机器学习概述

目录

Contents

学习目标

Learning Objectives

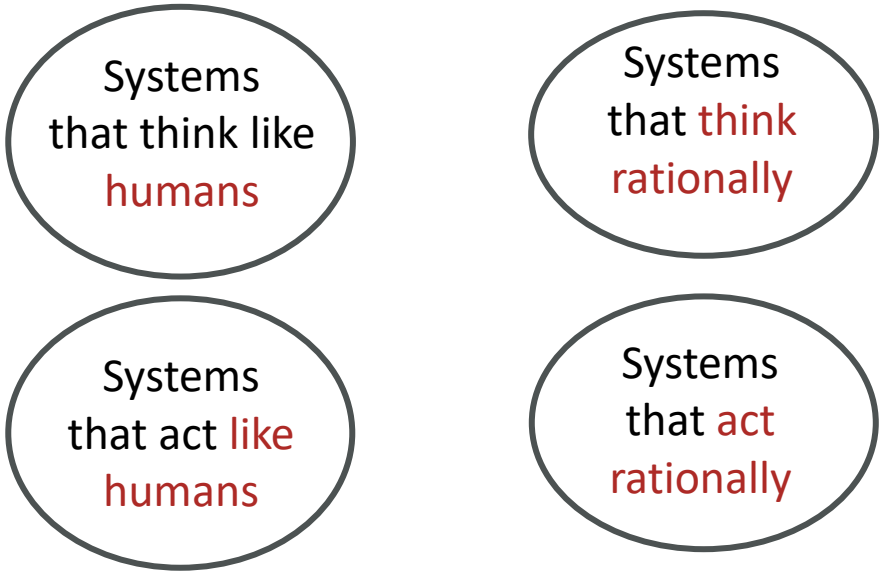
1. 知道AI，ML，DL是什么?
2. 了解AI、ML、DL之间的关系
3. 知道自动学习和规则编程的区别

人工智能的概念

1 什么是人工智能

- Artificial Intelligence 人工智能
- AI is the field that studies the synthesis and analysis of computational agents that act intelligently
- AI is to use computers to analog and instead of human brain
- 释义 - 仿智； 像人一样机器智能的综合与分析；机器模拟代替人类

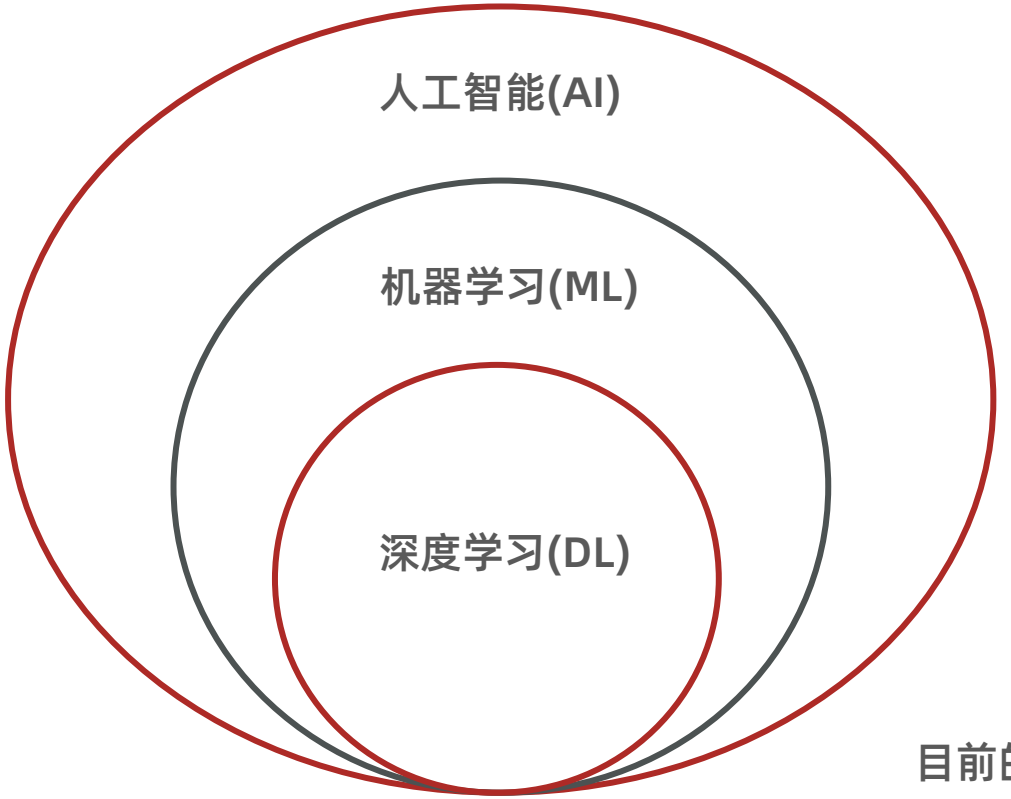
2 AI的期望



- 释义：是一个系统，像人那样思考 像人那样理性思考
- 释义：是一个系统，像人那样活动 像人那样合理行动

人工智能三大概念 - AI概念

- 三者关系



三者关系：

机器学习是**实现人工智能的一种途径**

深度学习是机器学习的**一种方法**发展而来的

目前的人工智能技术体系（软件开发角度）：

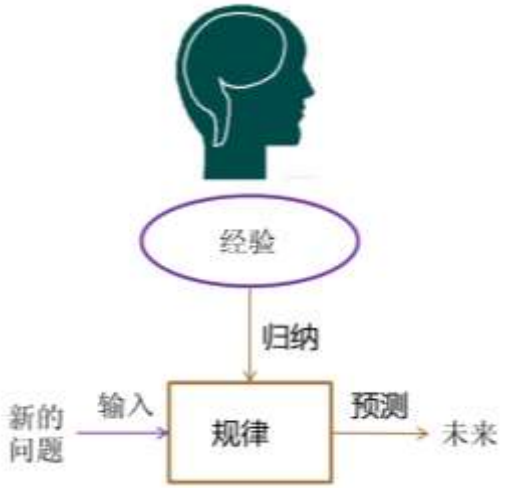
1. 基于**统计学的传统机器学习**方法
2. 基于**神经网络的深度学习**方法

机器学习

1 什么是机器学习

- Machine Learning　机器学习
- Field of study that gives computers the ability to learn without being explicitly programmed
- 释义：让机器自动学习，而不是基于规则的编程（不依赖特定规则编程）

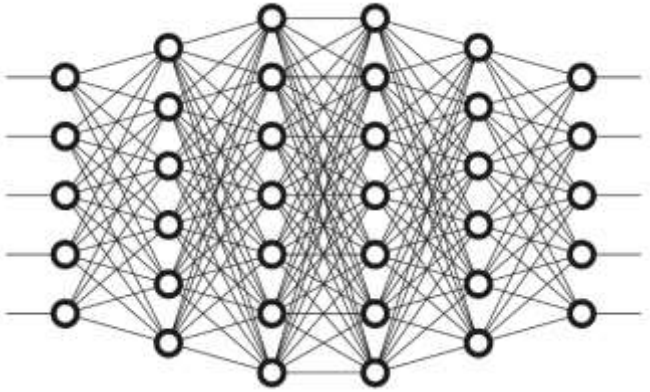
2 机器如何学习

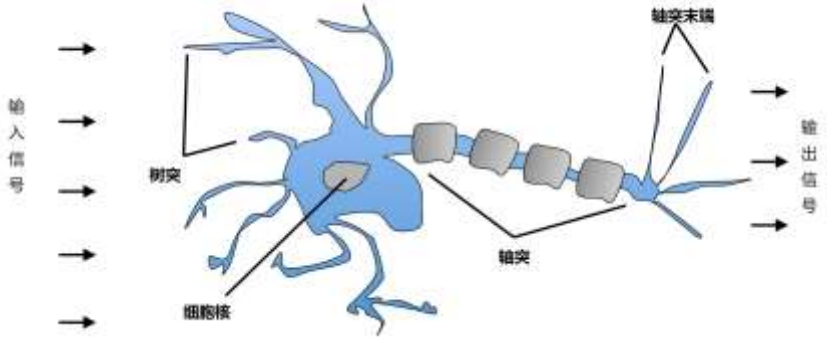


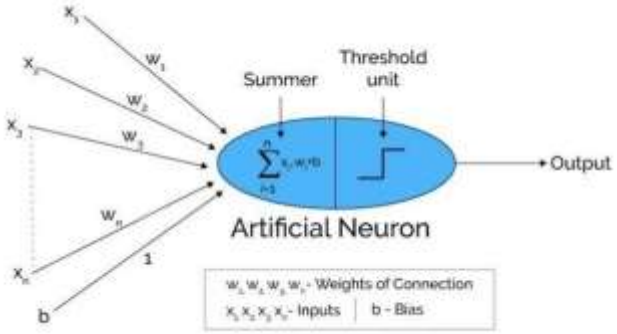
- 人类识别车：根据车的特征归纳出车的规律；来了一个新的图片，判断预测是否是车
- 机器学习识别车: 从数据中获取规律；来了一个新的数据，产生一个新的预测

深度学习

- 深度学习(DL, Deep Learning)：，也叫深度神经网络，大脑仿生，设计一层一层的神经元模拟万事万物

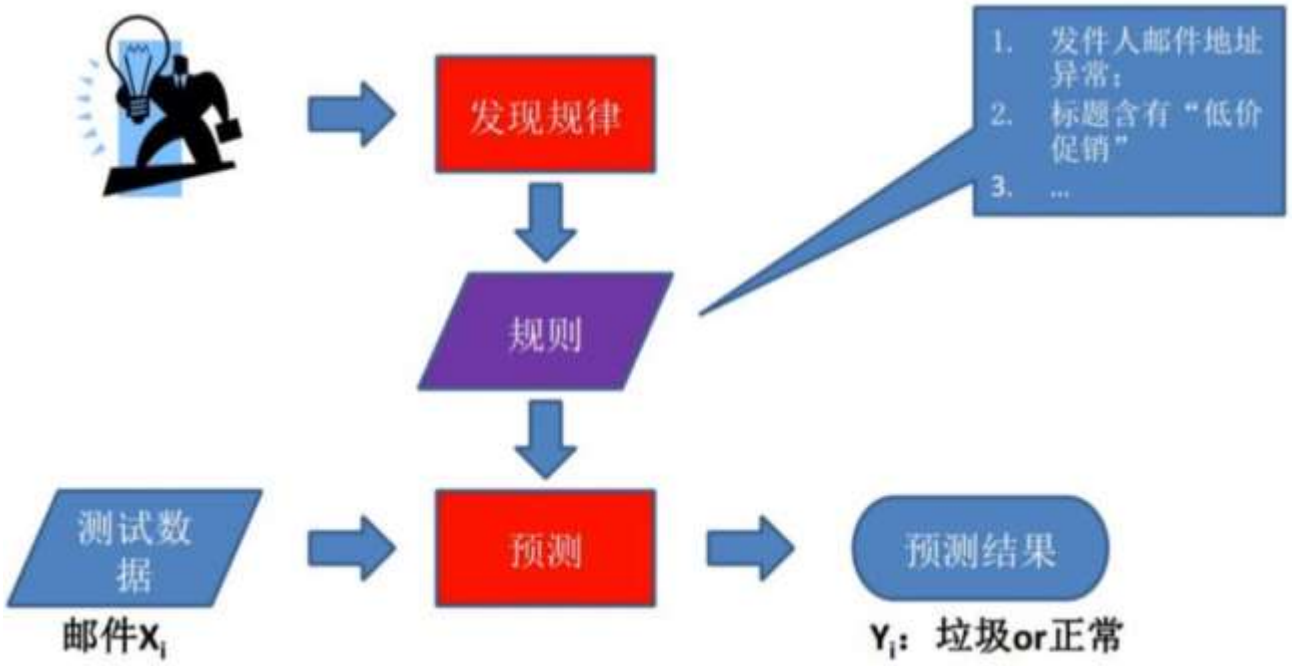






学习方式—基于规则的学习

- 基于规则的学习： 程序员根据经验利用手工的if-else方式进行预测

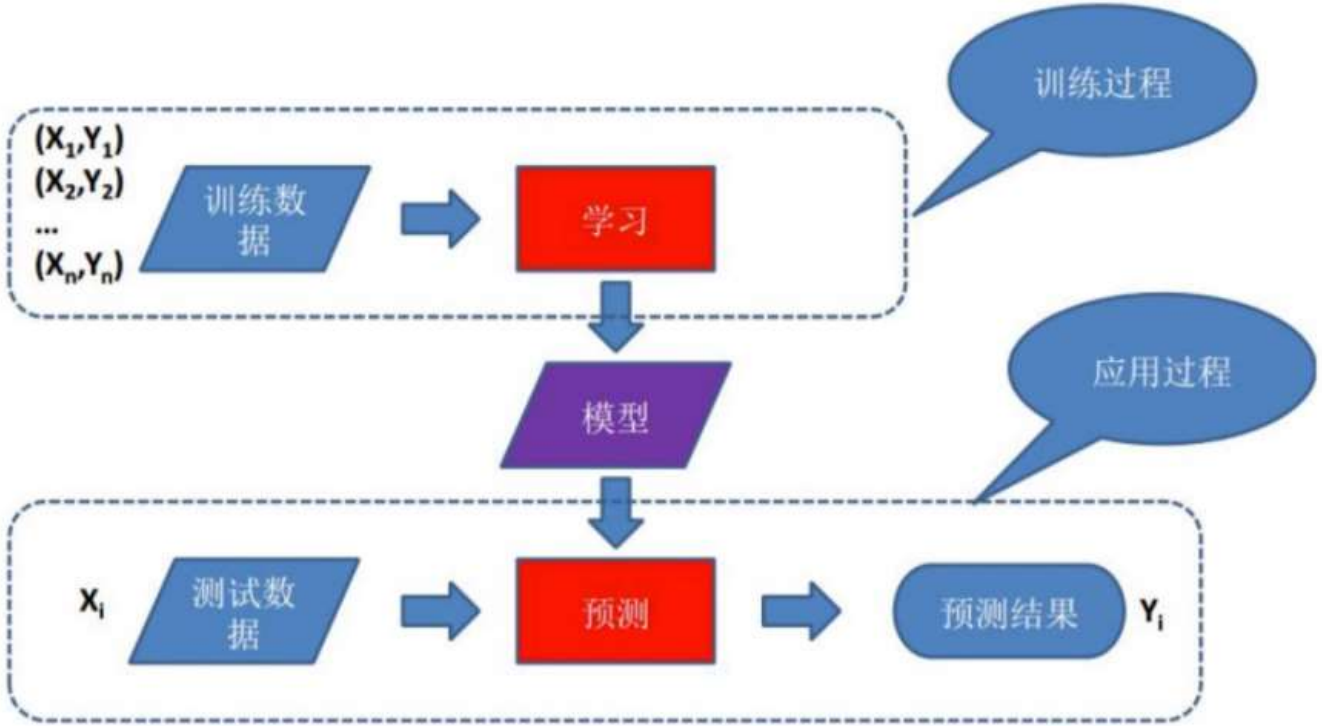


学习方式—基于规则的学习

有很多问题无法明确的写下规则，此时我们无法使用规则学习的方式来解决这一类问题，比如：图像和语音识别和自然语言处理

学习方式—基于模型的学习

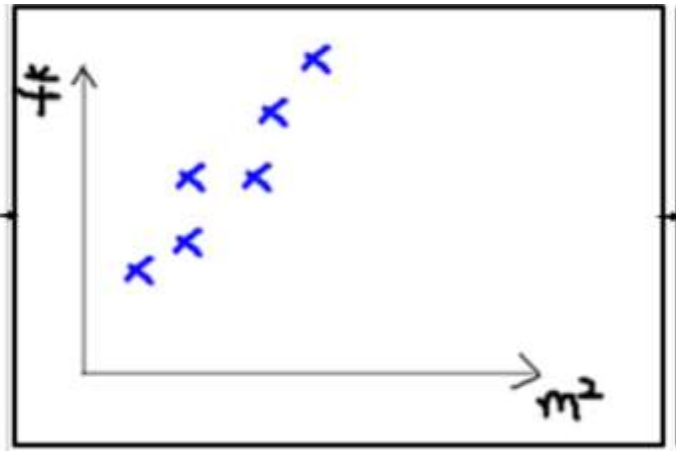
- 基于模型的学习：从数据中自动学出规律

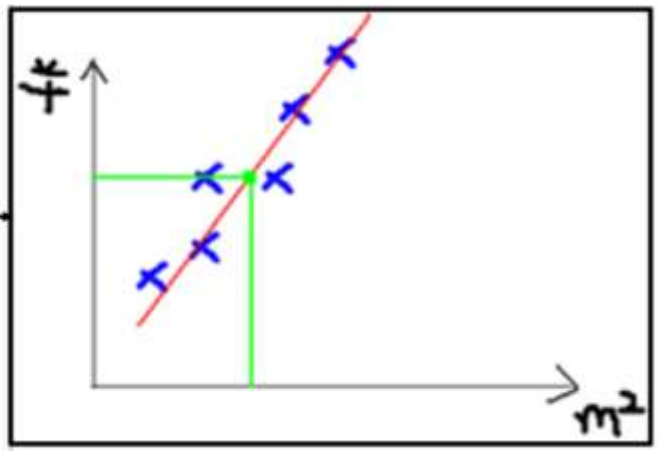


学习方式—基于模型的学习

- 基于模型的学习：房价预测

面积(m^2)	销售价钱（万元）
123	250
150	320
87	160
102	220
…	…





1 利用线性关系来模拟面积和房价之间的关系

eg：让直线尽可能多的经过这些点，不能经过的点分布直线两侧

2 机器学习模型

eg：直线记成y = ax + b 就是模型，其中 a、b 就是我们要训练的模型参数！

总结

1 人工智能

Artificial Intelligence(AI)：仿智，使用计算机来模拟或者代替人类

2 机器学习

Machine Learning(ML)：机器自动学习，不是人为规则编程

3 深度学习

Deep Learning (DL)：大脑仿生，设计一层一层的神经元模拟万事万物

4 AI、ML、DL三者之间的关系

- 机器学习是实现人工智能的一种途径
- 深度学习是机器学习的一种方法发展而来的
- 机器学习 = 传统机器学习 + 深度学习

5 算法的学习方式有哪两种？

- 基于规则的学习（专家系统——》跳棋、象棋、围棋）
- 基于模型的学习（机器学习、深度学习等）

练习

- **有关人工智能概念说法正确的？（多选）**

A）实现人工智能的方法很多，其中机器学习是实现人工智能一种途径、一种方法

B）广义上深度学习是从机器学习发展而来的，两者有区别还有联系

C）深度学习方法是大脑仿生，深度学习方法从机器学习发展而来

D）机器学习就是基于模型自动学习事物特征，而不是程序员手工的编写规则

E）深度学习和机器学习都有各自的应用场景。在研究领域中要根据待解决的问题来选择合理的方法。

答案：ABCDE

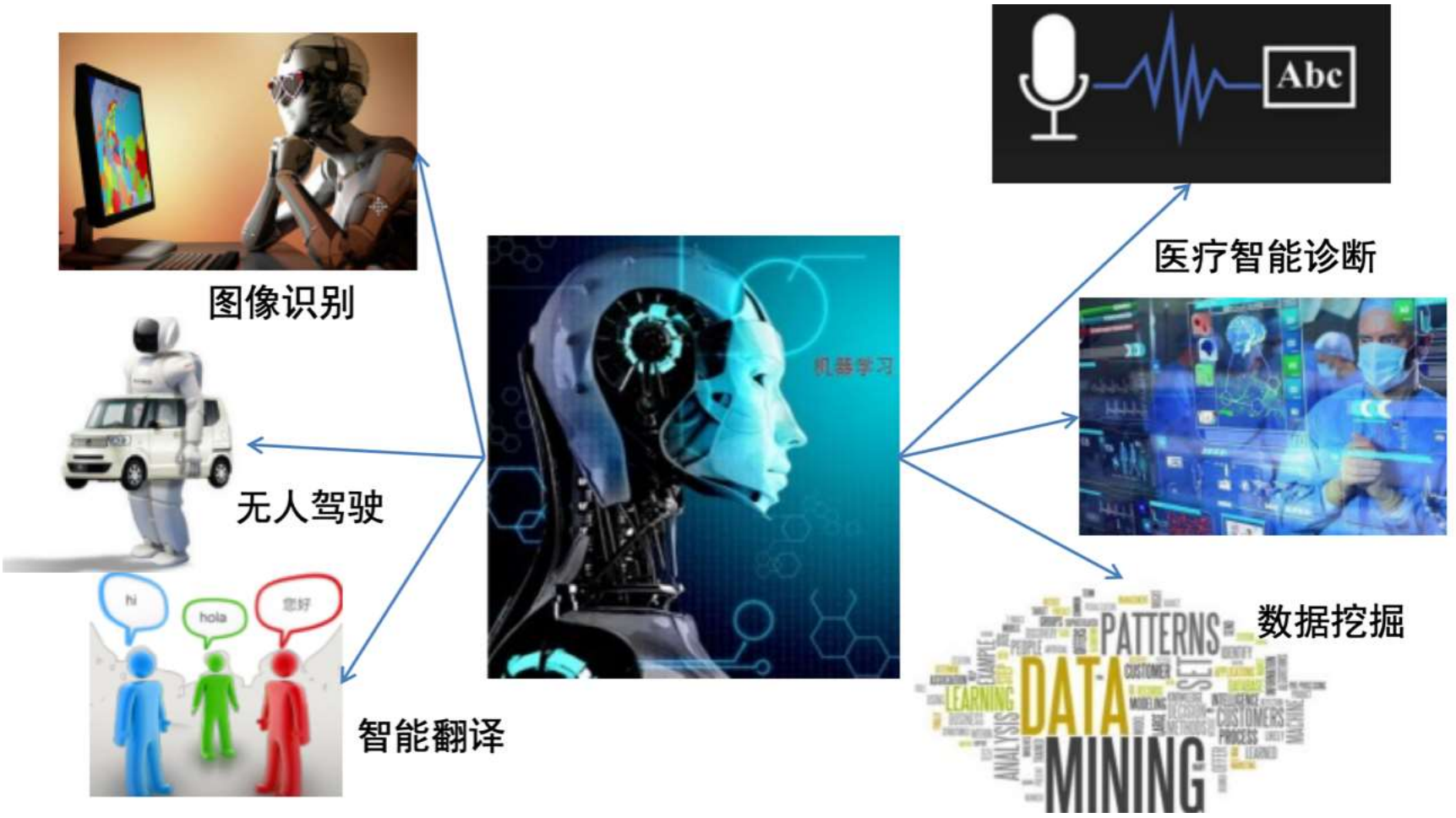
目录

Contents

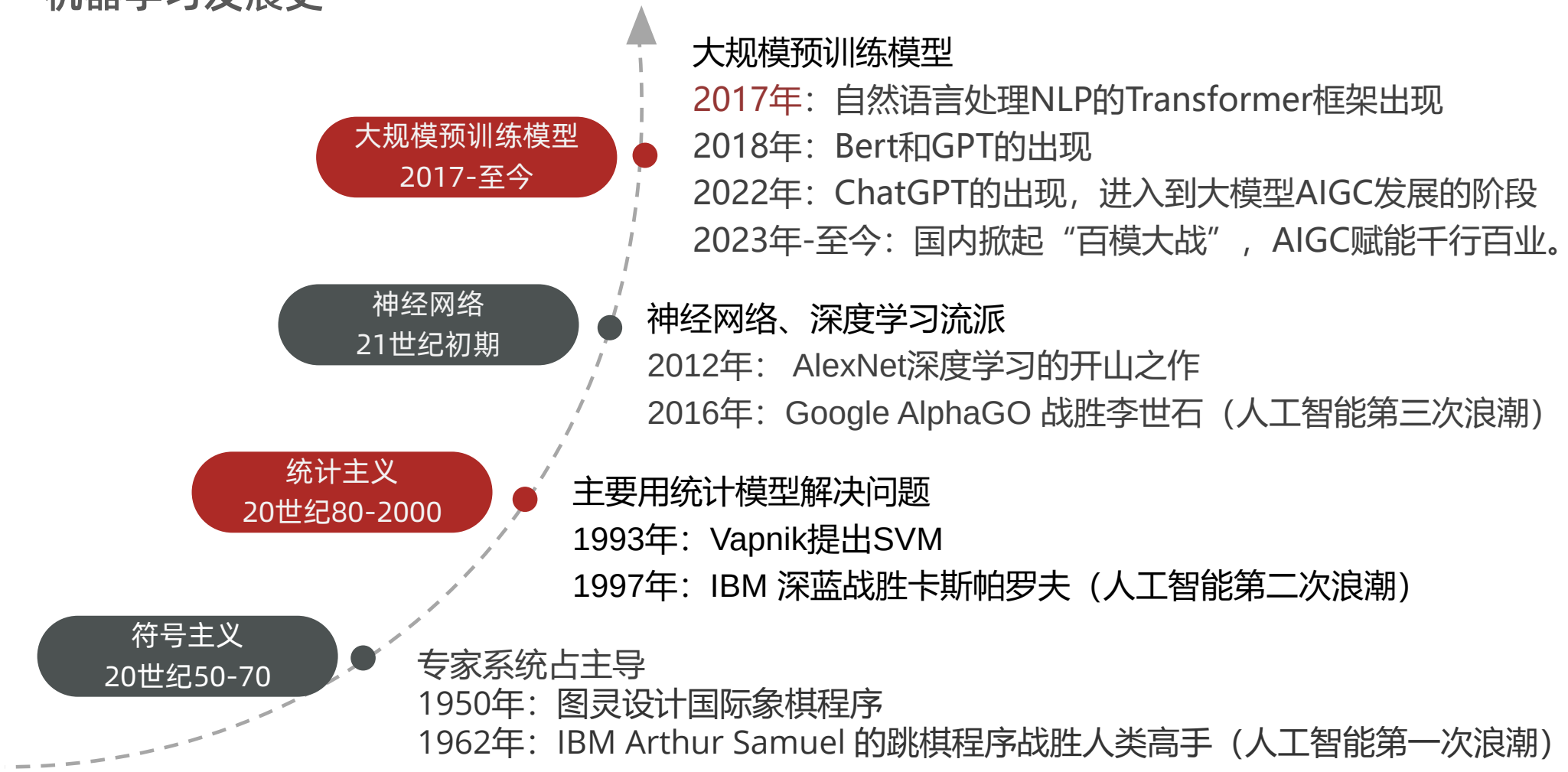
目录

Contents

机器学习的应用领域

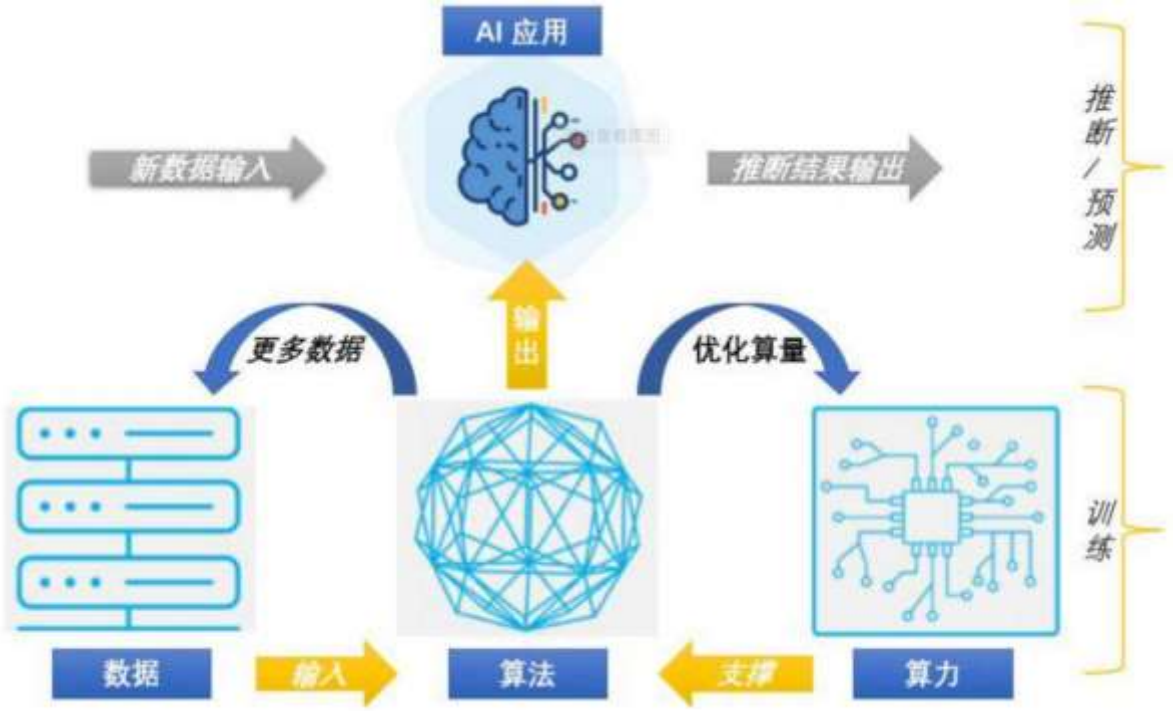

图像识别
无人驾驶
智能翻译
医疗智能诊断
数据挖掘
Abc
hi
hola
您好
PATTERNS
DATA
MINING
LEARNING
ANALYSIS
CUSTOMERS

机器学习发展史

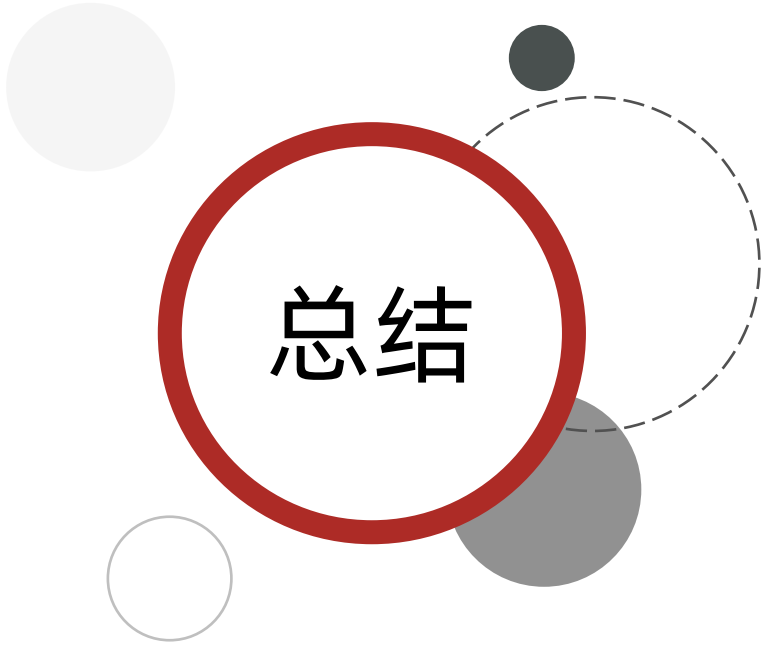

大规模预训练模型
2017-至今
大规模预训练模型
2017年：自然语言处理NLP的Transformer框架出现
2018年：Bert和GPT的出现
2022年：ChatGPT的出现，进入到大模型AIGC发展的阶段
2023年-至今：国内掀起“百模大战”，AIGC赋能千行百业。
神经网络
21世纪初期
神经网络、深度学习流派
2012年： AlexNet深度学习的开山之作
2016年：Google AlphaGO 战胜李世石（人工智能第三次浪潮）
统计主义
20世纪80-2000
主要用统计模型解决问题
1993年：Vapnik提出SVM
1997年：IBM 深蓝战胜卡斯帕罗夫（人工智能第二次浪潮）
符号主义
20世纪50-70
专家系统占主导
1950年：图灵设计国际象棋程序
1962年：IBM Arthur Samuel 的跳棋程序战胜人类高手（人工智能第一次浪潮）

AI发展三要素

- 数据、算法、算力三要素相互作用，是AI发展的基石



- **CPU**：主要适合I\O密集型的任务
- **GPU**：主要适合计算密集型任务
- **TPU**：专门针对大型网络训练而设计的一款处理器



1 机器学习的应用领域

- **计算机视觉CV**：对人看到的东西进行理解
- **自然语言处理**：对人交流的东西进行理解
- **数据挖掘和数据分析**：也属于人工智能的范畴

2 人工智能发展史

- **1956年**人工智能元年（符号主义时代）
- **1993年**Vapnik提出SVM（统计主义时代）
- **2012年**计算机视觉深度神经网络方法研究兴起（CNN）（神经网络时代）
- **2017年**自然语言处理应用大幕拉开（Transformer）（大规模预训练模型时代）
- **2022年**ChatGPT的出现，进入到大模型AIGC发展的阶段（大模型时代）

3 人工智能发展三要素

- **数据，算法，算力**
  - **CPU**：主要适合I\O密集型的任务
  - **GPU**：主要适合计算密集型任务
  - **TPU**：专门针对大型网络训练而设计的一款处理器

目录

Contents

学习目标

Learning Objectives

1. 知道样本是什么?
2. 知道特征是什么?
3. 知道标签/目标值是什么?
4. 理解数据集划分的方法

样本、特征、标签

- 黑马程序员同学就业薪资表

同学编号	培训学科	作业考试	学历	工作经验	工作地点	就业薪资
1	java	90	本科	1	北京	14k
2	java	80	本科	1	武汉	10k
3	AI	90	本科	0	北京	15k
4	AI	92	研究生	2	上海	25k
5	测试	95	本科	0	上海	11k
6	测试	80	专科	0	武汉	7k
…						
n	AI	91	本科	1	上海	?

样本(sample)：一行数据就是一个样本；多个样本组成数据集；有时一条样本被叫成一条记录

特征(feature)：一列数据一个特征，有时也被称为属性

标签/目标(label/target)：模型要预测的那一列数据。本场景是就业薪资

就业薪资 与 培训学科、作业考试、学历、工作经验、工作地点 5个特征有关系

特征如何理解（重点）：特征是从数据中抽取出来的，对结果预测有用的信息 eg:房价预测、车图片识别

数据集划分

- 黑马程序员同学就业薪资表

同学编号	培训学科	作业考试	学历	工作经验	工作地点	就业薪资
1	java	90	本科	1	北京	14k
2	java	80	本科	1	武汉	10k
3	AI	90	本科	0	北京	15k
4	AI	92	研究生	2	上海	25k
5	测试	95	本科	0	上海	11k
6	测试	80	专科	0	武汉	7k
…						
n	AI	91	本科	1	上海	?

训练集

测试集

数据集可划分两部分：训练集、测试集　比例：8 : 2，7 : 3

训练集(training set)：**用来训练模型（**model**）的数据集**

测试集(testing set)：**用来测试模型的数据集**

数据集划分

- 黑马程序员同学就业薪资表

同学编号	培训学科	作业考试	学历	工作经验	工作地点	就业薪资
1	java	90	本科	1	北京	14k
2	java	80	本科	1	武汉	10k
3	AI	90	本科	0	北京	15k
4	AI	92	研究生	2	上海	25k
5	测试	95	本科	0	上海	11k
6	测试	80	专科	0	武汉	7k
…						
n	AI	91	本科	1	上海	?

训练集

测试集

x_train 训练集中的数据（特征）　　y_train 训练集中的标签（目标值）

x_test　测试集中的数据（特征）　　y_test　测试集中的标签（目标值）

总结

1 样本和数据集

- 样本(sample)：一行数据就是一个样本
- 数据集dataset：多个样本组成数据集

2 特征

- 特征(feature)：一列数据一个特征，有时也被称为属性

3 标签

- 标签/目标(label/target)：模型要预测的那一列数据。

4 数据集划分

- 训练集用来训练模型、测试集用来测试评估模型 。
- 一般划分比例7:3 ~ 8:2

练习

- **西瓜数据集如下，可通过西瓜的色泽、根蒂、敲声确定一个西瓜是好瓜或坏瓜，数据集划分形成1/2/3/4 四个部分，其表示正确的是？**

A) 1:x_train, 2:x_test, 3:y_train, 4:y_test

B) 1:train_x, 2:test_x, 3:train_y, 4:test_y

C) 1:x_train, 2:y_train 3:x_test,, 4:y_test

D) 1:train_x, 2:train_y 3:test_x, 4:test_y

解析：x、y放在前头，train、test用来界定（修饰）x、y 。选A

目录

Contents

学习目标

Learning Objectives

1. 知道有监督学习是什么?
2. 知道无监督学习是什么?
3. 知道半监督学习是什么?
4. 了解强化学习是什么?
5. 掌握监督学习、无监督学习的数学表示

有监督学习 & 无监督学习

有监督学习

- 定义：输入数据是由输入特征值和目标值所组成，即输入的训练数据有标签的
- 数据集：需要标注数据的标签/目标值

序号	电影名称	搞笑镜头	拥抱镜头	打斗镜头	电影类型
1	功夫熊猫	39	0	31	喜剧片
2	叶问3	3	2	65	动作片
3	二次曝光	2	3	55	爱情片
4	代理情人	9	38	2	爱情片
5	新步步惊心	8	34	17	爱情片
6	谍影重重	5	2	57	动作片
7	美人鱼	21	17	5	喜剧片
8	宝贝当家	45	2	9	喜剧片
9	唐人街探案	23	3	17	?

无监督学习

- 定义：输入数据没有被标记，即样本数据类别未知，**没有标签**，根据样本间的相似性，对样本集聚类，以发现事物内部结构及相互关系。
- 数据集：不需要标注数据

有监督分类问题 & 回归问题

分类问题

- 目标值（标签值）是不连续的
- 分类种类：二分类、多分类

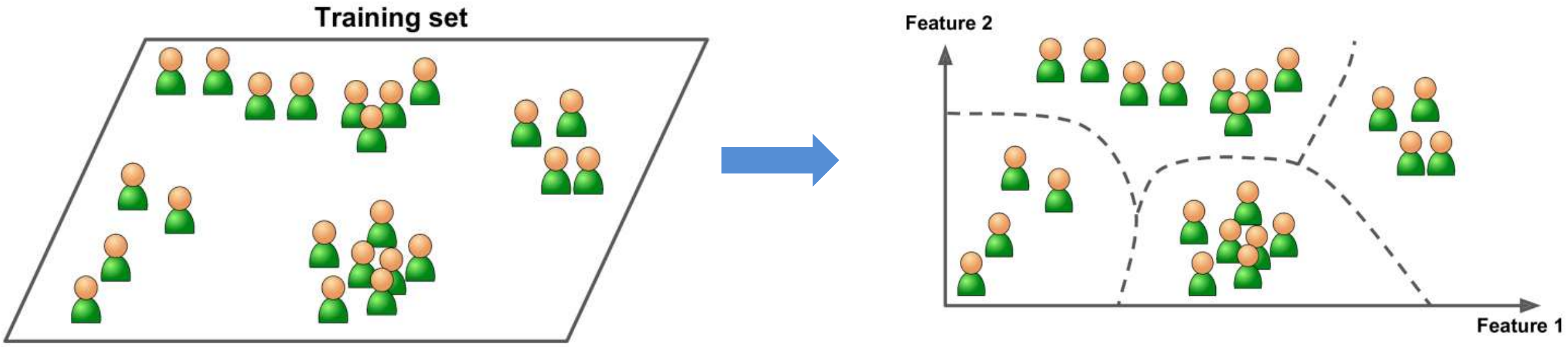
同学编号	培训学科	作业考试	学历	工作经验	工作地点	就业薪资
1	java	90	本科	1	北京	中
2	java	80	本科	1	武汉	中
3	AI	90	本科	0	北京	中
4	AI	92	研究生	2	上海	高
5	测试	95	本科	0	上海	低
6	测试	80	专科	0	武汉	低
...						
n	AI	91	本科	1	上海	?

回归问题

- 目标值（标签值）是连续的

	房子面积	房子位置	房子楼层	房子朝向	房子价格
数据1	80	1	3	0	81
数据2	100	2	5	1	121
数据3	80	3	3	0	102
...					
数据n	90	2	4	1	?

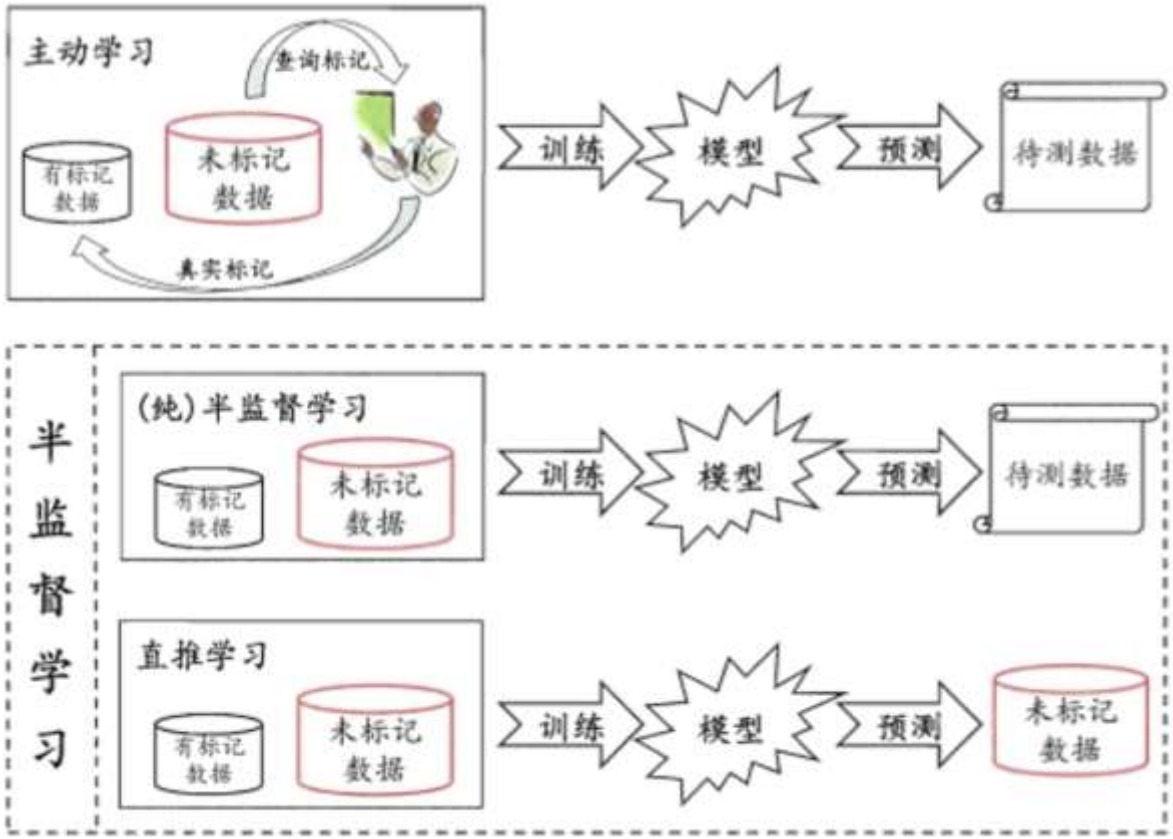
无监督再举例



无监督学习特点： 1 训练数据无标签

2 根据样本间的相似性对样本集进行聚类，发现事物内部结构及相互关系

半监督学习



工作原理：

1 让**专家标注少量数据**，利用已经标记的数据（也就是带有类标签）训练出一个模型

2 再利用**该模型去套用未标记的数据**

3 通过**询问领域专家**分类结果与模型分类结果做对比，从而对模型做进一步改善和提高

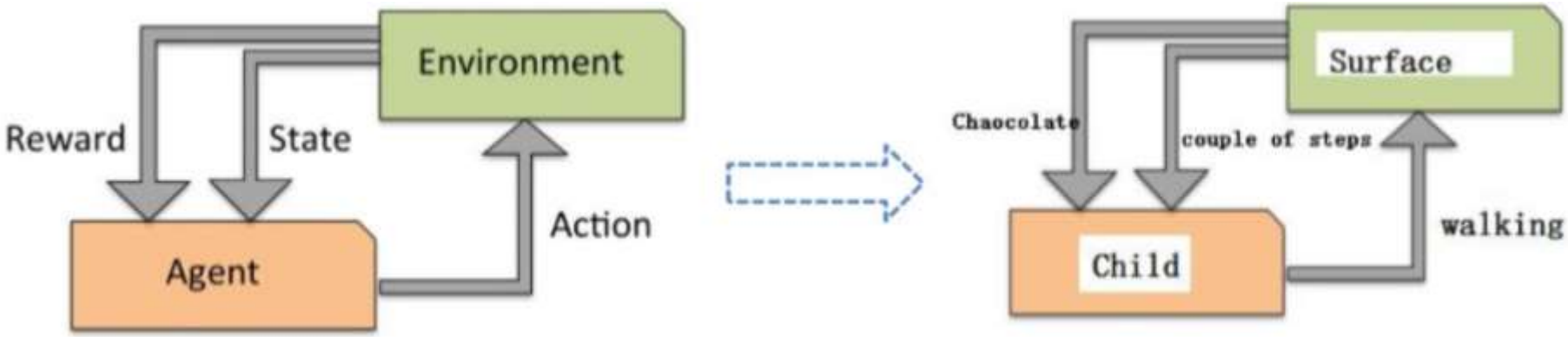
思考有什么好处？

半监督学习方式可**大幅降低标记成本**

机器学习算法分类 - 强化学习

1 强化学习（Reinforcement Learning）：机器学习的一个重要分支

2 应用场景：里程碑AlphaGo围棋、各类游戏、对抗比赛、无人驾驶场景



3 基本原理：通过构建四个要素：Agent，环境状态(State)，行动(Action)，奖励(Reward)，

Agent根据环境状态进行行动获得最多的累计奖励。

4 小孩子学走路：

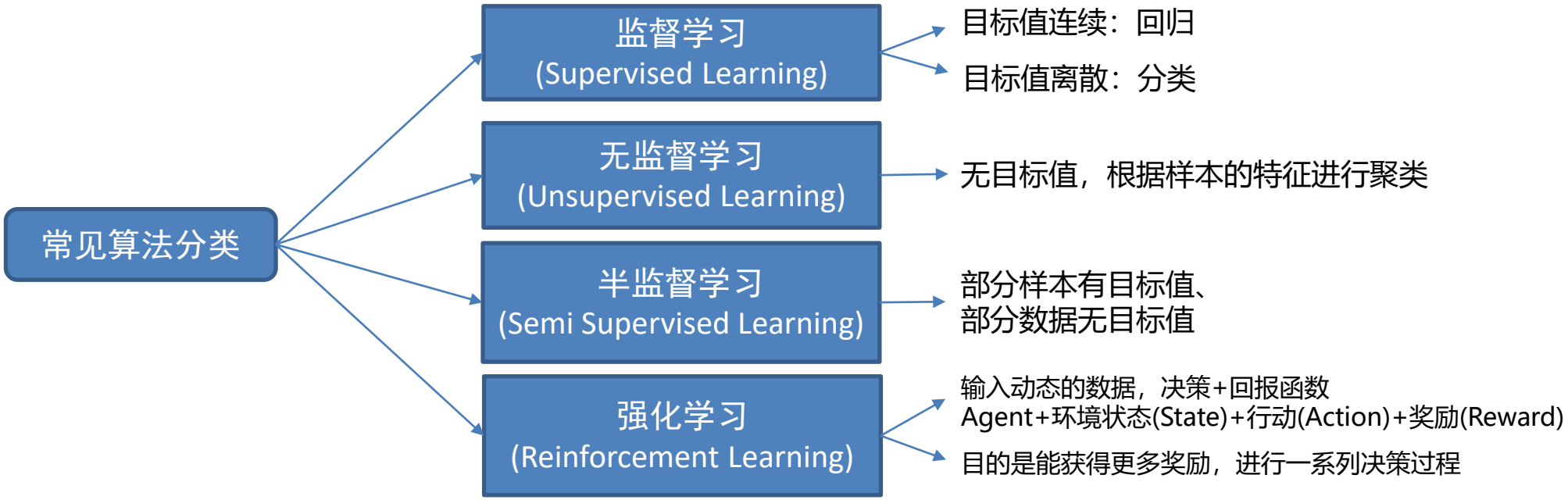
(1) 小孩就是 **Agent**，他试图通过采取**行动**（即行走）来操纵**环境**（行走的表面），

(2) 并且从**一个状态转变到另一个状态**（即他走的每一步），

(3) 当他完成任务的子任务（即走了几步）时，孩子得到**奖励**（给巧克力吃），

(4) 并且当他不能走路时，就不会给巧克力。

机器学习算法分类 - 总结


常见算法分类
监督学习
(Supervised Learning)
目标值连续：回归
目标值离散：分类
无监督学习
(Unsupervised Learning)
无目标值，根据样本的特征进行聚类
半监督学习
(Semi Supervised Learning)
部分样本有目标值、
部分数据无目标值
强化学习
(Reinforcement Learning)
输入动态的数据，决策+回报函数
Agent+环境状态(State)+行动(Action)+奖励(Reward)
目的是能获得更多奖励，进行一系列决策过程

总结

	Input	output	目的	案例
监督学习 (supervised learning)	有标签	有反馈	预测结果	猫狗分类 房价预测
无监督学习 (unsupervised learning)	无标签	无反馈	发现潜在结构	“物以类聚，人以群分”
半监督学习 (Semi-Supervised Learning)	部分有标签，部分无标签	有反馈	降低数据标记的难度	
强化学习 (reinforcement learning)	决策流程及激励系统	一系列行动	长期利益最大化	学下棋

练习

机器学习算法可分为哪些类别？分别说一说各自的特点？

按照学习方式分类可分为: 监督学习, 无监督学习, 半监督学习, 强化学习

- 监督学习(Supervised Learning): 输入训练集数据包含输入特征值和目标值

  回归: 函数的输出是一个连续的值

  分类: 函数的输出是有限个离散值

- 无监督学习(Unsupervised Learning): 输入训练集数据是由输入特征值组成，没有目标值

  比如：聚类根据样本间的相似性对样本集进行分类

- 半监督学习(Semi-Supervised Learning): 训练集同时包含有目标值的样本数据和不含有目标值的样本数据

- 强化学习(Reinforcement Learning): 智能体不断与环境进行交互，通过获取最大奖励的方式（试错的方式）来获得最佳策略；

  主要包含四个元素：Agent(智能体)，环境(State)，行动(Action)，奖励(reward)

目录

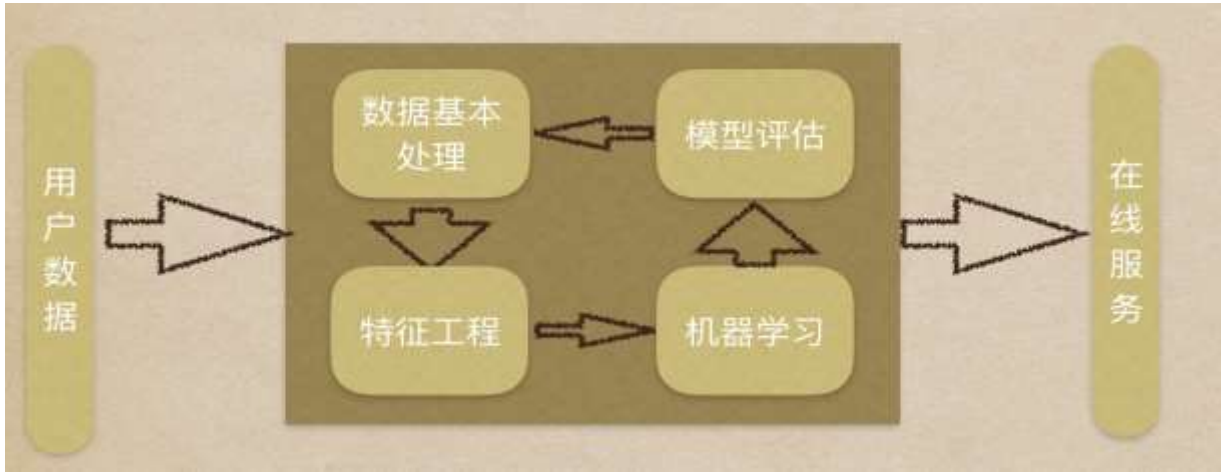
Contents

学习目标

Learning Objectives

1. 掌握机器学习建模流程

机器学习建模流程



1 获取数据

- 获取经验数据
- 图像数据
- 文本数据
- ……

2 数据基本处理

- 数据缺失值处理
- 异常值处理
- ……

3 特征工程

- 特征提取
- 特征预处理
- 特征降维
- ……

4 机器学习(模型训练)

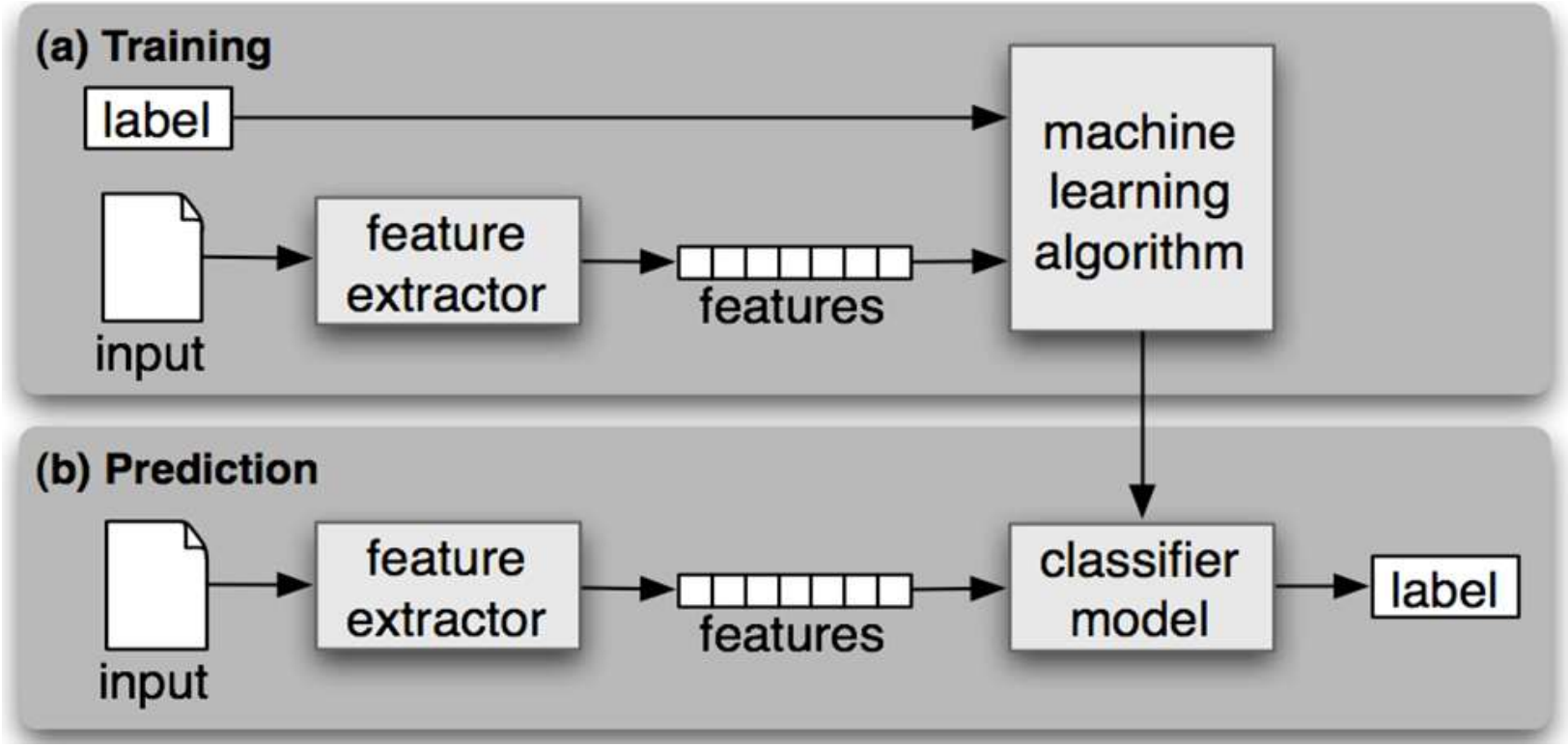
- 线性回归
- 逻辑回归
- 决策树
- GBDT
- ……

5 模型评估

- 回归评测指标
- 分类评测指标
- 聚类评测指标

注：在整个建模流程中，数据基本处理、特征工程一般是耗时、耗精力最多的。

有监督学习模型训练和模型预测



案例 机器学习三大经典任务API编程 - 线性回归模型预测学生成绩

- 已知数据：

学生	平时成绩	期末成绩
1	80	86
2	82	80
3	85	78
4	90	90
5	86	82
6	82	90
7	78	80
8	90	80

?

最终成绩
84.2
80.6
80.1
90
83.2
87.6
79.4
?

- 需求：学生平时成绩为90，80，请预测最终成绩？
- 分析：最终成绩有平时成绩、期末成绩来决定的，本质上是求两者的权重。

有了两者的权重，相当于自主学习了规律，就可以做预测了。

对于这个回归案例如何利用线性回归模型API快速求解呢？

线性回归API介绍

1 导入 线性回归包 → 2 准备 数据 → 3 实例化 线性回归模型 → 4 训练 线性回归模型 → 5 模型 预测

导包 from sklearn.linear_model import LinearRegression	X: 学生成绩 x = [[80,86], [82,80], ...] Y: 最终成绩 y = [84.2, 80.6, ...]	*使用类* LinearRegression 实例化对象 estimator	estimator.fit(x, y) 从数据中获取规律 查看模型参数 斜率 coef_ 截距 intercept_	estimator.predict ([[90, 80]])

案例

机器学习三大经典任务API编程 - 线性回归模型预测学生成绩

- 代码
- 分析

```
# 1 导入依赖包
from sklearn.linear_model import LinearRegression

def dm01_Regression_pred():
    # 2 准备数据 平时成绩 期末成绩 最终成绩
    x = [[80, 86], [82, 80], [85, 78], [90, 90], [86, 82], [82, 90], [78, 80], [92, 94]]
    y = [84.2, 80.6, 80.1, 90, 83.2, 87.6, 79.4, 93.4]

    # 3 实例化 线性回归模型
    estimator = LinearRegression()
    print('estimator-->', estimator)

    # 4 模型训练
    # 打印 线性回归模型参数 coef_ 、 intercept_
    estimator.fit(x, y)
    print('estimator.coef_-->', estimator.coef_)
    print('estimator.intercept_-->', estimator.intercept_)

    # 5 模型预测
    mypred = estimator.predict([[90, 80]])
    print('mypred-->', mypred)
```

```
estimator--> LinearRegression()
estimator.coef_--> [0.3 0.7]
estimator.intercept_--> -1.4210854715202004e-14
mypred--> [83.]
```

案例

机器学习三大经典任务API编程 - 线性回归模型预测学生成绩

- 代码
- 分析

```
# 保存模型 加载模型-恢复模型参数 再预测
# 6 模型保存 joblib.dump(estimator, xxpath)
# 7 模型加载 joblib.load(xxpath)
import joblib
def dm02_Regression_save_load():

    ... ...
    # 6 模型保存
    print('\n模型保存和模型重新加载')
    joblib.dump(estimator, './model/mylrmodel01.bin')

    # 7 模型加载
    myestimator2 = joblib.load('./model/mylrmodel01.bin')
    print('myestimator2-->', myestimator2)

    # 8 模型预测
    mypred2 = myestimator2.predict([[90, 80]])
    print('mypred2-->', mypred2)
```

```
estimator--> LinearRegression()
mypred--> [83.]

模型保存和模型重新加载
myestimator2--> LinearRegression()
mypred2--> [83.]
```

案例 机器学习三大经典任务API编程 - 使用KNN模型进行影片分类

- 已知数据：

序号	电影名称	搞笑镜头	拥抱镜头	打斗镜头	电影类型
1	功夫熊猫	39	0	31	喜剧片
2	叶问3	3	2	65	动作片
3	二次曝光	2	3	55	爱情片
4	代理情人	9	38	2	爱情片
5	新步步惊心	8	34	17	爱情片
6	谍影重重	5	2	57	动作片
7	美人鱼	21	17	5	喜剧片
8	宝贝当家	45	2	9	喜剧片
9	唐人街探案	23	3	17	?

- 需求：唐人街探案为[23, 3, 17]，请预测该影片属于什么类型？

- 分析：电影类型由搞笑、拥抱、打头镜头来影响的。

  利用KNN算法训练模型，进行预测。

  对于这个分类案例如何利用KNN模型API快速求解呢？

KNN模型API介绍

1 导入 线性回归包 → 2 准备 数据 → 3 实例化 线性回归模型 → 4 训练 线性回归模型 → 5 模型 预测

导包from sklearn.neighbors import KNeighborsClassifier	X: 影片特征 x = [[[39, 0, 31]], [3, 2, 65], ...] Y: 最终成绩 y = [0, 1, 2, ...]	*使用类* KNeighborsClassifier 实例化对象 estimator	estimator.fit(x, y) 从数据中获取规律	estimator.predict ([[23, 3, 17]])

案例

机器学习三大经典任务API编程 -使用KNN模型进行影片分类

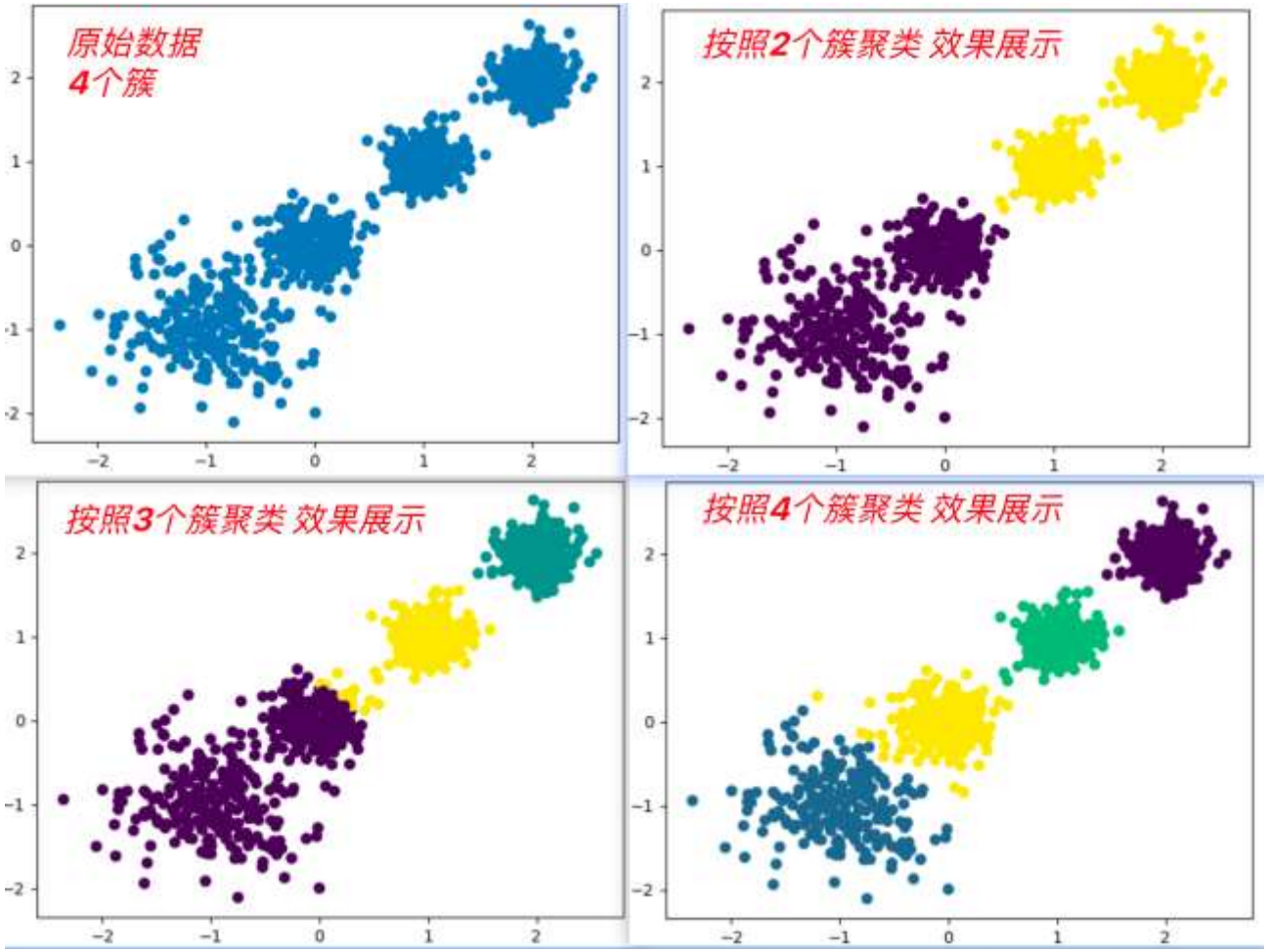
- 代码
- 分析

```
# 1 导入依赖包
from sklearn.neighbors import KNeighborsClassifier
def dm03_knn_clas():
    # 2 准备数据 # 0-喜剧片 1-动作片 2-爱情片
    x = [[39, 0, 31],  # 0
         [3, 2, 65],   # 1
         [2, 3, 55],   # 2
         [9, 38, 2],   # 2
         [8, 34, 17],  # 2
         [5, 2, 57],   # 1
         [21, 17, 5],  # 0
         [45, 2, 9]]   # 0
    y = [0, 1, 2, 2, 2, 1, 0, 0]

    # 3 实例化模型
    estimator = KNeighborsClassifier(n_neighbors=3)
    print('estimator-->', estimator)

    # 4 模型训练.fit()
    estimator.fit(x, y)

    # 5 模型预测.predict() 搞笑镜头23 拥抱镜头3 打动镜头17
    mypre = estimator.predict([[23, 3, 17]])
    print('mypre-->', mypre)
```

```
estimator--> KNeighborsClassifier(n_neighbors=3)
mypre--> [0]
```

案例 机器学习三大经典任务API编程 - 使用KMeans模型数据探索聚类

- 已知数据
- 模型效果



- 分析：根据样本间的相似性对样本集进行聚类，发现事物内部结构及相互关系

案例

机器学习三大经典任务API编程 -使用KMeans模型数据探索聚类

- 代码
- 分析

```
# 1 导包 sklearn.cluster.KMeans sklearn.datasets.make_blobs
# 2 创建数据集
# 2-1 展示数据效果
# 3 实例化Kmeans模型并预测
# 4 展示聚类效果
# 5 评估3种聚类效果好坏

from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
from sklearn.datasets import make_blobs
# make系列-自己构造数据集 fetch系列大数据-从网络加载 load系列小数据集-从本地数据集
from sklearn.metrics import calinski_harabasz_score  # calinski_harabaz_score 废弃
```

案例

- 代码
- 分析

```
def dm04_kmeans():
    # 2 创建数据集 1000个样本,每个样本2个特征 4个质心 簇数据标准差[0.4, 0.2, 0.2, 0.2]
    x, y = make_blobs(n_samples=1000, n_features=2, centers=[[-1,-1], [0,0], [1,1], [2,2]],
                      cluster_std = [0.4, 0.2, 0.2, 0.2], random_state=22)
    plt.figure()
    plt.scatter(x[:, 0], x[:, 1], marker='o')
    plt.show()

    # 3 使用k-means进行聚类, 并使用CH方法评估
    # 3-1 n_clusters=2
    y_pred = KMeans(n_clusters=2, random_state=22, init='k-means++').fit_predict(x)
    plt.scatter(x[:, 0], x[:, 1], c=y_pred)
    plt.show()
    print('1-->', calinski_harabasz_score(x, y_pred))

    # 3-2 n_clusters=3
    y_pred = KMeans(n_clusters=3, random_state=22).fit_predict(x)
    plt.scatter(x[:, 0], x[:, 1], c=y_pred)
    plt.show()
    print('2-->', calinski_harabasz_score(x, y_pred))

    # 3-3 n_clusters=4
    y_pred = KMeans(n_clusters=4, random_state=22).fit_predict(x)
    plt.scatter(x[:, 0], x[:, 1], c=y_pred)
    plt.show()
    # 4 模型评估
    print('3-->', calinski_harabasz_score(x, y_pred))
```

```
# CH值越大聚类效果越好，从数据来看聚成4类效果最好
# 3125.9400435605726     # n_clusters = 2
# 2964.3137148168053     # n_clusters = 3
# 5813.930875534541      # n_clusters = 4
```

总结

1 机器学习建模的一般步骤

- 获取数据：搜集与完成机器学习任务相关的数据集
- 数据基本处理：数据集中异常值,缺失值的处理等
- 特征工程：对数据特征进行提取、转成向量，让模型达到最好的效果
- 机器学习(模型训练)：选择合适的算法对模型进行训练

  根据不同的任务来选中不同的算法；有监督学习,无监督学习,半监督学习,强化学习
- 模型评估：评估效果好上线服务,评估效果不好则重复上述步骤

2 机器学习三大经典任务: 回归、分类、聚类问题

- 每一类问题都会用不同的算法来求解
- 算法没有好坏：要根据应用场景（数据要求、性能要求等）来选择不同的算法
- 每一类算法都有不同的评估指标来评测模型的效果如何。

练习

下面关于机器学习建模的流程每个步骤表示如下：
获取数据 (3)、数据基本处理 (1)、 特征工程 (6)、 机器学习(模型训练) (5)、 模型评估 (4)、在线服务模型预测 (2)。
下列流程正确的是：

A) 1 -> 2 -> 3 -> 4 -> 5 -> 6

B) 3 -> 1 -> 6 -> 5 -> 4 -> 2

C) 3 -> 1 -> 6 -> 2 -> 5 -> 4

D) 1 -> 3 -> 6 -> 5 -> 4 -> 2

解析：最后是在线服务模型预测 正确答案 B

目录

Contents

学习目标

Learning Objectives

1. 知道特征工程是什么?
2. 理解特征提取的作用
3. 理解特征预处理的作用
4. 了解特征降维、特征选择、特征组合

特征工程概念入门

- 特征(feature)

- 特征工程

(Feature Engineering)

- 利用专业背景知识和技巧处理数据，让机器学习算法效果最好。这个过程就是特征工程

Coming up with features is difficult, time-consuming, requires expert knowledge.

“Applied machine learning” is basically feature engineering.”

数据和特征决定了机器学习的上限，而模型和算法只是逼近这个上限而已

特征工程概念入门 - 涉及内容

1. **特征提取**
   feature extraction
   原始数据中提取与任务相关的特征，构成特征向量

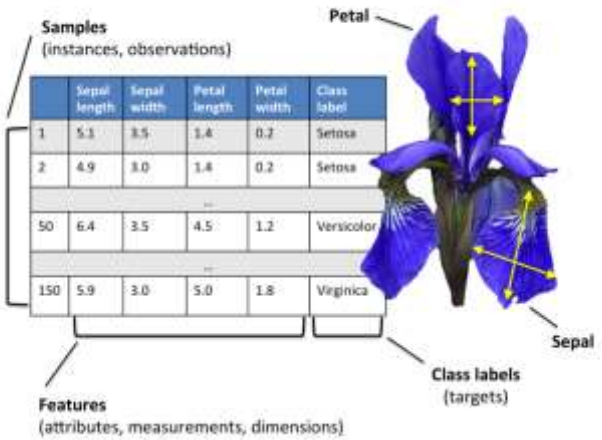
Iris Versicolor　Iris Setosa　Iris Virginica

2. **特征预处理**
   feature preprocessing

3. **特征降维**
   Feature decomposition

4. **特征选择**
   feature selection

5. **特征组合**
   feature crosses



	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

特征工程概念入门 - 涉及内容

1. **特征提取 feature extraction**
2. **特征预处理 feature preprocessing**
3. **特征降维 Feature decomposition**
4. **特征选择 feature selection**
5. **特征组合 feature crosses**

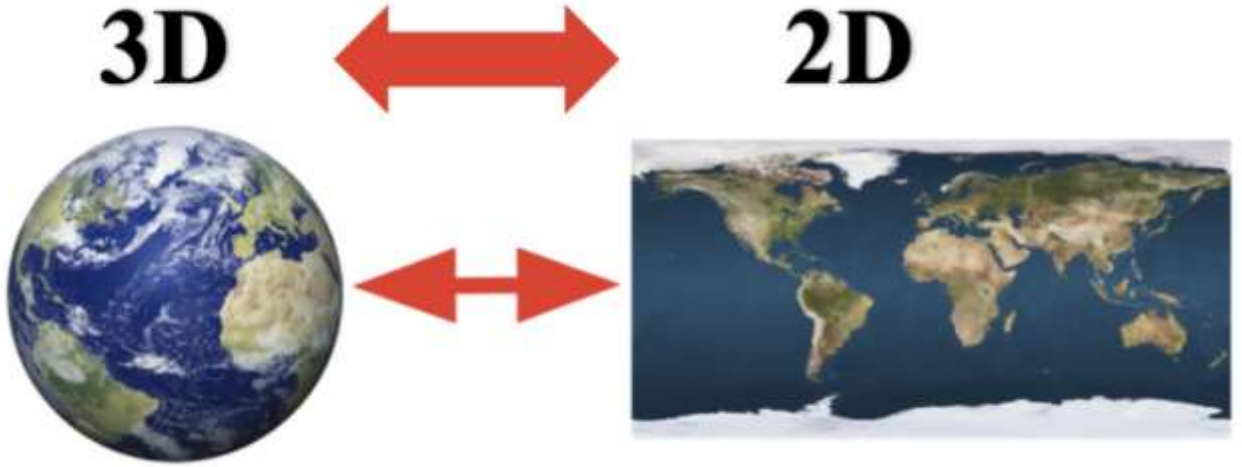
特征对模型产生影响；因量纲问题，有些特征对模型影响大、有些影响小

特征1	特征2	特征3	特征4
90	2	10	40
60	4	15	45
75	3	13	46

特征1	特征2	特征3	特征4
1.	0.	0.	0.
0.	1.	1.	0.83
0.5	0.5	0.6	1.

特征工程概念入门 - 涉及内容

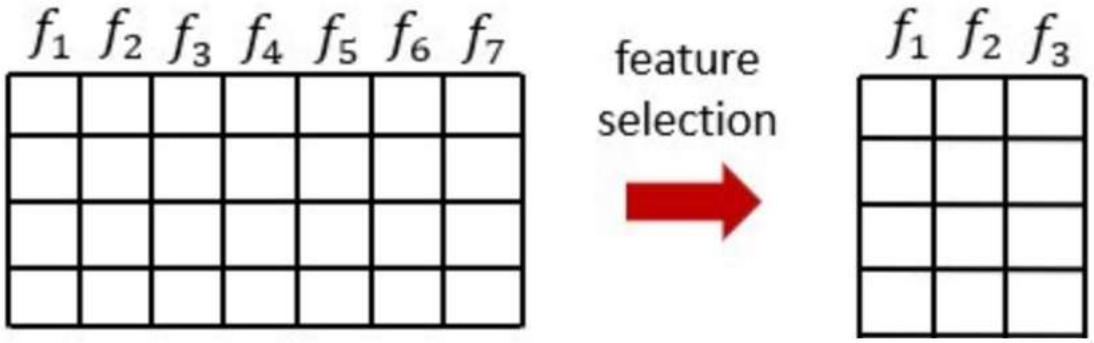
1. 特征提取 feature extraction
2. 特征预处理 feature preprocessing
3. 特征降维 Feature decomposition
4. 特征选择 feature selection
5. 特征组合 feature crosses



将原始数据的维度降低，叫做特征降维，一般会对原始数据产生影响

特征工程概念入门 - 涉及内容

1. 特征提取 feature extraction
2. 特征预处理 feature preprocessing
3. 特征降维 Feature decomposition
4. 特征选择 feature selection
5. 特征组合 feature crosses



原始数据特征很多，与任务相关是其中一个特征集合子集，不会改变原数据

特征工程概念入门 - 涉及内容

1. **特征提取 feature extraction**
2. **特征预处理 feature preprocessing**
3. **特征降维 Feature decomposition**
4. **特征选择 feature selection**
5. **特征组合 feature crosses**

`[A X B]`：将两个特征的值**相乘**形成的特征组合。

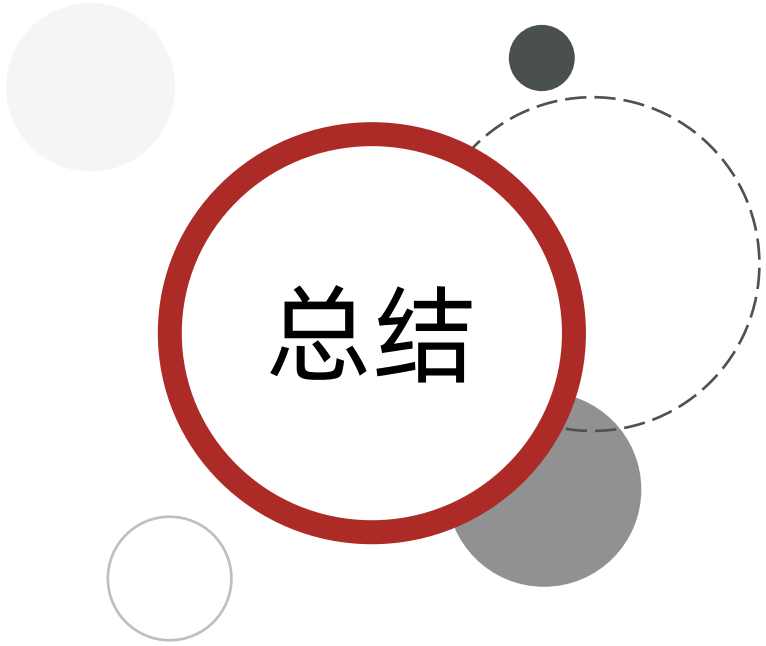
`[A x B x C x D x E]`：将五个特征的值**相乘**形成的特征组合。

`[A x A]`：对单个特征的值**求平方**形成的特征组合。

把多个的特征合并成一个特征。利用乘法或加法来完成

特征工程概念入门 - 涉及内容

1. **特征提取 feature extraction** — 从原始数据中提取与任务相关的特征
2. **特征预处理 feature preprocessing** — 特征对模型产生影响；因量纲问题，有些特征对模型影响大、有些影响小
3. **特征降维 Feature decomposition** — 将原始数据的维度降低，叫做特征降维
4. **特征选择 feature selection** — 原始数据特征很多，但是对模型训练相关是其中一个特征集合子集。
5. **特征组合 feature crosses** — 把多个的特征合并成一个特征。一般利用乘法或加法来完成



1 特征工程 Feature Engineering

- 特征**Feature**：对任务有用的属性信息
- 特征工程：利用专业背景知识和技巧处理数据，让模型效果更好

2 特征工程的内容

- 特征提取 Feature extraction ： 特征向量
- 特征预处理 Feature preprocessing：不同特征对模型影响一致性
- 特征降维 Feature decomposition：保证数据的主要信息要保留下来
- 特征选择 Feature selection ： 从特征中选择出一些重要特征训练模型
- 特征组合 Feature crosses：把多个特征合并组合成一个特征

练习

- **有关特征工程说法正确的？（多选）**

A）在机器学习整个工程项目中，一般情况下特征工程往往是耗时、耗精力最多工作

B）特征工程就是处理数据，不重要

C）特征提取一般是做数据的标准化、归一化等工作

D）特征降维会修改原始数据，特征选择不会修改原始数据

E）特征工程的好坏会影响模型的上限，是一项专项的工作；开发者需要掌握

解析：　特征工程是很重要的B描述错误；

特征提取从无到有的做行列向量数据，C描述错误。

特征预处理做数据标准化、归一化前置处理工作。

答案 (ADE)

目录

Contents

学习目标

Learning Objectives

1. 知道拟合是什么?
2. 理解过拟合、欠拟合是什么?
3. 知道过拟合、欠拟合出现的原因
4. 理解泛化是什么?

拟合

- 拟合 fitting

用在机器学习领域，用来表示模型对样本点的拟合情况

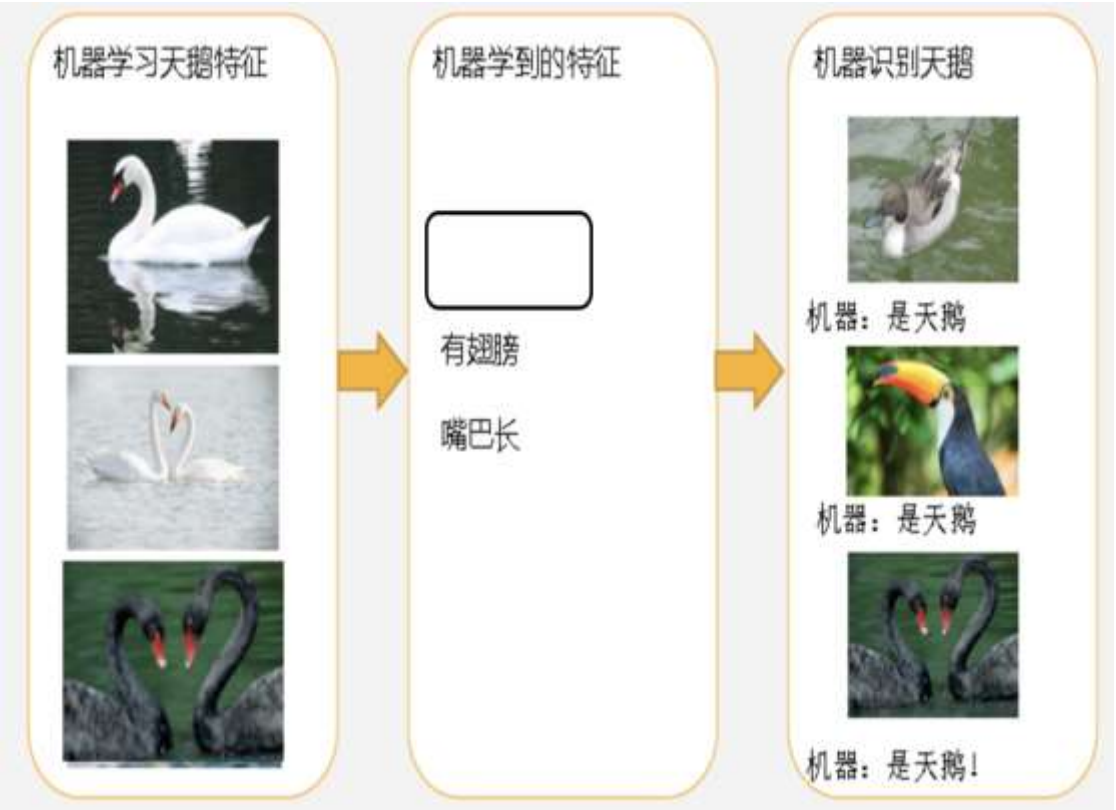
- 欠拟合 under-fitting

模型在训练集上表现很差、在测试集表现也很差

- 过拟合 over-fitting

模型在训练集上表现很好、在测试集表现很差

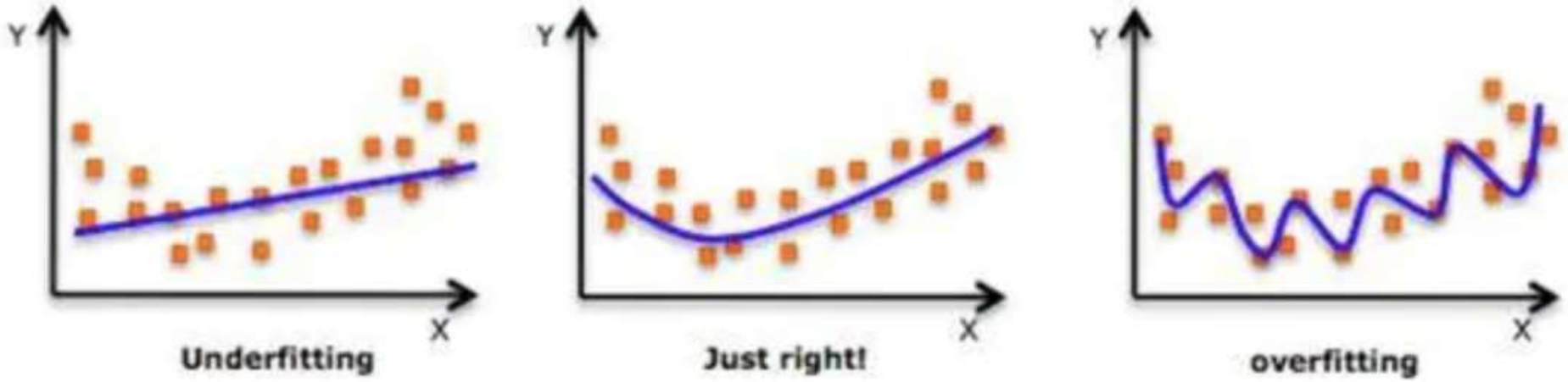
拟合-欠拟合/过拟合



机器学习到的天鹅特征太少了，
导致区分标准太粗糙，不能准确识别出天鹅

机器已基本能区别天鹅和其他动物了
很不巧已有的天鹅图片全是白天鹅的。黑色的天鹅不能识别

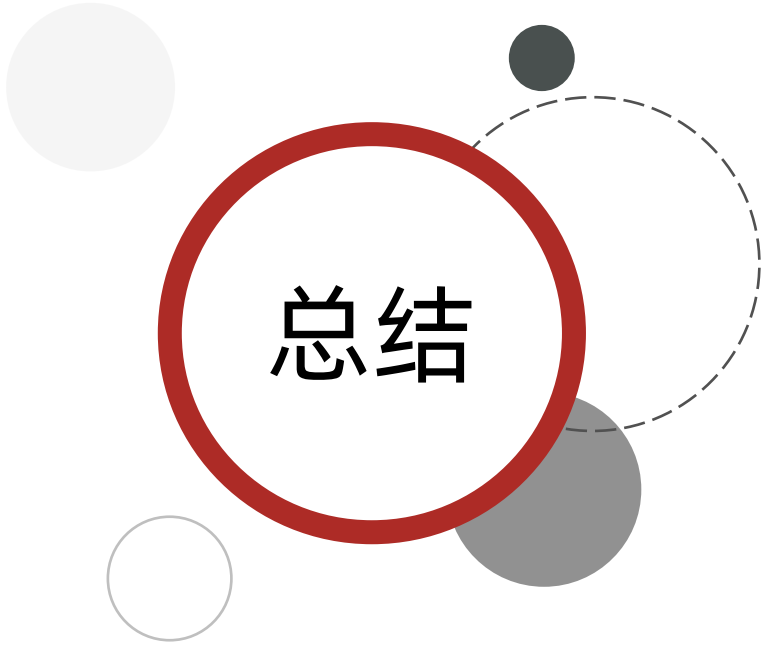
模型表现效果 - 欠拟合过拟合 - 从样本分布角度看



- 欠拟合产生的原因：模型过于简单
- 过拟合产的原因：模型太过于复杂、数据不纯、训练数据太少

- 泛化 Generalization ： 模型在新数据集（非训练数据）上的表现好坏的能力。
- 奥卡姆剃刀原则：给定两个具有相同泛化误差的模型，较简单的模型比较复杂的模型更可取

(如无必要，勿增实体/简单有效原理)



1 过拟合欠拟合？

- 拟合：用来表示模型对样本分布点的模拟情况
- 模型在训练集上表现很差、在测试集表现也很差，是欠拟合
- 模型在训练集上表现很好、在测试集表现很差，是过拟合

2 过拟合欠拟合产生的原因

- 欠拟合产生的原因：模型过于简单
- 过拟合产生的原因：模型太过于复杂、数据不纯、训练数据太少

3 泛化概念

- 泛化 Generalization ： 具体的、个别的扩大为一般的能力
- 奥卡姆剃刀原则：给定两个具有相同泛化误差的模型，倾向选择较简单的模型

(如无必要，勿增实体/简单有效原理)

练习

- 下列有关过拟合欠拟合说法正确的？（多选）

A）欠拟合：模型学习到的特征过少，无法准确的预测未知样本

B）过拟合：模型学习到的特征过多，导致模型只能在训练样本上得到较好的预测结果，而在未知样本上的效果不好

C）欠拟合可以通过增加特征来解决

D）过拟合可以通过正则化、异常值检测、特征降维等方法来解决

解析：A欠拟合出现的原因 B过拟合出现的原因 C增加模型的复杂度 D降低模型复杂度。

答案：ABCD

目录

Contents

基于Python的 scikit-learn 库

1. 简单高效的数据挖掘和数据分析工具
2. 可供大家使用，可在各种环境中重复使用
3. 建立在NumPy，SciPy和matplotlib上
4. 开源，可商业使用-获取BSD许可证

安装方法：

pip install scikit-learn

官网：

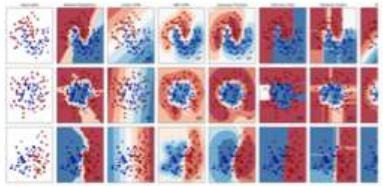
https://scikit-learn.org/stable/

基于Python的 scikit-learn 库

Classification

Identifying which category an object belongs to.

Applications: Spam detection, image recognition.
Algorithms: SVM, nearest neighbors, random forest, and more...

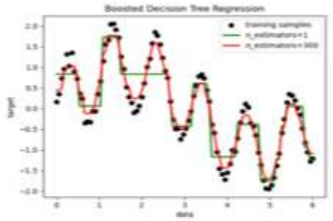


Examples

Regression

Predicting a continuous-valued attribute associated with an object.

Applications: Drug response, Stock prices.
Algorithms: SVR, nearest neighbors, random forest, and more...



Examples

Clustering

Automatic grouping of similar objects into sets.

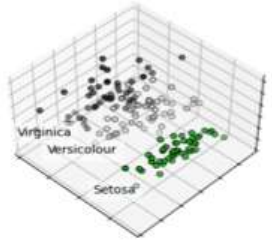
Applications: Customer segmentation, Grouping experiment outcomes
Algorithms: k-Means, spectral clustering, mean-shift, and more...

Examples

Dimensionality reduction

Reducing the number of random variables to consider.

Applications: Visualization, Increased efficiency
Algorithms: k-Means, feature selection, non-negative matrix factorization, and more...

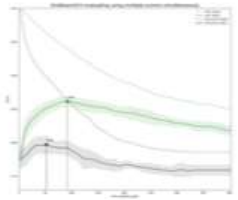


Examples

Model selection

Comparing, validating and choosing parameters and models.

Applications: Improved accuracy via parameter tuning
Algorithms: grid search, cross validation, metrics, and more...

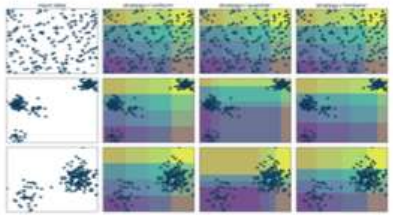


Examples

Preprocessing

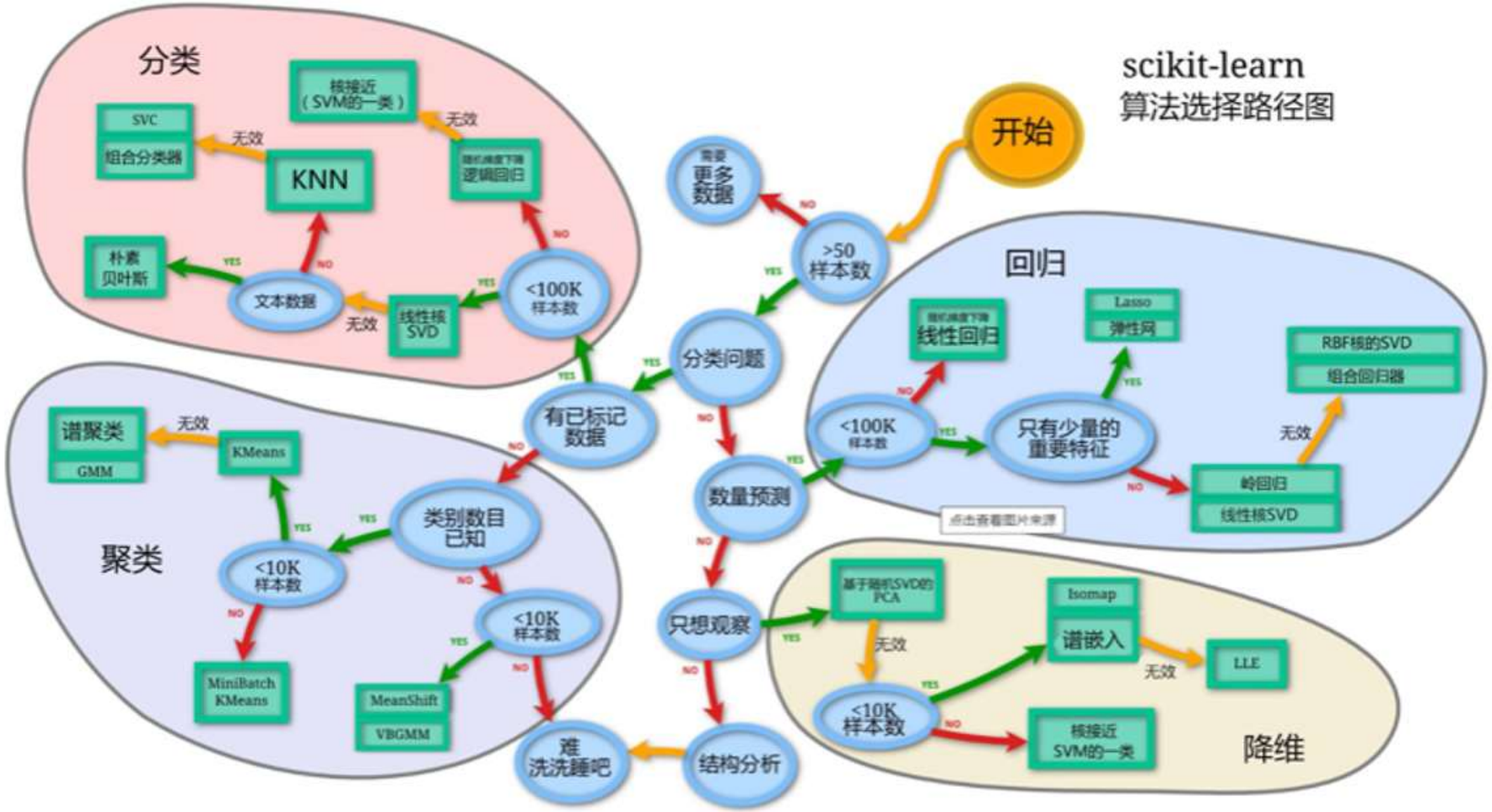
Feature extraction and normalization.

Applications: Transforming input data such as text for use with machine learning algorithms.
Algorithms: preprocessing, feature extraction, and more...



Examples

基于Python的 scikit-learn 库




黑马程序员
www.itheima.com
传智教育旗下高端IT教育品牌