上海财经大学
SHANGHAI UNIVERSITY OF FINANCE AND ECONOMICS

高级机器学习理论与算法

授课老师：江敏祺
jiangminqi@mail.shufe.edu.cn

答疑时间：周三下午3:30~5:30

个人简介

- 江敏祺，助理教授，计算机与人工智能学院
- 研究方向：异常检测，时间序列预测，大模型，量化投资
- 联系方式：jiangminqi@mail.shufe.edu.cn
- 办公室：信管学院306
- 答疑时间：预约或周三下午3:30-5:30

基本调研

上课注意事项

- 随时提问，有问必答
- 课间不休息
- 关于点名问题...
- 课堂尽量保持安静（感谢

课程安排

- 考勤，占总成绩**10%**
- 课堂参与，占总成绩**5%**
- 作业3次，占总成绩**35%**
  - 包含一次研读人工智能顶会论文
- 期末项目，占总成绩**50%**
  - 以组队的形式，将课程汇报论文相关算法与模型复现，应用在量化金融预测问题中并改进 (提供代码辅助、算力支持)。最后得分组内基本相同
  - 可能会提前开始，给大家留充足时间
  - 最后采取kaggle打榜的形式，详见：https://www.kaggle.com/competitions/time-series-forecasting-sufe/overview

课程安排

上海财经大学 SHANGHAI UNIVERSITY OF FINANCE AND ECONOMICS

第一章 机器学习初识

§1.1什么是机器学习

§1.2 机器学习相关应用与基本术语

§1.3 线性回归介绍与理论

§1.4 从线性回归到逻辑回归

第二章 机器学习进阶

§2.1 机器学习中的特征工程

§2.2 决策树与集成算法

§2.3 异常检测 (开启深度学习篇章)

第三章 深度学习初识

§3.1 神经网络基础

§3.2 卷积神经网络

§3.3 循环神经网络

§3.4 梯度反向传播与神经网络优化 (了解NN工作原理)

第四章 深度学习进阶

§4.1 Transformer模型

§4.2 大语言模型

§4.3 Time-series Foundation Model

§4.4 异常检测进阶与量化金融预测 (最终考验~)

课程安排

上海财经大学
SHANGHAI UNIVERSITY OF FINANCE AND ECONOMICS

课程定位
这是一门想让各位对人工智能产生兴趣，同时对想深入了解AI技术同学有帮助的课程

课程难度
适中，可能会有部分理论推导，例如Transformer中的Attention推导

课程知识面
偏向深度学习+大模型，但兼顾统计机器学习方法；可能会根据课程进度引入类似强化学习等知识

课程数据结构
Tabular表格/Time-series时间序列
Image图像
Video视频
Graph图结构

参考资料

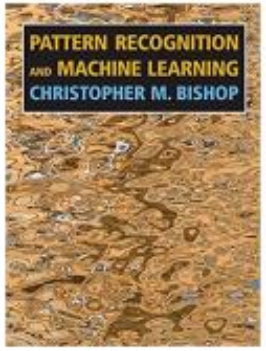
指定教材：[美] 塞巴斯蒂安·拉施卡 / [美] 刘玉溪 / [美] 瓦希德·米尔贾利利著，
《Python机器学习：基于 PyTorch 和 Scikit-Learn》，机械工业出版社

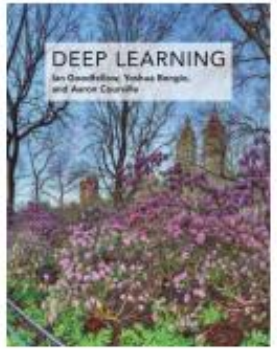
参考书目：

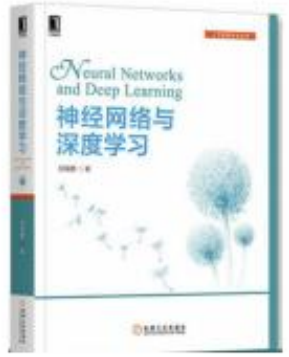
李航著，《机器学习方法》

周志华著，《机器学习》aka 西瓜书

Ian Goodfellow and Yoshua Bengio and Aaron Courville，《Deep Learning》







上海财经大学
SHANGHAI UNIVERSITY OF FINANCE AND ECONOMICS

机器学习

初识

什么是机器学习

- 人类 PK 机器

微湿路面

感到和风

看到晚霞

什么是机器学习

- 人类 PK 机器

色泽青绿

根蒂蜷缩

敲声浊响

什么是机器学习

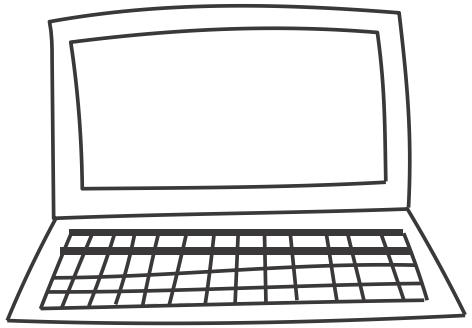
- 人类 PK 机器

人类专家积累了许多经验，而通过对经验的利用，
就能对新情况做出有效的决策

什么是机器学习

- 人类 PK 机器

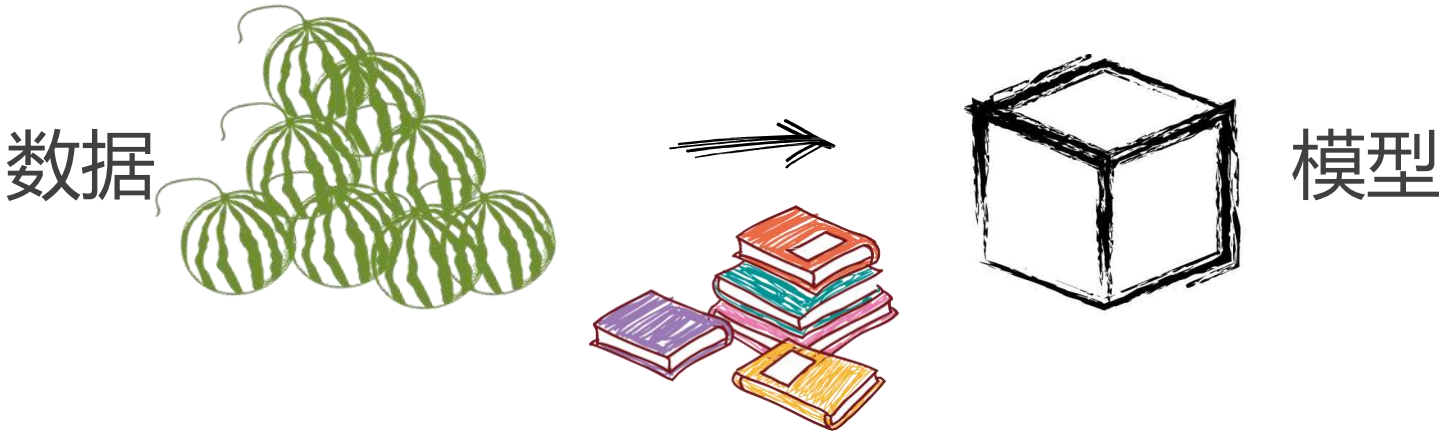
VS



对经验的利用是靠我们人类自身完成的. 计算机能帮忙吗?

什么是机器学习

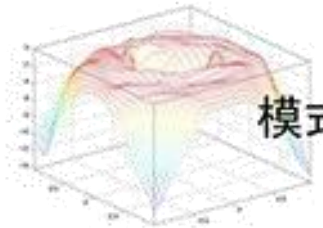
- 定义



“经验”通常以“数据”形式存在，因此，机器学习所研究的主要内容，是关于在计算机上从数据中产生“模型”的算法，即“学习算法”．可以说机器学习是研究关于“学习算法”的学问。

Q: 数据驱动的前提下，如果数据质量很差，算法能学习到吗?

机器学习应用



基本术语

- “数据集”

数据 

西瓜记录1：(色泽=青绿；根蒂=蜷缩；敲声=浊响)
西瓜记录2：(色泽=乌黑；根蒂=稍蜷；敲声=沉闷)
西瓜记录3：(色泽=浅白；根蒂=硬挺；敲声=清脆)

记录的集合称为一个“数据集”(data set)

基本术语

- “示例” / “样本”

西瓜记录1：(色泽=青绿；根蒂=蜷缩；敲声=浊响)
西瓜记录2：(色泽=乌黑；根蒂=稍蜷；敲声=沉闷)
西瓜记录3：(色泽=浅白；根蒂=硬挺；敲声=清脆)

每条记录是关于一个事件或对象(这里是一个西瓜)的描述，称为一个“示例” 或“样本”

基本术语

- “属性” / “特征”

数据

西瓜记录1：(色泽=青绿；根蒂=蜷缩；敲声=浊响)
西瓜记录2：(色泽=乌黑；根蒂=稍蜷；敲声=沉闷)
西瓜记录3：(色泽=浅白；根蒂=硬挺；敲声=清脆)

反映事件或对象在某方面的表现或性质的事项，例如“色泽” “根蒂” “敲声”，称为“属性”或“特征”

基本术语

- “属性值” / “特征值”

数据 

西瓜记录1：(色泽=青绿；根蒂=蜷缩；敲声=浊响)
西瓜记录2：(色泽=乌黑；根蒂=稍蜷；敲声=沉闷)
西瓜记录3：(色泽=浅白；根蒂=硬挺；敲声=清脆)

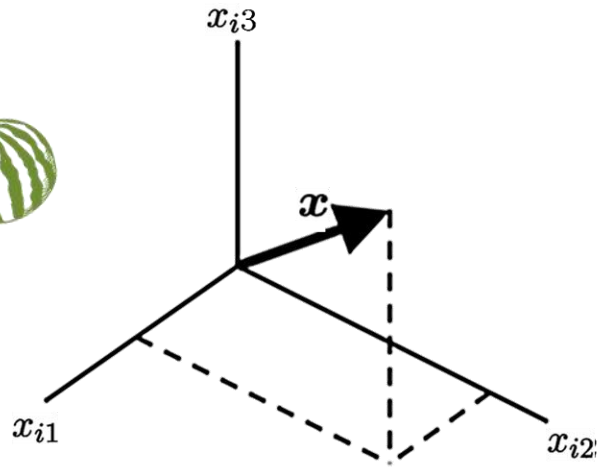
属性上的取值，例如“青绿” “乌黑”，称为“属性值”或“特征值”

基本术语

- “特征向量”

数据

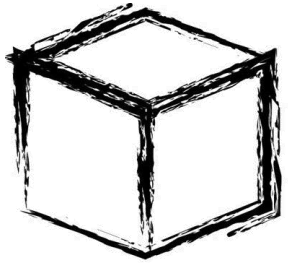
西瓜记录1：(色泽=青绿；根蒂=蜷缩；敲声=浊响)
西瓜记录2：(色泽=乌黑；根蒂=稍蜷；敲声=沉闷)
西瓜记录3：(色泽=浅白；根蒂=硬挺；敲声=清脆)



由于空间中的每个点对应一个坐标向量，
因此我们也把一个示例称为一个
“特征向量”

基本术语

- “训练”/“学习”

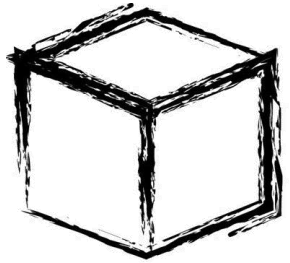
数据    模型

从数据中学得模型的过程称为“学习”或“训练”，这个过程通过执行某个学习算法来完成。

训练过程中使用的数据称为“训练数据”，其中每个样本称为一个“训练样本”，训练样本组成的集合称为“训练集”。

基本术语

- “标记” / “标签”

数据   模型

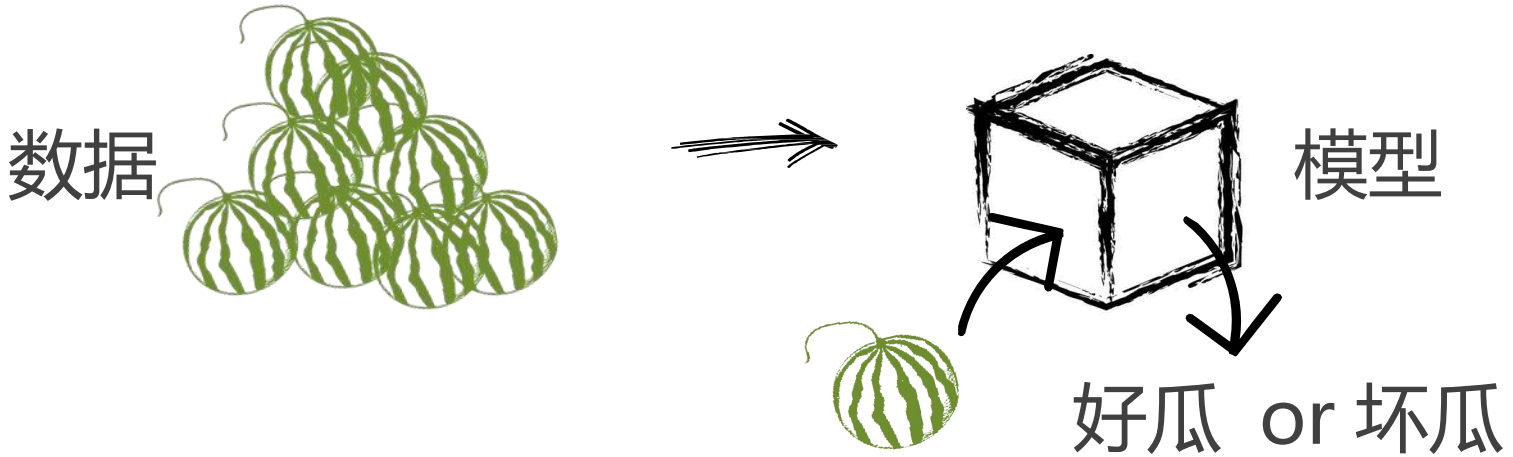
训练出的模型，要能够给出是不是好瓜的预测，通常需要训练集里是包含结果信息的：

西瓜记录1：((色泽=青绿；根蒂=蜷缩；敲声=浊响)，好瓜)
西瓜记录2：(色泽=乌黑；根蒂= 稍蜷；敲声=沉闷)，坏瓜)
西瓜记录3：(色泽=浅白; 根蒂=硬挺；敲声=清脆)，好瓜)

这里关于示例结果的信息，例如“好瓜”，称为“标记”或“标签”

基本术语

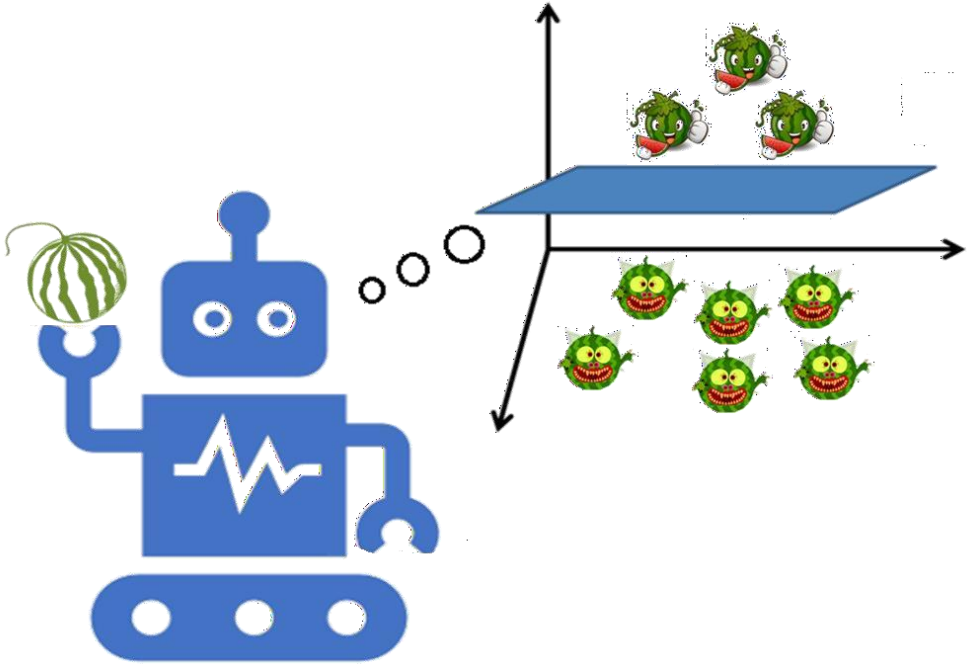
- “测试”

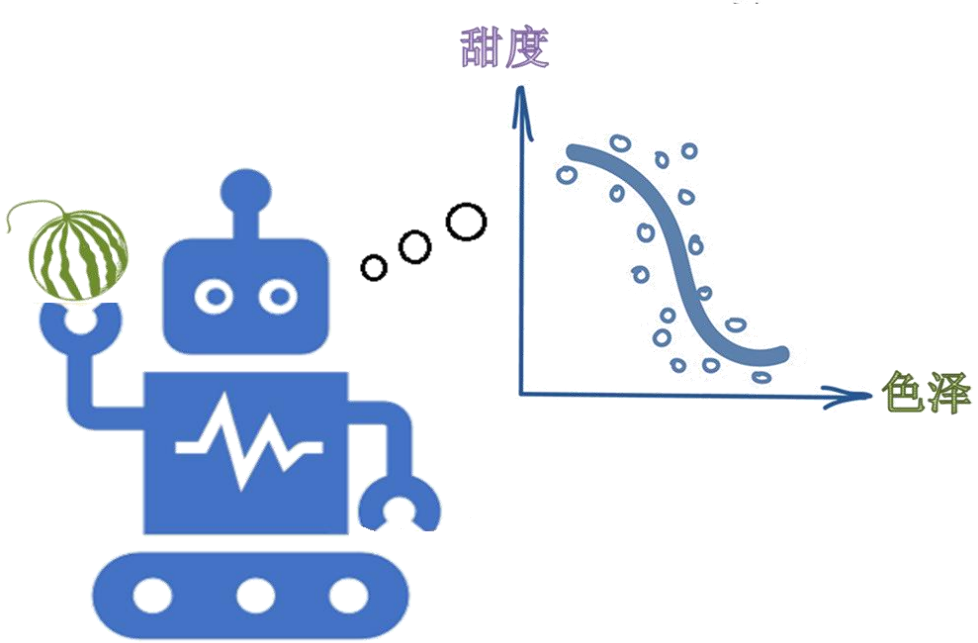


学得模型后，使用其进行预测的过程称为“测试”，
被预测的样本称为“测试样本”。

基本术语

- “分类”&“回归”



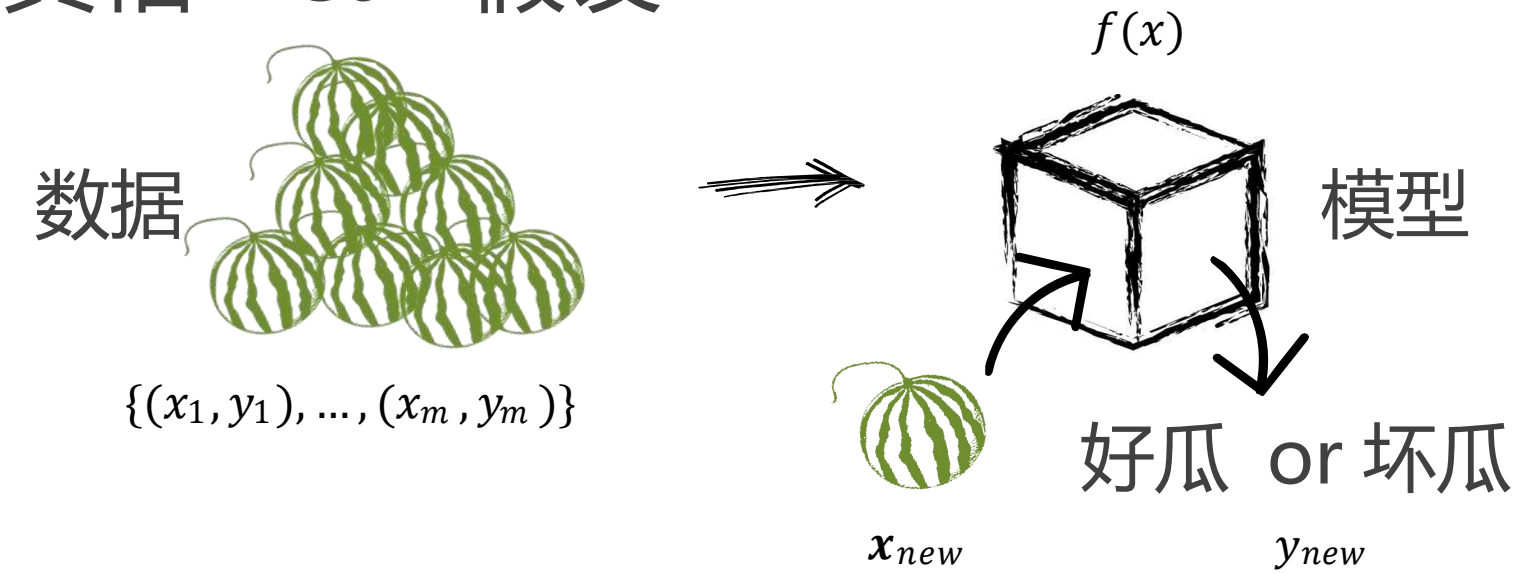


若我们欲预测的是离散值，例如“好瓜” “坏瓜”，此类学习任务称为“分类”

若欲预测的是连续值，例如西瓜甜度 0.95、0.37,此类学习任务称为“回归”

基本术语

- “真相”&“假设”



$f(x)$ 学得模型对应了关于数据的某种潜在的规律，因此亦称“假设”(hypothesis)

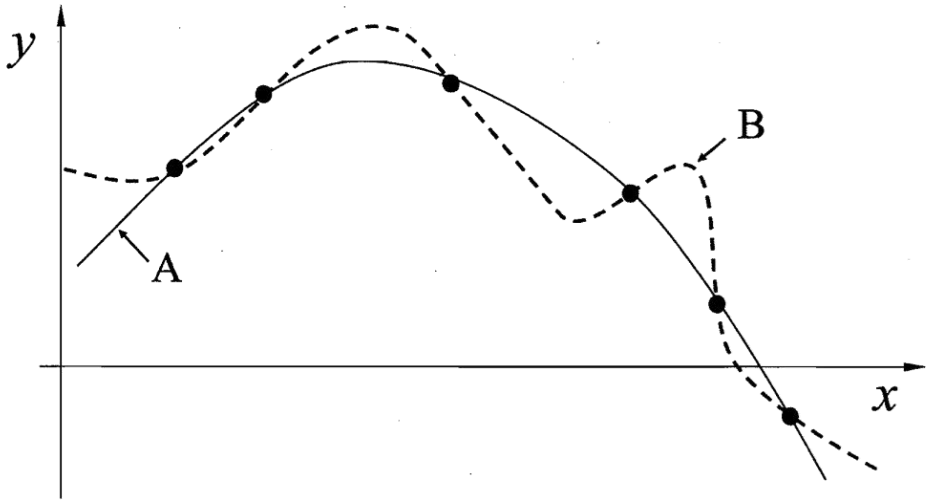
$g(x)$ 潜在规律自身，则称为“真相”或“真实”(ground-truth)

特征 ↑　　标记 ↑

编号	色泽	根蒂	敲声	好瓜
1	青绿	蜷缩	浊响	是
2	乌黑	蜷缩	沉闷	是
3	青绿	硬挺	清脆	否
4	乌黑	稍蜷	沉闷	否
1	青绿	蜷缩	沉闷	?

训练集 ←　　测试集 ←

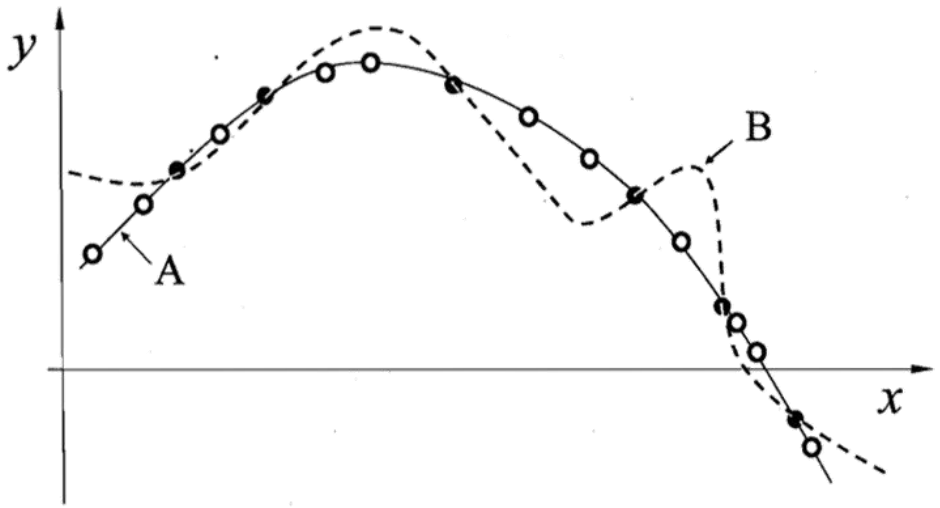
没有免费的午餐



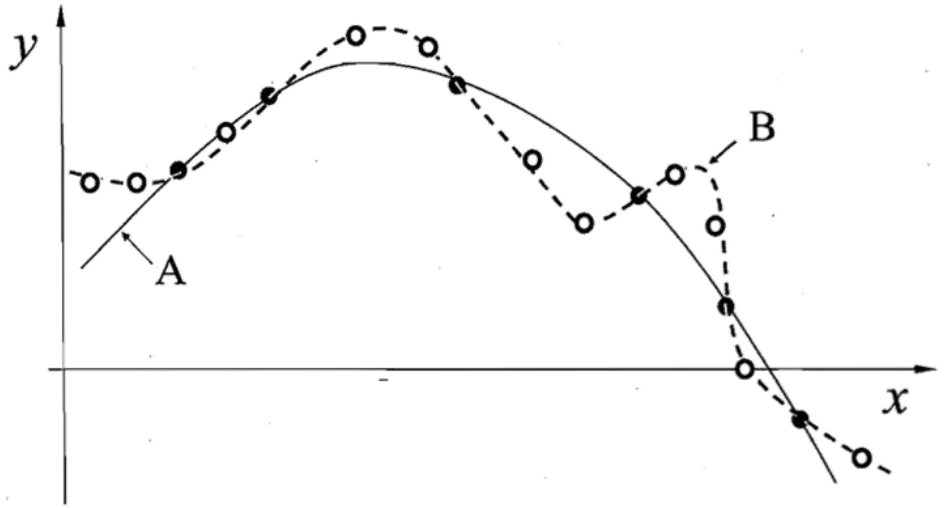
有限样本训练集 ⟶ 多条曲线（假设）

“奥卡姆剃刀”(Occam’s razor)
“若有多个假设与观察一致，则选最简单的那个”

没有免费的午餐



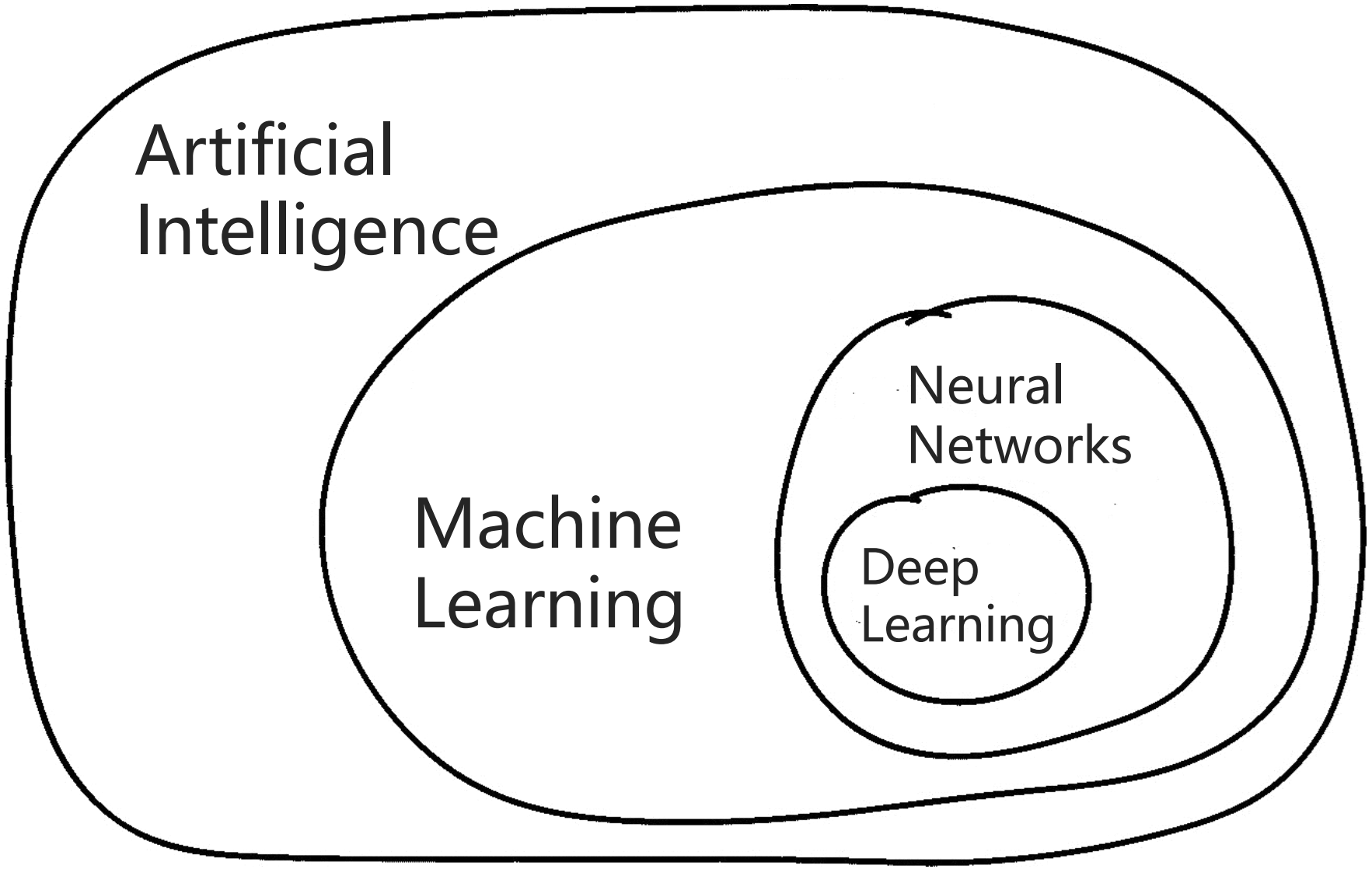
$g(x)$与A更相似，A优于B



$g(x)$与B更相似，B优于A

1、一种算法（算法A）在特定数据集上的表现优于另一种算法（算法B）的同时，一定伴随着算法A在另外某一个特定的数据集上有着不如算法B的表现；
2、具体问题需要具体分析，没有通用的最优机器学习算法。

学科对比

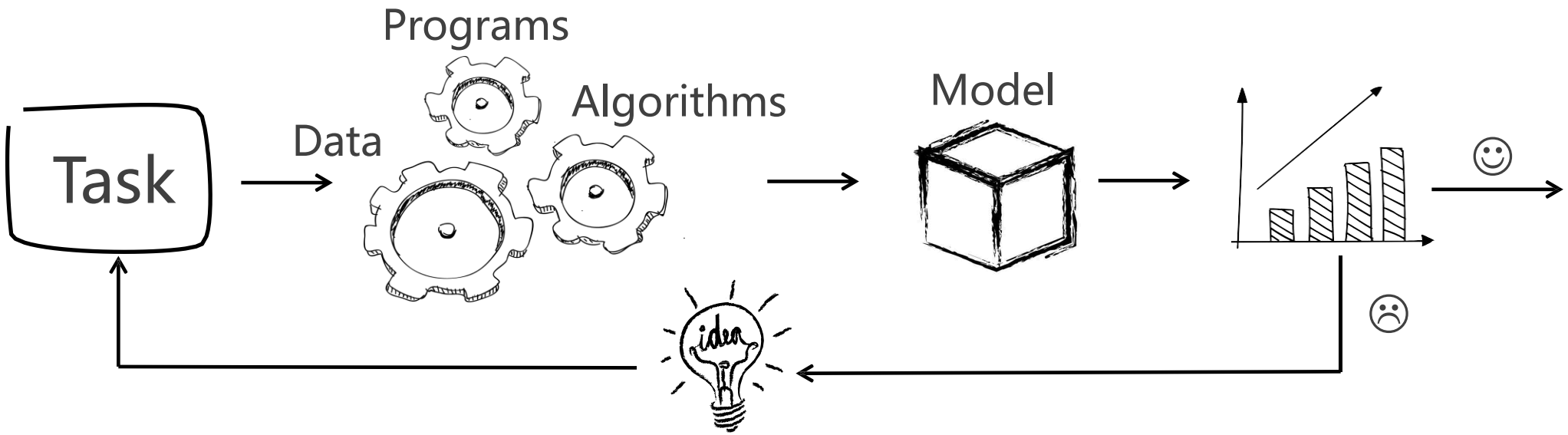

Artificial
Intelligence
Machine
Learning
Neural
Networks
Deep
Learning

数据模态

- Tabular
- Time-series
- Images (CV, Computer Vision)
- Text (NLP, Natural Language Processing)
- Video
- Speech
- …

总结

机器学习研究了一套算法：这套算法能够令到计算机通过机器学习算法通过和某个任务T相关的经验数据E来学习/训练模型。模型能够在任务T上在评估准则P上获得性能改善。



Q: 数据驱动的前提下，如果数据质量很差，算法能学习到吗?

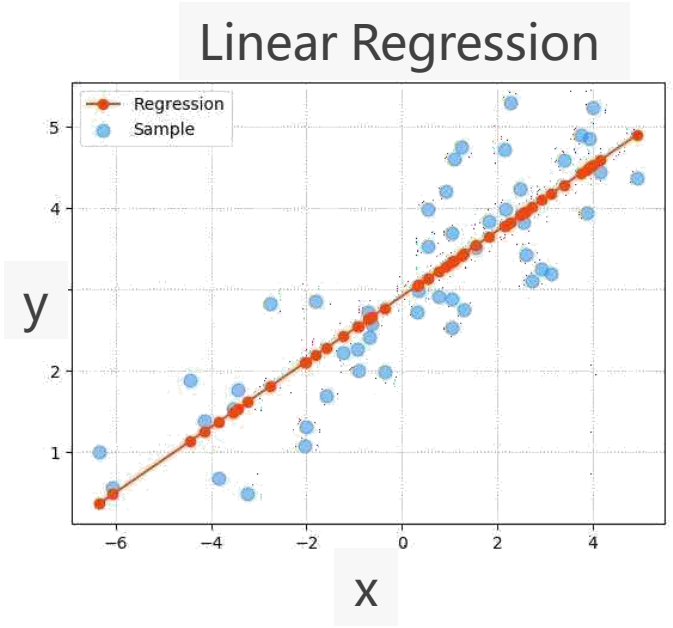
上海财经大学
SHANGHAI UNIVERSITY OF FINANCE AND ECONOMICS

机器学习

线性回归

线性回归

- 定义



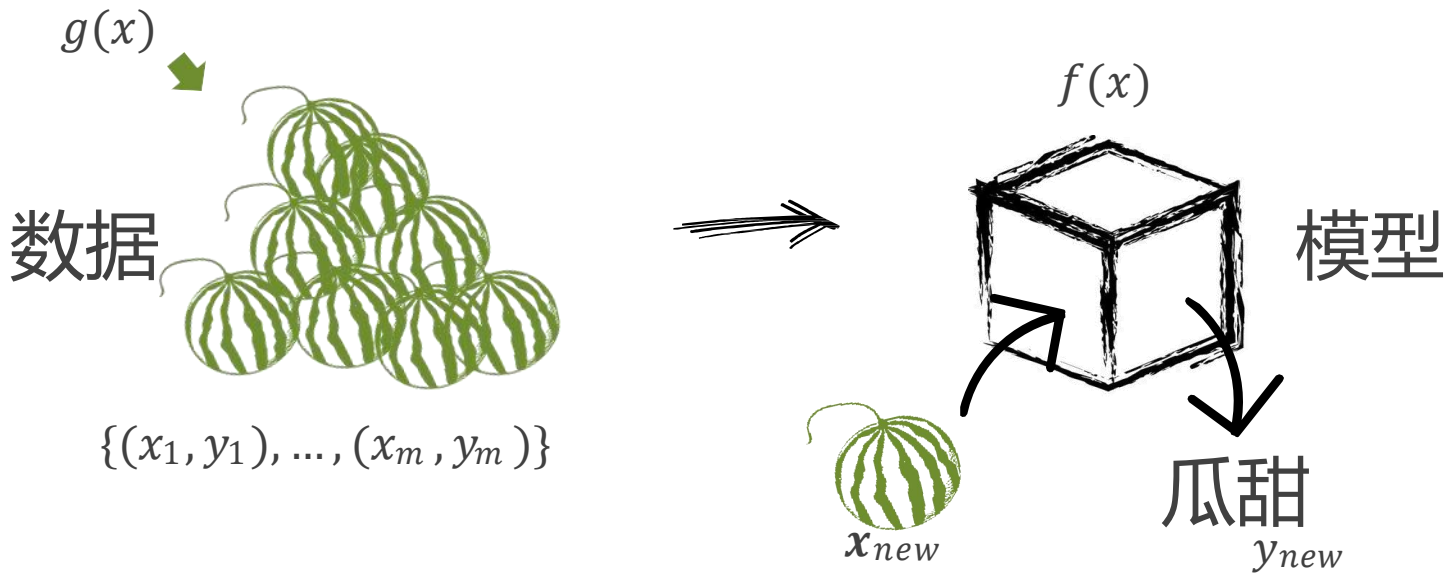
Q: 为什么还多了一个截距项?
适应数据的真实分布
（否则刚性约束是零输入→零输出）

$$f_{瓜甜}(\boldsymbol{x}) = 0.2 \cdot x_{色泽} + 0.5 \cdot x_{根蒂} + 0.3 \cdot x_{敲声} + 1$$

线性回归

◦ 最小二乘法

设定模型的形式：$f(\boldsymbol{x}) = w_1x_1 + w_2x_2 + \cdots + w_dx_d + b$



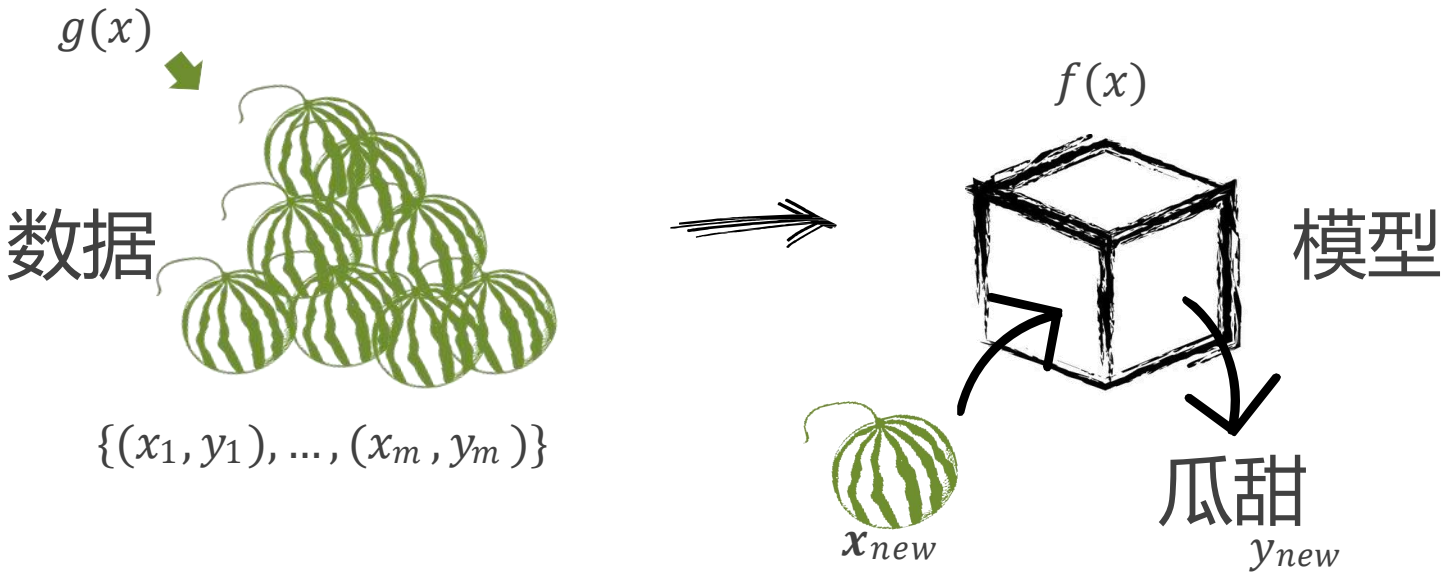
线性回归

。最小二乘法

设定模型的形式：$f(\boldsymbol{x}) = w_1x_1 + w_2x_2 + \cdots + w_dx_d + b$

设定误差的形式（均方误差MSE）：$\ell(f(\boldsymbol{x}_i), y_i) = (f(\boldsymbol{x}_i) - y_i)^2$

利用最小化训练误差求解模型参数：$\underset{(w,b)}{\operatorname{argmin}} \sum_{i=1}^{m} (f(\boldsymbol{x}_i) - y_i)^2 = \underset{(w,b)}{\operatorname{argmin}} \sum_{i=1}^{m} (y_i - w\boldsymbol{x}_i - b)^2$



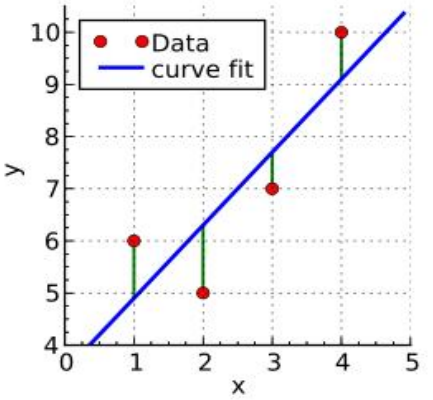
在欧几里得空间中，点$x = (x_1,\ldots,x_n)$和 $y = (y_1,\ldots,y_n)$之间的欧氏距离为

$$d(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots + (x_n - y_n)^2}$$

向量$\vec{x}$的自然长度，即该点到原点的距离为

$$\|\vec{x}\|_2 = \sqrt{|x_1|^2 + \cdots + |x_n|^2}$$

让f(x)=wx+b和y尽可能的靠近



线性回归

◦最小二乘法（二元）

设定模型的形式：$f(\boldsymbol{x}) = w_1x_1 + w_2x_2 + \cdots + w_dx_d + b$ 简化→$f(x) = wx + b$

设定误差的形式：$\ell(f(x_i), y_i) = (f(x_i) - y_i)^2$

利用最小化训练误差求解模型参数：$\underset{(w,b)}{\operatorname{argmin}} \sum_{i=1}^{m} (f(x_i) - y_i)^2 = \underset{(w,b)}{\operatorname{argmin}} \sum_{i=1}^{m} (y_i - wx_i - b)^2$

解析解：

$$w = \frac{\sum_{i=1}^{m} y_i(x_i - \bar{x})}{\sum_{i=1}^{m} x_i^2 - \frac{1}{m}\left(\sum_{i=1}^{m} x_i\right)^2}, \bar{x} = \frac{1}{m}\sum_{i=1}^{m} x_i, \quad b = \frac{1}{m}\sum_{i=1}^{m} (y_i - wx_i)$$

线性回归

利用最小化训练误差求解模型参数：$\underset{(w,b)}{\operatorname{argmin}} \sum_{i=1}^{m}(f(x_i)-y_i)^2 = \underset{(w,b)}{\operatorname{argmin}} \sum_{i=1}^{m}(y_i-wx_i-b)^2$

记$\mathrm{E}(w,b) = \sum_{i=1}^{m}(y_i-wx_i-b)^2$

对参数w求偏导

$\frac{\partial \mathrm{E}(w,b)}{\partial w} = \sum_{i=1}^{m} 2(y_i-wx_i-b)(-x_i) = 2\left(w\sum_{i=1}^{m}x_i^2 - \sum_{i=1}^{m}(y_i-b)x_i\right)$，令其为0

对参数b求偏导

$\frac{\partial \mathrm{E}(w,b)}{\partial b} = \sum_{i=1}^{m} 2(y_i-wx_i-b)(-1) = 2\left(mb - \sum_{i=1}^{m}(y_i-wx_i)\right)$，令其为0 ⇒ $b = \frac{1}{m}\sum_{i=1}^{m}(y_i-wx_i)$

记为$b = \bar{y} - w\bar{x}$，代入上式

$$w\sum_{i=1}^{m}x_i^2 = \sum_{i=1}^{m}(y_i-b)x_i$$

$$w\sum_{i=1}^{m}x_i^2 = \sum_{i=1}^{m}(y_i-\bar{y}+w\bar{x})x_i$$

线性回归

$$w\left(\sum_{i=1}^{m} x_i^2 - \sum_{i=1}^{m} \bar{x}x_i\right) = \sum_{i=1}^{m} (y_i - \bar{y})x_i$$

$$w = \frac{\sum_{i=1}^{m} (y_i - \bar{y})\, x_i}{\sum_{i=1}^{m} x_i^2 - \sum_{i=1}^{m} \bar{x}x_i}$$

然后和书上结果不一样....

变换一下

$$w = \frac{\sum_{i=1}^{m} \left(y_i - \frac{1}{m}\sum_{i=1}^{m} y_i\right) x_i}{\sum_{i=1}^{m} x_i^2 - \bar{x}\sum_{i=1}^{m} x_i}$$

$$w = \frac{\sum_{i=1}^{m} \left(y_i x_i - \frac{1}{m}\sum_{i=1}^{m} x_i y_i\right)}{\sum_{i=1}^{m} x_i^2 - \frac{1}{m}\sum_{i=1}^{m} x_i \sum_{i=1}^{m} x_i}$$

$$w = \frac{\sum_{i=1}^{m} y_i (x_i - \bar{x})}{\sum_{i=1}^{m} x_i^2 - \frac{1}{m}\left(\sum_{i=1}^{m} x_i\right)^2}$$

线性回归

利用最小化训练误差求解模型参数：$\underset{(w,b)}{\operatorname{argmin}} \sum_{i=1}^{m}(f(x_i)-y_i)^2 = \underset{(w,b)}{\operatorname{argmin}} \sum_{i=1}^{m}(y_i-wx_i-b)^2$

记$\mathrm{E}(w,b) = \sum_{i=1}^{m}(y_i-wx_i-b)^2$

对参数w求偏导

$\frac{\partial \mathrm{E}(w,b)}{\partial w} = \sum_{i=1}^{m} 2(y_i-wx_i-b)(-x_i) = 2\left(w\sum_{i=1}^{m}x_i^2 - \sum_{i=1}^{m}(y_i-b)x_i\right)$，令其为0

对参数b求偏导

$\frac{\partial \mathrm{E}(w,b)}{\partial b} = \sum_{i=1}^{m} 2(y_i-wx_i-b)(-1) = 2\left(mb - \sum_{i=1}^{m}(y_i-wx_i)\right)$，令其为0 ⇒ $b = \frac{1}{m}\sum_{i=1}^{m}(y_i-wx_i)$

代入上式

$$w = \frac{\sum_{i=1}^{m} y_i(x_i-\bar{x})}{\sum_{i=1}^{m}x_i^2 - \frac{1}{m}\left(\sum_{i=1}^{m}x_i\right)^2}, \bar{x} = \frac{1}{m}\sum_{i=1}^{m}x_i$$

线性回归

- 最小二乘法(多元、矩阵形式)

$$\mathbf{X}=\begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} & 1 \\ x_{21} & x_{22} & \cdots & x_{2d} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{md} & 1 \end{pmatrix}=\begin{pmatrix} \boldsymbol{x}_1^{\mathrm{T}} & 1 \\ \boldsymbol{x}_2^{\mathrm{T}} & 1 \\ \vdots & \vdots \\ \boldsymbol{x}_m^{\mathrm{T}} & 1 \end{pmatrix} \qquad \boldsymbol{y}=(y_1;y_2;\ldots;y_m)$$

$$\hat{\boldsymbol{w}}^*=arg\min_{\hat{\boldsymbol{w}}}(\boldsymbol{y}-\mathbf{X}\hat{\boldsymbol{w}})^T(\boldsymbol{y}-\mathbf{X}\hat{\boldsymbol{w}})$$

$$E_{\hat{\boldsymbol{w}}}=(\boldsymbol{y}-\mathbf{X}\hat{\boldsymbol{w}})^T(\boldsymbol{y}-\mathbf{X}\hat{\boldsymbol{w}})$$

$$\frac{\partial \boldsymbol{a}^T\mathbf{X}}{\partial \mathbf{X}}=\boldsymbol{a},\frac{\partial \mathbf{X}^T\mathbf{X}}{\partial \mathbf{X}}=2\mathbf{X}$$

$$\frac{\partial E_{\hat{\boldsymbol{w}}}}{\partial \hat{\boldsymbol{w}}}=-2(\boldsymbol{y}-\mathbf{X}\hat{\boldsymbol{w}})\mathbf{X}^T=2\mathbf{X}^T(\mathbf{X}\hat{\boldsymbol{w}}-\boldsymbol{y})$$

$$\hat{\boldsymbol{w}}^*=(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{y}$$

$$f(\hat{\boldsymbol{x}}_i)=\hat{\boldsymbol{x}}_i{}^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{y},\hat{\boldsymbol{x}}_i=(\boldsymbol{x}_i,1)$$

现实任务中 $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ 往往不是满秩矩阵，此时可解出多个$\hat{\boldsymbol{w}}$都能使均方误差最小化. 选择哪一个解作为输出将由学习算法的归纳偏好决定， 常见的做法是引入正则化(regularization)项，例如Lasso, Ridge.

变量$\mathbf{X}$ 矩阵中不同列可能存在线性相关性，或者列数 > 行数（而行秩=列秩），因此很可能不是满秩。

归纳偏好（inductive bias）是指学习算法在面对数据时所隐含的偏好或者假设。

线性回归

梯度下降法

考虑无约束优化问题$minf(\boldsymbol{\theta})$

若能构造一个序列$\boldsymbol{\theta}^0, \boldsymbol{\theta}^1, \boldsymbol{\theta}^2, \ldots$
满足$f(\boldsymbol{\theta}^{t+1}) < f(\boldsymbol{\theta}^t)$, t=0,1,2 , ...
则不断执行该过程即可收敛到局部极小点
根据泰勒展式有$f(\boldsymbol{\theta} + \Delta\boldsymbol{\theta}) \simeq f(\boldsymbol{\theta}) + \Delta\boldsymbol{\theta}^{\mathrm{T}}\nabla f(\boldsymbol{\theta})$
于是, 欲满足$f(\boldsymbol{\theta} + \Delta\boldsymbol{\theta}) < f(\boldsymbol{\theta})$
可选择$\Delta\boldsymbol{\theta} = -\eta\nabla f(\boldsymbol{\theta})$
其中步长η是一个小常数. 这就是梯度下降法

ps: $\Delta\theta^{\mathrm{T}}\nabla f(\theta) = |\Delta\theta|\cdot|\nabla f(\theta)|\cdot\cos\varphi$
最负的情况就是 $\cos\varphi = -1$，即 $\Delta\theta$ 与 $\nabla f(\theta)$ 反向。

p.s.梯度的每个分量是函数相对于每个参数的偏导数

Q: 为什么要用梯度下降法？不直接求解析解？

线性回归

- 梯度下降法求解最小二乘法

> 批量梯度下降

$$w^{(t+1)} = w^{(t)} - \eta \nabla \mathcal{L}$$

$$w^{(t+1)} = w^{(t)} + \eta \sum_{i=1}^{m} \left(y_i - w^{(t)T} x_i\right) x_i$$

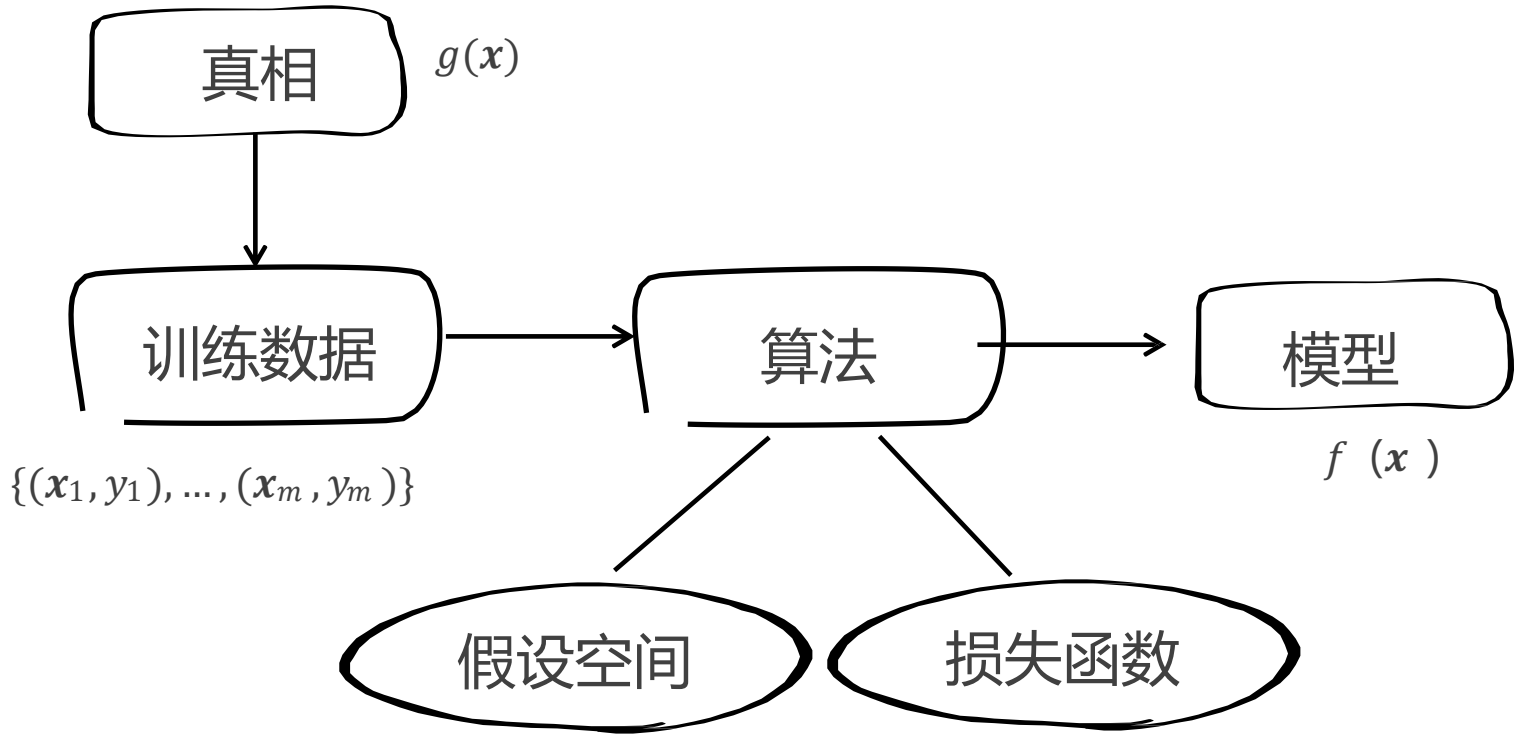
> 随机梯度下降

$$w^{(t+1)} = w^{(t)} - \eta \nabla \mathcal{L}_i$$

$$w^{(t+1)} = w^{(t)} + \eta (y_i - w^{(t)T} x_i) \quad x_i$$

线性回归

- 机器学习框架



e. $g, f(\boldsymbol{x}) = w_1 x_1 + w_2 x_2 + \cdots + w_d x_d + b$

$f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + b$

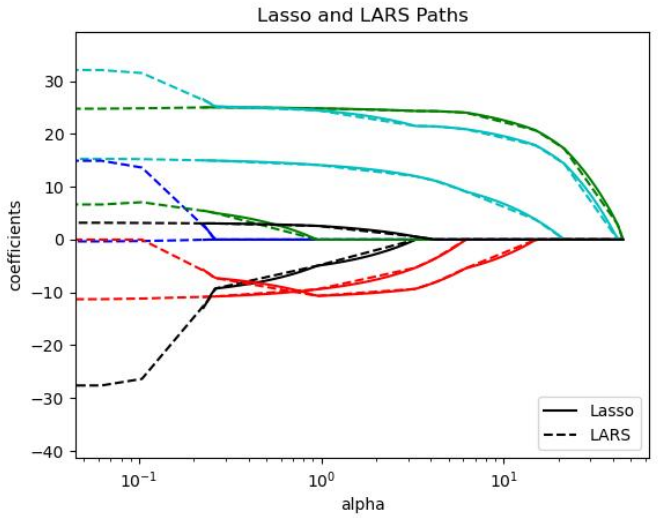
e.g., $P(f(\boldsymbol{x}_i), y_i) = (f(\boldsymbol{x}_i) - y_i)^2$

实践

- Sklearn工具包简介

[官方文档]
[中文文档（非官方）]

Linear Regression
- w/ lasso
- w/ ridge



- ChatGPT
- Gemini
- 通义千问
- KIMI..