Petal：花瓣, Sepal：花萼

回归

- **预测一个连续值**，例如温度：
  - 目标变量是数值型的。
  - 一些算法可以返回置信区间。
- 寻找预测因子和目标变量之间的关系。



Machine Learning Problems



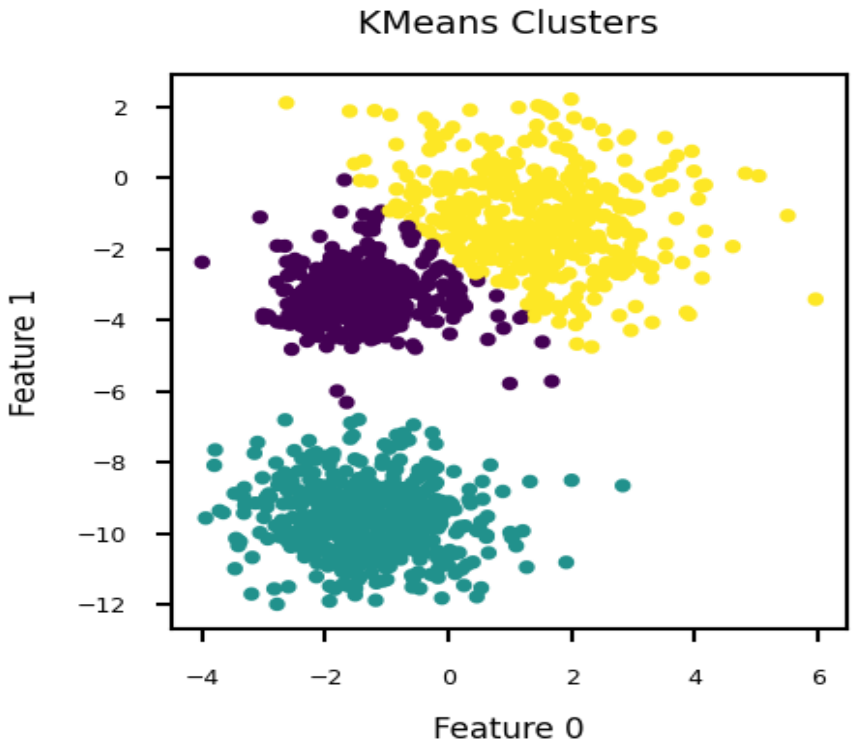
无监督学习 (Unsupervised Learning)

- **未标记的数据**，或者具有未知结构的数据。
- **探索数据的结构**以提取信息。
- 有许多类型，我们只讨论其中两种。

聚类

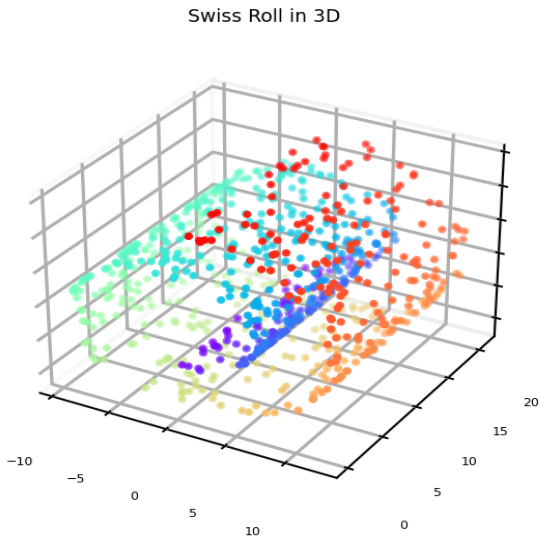
- 将信息组织成有意义的**子群（簇）**。
- 同一簇中的对象具有**一定程度的相似性**（并且与其他簇的不相似性）。

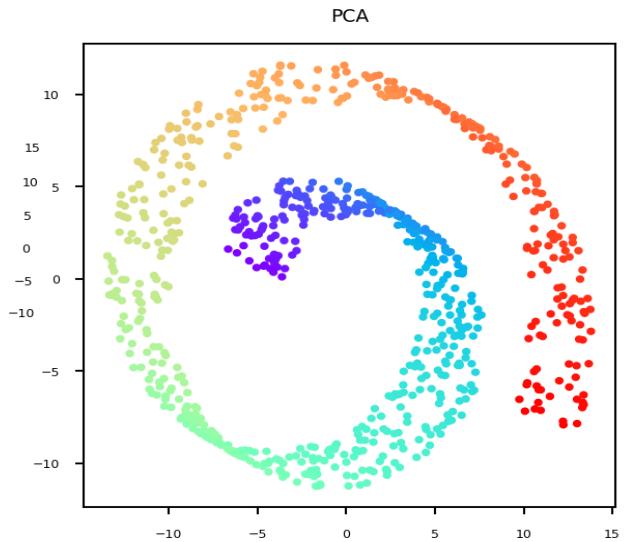
  例如：区分不同类型的客户。

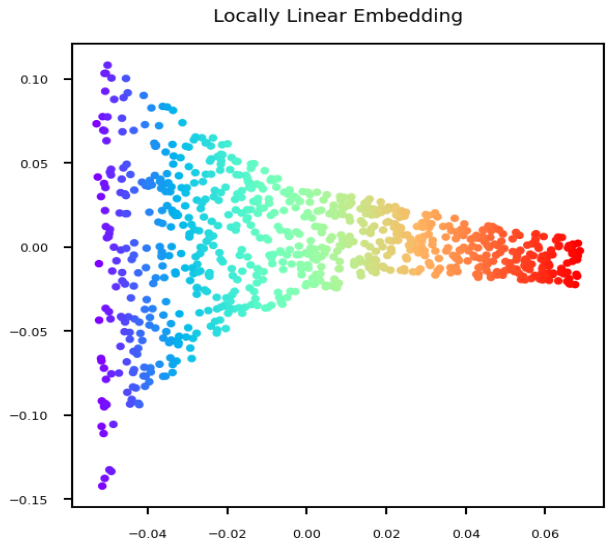


降维

- 数据可能具有非常高的维度，难以理解、学习和存储。
- 降维可以将数据压缩到**较少的维度**，同时**保留大部分信息。**
- 与特征选择相反，新特征失去了（原始）的含义。
- 新的表示形式可能更容易建模（和可视化）。

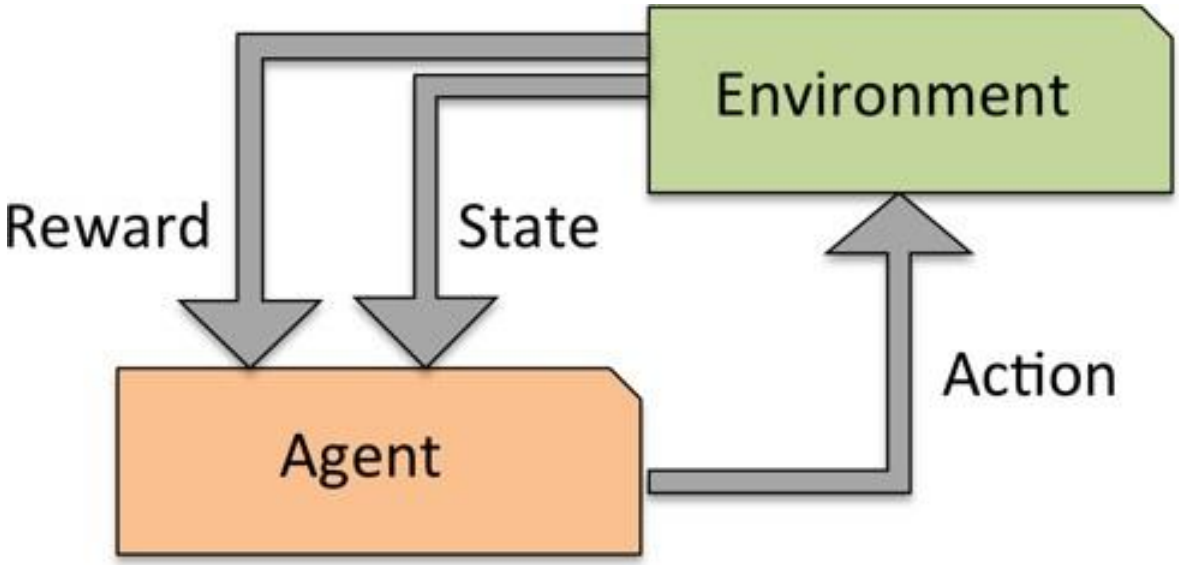






强化学习

- 部署一个基于**与环境的交互**来提高性能的智能体。 例如：象棋、围棋等游戏。
  - 搜索一个（大型）的行动和状态空间。
- 奖励函数定义了某一或某一系列行动的效果如何。
- 通过探索，学习一系列最大化奖励的行动（策略）。



人工智能

机器学习

神经网络（NN）

- 卷积神经网络
- 递归神经网络
- 全连接神经网络

多选题 1分

设置

神经网络能解决以上哪些问题?

A Classification （分类）

B Regression (回归)

C Clustering （聚类）

D Dimensionality reduction（降维）

提交

学习 = 表示 + 评估 + 优化

- 所有机器学习算法都包含三个组件：
- **表示**（Representation）：模型 f_θ 必须以计算机可以处理的语言形式表示出来。
  - 这定义了它可以学习的“概念”，即假设空间。
  - 例如，决策树、神经网络、一组带有注释的数据点。
- **评估**（Evaluation）：一种内部方式，用于在不同假设之间进行选择。
  - 目标函数、评分函数、损失函数 $L(f_\theta)$
  - 例如：正确输出与预测之间的差异。
- **优化**（Optimization）：一种高效搜索假设空间的方法。
  - 从简单假设开始，如果不符合数据，则进行扩展（放宽条件）。
  - 从一组初始模型参数开始，逐步细化它们。
  - 方法多样，学习速度、最优解的数量等方面各不相同。
- 一个强大/灵活的模型只有在能够高效优化的情况下才有用。

泛化

- 如何评估一个学习到的模型**从其训练数据泛化到新测试集**的表现好坏?

训练数据集 (标签已知)

测试数据集(标签未知)

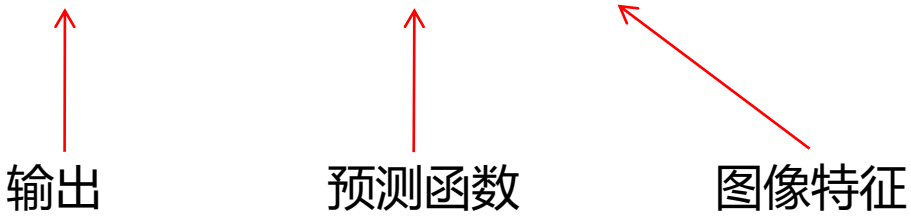
泛化

- 泛化误差的组成部分：
  - **偏差（Bias）**：平均模型在所有训练集上与真实模型之间的差异程度
    - 由模型所做的不准确假设/简化造成的误差。
  - **方差**（Variance）：从不同训练集估计的模型之间的差异程度。

- **欠拟合**（Underfitting）：模型过于“简单”，无法表示所有相关的类别特征。
  - 高偏差和低方差；
  - 训练误差和测试误差都较高；
- **过拟合**（Overfitting）：模型过于“复杂”，拟合了数据中的无关特征（噪音）。
  - 低偏差和高方差，
  - 训练误差较低，但测试误差较高。

机器学习框架

$$y = f(x)$$

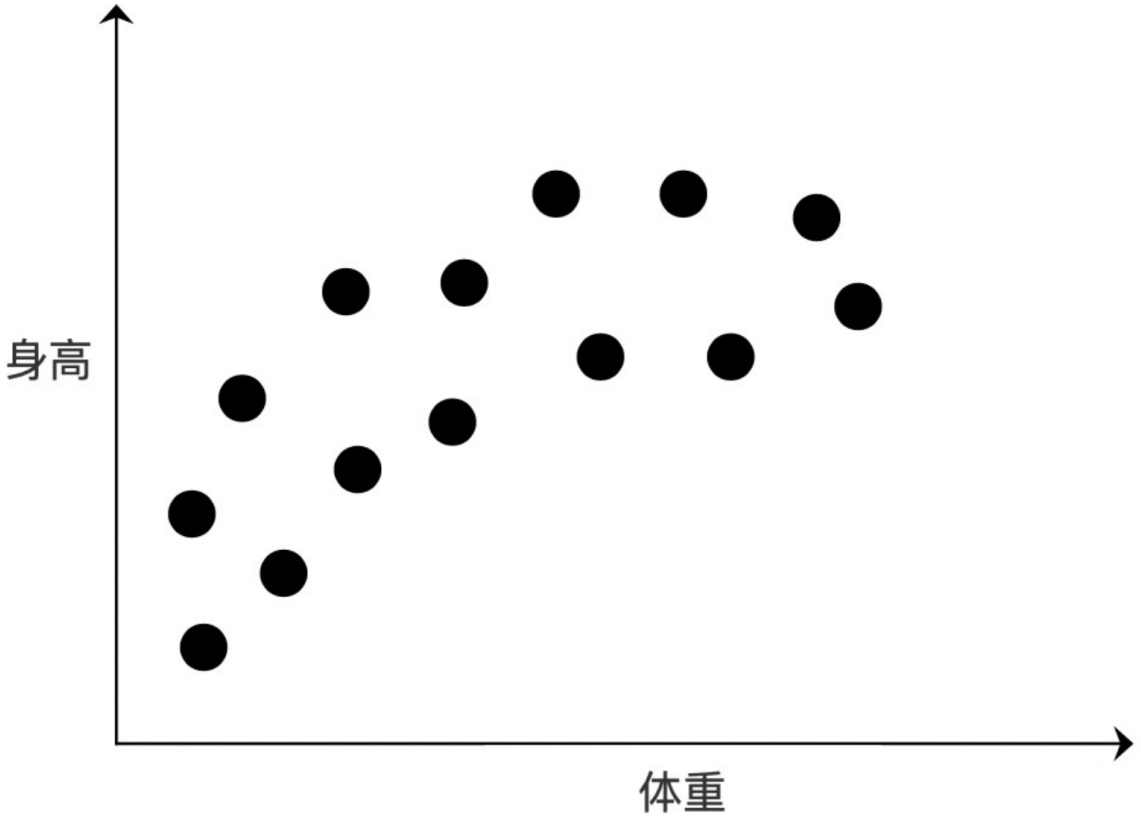


- 训练：给定一个带标签的训练集D_{train} ={($\mathbf{x}_1$,y_1), …, ($\mathbf{x}_n$,y_n)}，通过在训练集上最小化预测误差来估计预测函数f。
- 测试：将f 应用于之前从未见过的测试集 D_{test} = ={($\mathbf{x}_1$,y_1), …, ($\mathbf{x}_m$,ym)}，并输出预测值$y = f(x)$
- **误差：** $E(f(x), y)$
- **错误率:** $E(f, D_{test}) = \frac{1}{m}\sum_{i=1}^{m}(f(x_i) \neq y_i)$ **准确率：** $acc = 1 - E(f, D_{test})$

偏差(Bias)-方差(Variance)平衡(Trade-off)

例如：

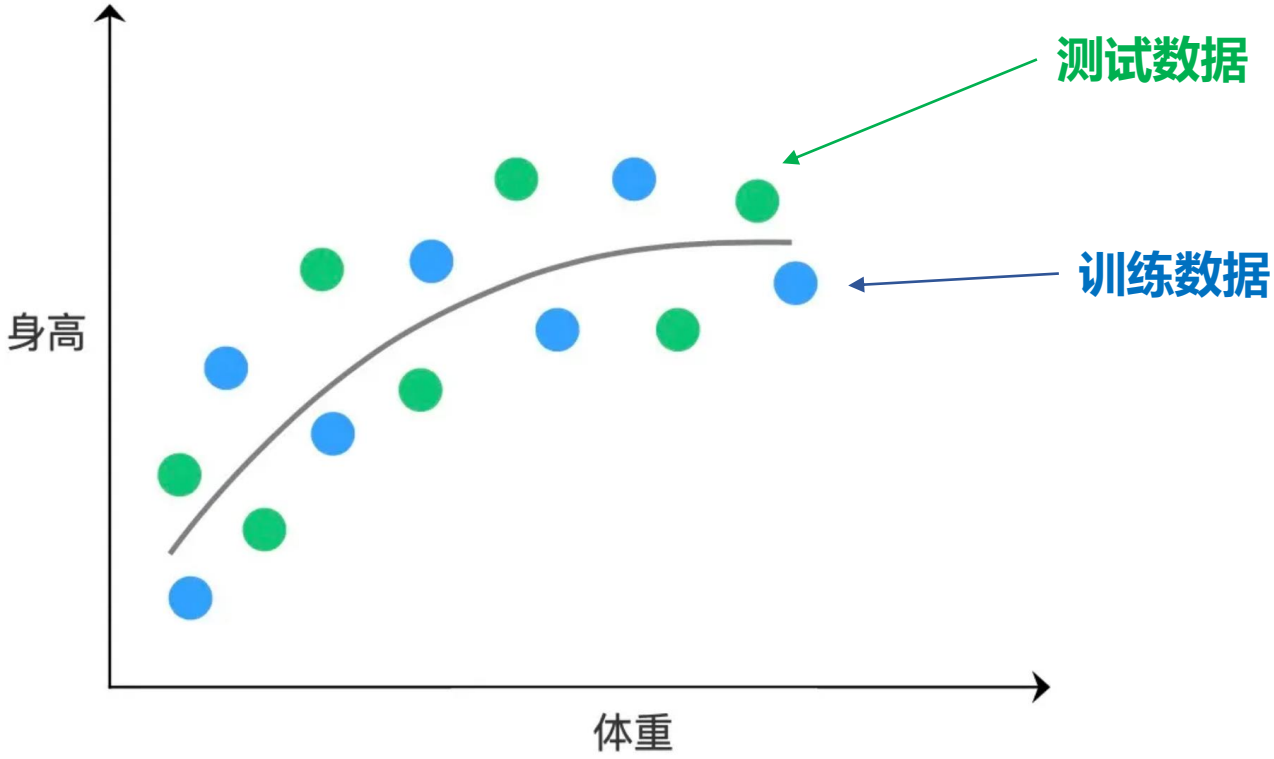
- 一个建模任务



偏差-方差权衡

例如：

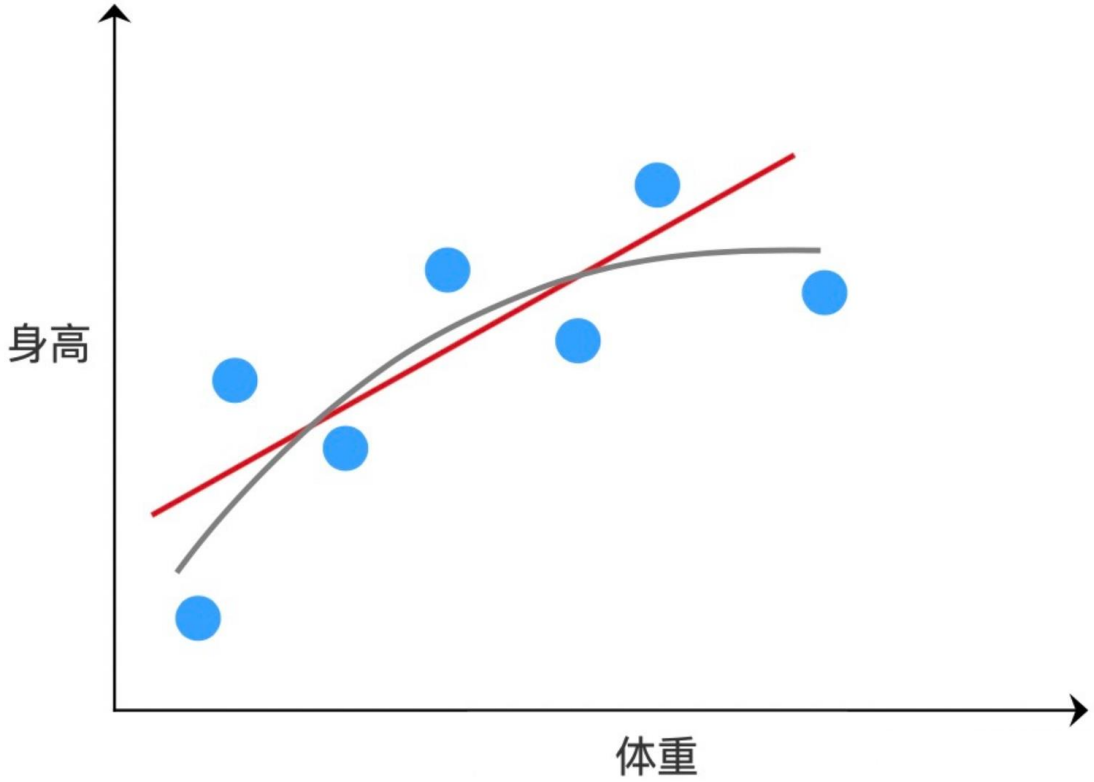
- 一个建模任务
- 将数据分割为训练集和测试集



偏差-方差权衡

例如：

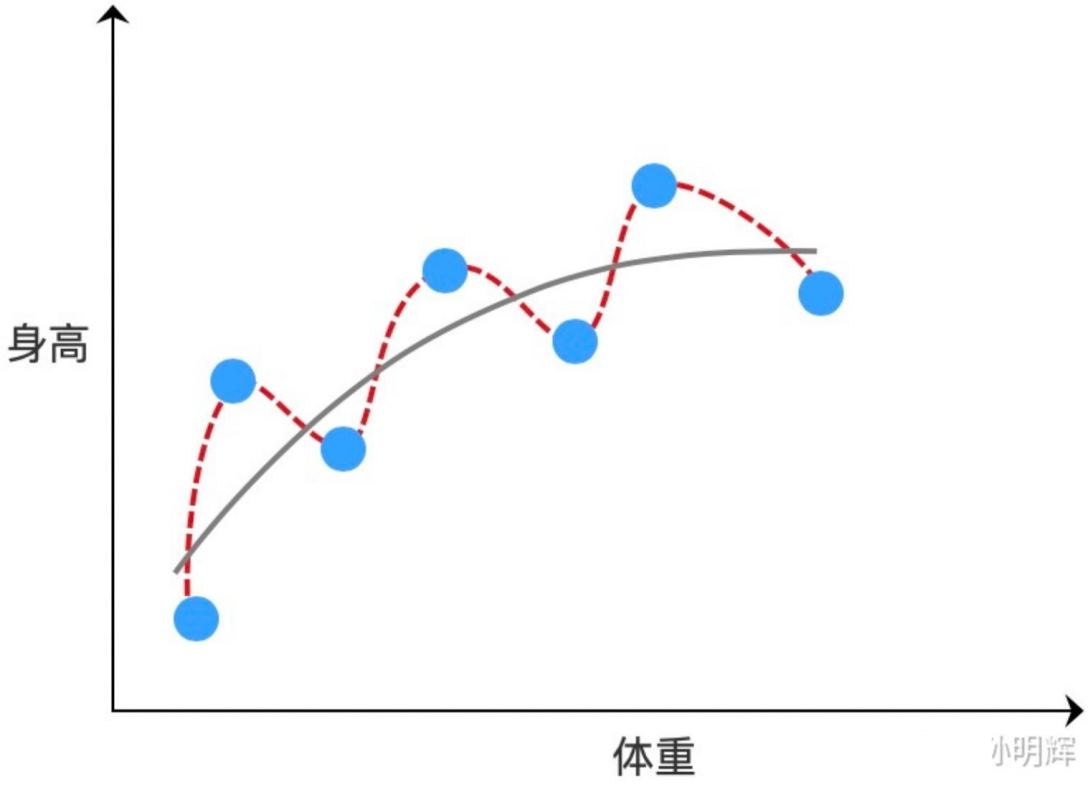
- 一个建模任务
- 将数据分割为训练集和测试集
- 选择一个简单的模型



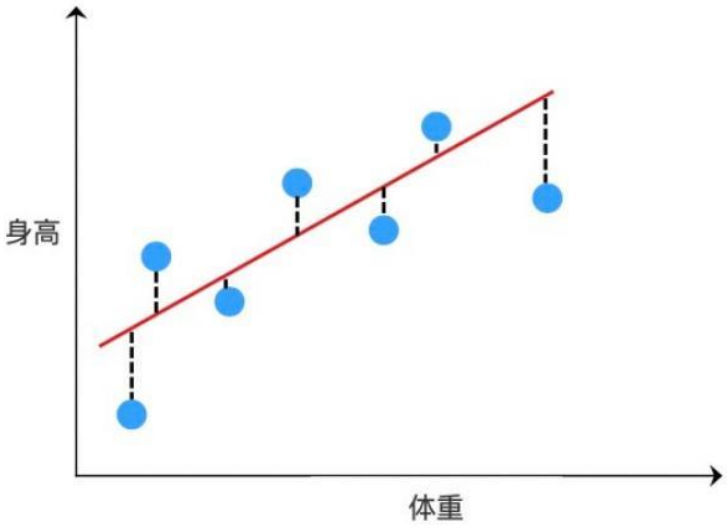
偏差-方差权衡

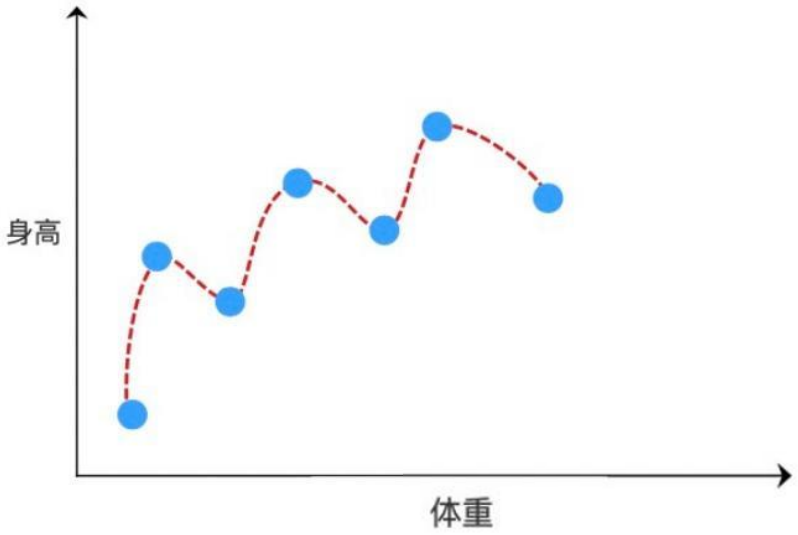
例如：

- 一个建模任务
- 将数据分割为训练集和测试集
- 选择一个简单的模型
- 尝试一个复杂的模型



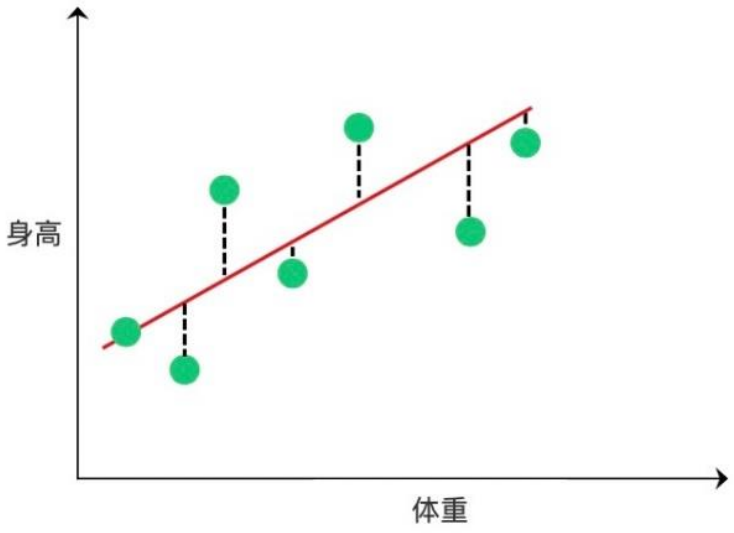
偏差-方差权衡

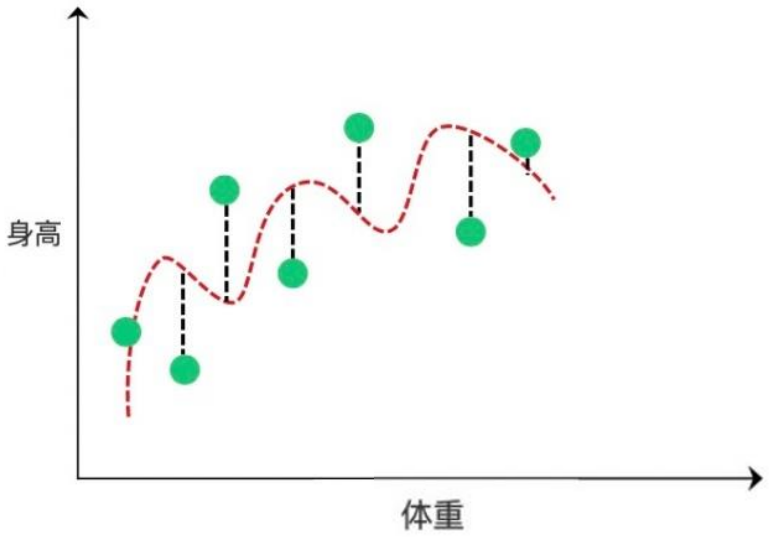




Error > Error

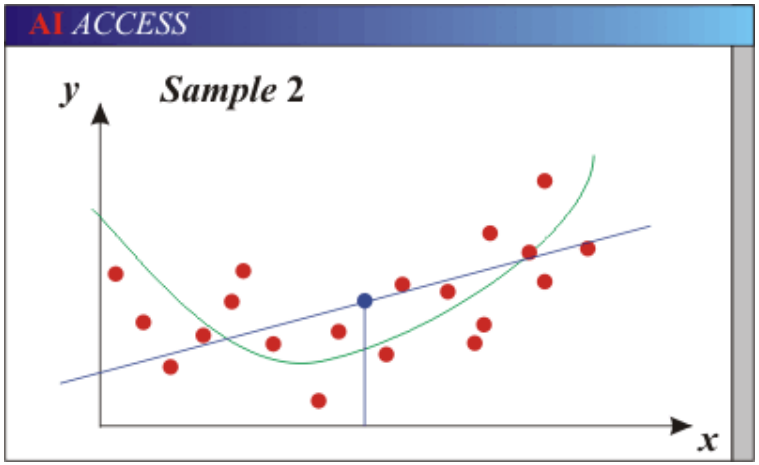
偏差-方差权衡



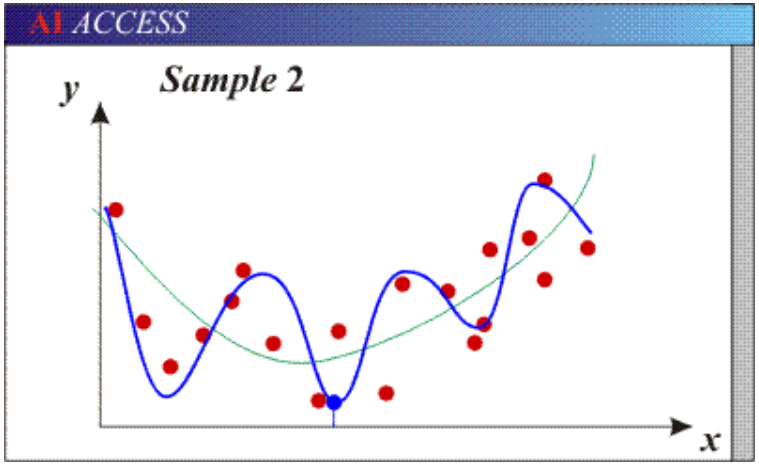


Error < Error

偏差-方差权衡



- 参数过少的模型不准确，因为存在较大的偏差（灵活性不足）。



- 参数过多的模型不准确，因为存在较大的方差（对样本过于敏感）。

Slide credit: D. Hoiem

偏差(Bias)-方差(Variance)平衡(Trade-off)

$$\text{Error} = \text{noise}^2 + \text{bias}^2 + \text{variance}$$

Unavoidable error

Error due to incorrect assumptions

Error due to variance of training samples

Error: 误差
Bias: 偏差
Variance: 方差
Noise: 噪声

你可以查看以下资源以了解偏差-方差（还有Bishop的“神经网络”书）：

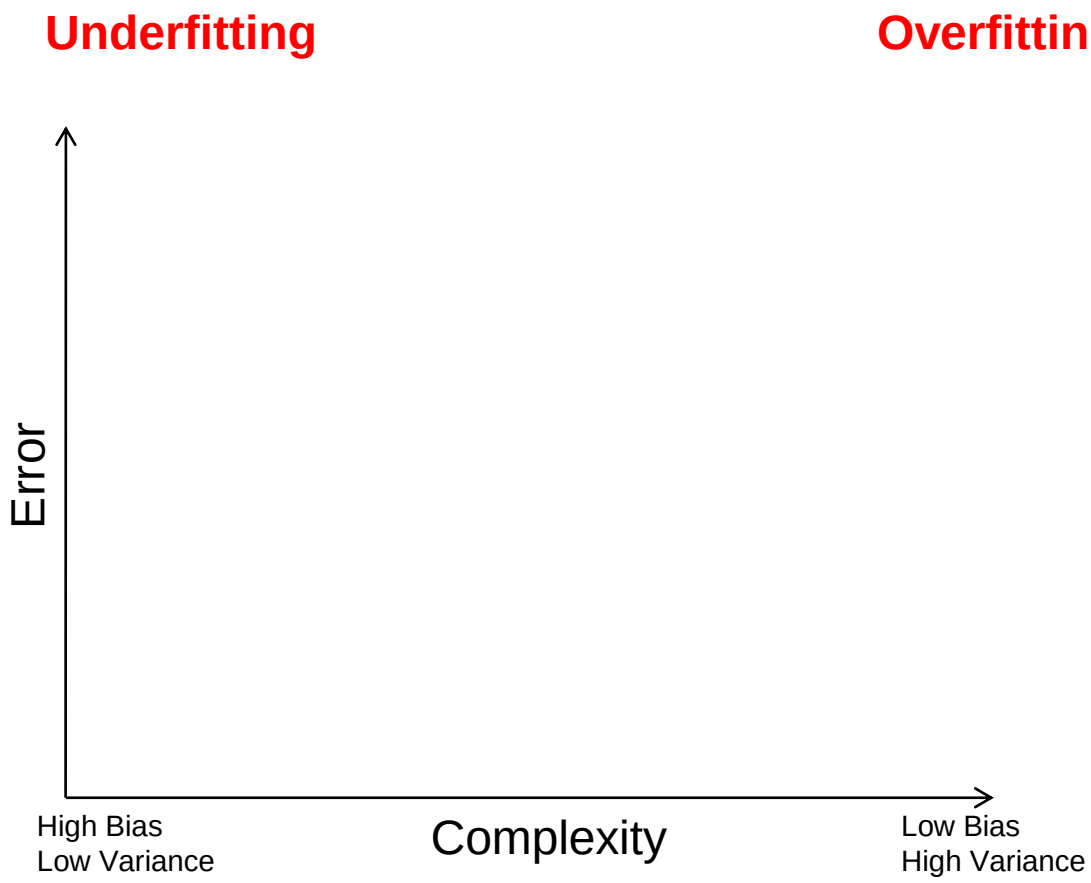
- http://www.inf.ed.ac.uk/teaching/courses/mlsc/Notes/Lecture4/BiasVariance.pdf

Slide credit: D. Hoiem

偏差-方差权衡

- $Error(x) = E[y - f(x)]^2$
- $Error(x) = E[y - E\big(f(x)\big) + E\big(f(x)\big) - f(x)]^2$
- $Error(x) = E[y - E\big(f(x)\big)]^2 + E[f(x) - E\big(f(x)\big)]^2$
- $Error(x) = Bias^2 + Variance$

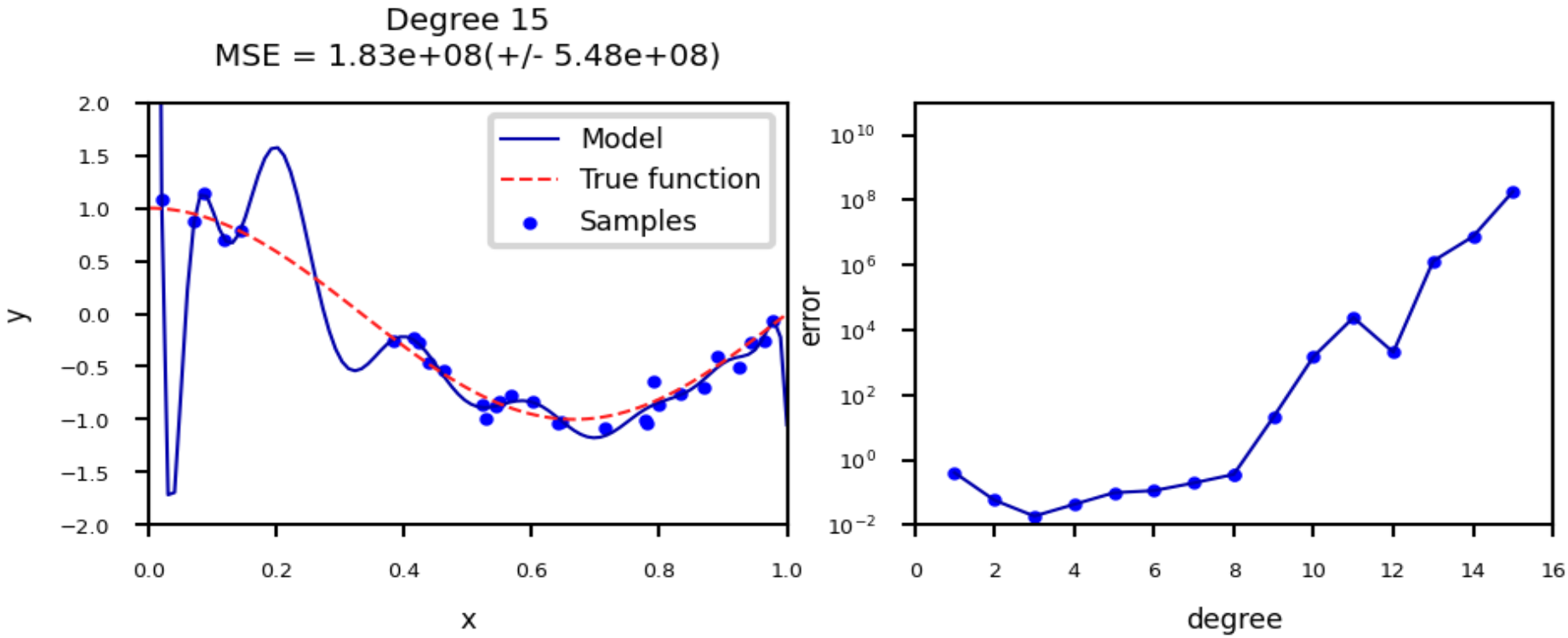
偏差-方差权衡



Slide credit: D. Hoiem

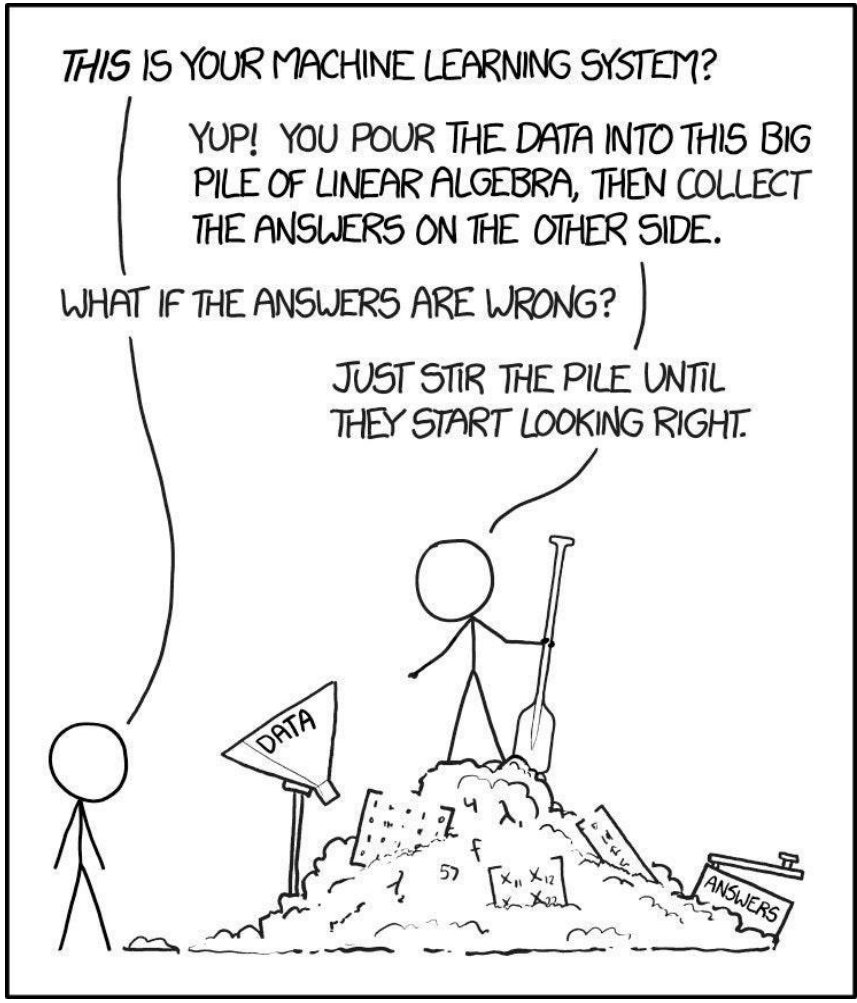
过拟合和欠拟合

- 通常，你需要通过优化算法和超参数的选择，或使用更多数据，来找到一个最佳点。
- 例如：使用多项式函数进行回归。



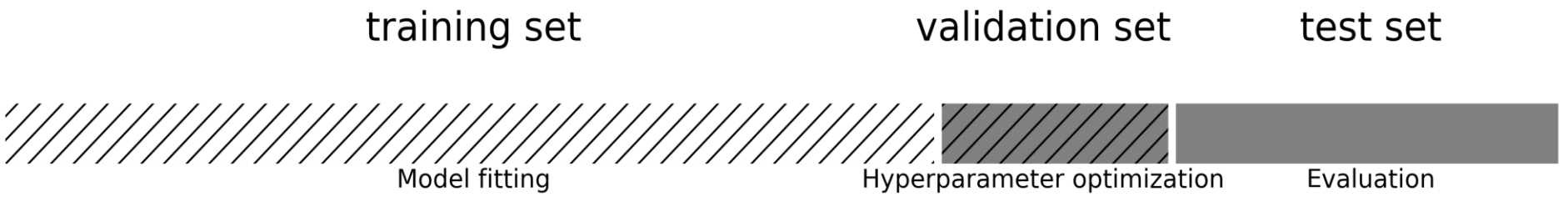
模型选择

- 除了（内部的）损失函数之外，我们还需要一个（外部的）评估函数。
- 反馈信号：我们是否真正学到了正确的东西?
- 我们是否欠拟合/过拟合?
- 仔细选择适合应用的模型。
- 需要在模型（和超参数设置）之间进行选择。



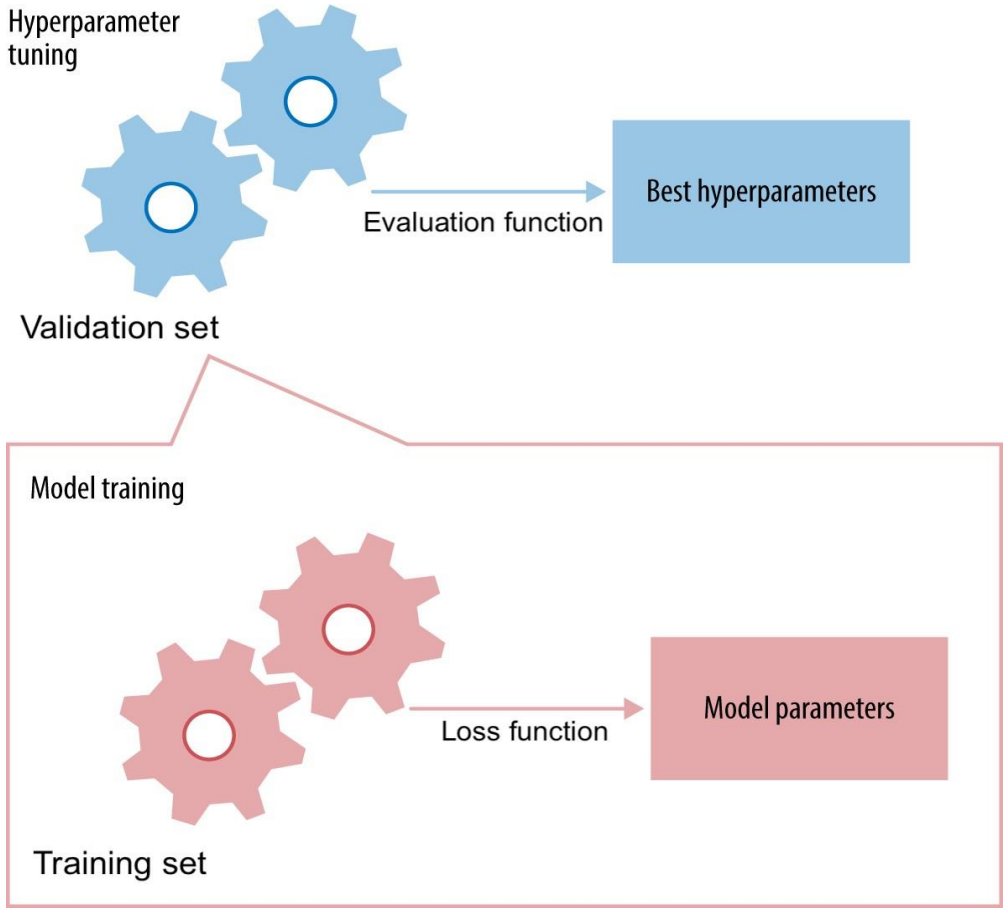
模型选择

- 数据需要分成训练集和测试集。
  - 在训练集上优化模型参数，然后在独立的测试集上进行评估。
- 避免数据泄露:
  - 永远不要在测试数据上优化超参数设置。
  - 永远不要根据测试数据选择预处理技术。
- 为了优化超参数和预处理，将部分训练集设置为验证集。
  - 在所有训练期间，保持测试集隐藏。



模型选择

- 对于给定的超参数设置，在训练集上学习模型参数。
  - 最小化损失
- 在验证集上评估训练好的模型。
  - 调整超参数以最大化某个指标（例如准确率）。



模型选择

- **只有泛化能力才重要！**
- 除了以下情况外，永远不要在训练数据上评估最终模型：
  - 追踪优化器是否收敛（学习曲线）
  - 诊断欠拟合/过拟合：
    - 较低的训练和测试得分：欠拟合
    - 高训练得分，低测试得分：过拟合
- 始终保留一个完全独立的测试集。
- 对于小数据集，使用多个训练-测试分割以避免抽样偏差。
  - 可能会意外地抽取一个“简单”的测试集。
  - 例如，使用交叉验证（见后文）。

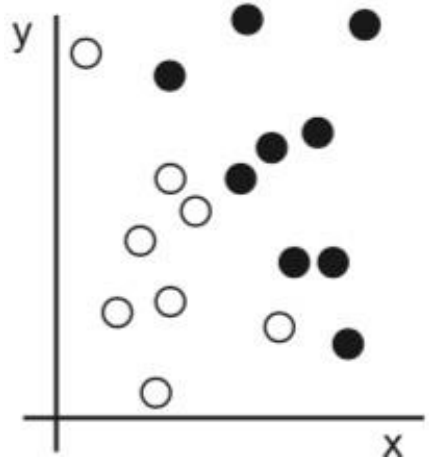
更好的数据表示，更好的模型

算法需要正确地将输入转换为正确的输出。

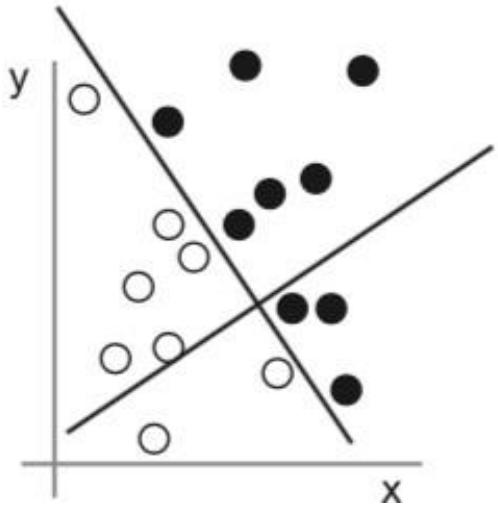
这很大程度上取决于我们如何将数据呈现给算法。

- 将数据转换为**更好的表示**（也称为**编码**或**嵌入**）。
- 可以通过**端到端**（例如深度学习）进行，也可以首先**对数据进行“预处理”**（例如特征选择/生成）。

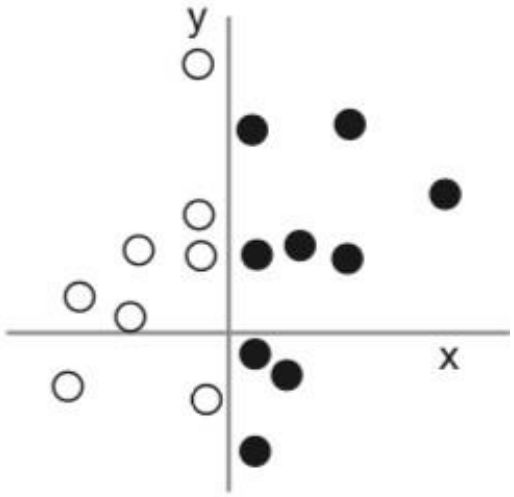
1: Raw data



2: Coordinate change

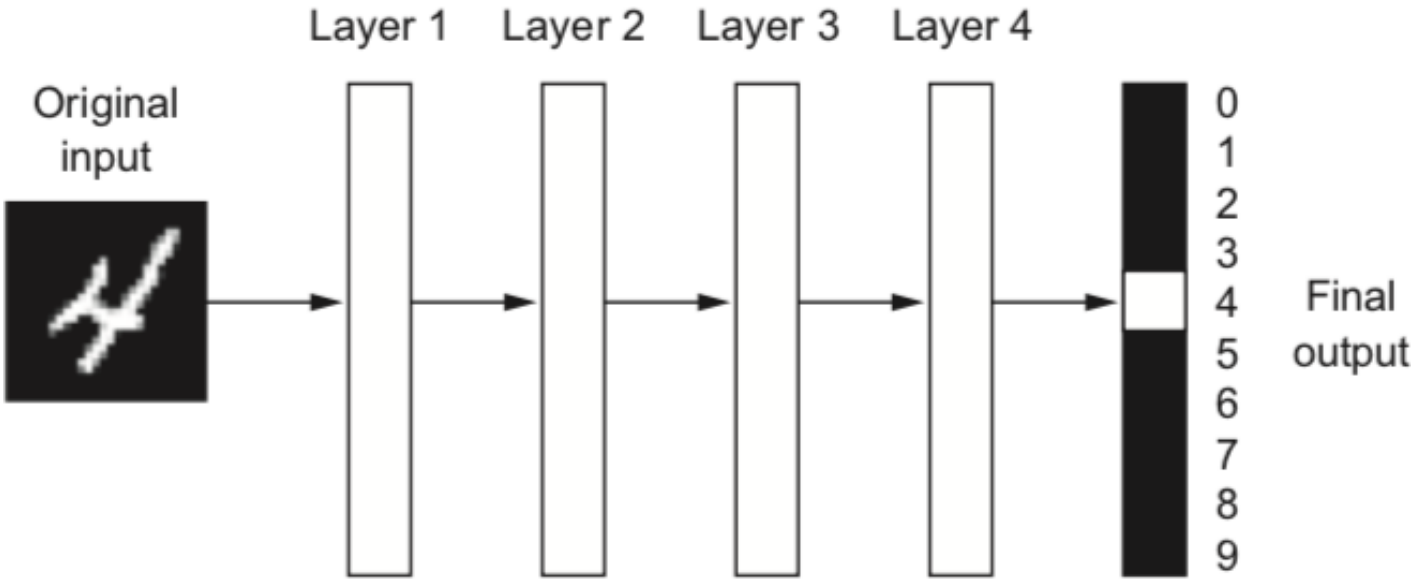


3: Better representation



端到端学习数据转换

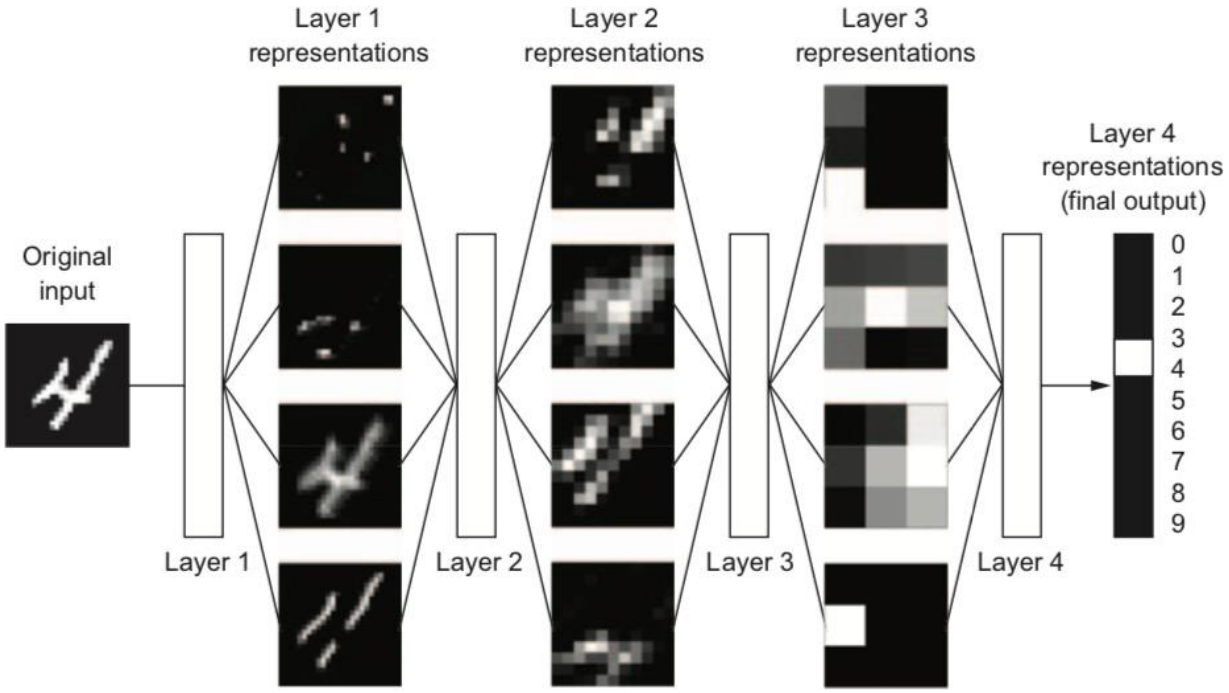
- 对于非结构化数据（例如图像、文本），**很难提取出好的特征。**
- 深度学习：**学习数据自己的表示**（嵌入）。
  - 通过多层表示（例如神经元层）。
  - 每一层根据减少误差的原则对数据进行一些转换。



例如：数字分类

- 像素作为输入进入，每一层针对给定任务将它们转换为越来越丰富的表示。
- 这通常对人类来说不太直观。

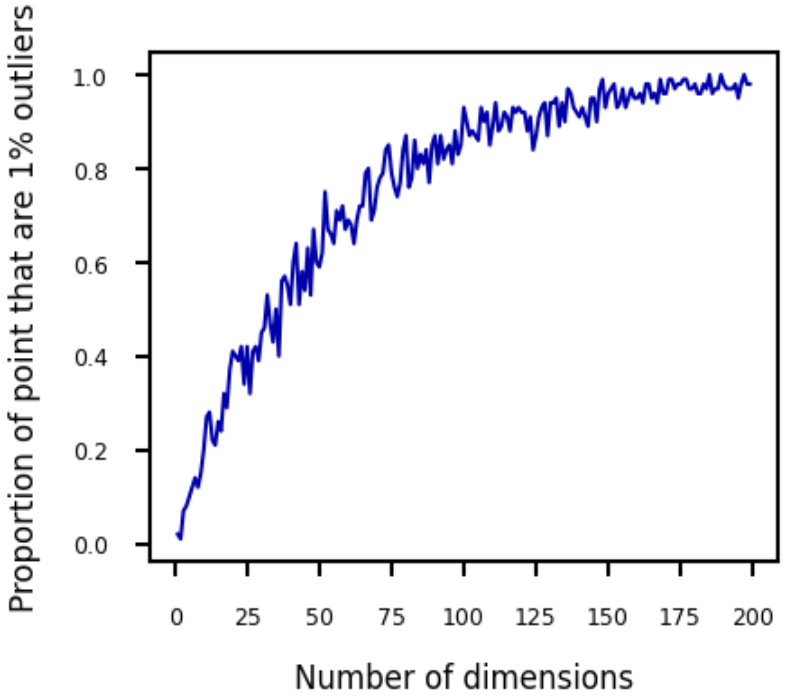
MNIST



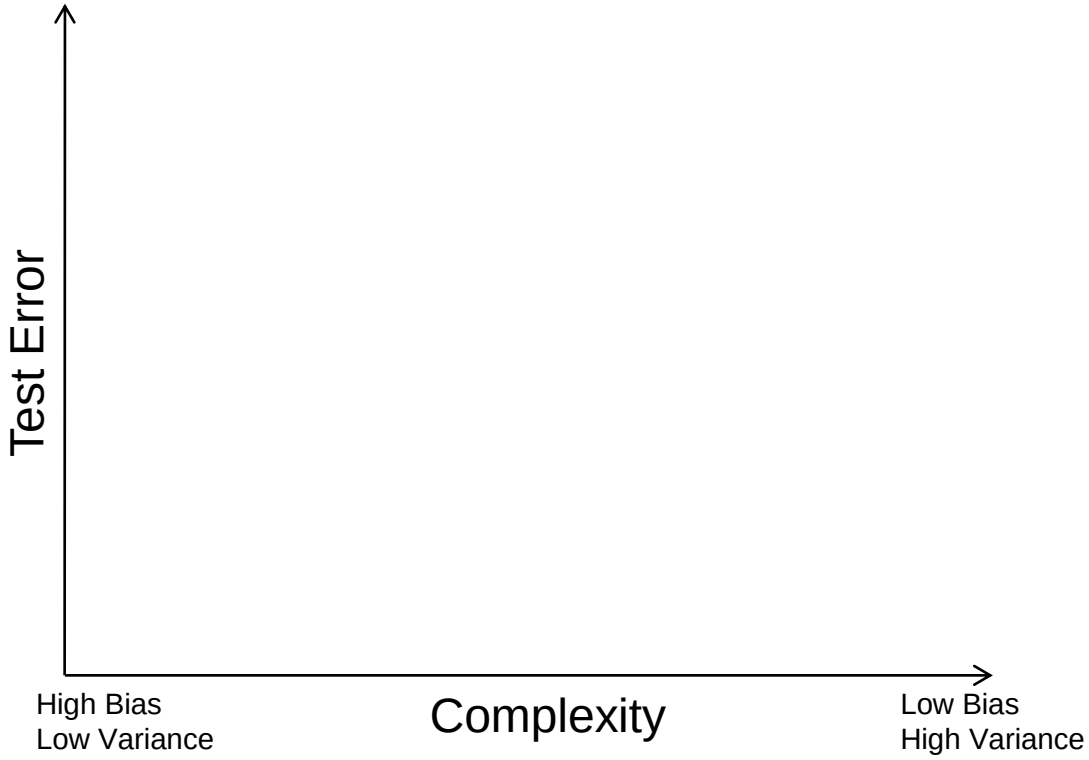
维度灾难

- 只是**添加大量特征**并让**模型自行处理**是**行不通**的。

- 我们的假设在高维度下经常失败：
  - 在n维空间中随机采样点（例如单位超立方体）。
  - 几乎所有点都会成为空间边缘的异常值。
  - 任意两点之间的距离将变得几乎相同。

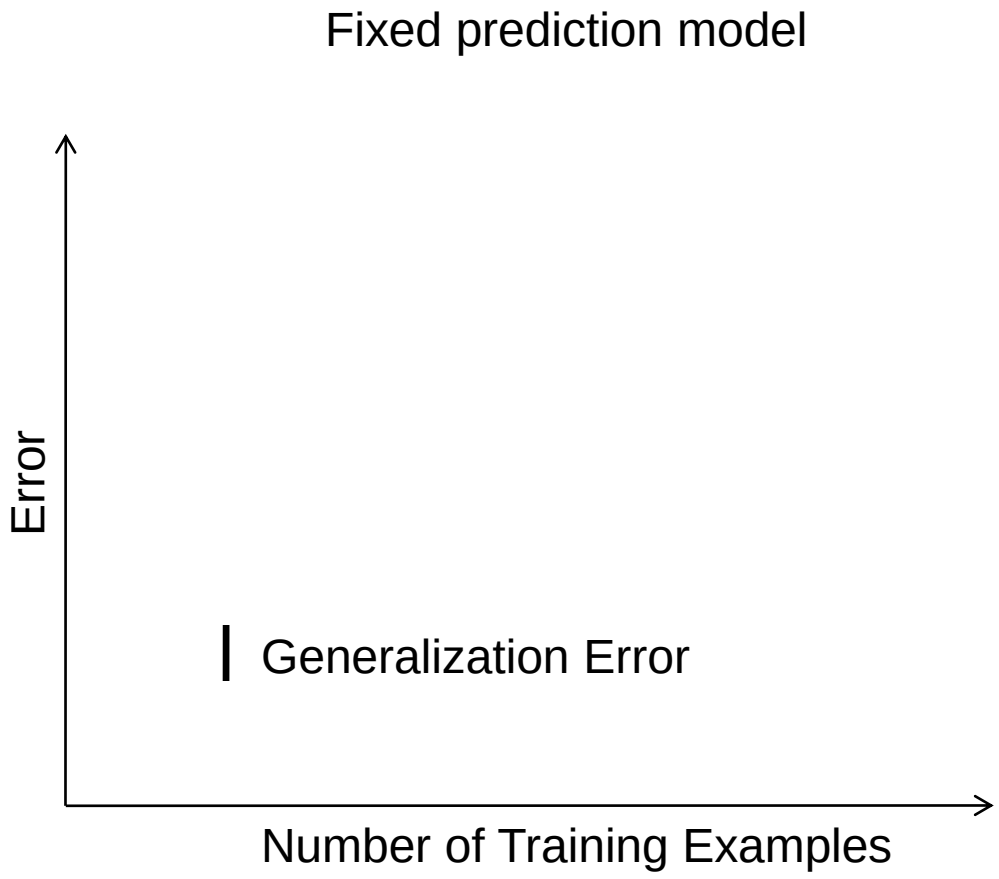


训练集大小的影响



Slide credit: D. Hoiem

训练集大小的影响


Fixed prediction model
Error
Generalization Error
Number of Training Examples

"更多的数据可以战胜一个更聪明的算法"

- (但是实际上你都需要)
- **更多的数据**可以**降低过拟合**的几率。
- 更**稀疏的数据**可以**降低维度诅咒。**

- **非参数模型**：模型参数的数量随着数据量的增加而增加。
  - 例如：基于树的技术、k-最近邻算法、支持向量机等。
  - 它们可以在足够的数据下学习任何模型（但可能会陷入局部最小值）。
- **参数模型**（固定大小）：固定数量的模型参数。
  - 例如：线性模型、神经网络等。
  - 可以给予大量的参数以利用更多的数据。
  - 深度学习模型可以有数百万个权重，学习几乎任何函数。
- 瓶颈在于从数据到计算/可扩展性的转移。

构建机器学习系统

- 一个典型的机器学习系统有多个组件，我们将在接下来的讲座中介绍：
- 预处理：
  - 原始数据很少是理想的学习数据。
  - 特征缩放：将值缩放到相同的范围内。
  - 编码：将分类特征转换为数值特征。
  - 离散化：将数值特征转换为分类特征。
  - 标签不平衡的纠正（例如下采样）。
  - 特征选择：删除不相关或相关的特征。
  - 降维也可以使数据更易于学习。

构建机器学习系统

学习和评估

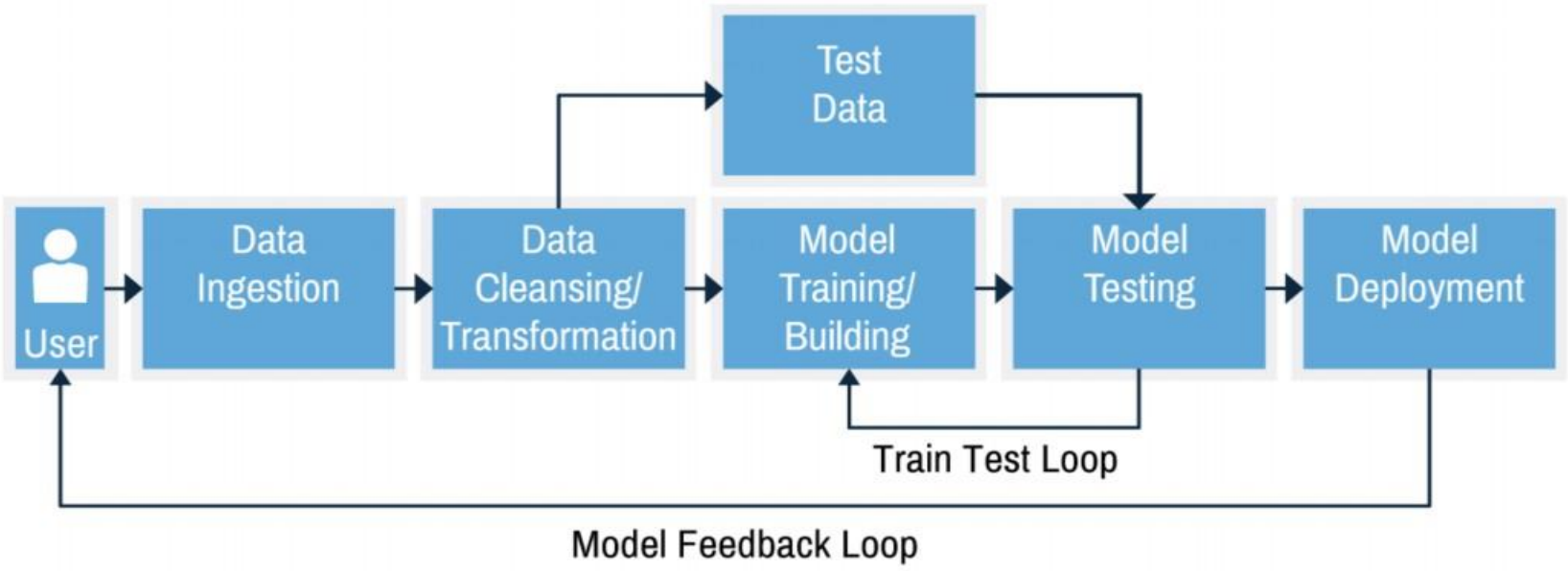
- 每个算法都有其自身的偏差。没有唯一的算法始终是最好的。
- 模型选择比较和选择最佳模型（不同的算法和不同的超参数设置）。
- 将数据分成训练、验证和测试集。

预测

- 最终优化的模型可用于预测。
- 期望的性能是在独立测试集上测量的性能。

构建机器学习系统

- 机器学习的流程由这些部分共同构成。
- 机器学习方法可以自动构建和调整这些流程。
- 你需要不断优化系统流程。
  - 概念漂移：建模的环境可能会随时间变化。
  - 反馈：模型的预测可能会改变未来的数据。



总结

- 学习算法包含三个组成部分:
  - 表征（Representation）：一个模型 f，将输入数据 X 映射到期望的输出 y。
  - 包含可以调整以适应数据 X 的模型参数 θ。损失函数 $L(f\_\theta(X))$ 用于衡量模型拟合数据的程度。
  - 利用优化技术找到最优的 θ，即: $\boldsymbol{\arg\min_{\theta} L(f_{\theta}(X))}$

- 过拟合：模型在训练数据上拟合得很好，但在新的（测试）数据上表现不佳。
  - 将数据分成（多个）训练-验证-测试集。
  - 正则化：调整超参数（在验证集上）以简化模型。
  - 收集更多数据，或构建模型集成。

- 机器学习*流程*：预处理 + 学习 + 部署

致谢

- Lecture partially from Joaquin Vanschoren's course materials

- https://zhuanlan.zhihu.com/p/600015111