AI007 《人工智能》Week04

机器学习

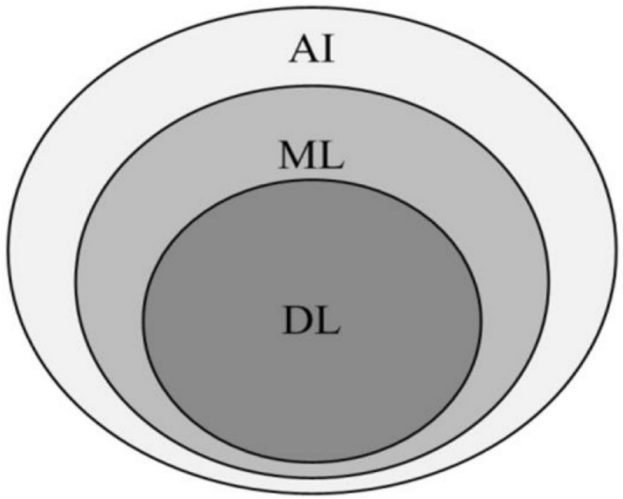
Machine Learning

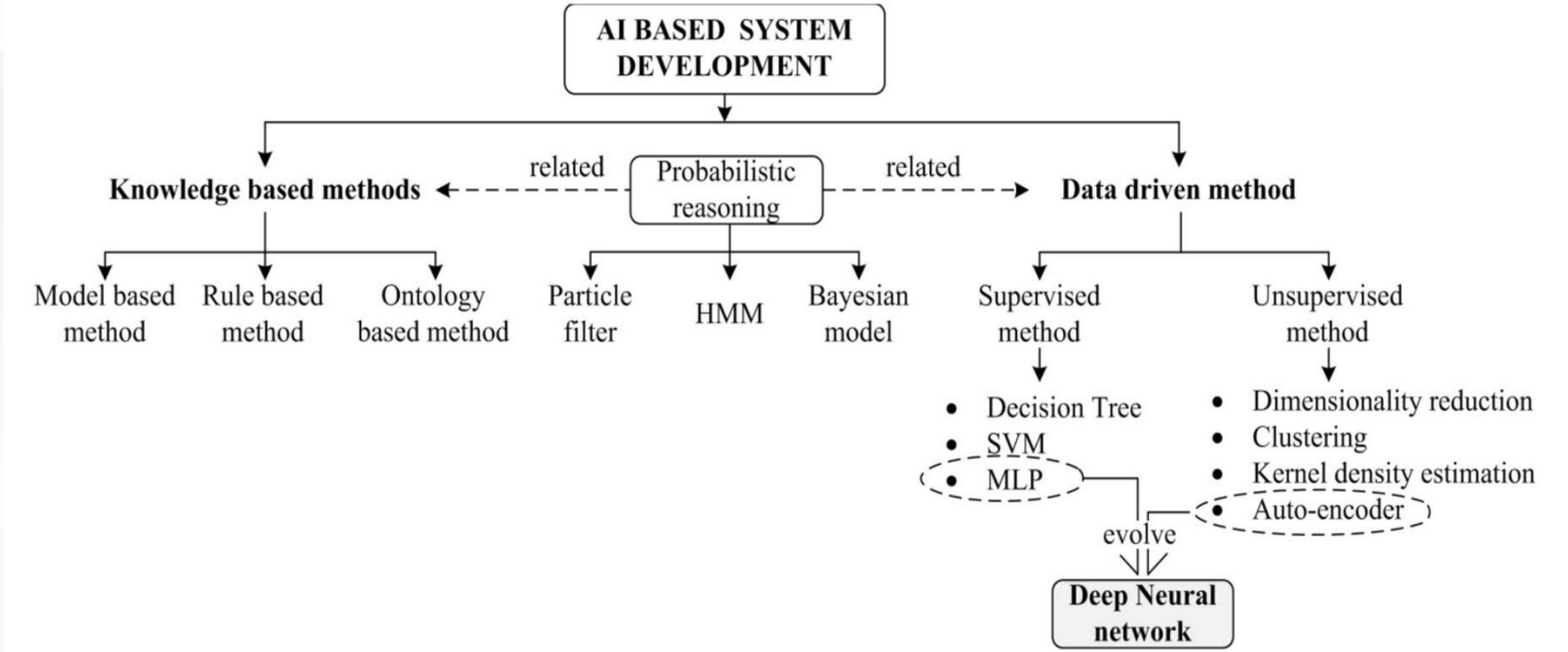
许岩岩

上海交通大学 · 人工智能研究院

2021年10月09日

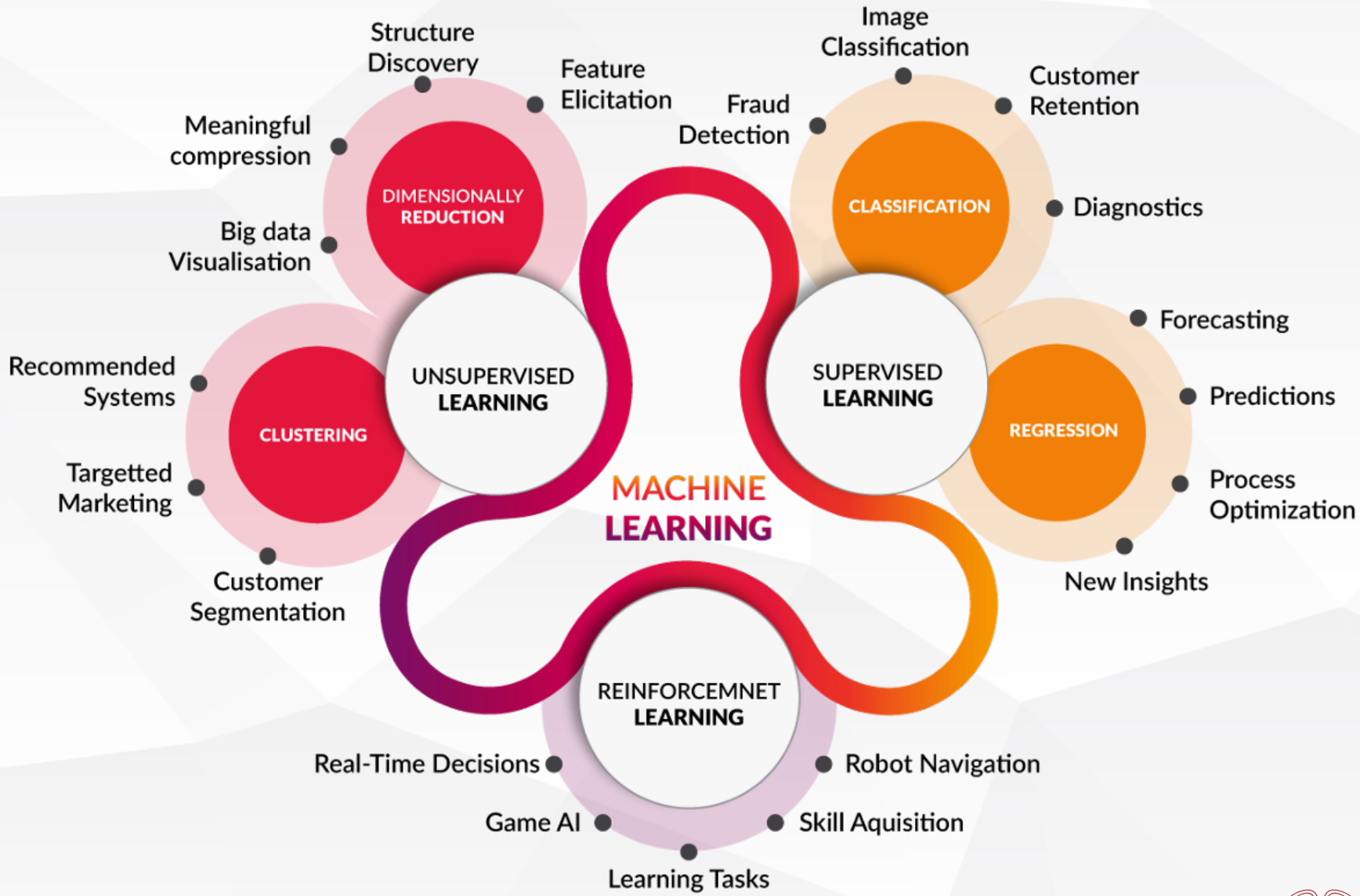
人工智能 vs 机器学习





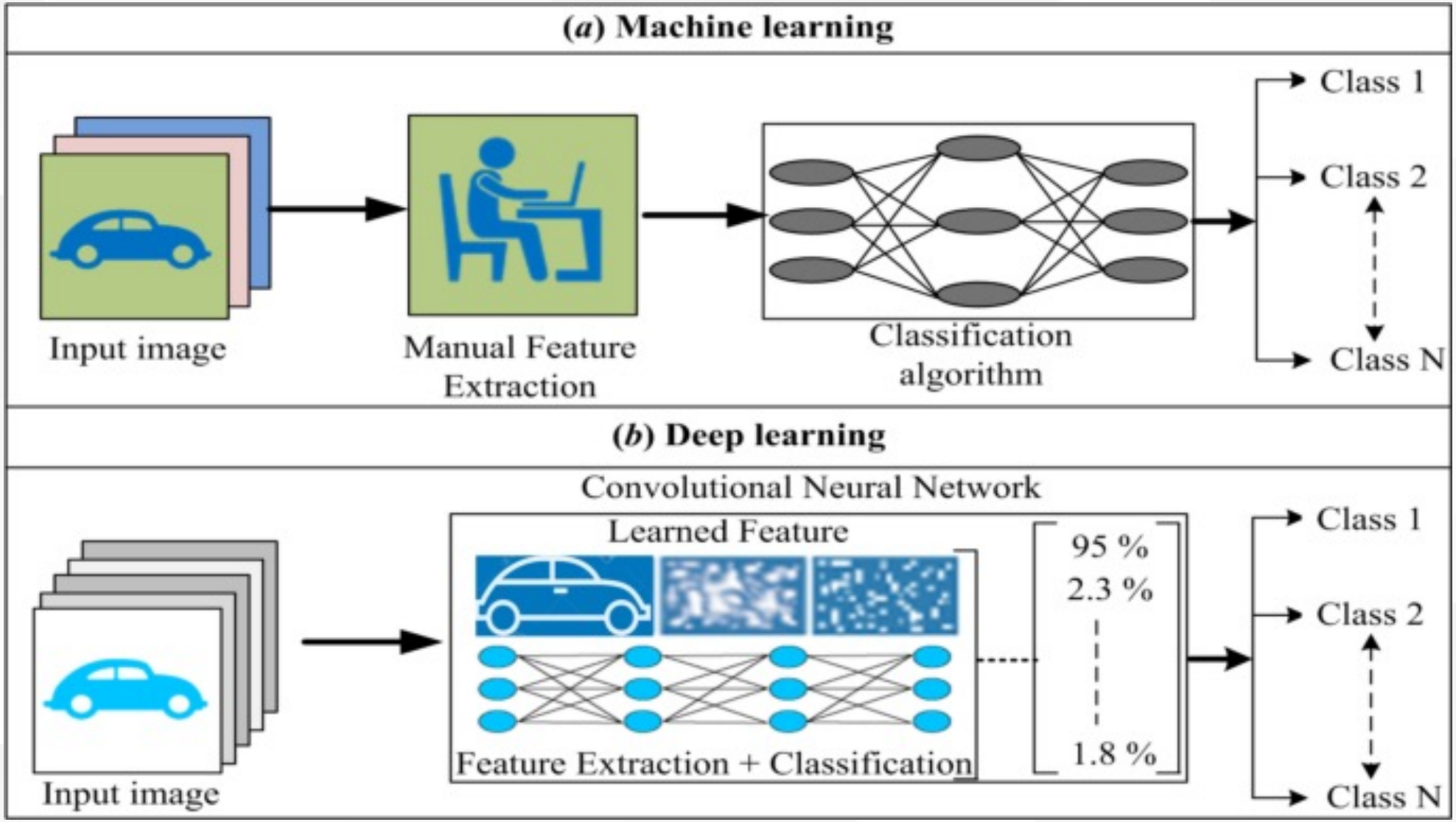
Source: Bhatt, Chandradeep, et al. "The state of the art of deep learning models in medical science and their challenges." *Multimedia Systems* 27.4 (2021): 599-613.

机器学习分类

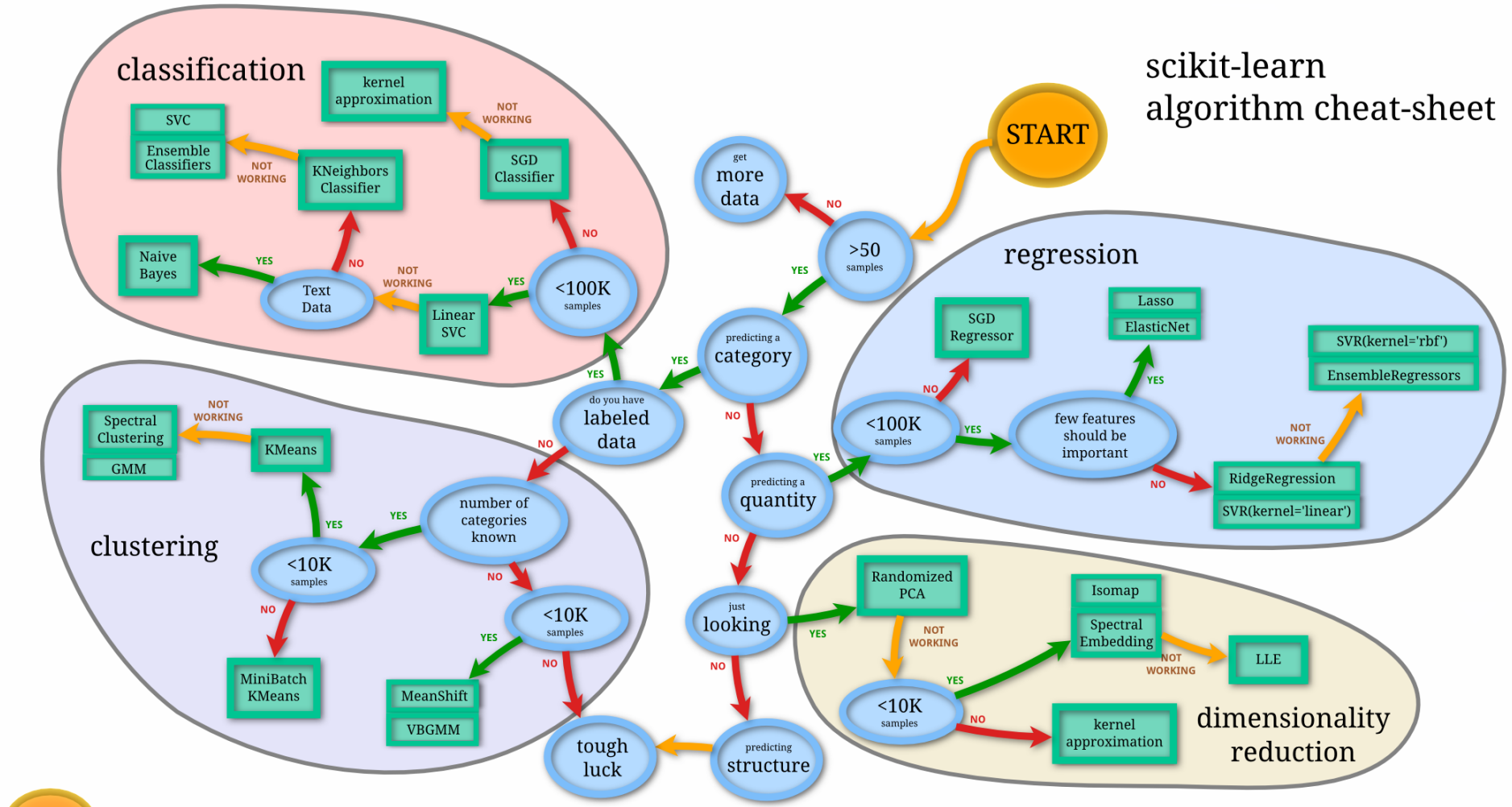


Source: https://towardsdatascience.com/coding-deep-learning-for-beginners-types-of-machine-learning-b9e651e1ed9d

经典机器学习 vs 深度学习

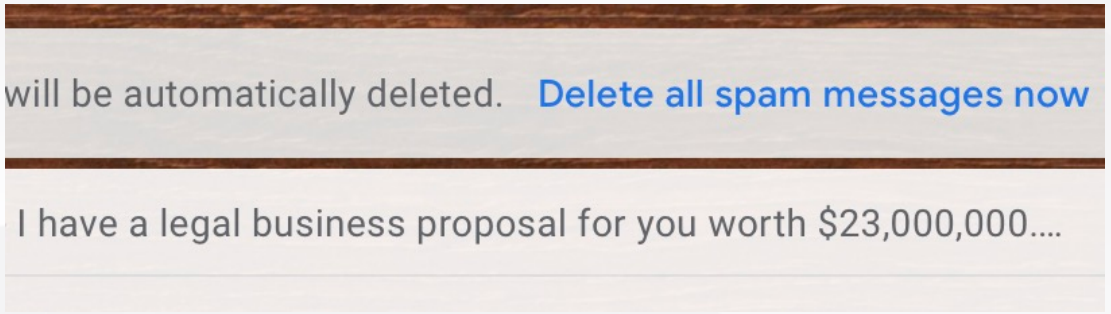


Source: Bhatt, Chandradeep, et al. "The state of the art of deep learning models in medical science and their challenges." *Multimedia Systems* 27.4 (2021): 599-613.

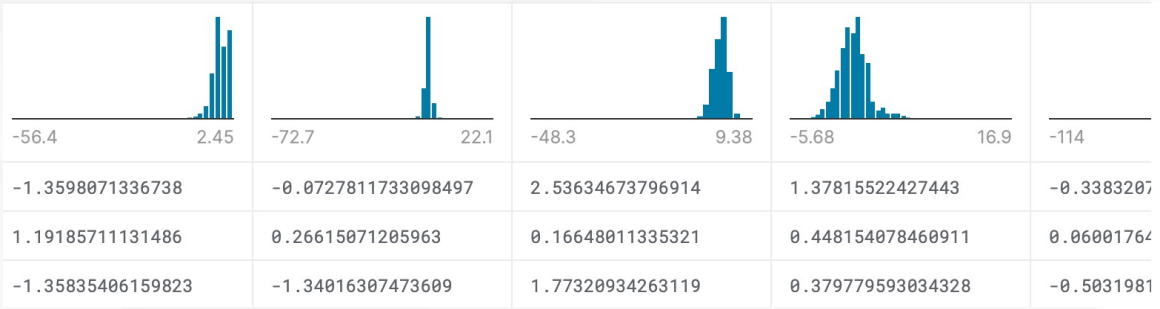


Source: https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

分类问题

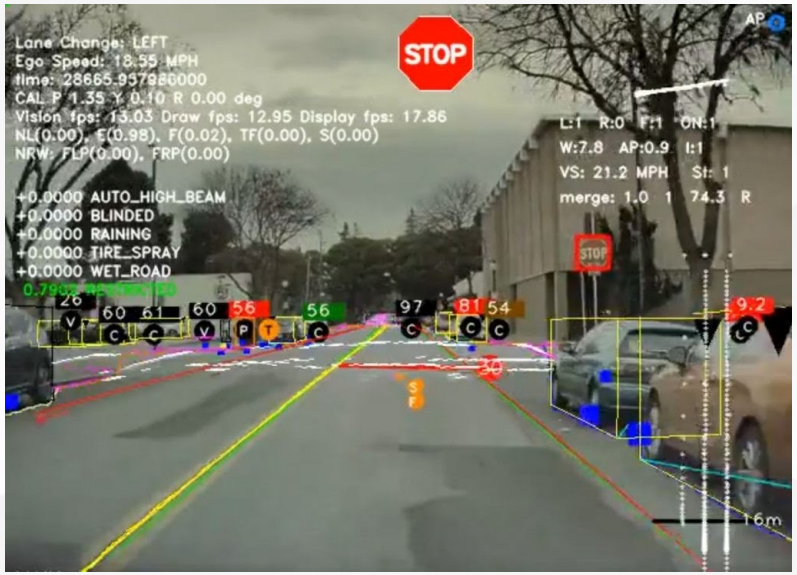


自然语言中的分类问题：spam、not spam



-56.4 2.45	-72.7 22.1	-48.3 9.38	-5.68 16.9	-114
-1.3598071336738	-0.0727811733098497	2.53634673796914	1.37815522427443	-0.3383207
1.19185711131486	0.26615071205963	0.16648011335321	0.448154078460911	0.06001764
-1.35835406159823	-1.34016307473609	1.77320934263119	0.379779593034328	-0.5031981

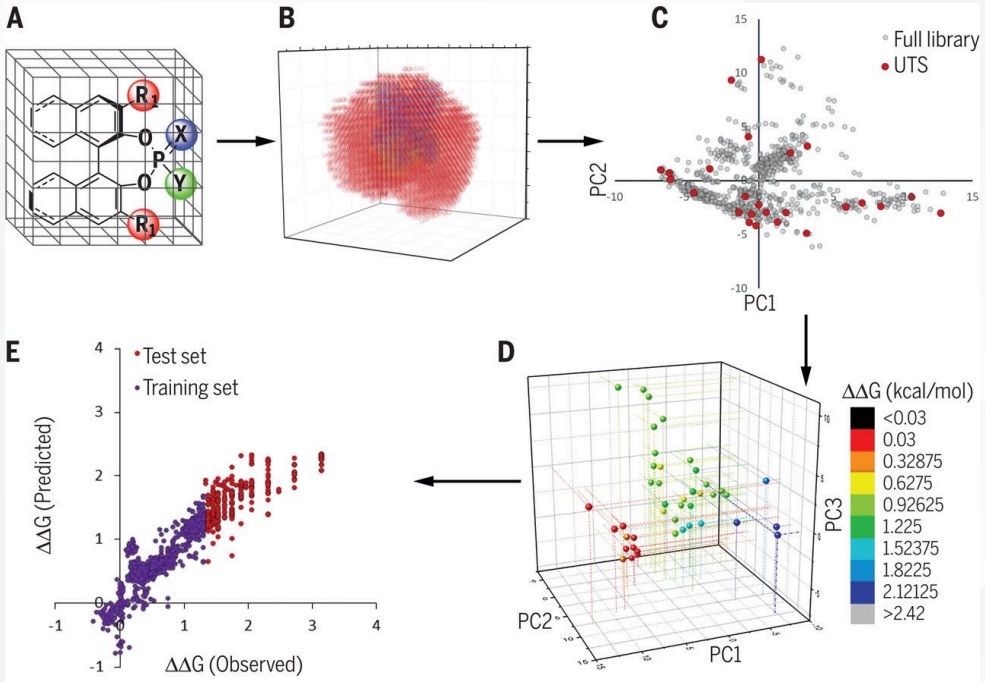
表格数据中的分类问题：欺诈、非欺诈



图像中的分类问题：车、人、道路、天空

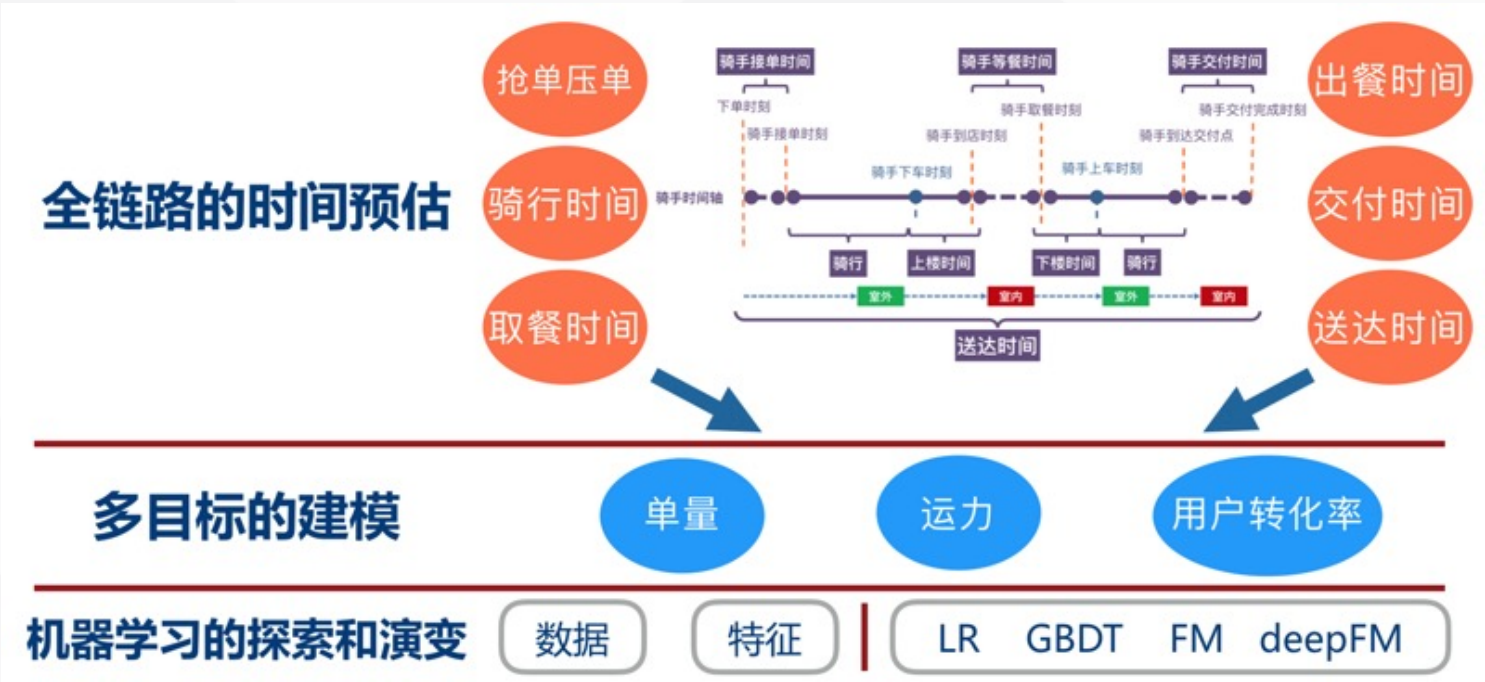
视频中的分类问题：in、out

回归问题



催化剂效率预测

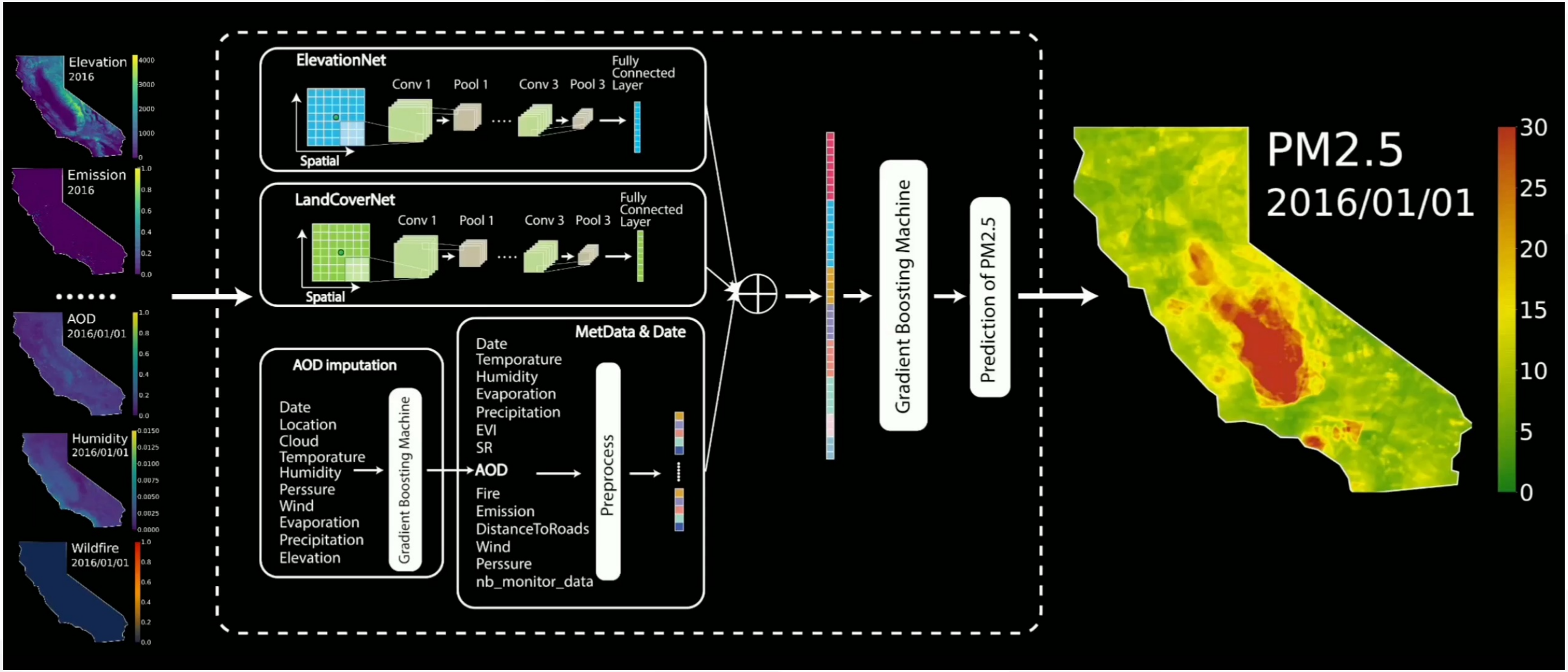
Source: Zahrt, Andrew F., et al. "Prediction of higher-selectivity catalysts by computer-driven workflow and machine learning." *Science* 363.6424 (2019).



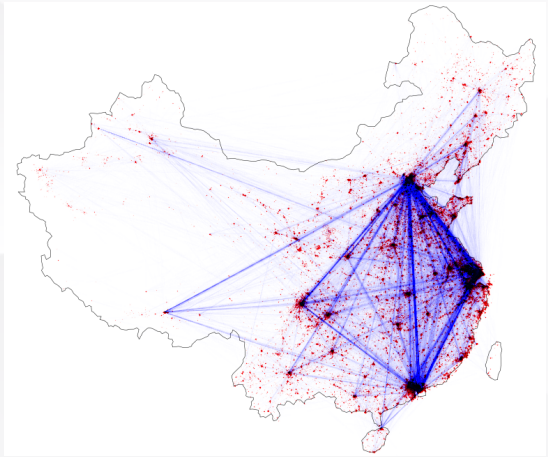
外卖送达时间预测

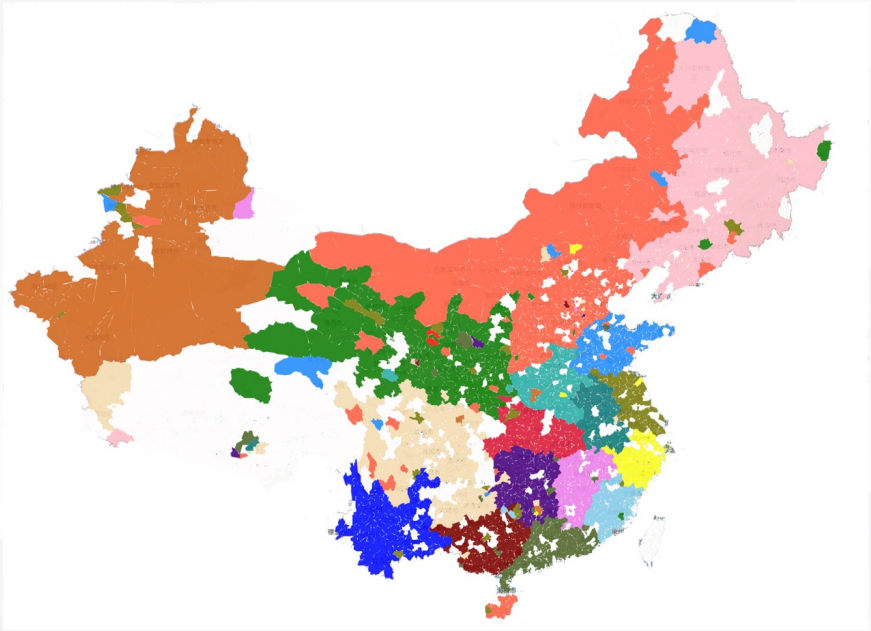
Source: https://tech.meituan.com/2019/02/21/meituan-delivery-eta-estimation-in-the-practice-of-deep-learning.html

回归问题

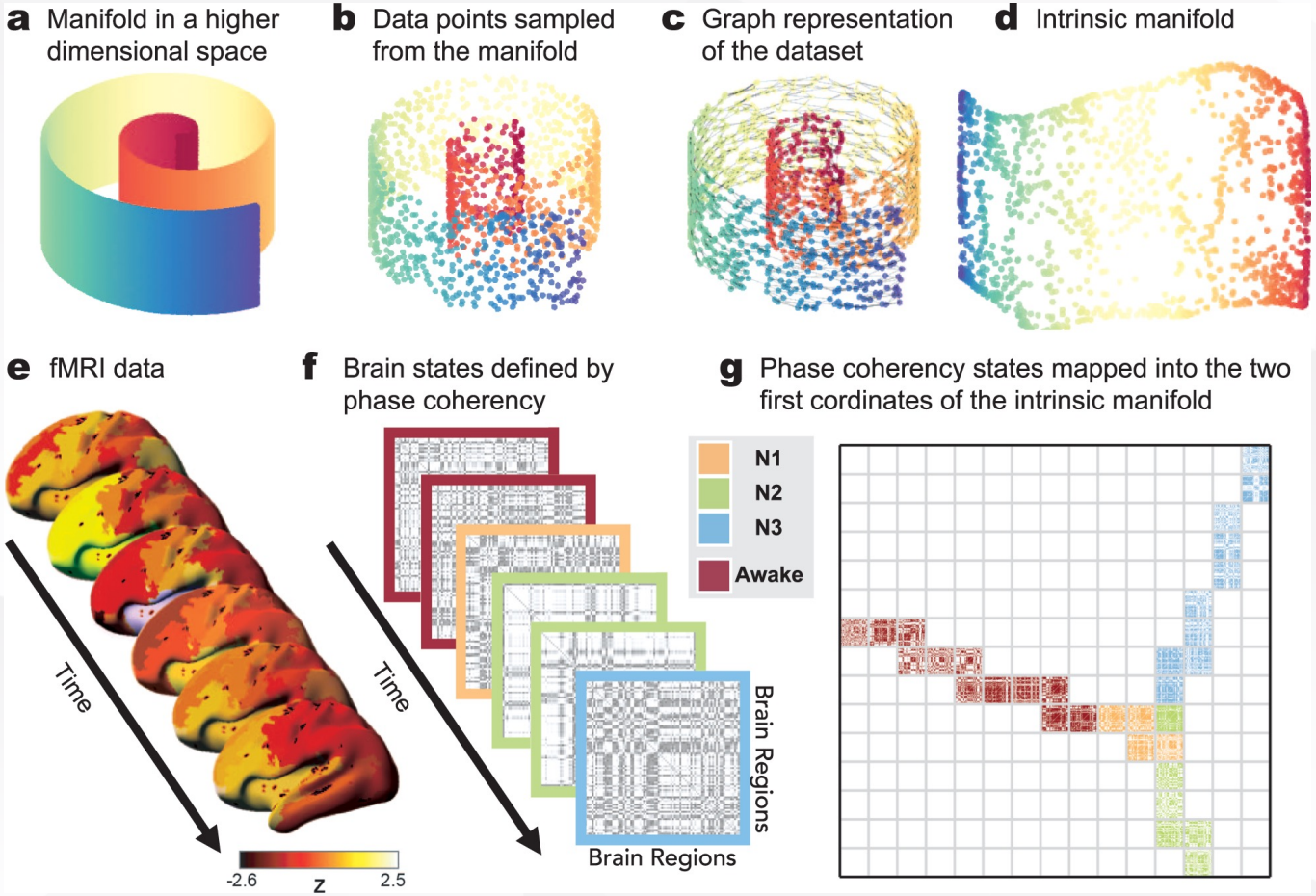


PM2.5浓度空间推测





区域（县）投资网络社区检测

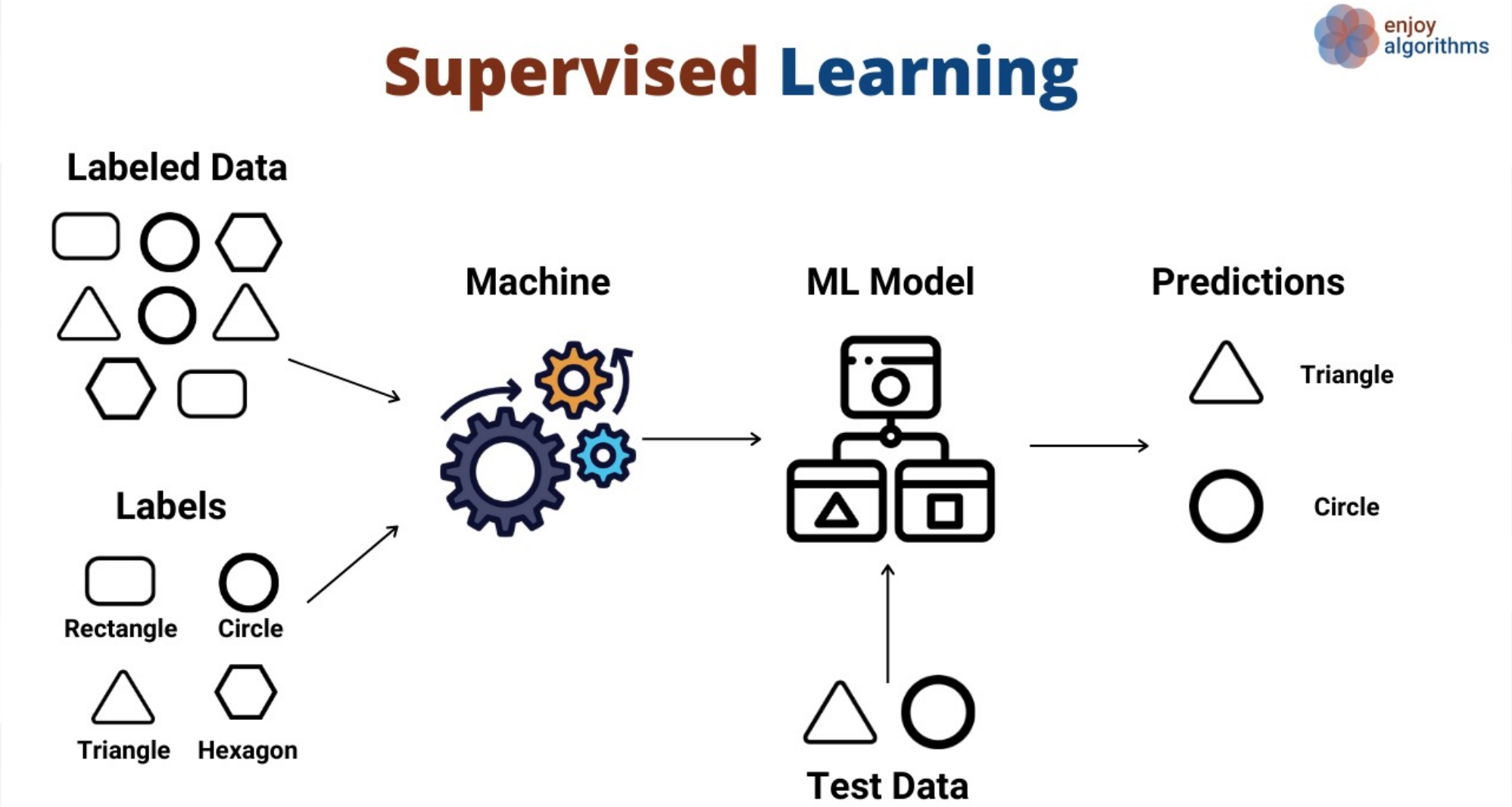


大脑状态数据降维

Source: Rue-Queralt, Joan, et al. "Decoding brain states on the intrinsic manifold of human brain dynamics across wakefulness and sleep." *bioRxiv* (2021).

提纲

1. 监督学习
2. 无监督学习
3. 弱监督学习
4. 知识点回顾


Supervised Learning
enjoy algorithms
Labeled Data
Labels
Rectangle
Circle
Triangle
Hexagon
Machine
ML Model
Predictions
Triangle
Circle
Test Data

K-近邻算法

K-近邻算法（K-nearest neighbors, KNN）

给定训练集，对于待分类的样本点，计算待预测样本和训练集中所有数据点的距离，将距离从小到大排序，选择前K个样本中数据最大的类别，作为预测样本类别

优点：容易实现，支持多分类，不需要模型训练

缺点：对参数选择敏感，计算量大

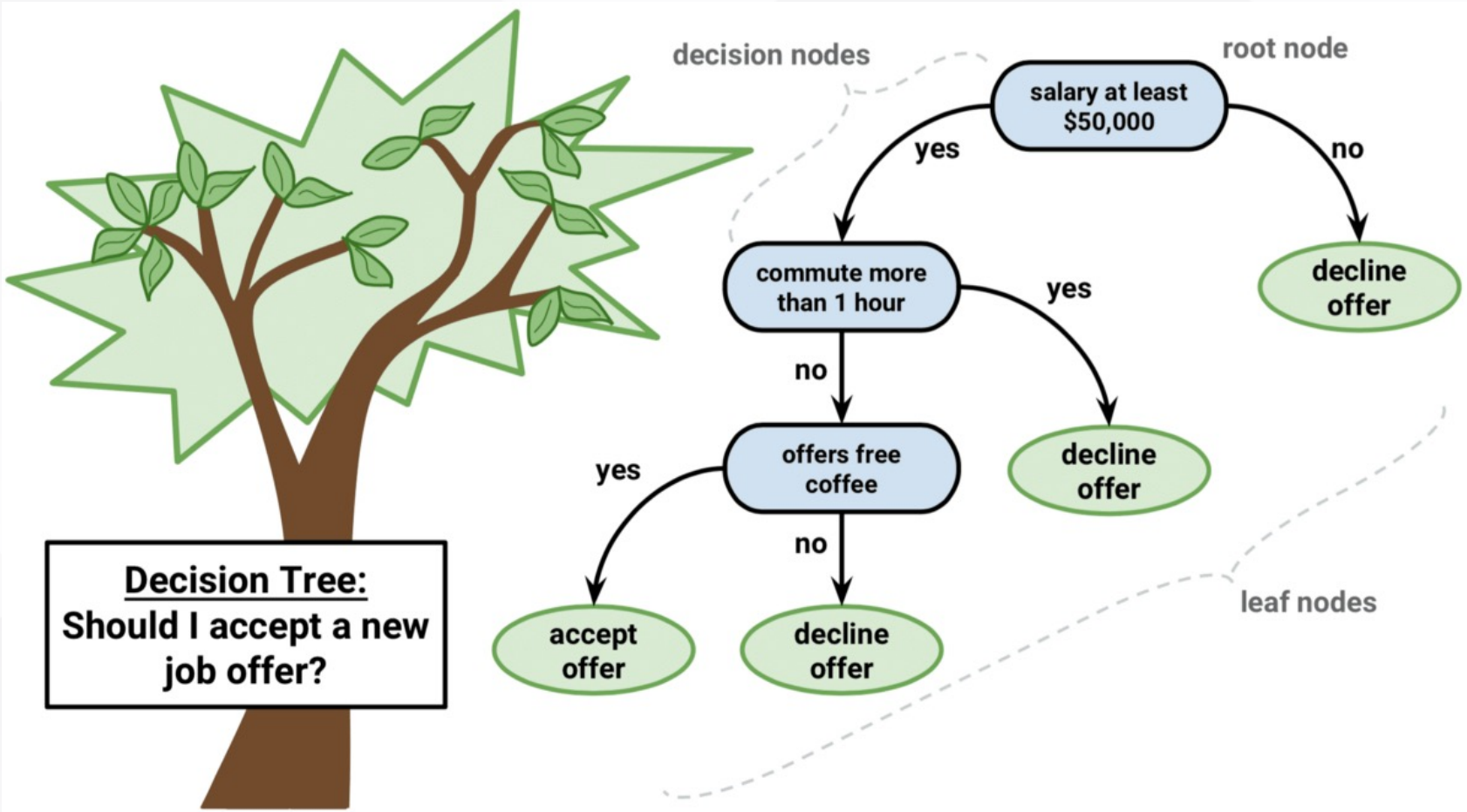
决策树（Decision Tree）

人类决策的“分而治之（divide-and-conquer）”处理机制；

包含一个跟节点、若干内部节点、若干叶节点。

可适用于分类、回归任务。

可以处理连续型、离散型变量。



决策树

目的：产生一颗泛化能力强的决策树；

学习关键：如何选择最优划分属性。

著名决策树算法：ID3算法、C4.5算法和CART算法

学习过程：

- **特征选择：**选取对训练数据有分类能力的特征
- **决策树的生成：**不断进行特征选择并划分训练数据的过程，直到叶结点结束
- **决策树的剪枝：**对生成的决策树进行简化，即减少叶结点的数量，避免出现过拟合现象，即生成的决策树往往会对训练数据分类准确，而对测试数据分类效果不好

不同算法特征选择的准则不同：ID3: 最大信息增益准则；ID4.5: 最大信息增益比；CART: 最大基尼指数

为了说明信息增益是什么，首先要给出经验熵和经验条件熵的定义。在信息论与概率统计中，熵(entropy)是表示随机变量不确定性的度量。假设数据集为D，$|D|$表示数据集中的样本个数，D中的样本共有K个类别，则数据集D的**经验熵** $H(D)$的计算公式为：

$$H(D) = -\sum_{k=1}^{K} \frac{|C_k|}{|D|} \log_2 \frac{|C_k|}{|D|}$$

其中C_k是数据集D中属于第k类的样本子集，$|C_k|$则表示该子集的样本个数。

假设样本的某一个特征为A，且特征A有n个不同的取值$a_1, a_2, \ldots, a_n$，根据特征A将数据集D划分为n个子集$D_1, D_2, \ldots, D_n$，$|D_i|$为子集D_i中的样本个数，子集D_i中属于C_k类的样本集合为D_{ik}，$|D_{ik}|$为D_{ik}中的样本个数，则特征A对于数据集D的**经验条件熵** 的计算公式为：

$$H(D|A) = \sum_{i=1}^{n} \frac{|D_i|}{|D|} H(D_i)$$

$$= -\sum_{i=1}^{n} \frac{|D_i|}{|D|} \sum_{k=1}^{K} \frac{|D_{ik}|}{|D_i|} \log_2 \frac{|D_{ik}|}{|D_i|}$$

有了经验熵和经验条件熵之后便可以得到信息增益(information gain)了，信息增益$g(D|A)$即为二者之差：

$$g(D|A) = H(D) - H(D|A)$$

ID3算法选择特征的过程就是计算出所有特征对数据集D的信息增益，然后选择信息增益最大的特征。

ID3算法采用的最大信息增益准则存在偏向于选择取值较多的特征的问题，信息增益比则是对这个问题进行了改进，信息增益比记为$g_R(D|A)$,其计算公式如下：

$$g_R(D|A) = \frac{g(D|A)}{H_A(D)}$$

其中，$H_A(D)$为数据集D关于特征A的取值的熵，其计算公式如下：

$$H_A(D) = -\sum_{i=1}^{n} \frac{|D_i|}{|D|} \log_2 \frac{|D_i|}{|D|}$$

其中n为特征A的取值个数。

决策树的剪枝

为了提高决策树模型的泛化能力，避免过拟合，往往需要对生成的决策树进行剪枝。决策树的剪枝通常有两种方法，分别为预剪枝(pre-pruning)和后剪枝(post-pruning)。

预剪枝的核心思想是在树中结点进行扩展之前，先计算当前的划分是否能够带来模型泛化能力的提升，如果不能则不在继续生长子树。此时可能存在不同类别的样本同时存在叶结点中，按照多数投票的原则判断该结点所属类别。预剪枝对于何时停止树的生长有以下几种方法：

- **当树到达一定深度的时候，停止树的生长；**
- **当到达当前结点的样本数量小于某个阈值的时候，停止树的生长；**
- **计算每次分裂对测试集的准确度提升，当小于某个阈值的时候，不再继续扩展。**

后剪枝的核心思想是让算法生成一颗完全生长的决策树，然后自下而上计算是否剪枝。剪枝的过程是将子树删除，用一个叶结点替代，该结点的类别同样按照多数投票的原则进行判断。相比于预剪枝，后剪枝通常可以得到泛化能力更强的决策树，但时间开销会更大。

其他决策树模型

分类回归树 CART：可应用于分类、回归

随机森林 Random Forest：用随机的方式建立一个森林。RF 算法由很多决策树组成，每一棵决策树之间没有关联。建立完森林后，当有新样本进入时，每棵决策树都会分别进行判断，然后基于投票法给出分类结果。

自适应提升数Adaboost：前一个基本分类器分错的样本会得到加强，加权后的全体样本再次被用来训练下一个基本分类器。同时，在每一轮中加入一个新的弱分类器，直到达到某个预定的足够小的错误率或达到预先指定的最大迭代次数。

梯度提升决策树GBDT：是一种迭代的决策树算法，该算法由多棵决策树组成，从名字中我们可以看出来它是属于 Boosting 策略。GBDT 是被公认的泛化能力较强的算法。

XGBoost：XGBoost 是大规模并行 boosting tree 的工具，它是目前最快最好的开源 boosting tree 工具包，比常见的工具包快 10 倍以上。Xgboost 和 GBDT 两者都是 boosting 方法，除了工程实现、解决问题上的一些差异外，最大的不同就是目标函数的定义。

LightGBM：LightGBM 由微软提出，主要用于解决 GDBT 在海量数据中遇到的问题，以便其可以更好更快地用于工业实践中。

逻辑回归

逻辑回归（Logistic Regression）模型虽然名字是回归，但是实际上是一种二类(0或1)分类模型，0代表负类，1代表正类。逻辑模型的输入为样本的特征向量 ***x***，输出为 P(y=1|***x***) 和 P(y=0|***x***) ，即在输入特征向量 ***x*** 为条件下，样本标签属于某一类的概率。

首先我们来说一下线性回归，我们知道线性回归是用一个一元方程来拟合输入$\boldsymbol{x}$与输出Y之间的关系：

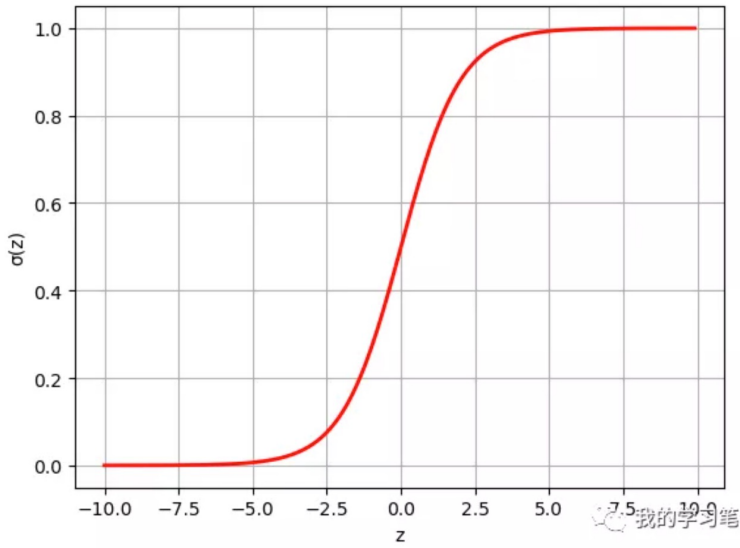
$$y = \boldsymbol{w} \cdot \boldsymbol{x} + b$$

那我们能不能直接将$P(y=1|\boldsymbol{x})$看作Y带入上式来构成逻辑回归模型呢？即：

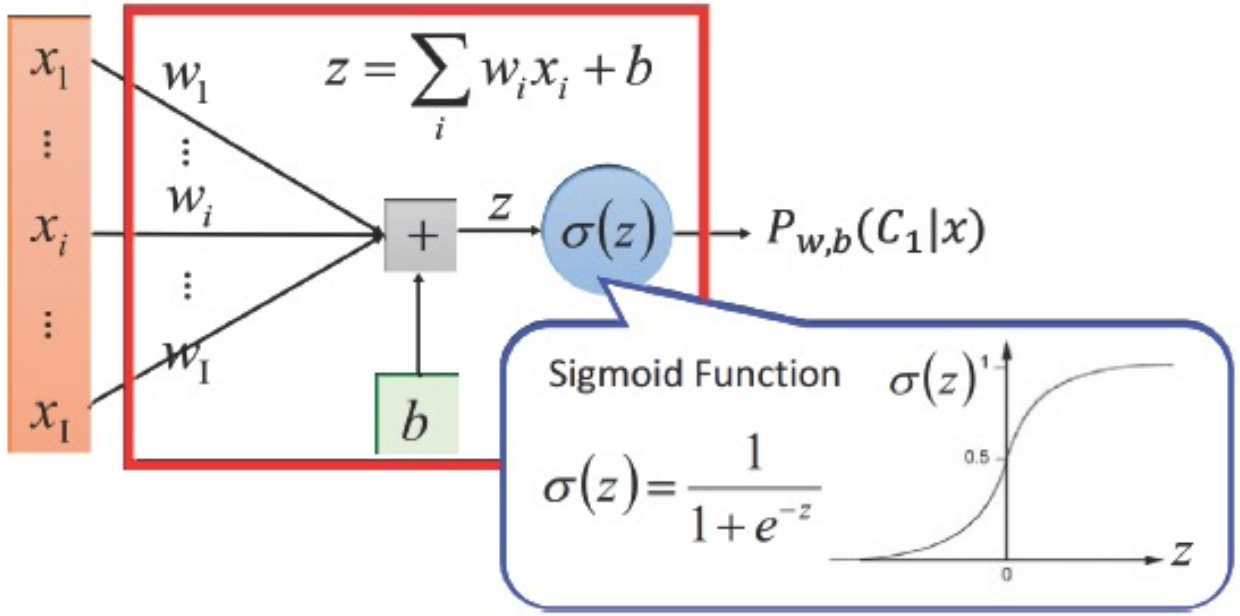
$$P(y=1|\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{x} + b$$

这样显然不合理，因为等式的左边为概率，取值应该为$0 \leq P(y=1|\boldsymbol{x}) \leq 1$，而等式右边的取值范围为$[-\infty, +\infty]$。所以应该找一个函数，将等式右边的取值范围从$[-\infty, +\infty]$映射到$[0, 1]$上，这个函数就是sigmoid函数，其数学表达式如下，其函数图像见图1：

$$\sigma(\boldsymbol{z}) = \frac{1}{1+\exp(-z)} = \frac{\exp(z)}{1+\exp(z)}$$



Sigmoid激活函数



损失函数：交叉熵损失函数

参数求解：梯度下降法、牛顿法

$$L(\boldsymbol{w}) = \sum_{i=1}^{N}[-y_i \ln(\sigma(\boldsymbol{x}_i)) - (1-y_i)\ln(1-\sigma(\boldsymbol{x}_i))]$$

感知机（Perceptron）

感知机在1957年由Rosenblatt提出，是支持向量机和神经网络的基础。感知机是一种二类分类的线性分类模型，输入为实例的特征向量，输出为实例的类别，正类取1，负类取-1。感知机是一种判别模型，其**目标是求得一个能够将数据集中的正实例点和负实例点完全分开的分离超平面**。如果数据不是线性可分的，则最后无法获得分离超平面。

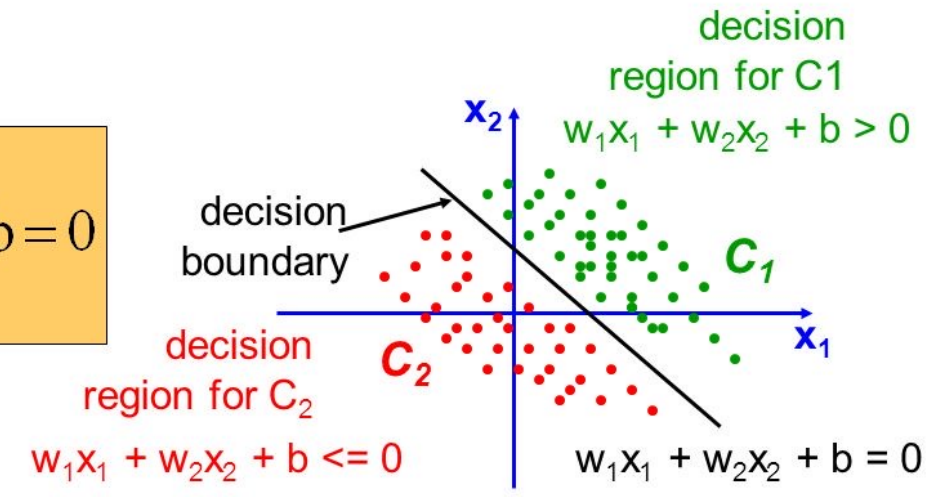
假设模型的输入为$\boldsymbol{x} \in R^n$，输出为{+1，-1}，则模型可以表示为：

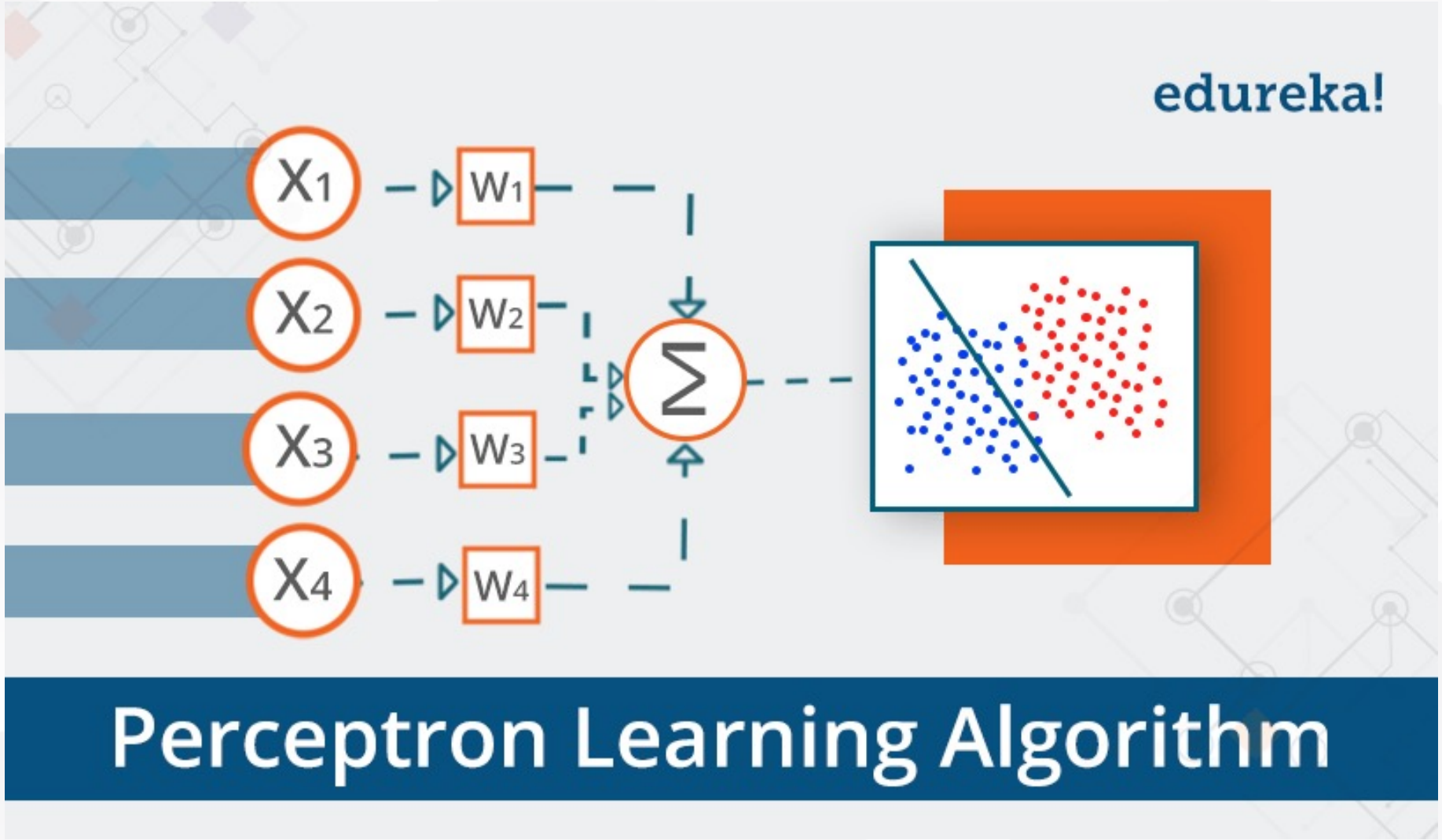
$$f(\boldsymbol{x}) = \text{sign}(\boldsymbol{w} \cdot \boldsymbol{x} + b)$$

$$\text{sign}(z) = \begin{cases} +1, & \text{if } z \geq 0 \\ -1, & \text{if } z < 0 \end{cases}$$

其中$\boldsymbol{w}$和b为感知机的模型参数，$f(x)$即为要求的分离超平面。图1为感知机模型的示意图，$\boldsymbol{w} \cdot \boldsymbol{x} + b = 0$即为一个超平面。

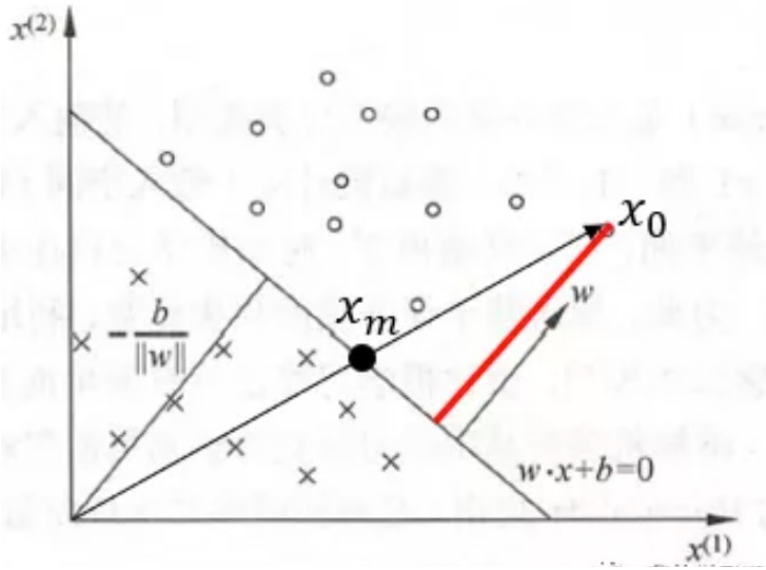
$$\sum_{i=1}^{m} w_i x_i + b = 0$$




edureka!
X1
X2
X3
X4
W1
W2
W3
W4
Σ
Perceptron Learning Algorithm

损失函数：

为了求得分离超平面，需要构建一个损失函数，并通过极小化此损失函数来求得模型参数$\boldsymbol{w}$和b。**感知机的损失函数为所有误分类点到超平面的距离之和**。首先写出输入空间R^n任一实例点到$\boldsymbol{x_0}$分离超平面的距离d，即图2中的红线长度：



感知机的损失函数：

$$L(\boldsymbol{w}, b) = -\sum_{\boldsymbol{x_i} \in M} y_i(\boldsymbol{w} \cdot \boldsymbol{x_i} + b)$$

损失函数：

要求得模型参数$\boldsymbol{w}$和b，需要极小化损失函数，即：

$$min_{\boldsymbol{w},b} L(\boldsymbol{w}, b) = -\sum_{\boldsymbol{x_i} \in M} y_i(\boldsymbol{w} \cdot \boldsymbol{x_i} + b)$$

极小化的过程采用随机梯度下降算法(stochastic gradient descent, SGD)来实现，首先任意选取一个超平面$\boldsymbol{w_0}, b_0$，然后找出误分类点，每次随机选取一个误分类点来进行梯度下降，进而对$\boldsymbol{w}$和b进行更新，直到没有误分类点：

$$\boldsymbol{w} = \boldsymbol{w} + \eta y_i \boldsymbol{x_i}$$
$$b = b + \eta y_i$$

其中，$\eta(0 < \eta \leq 1)$为学习率。

支持向量机（SVM）

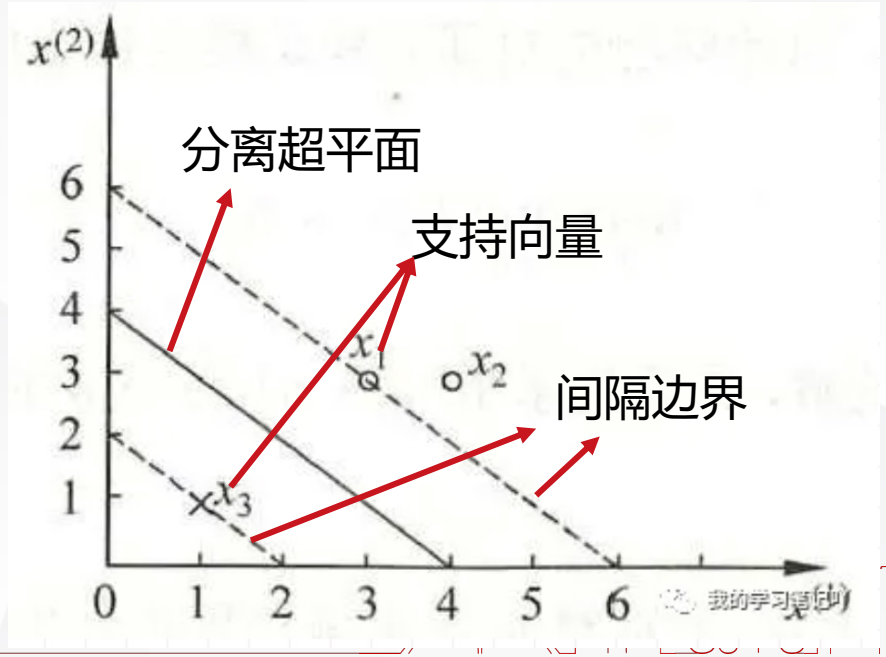
支持向量机(support vector machines, SVM)也是一种二类分类模型，感知机的目标是求得一个能够将数据正确分类的分离超平面，而SVM则是要**「找到一个能将数据正确分类且间隔最大的分离超平面」**，即SVM不仅要找到一个分离超平面，而且要求以尽可能大的确信度将数据正确分类。SVM与感知机的另一个区别在于，当数据是线性可分的时候，感知机求得的分离超平面可能有无穷多个，而SVM有了间隔最大化的约束，最后求得的分离超平面解是唯一的。

支持向量：

在线性可分的情况下，训练数据的样本点中与分离超平面距离最近的样本点称为支持向量。

模型求解：

在实际应用过程中，一般不直接对原始问题进行求解，而是通过拉格朗日对偶性，**「先求解对偶问题的解，进而求得原始问题的解」**，这便是线性可分SVM的对偶算法。



Python实现：scikit-learn

SVM模型解决手写数字识别问题：

http://172.28.19.228:8899/notebooks/Cources/scikit-learn/examples/classification/plot_digits_classification.ipynb

分类方法比较：

http://172.28.19.228:8899/notebooks/Cources/scikit-learn/examples/classification/plot_classifier_comparison.ipynb

提纲

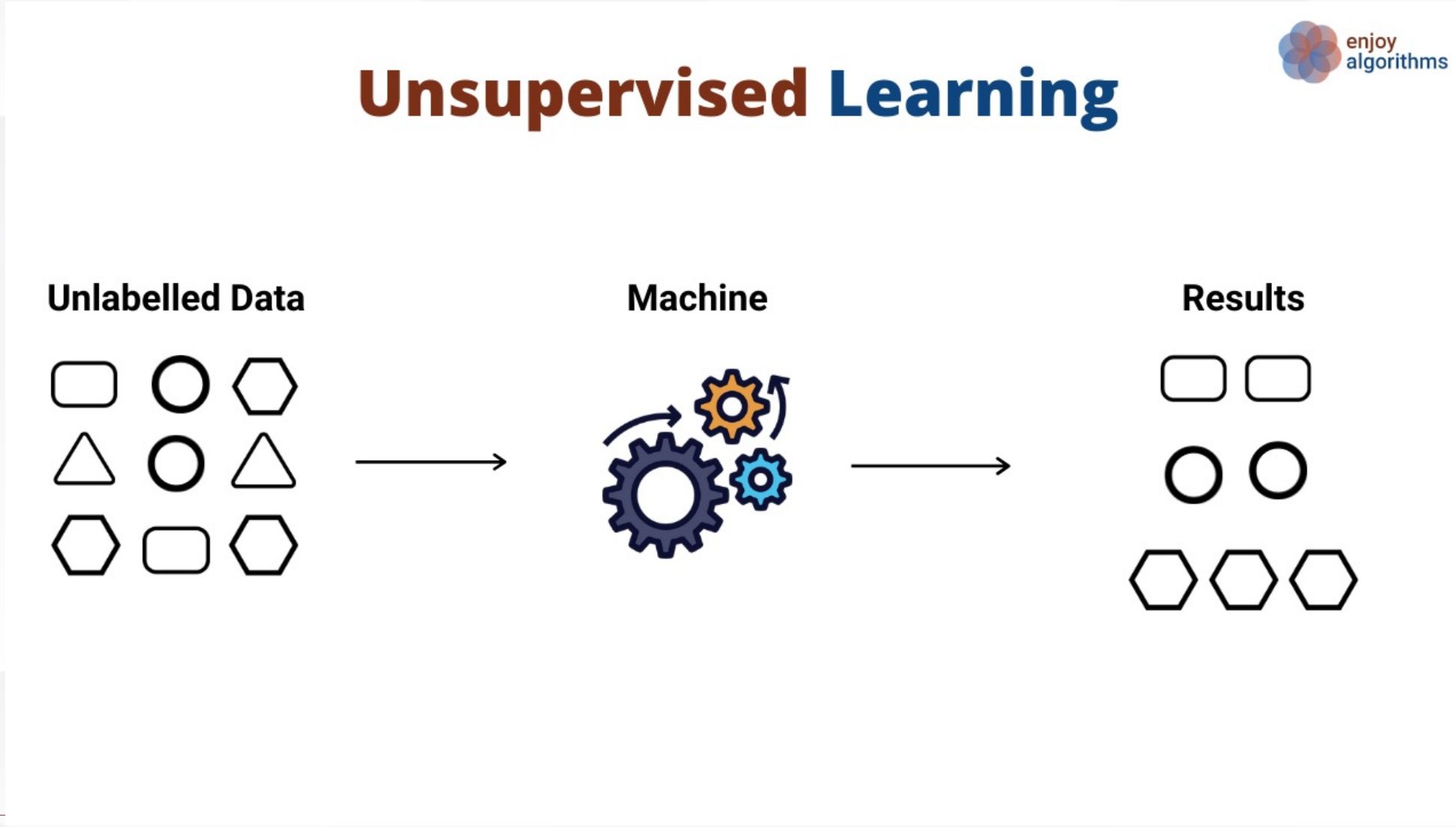
1 监督学习

2 无监督学习

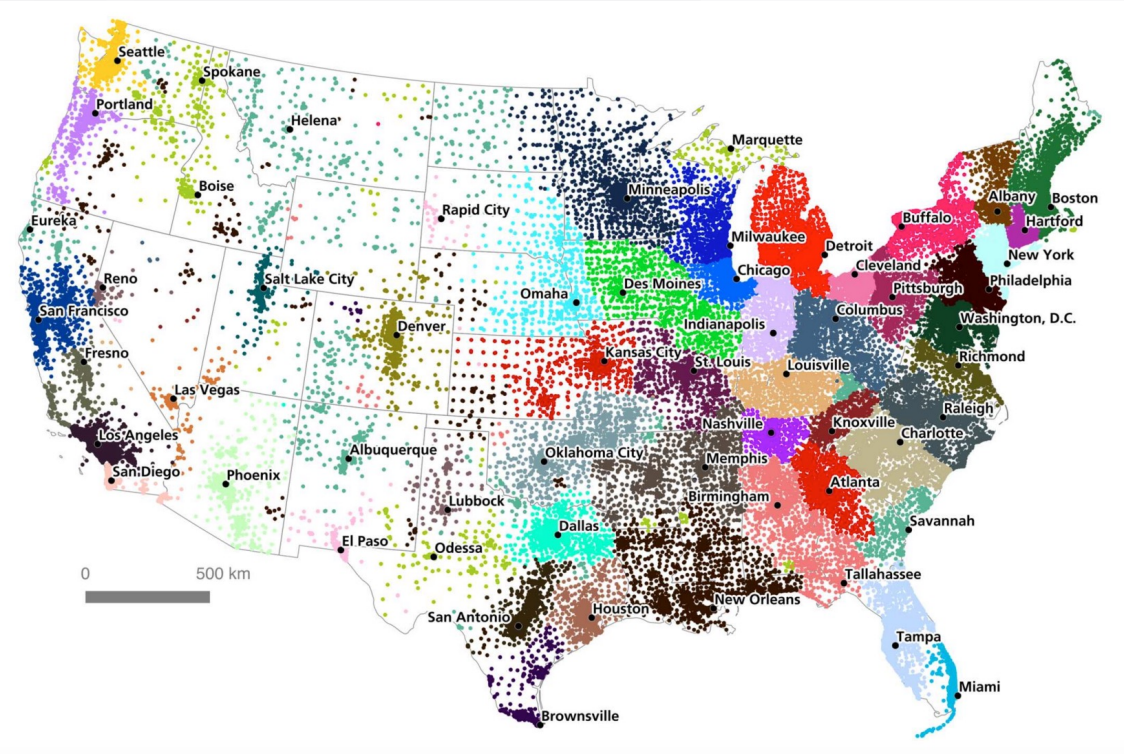
3 弱监督学习

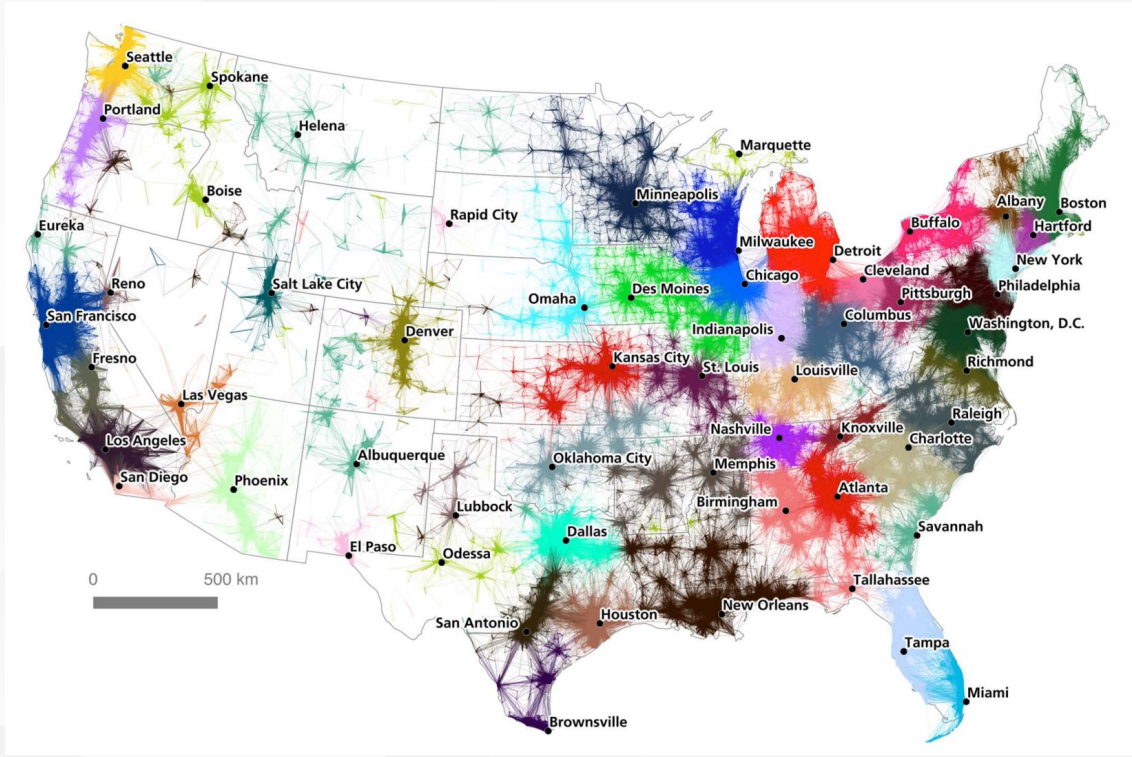
4 知识点回顾

无监督学习（Unspuervised Learning），又可称为自监督学习（Self-supervised Learning）



训练样本的标记信息是未知的，给定一个数据集，聚类的目的是通过对无标签数据的学习来揭示数据的内在性质和规律，将样本点划分为若干类，使得属性相似的样本归属于同一类，即类内相似度（intra-cluster similarity）高，但是簇间相似度（inter-cluster similarity）低





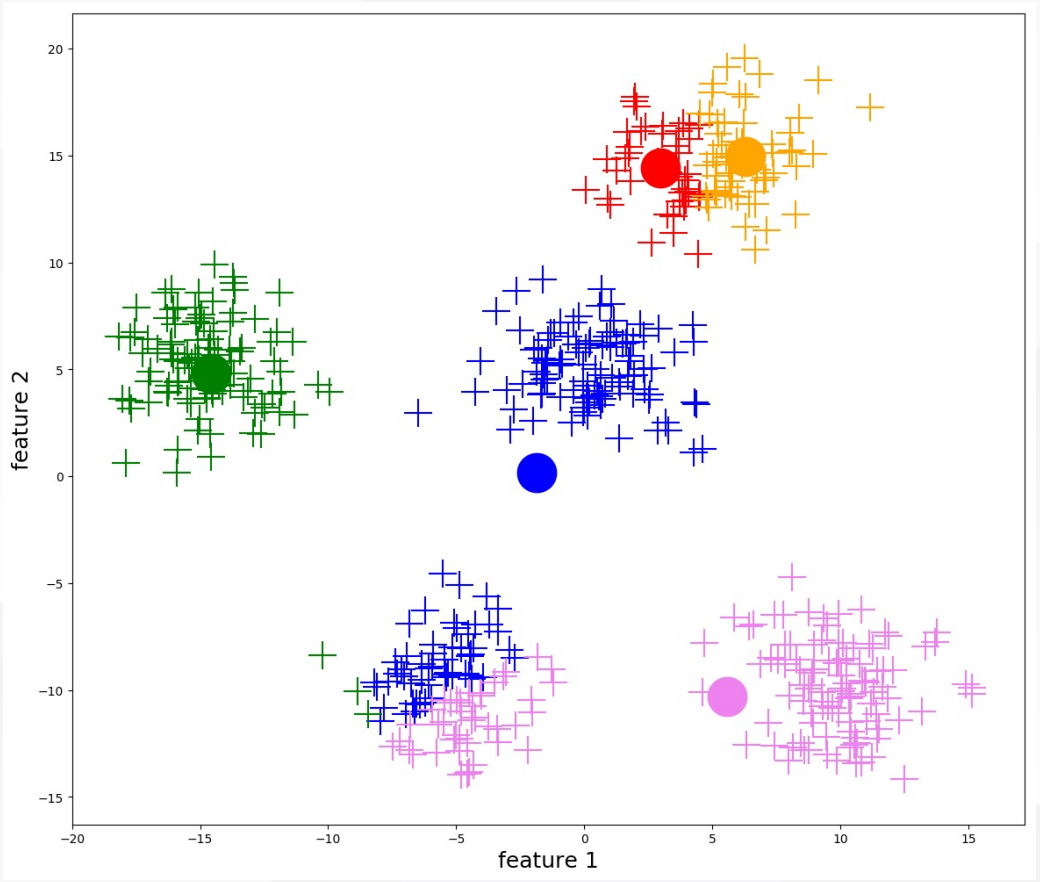
Source: Nelson, Garrett Dash, and Alasdair Rae. "An economic geography of the United States: From commutes to megaregions." *PloS one* 11.11 (2016): e0166083.

K-means 算法

牧师—村民模型：

- 有四个牧师去郊区布道，一开始牧师们随意选了几个布道点，并且把这几个布道点的情况公告给了郊区所有的村民，于是每个村民到离自己家最近的布道点去听课。
- 听课之后，大家觉得距离太远了，于是每个牧师统计了一下自己的课上所有的村民的地址，搬到了所有地址的中心地带，并且在海报上更新了自己的布道点的位置。
- 牧师每一次移动不可能离所有人都更近，有的人发现A牧师移动以后自己还不如去B牧师处听课更近，于是每个村民又去了离自己最近的布道点……
- 就这样，牧师每个礼拜更新自己的位置，村民根据自己的情况选择布道点，最终稳定了下来。

我们可以看到该牧师的目的是为了让每个村民到其最近中心点的距离和最小。



K-means 算法

K-means算法流程：

1. 选择初始化的 k 个样本作为初始聚类中心 $a = a_1, a_2, \ldots a_k$ ；
2. 针对数据集中每个样本 x_i 计算它到 k 个聚类中心的距离并将其分到距离最小的聚类中心所对应的类中；
3. 针对每个类别 a_j ，重新计算它的聚类中心 $a_j = \frac{1}{|c_i|}\sum_{x \in c_i} x$ （即属于该类的所有样本的质心）；
4. 重复上面 2 3 两步操作，直到达到某个中止条件（迭代次数、最小误差变化等）。

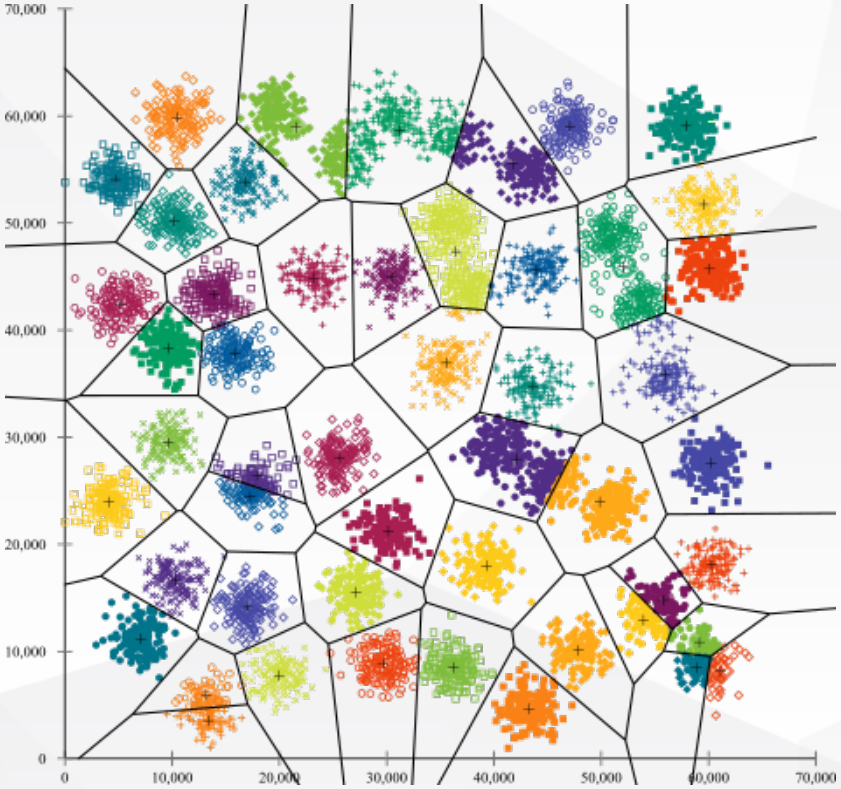
```
获取数据 n 个 m 维的数据
随机生成 K 个 m 维的点
while(t)
    for(int i=0;i < n;i++)
        for(int j=0;j < k;j++)
            计算点 i 到类 j 的距离
    for(int i=0;i < k;i++)
        1. 找出所有属于自己这一类的所有数据点
        2. 把自己的坐标修改为这些数据点的中心点坐标
end
```

时间复杂度：O(tknm) ，其中，t 为迭代次数，k 为簇的数目，n 为样本点数，m 为样本点维度。

K-means 算法

优点

- 容易理解，聚类效果不错，虽然是局部最优， 但往往局部最优就够了；
- 处理大数据集的时候，该算法可以保证较好的伸缩性；
- 当簇近似高斯分布的时候，效果非常不错；
- 算法复杂度低。

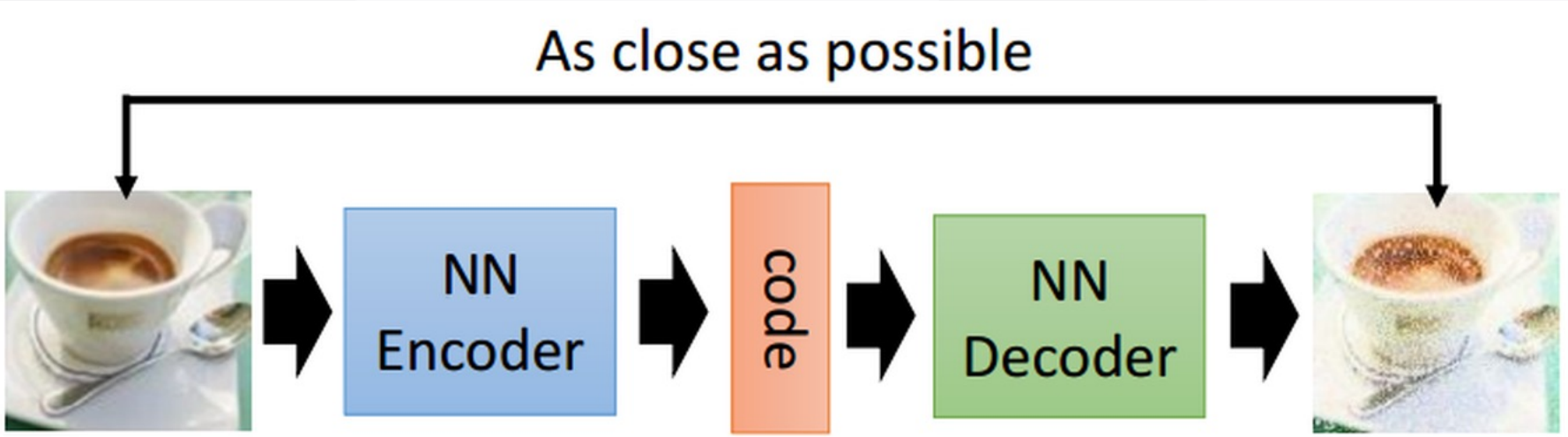


缺点

- K 值需要人为设定，不同 K 值得到的结果不一样；
- 对初始的簇中心敏感，不同选取方式会得到不同结果；
- 对异常值敏感；
- 样本只能归为一类，不适合多分类任务；
- 不适合太离散的分类、样本类别不平衡的分类、非凸形状的分类。

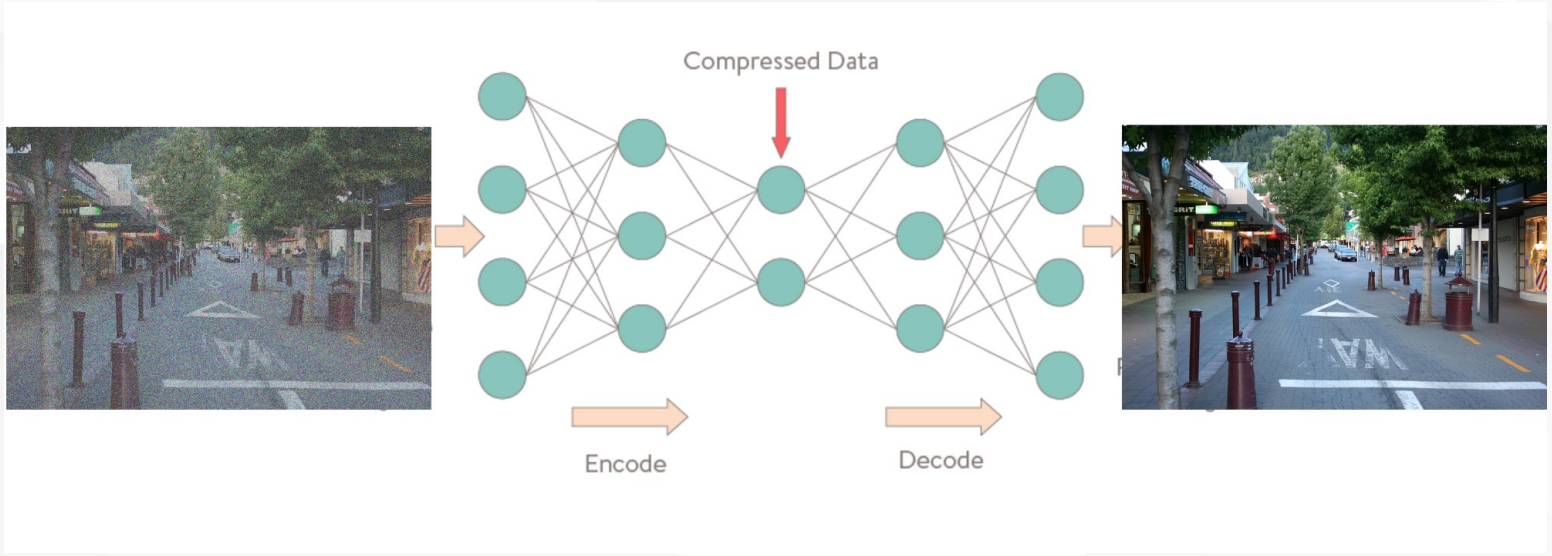
自编码器 AutoEncoder

自编码器（autoencoder, AE）是一类在半监督学习和非监督学习中使用的人工神经网络（Artificial Neural Networks, ANNs），其功能是通过将输入信息作为学习目标，对输入信息进行表征学习（representation learning）

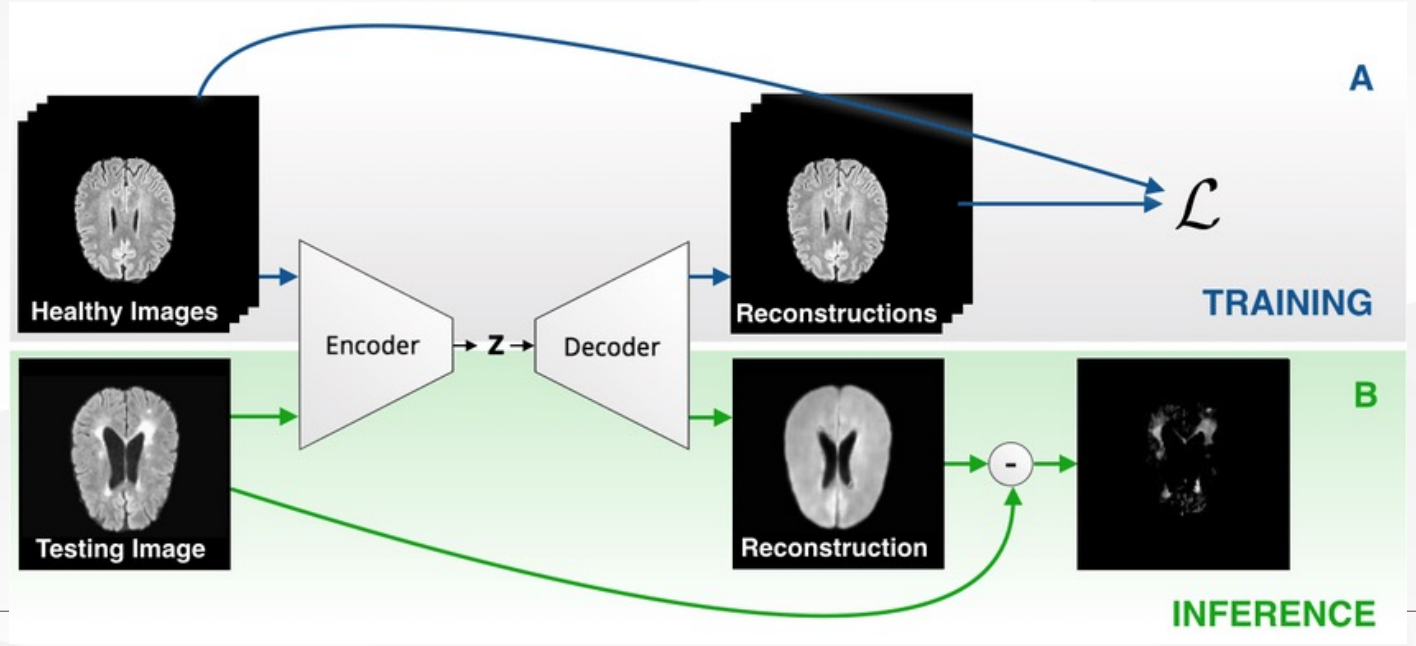


自编码器 AutoEncoder

图像去噪



异常检测



提纲

1 监督学习

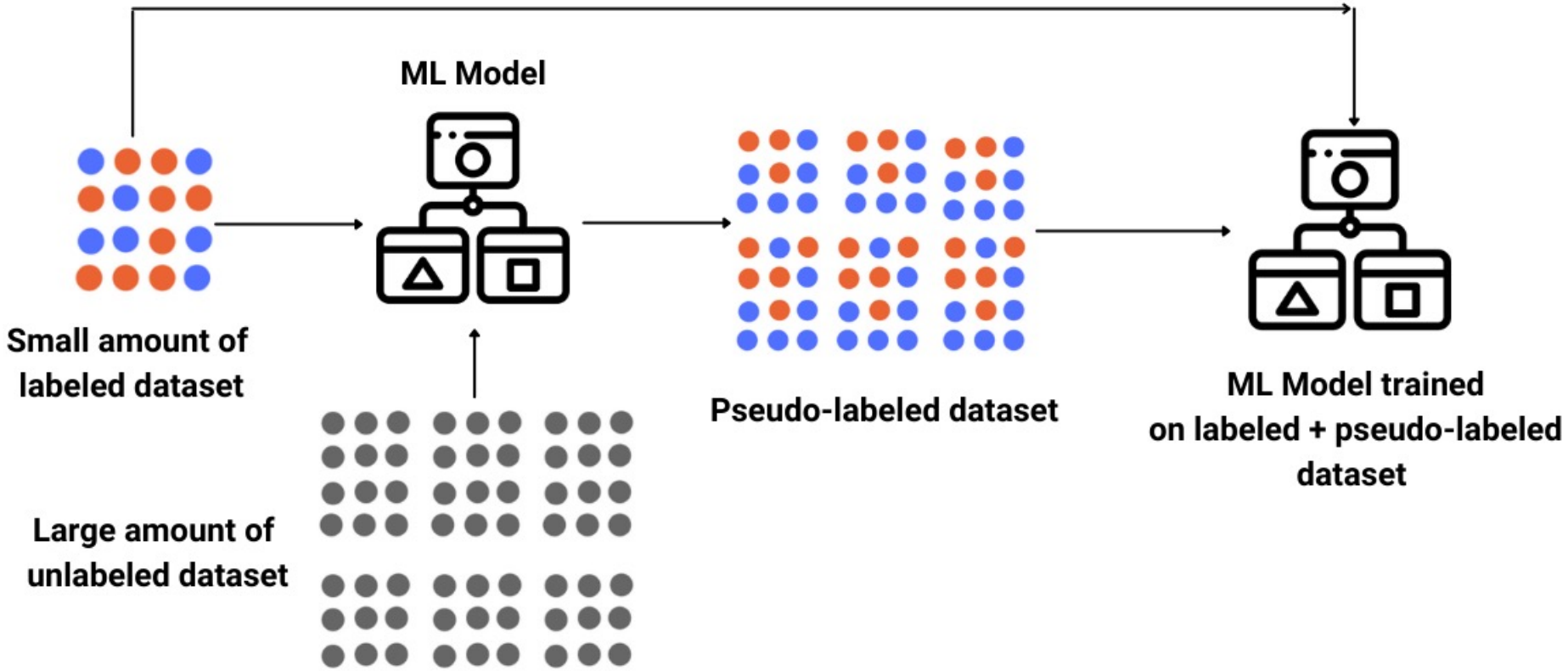
2 无监督学习

3 弱监督学习

4 知识点回顾

Semi-supervised learning use-case

enjoy algorithms



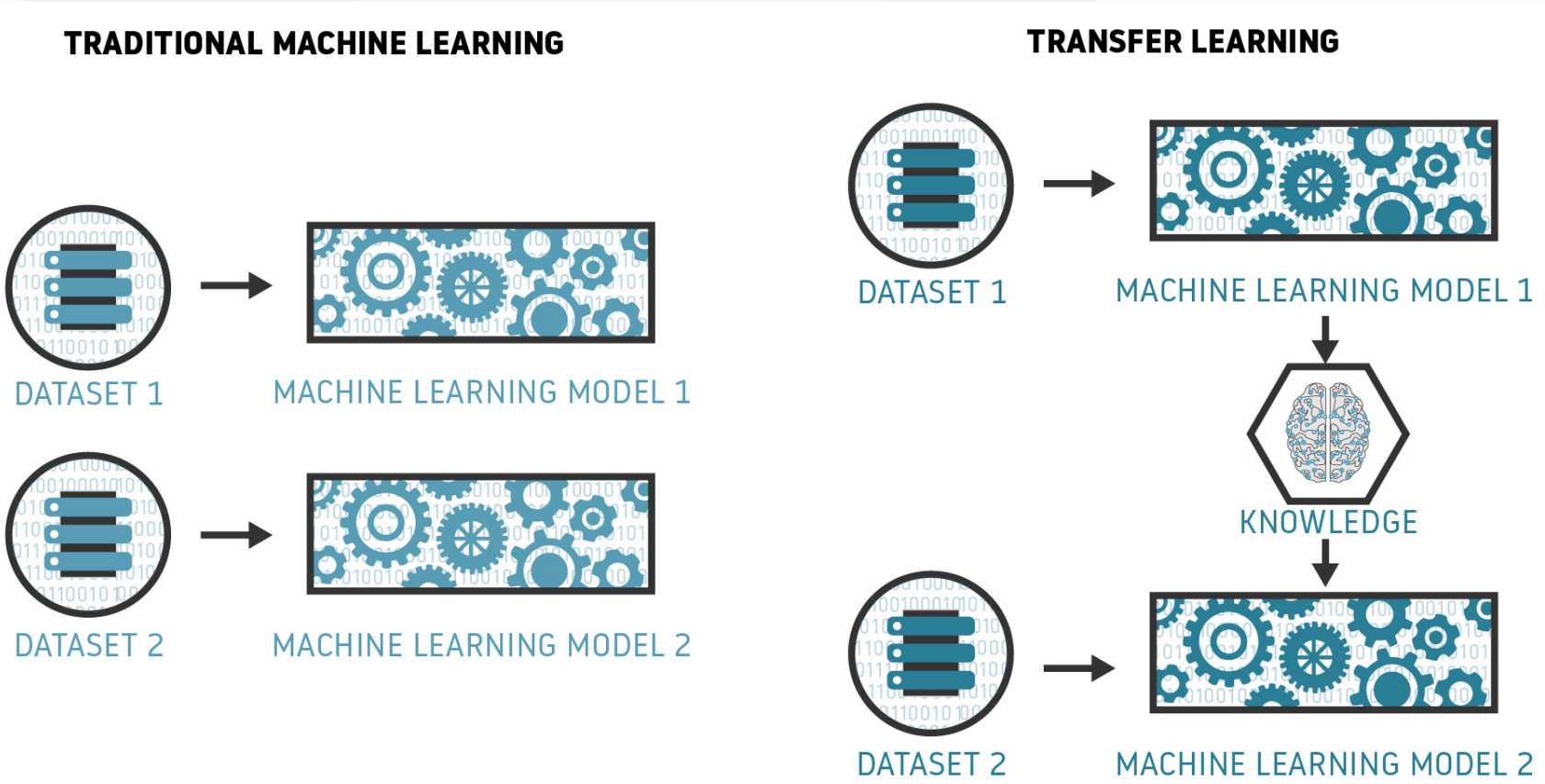
迁移学习

传统机器学习(主要指监督学习)

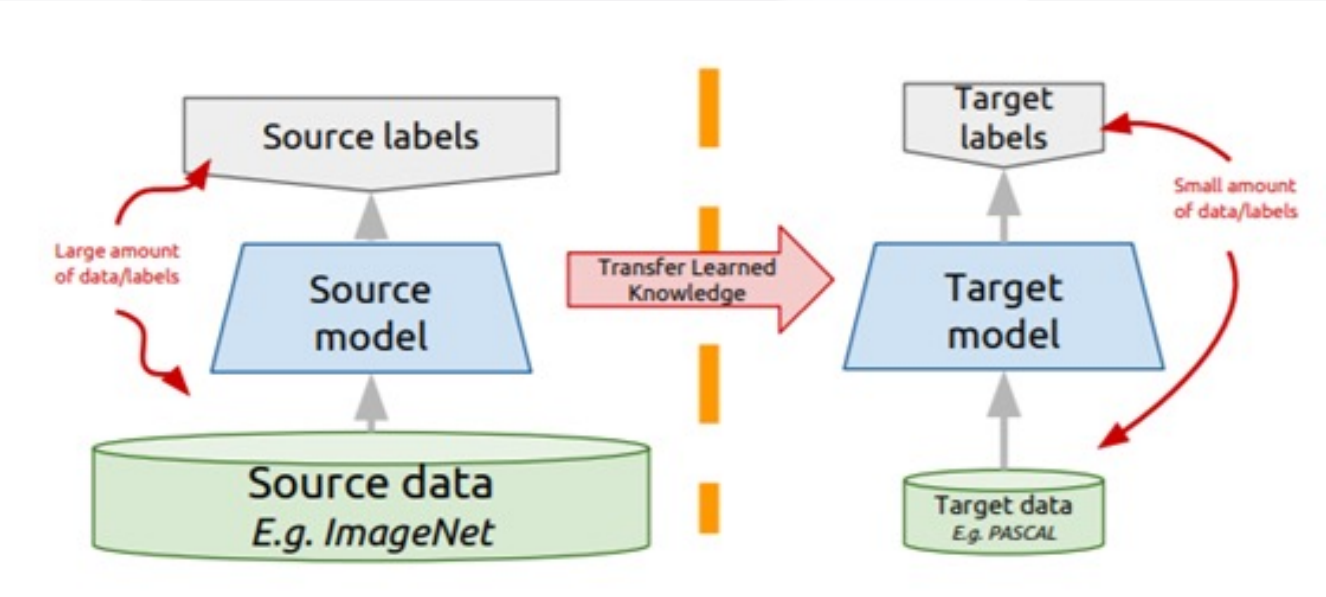
- 基于同分布假设
- 需要大量标注数据

然而实际使用过程中不同数据集可能存在一些问题，比如

- 数据分布差异
- 标注数据、训练数据过期



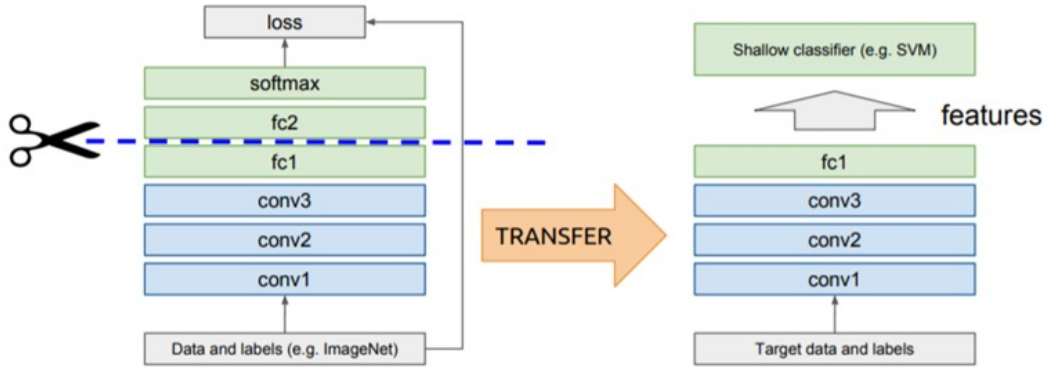
Idea 1: 模型参数更新

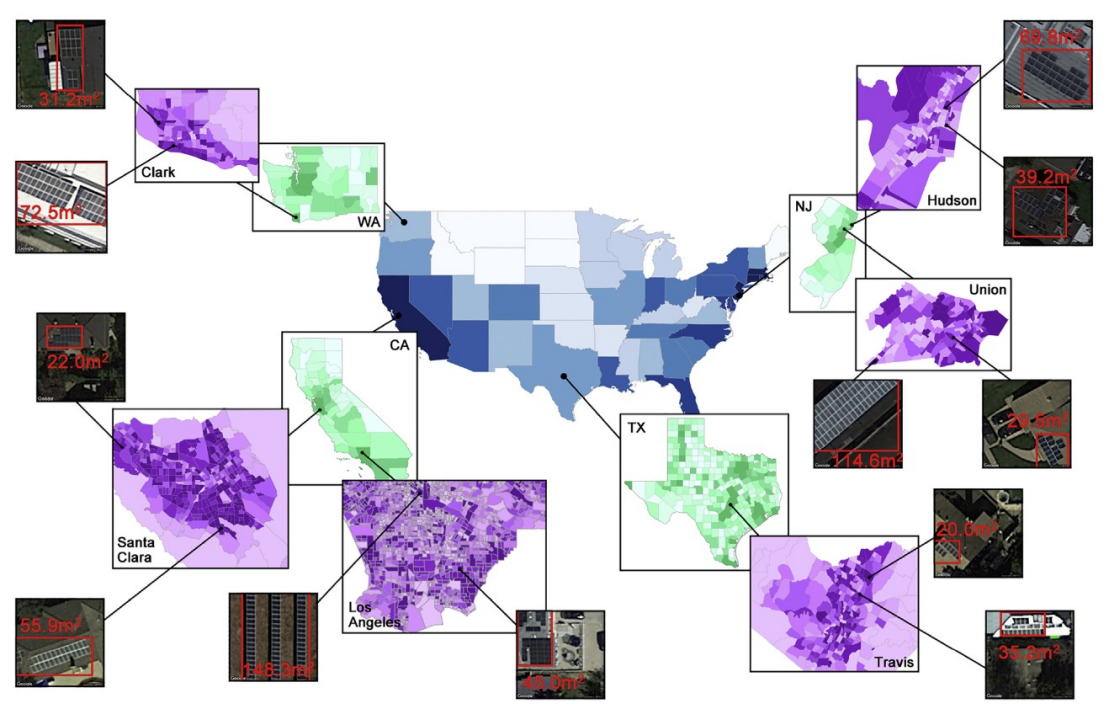


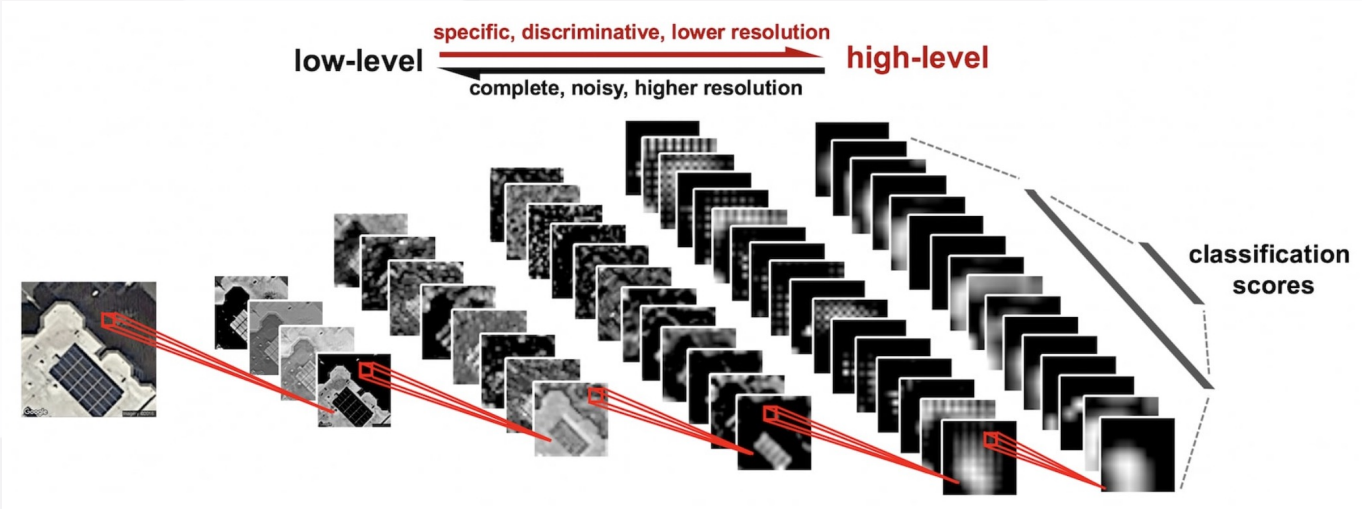
Idea 2: 模型拼接

Idea: use outputs of one or more layers of a network trained on a different task as generic feature detectors. Train a new shallow model on these features.

Assumes that $D_S = D_T$



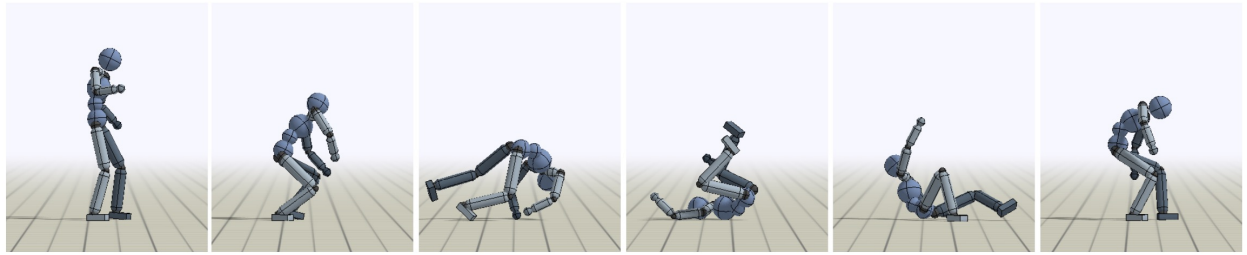


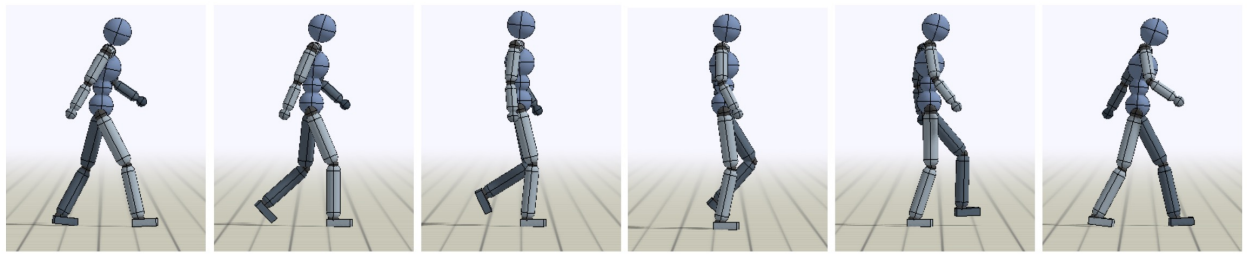


Source: Yu, Jiafan, et al. "DeepSolar: A machine learning framework to efficiently construct a solar deployment database in the United States." *Joule* 2.12 (2018): 2605-2617.

强化学习

强化学习（Reinforcement Learning，RL）的中心思想，就是**让智能体在环境里学习。**用于描述和解决智能体（agent）在与环境的交互过程中通过学习策略以达成回报最大化或实现特定目标的问题。一般采用马尔可夫决策过程（Markov Decision Processes,MDPs）进行建模。





Source: Song, Seungmoon, et al. *Journal of NeuroEngineering and Rehabilitation*, 2021

RL的四个基本元素：

- **智能体（Agent）**：程序中被训练、控制的对象，完成某个任务的个体；
- **环境（Environment）**：真实或虚拟世界，智能体在其中完成动作；
- **动作（Action）**：智能体的动作行为，动作会引起环境发生变化；
- **奖励（Reward）**：对动作的评价，可以是正或者负值

强化学习过程可以用一个循环 (loop) 来表示

- 智能体在环境 (超级马里奥) 里获得初始状态**S0** (游戏的第一帧) ；
- 在S0的基础上，agent会做出第一个行动**A0** (如向右走) ；
- 环境变化，获得新的状态**S1** (A0发生后的某一帧) ；
- 环境给出了第一个奖励**R1** (没死：+1) ；

这个loop输出的就是一个**由状态、奖励和行动组成的序列**。而智能体的**目标**就是让**预期累积奖励最大化**。

数学化表示

在时刻 t，由于当前对不可全知的环境，只能采取对环境的 observation **观察**O_t，选择一个action **行为**A_t，得到一个reward **奖励打分** R_{t+1}，然后继续选择动作，循环往复。（当A_t的行为采用后，环境将会改变，而采取了A_t得到的奖励是在下一时刻出现的，即R_{t+1}）

所以 t 时刻所包含的整个历史过程会是：$H_t = O_1, R_1, A_1, ..., O_{t-1}, R_{t-1}, A_{t-1}, O_t, R_t, A_t$

当前的 **状态**S_t 是所有已有信息序列的函数$S_t = f(H_t)$，用于决定未来的发展。

Policy π：表示所采用的行动。它是从状态 S到行为A的一个映射，可以是确定性的，也可以是不确定性的。即动作将会根据策略 π(a|s) 这一概率分布来进行选择。

Value：状态价值函数，每种动作可能带来的后续奖励，用于评估策略。因为只有一个当前奖励 reward 是远远不够的，当前奖励大并不代表其未来的最终奖励一定是最大的，所以需要使用 Value期望函数来评价未来。

$$v_\pi(s) = E_\pi(R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + ...|S_t = s)$$

即是在s状态和策略π时，采取行动后的价值由接下来的时刻的R的期望组成，其中γ是奖励衰减因子，用于控制当前奖励和未来奖励的比重。越小越重视现在，由于环境不可全知，未来具有更多的不确定性。

Model：用来感知场景的变化。模型要解决两个问题：一是**状态转化概率** $P^a_{ss\prime}$，即预测在s状态下，采取动作a,转到下一个状态s'的概率。另一个是预测可能获得的即时奖励R^a_s。

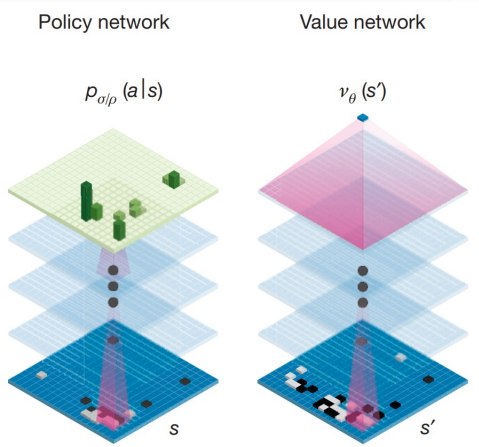
深度强化学习（Deep Reinforcement Learning，DRL），其所带来的推理能力 是智能的一个关键特征衡量，真正的让机器有了自我学习、自我思考的能力。

DRL 本质上属于采用神经网络作为值函数估计器的一类方法，其主要优势在于它能够利用深度神经网络对状态特征进行自动抽取，避免了人工 定义状态特征带来的不准确性，使得Agent能够在更原始的状态上进行学习。

AlphaGo将深度学习方法引MCTS，设计了两个深度学习网络：

策略网络：选择若干认为最好的下子点

估值网络：对给定棋局进行估值，判断棋局是否有利



提纲

1 监督学习

2 无监督学习

3 弱监督学习

4 知识点回顾

4. 线性回归中，总体误差平方和(TSS)、残差平方和(RSS)与回归误差平方和(ESS)三者的关系是_____。 **答案：B 知识点：【38040104】 知识考核要求：【3】 能力考核要求：【1】 难度系数：【C】**

A. RSS=TSS+ESS

B. TSS=RSS+ESS

C. ESS=RSS-TSS

D. ESS=TSS+RSS

5. 过拟合模型表现在训练集上的特点是_____。 **答案：B 知识点：【38040106】 知识考核要求：【3】 能力考核要求：【1】 难度系数：【C】**

A. 低方差和低偏差

B. 高方差和低偏差

C. 低方差和高偏差

D. 高方差和高偏差

8. 在一个分类问题中，如果样本的特征有限，且都取离散值。最适合的分类模型为_____。 **答案：A 知识点：【38040402】 知识考核要求：【3】 能力考核要求：【1】 难度系数：【C】**

A. 决策树

B. 线性回归模型

C. 生成对抗网络

D. K 近邻分类器

9. 不适合使用 Logistic 回归的任务有_____。 **答案：A 知识点：【38040404】 知识考核要求：【3】 能力考核要求：【1】 难度系数：【D】**

A. 电影票房预测

B. 人脸检测

C. 情感分类

D. 垃圾邮件分类

4. 机器学习/深度学习项目所需的步骤主要有_____。 **答案：ABCD 知识点：【38050101】 知识考核要求：【3】 能力考核要求：【1】 难度系数：【B】**

A. 采集数据

B. 数据预处理与特征选择

C. 选择模型、训练模型、评估模型

D. 预测、上线运行

4. _____可以用来评估最终模型的泛化能力，但不能作为调参、选择特征等算法相关的选择的依据。 **答案：B 知识点：【38040102】 知识考核要求：【3】 能力考核要求：【1】 难度系数：【B】**

A. 训练集

B. 测试集

C. 验证集

D. 参数集

6. 在机器学习中，损失函数（loss 函数）的作用是_____。 答案：A 知识点：【38040104】 知识考核要求：【3】 能力考核要求：【1】 难度系数：【C】

A. 牵引模型参数的更新

B. 增加网络层次

C. 修改模型参数

D. 防止过拟合

5. 在机器学习中，用于学习的经验数据集合称为_____。 答案：A 知识点：【38040102】 知识考核要求：【3】 能力考核要求：【1】 难度系数：【A】

A. 训练集

B. 测试集

C. 验证集

D. 标签集

4. 某机器学习模型对训练集的准确率很高，但对测试集则效果不佳，其原因可能是_____。 答案：B 知识点：【38040106】 知识考核要求：【3】 能力考核要求：【1】 难度系数：【C】

A. 欠拟合

B. 过拟合

C. 参数过少

D. 机器性能问题

5. 在决策树中，_____属于特征重要性评估的方法。 答案：C 知识点：【38040402】 知识考核要求：【3】 能力考核要求：【1】 难度系数：【C】

A. 交叉熵

B. 布朗运动

C. 信息熵

D. 核方法

2. 机器学习的关键要素包括_____。 答案：ABCD 知识点：【38040101】 知识考核要求：【3】 能力考核要求：【1】 难度系数：【C】

A. 数据

B. 模型

C. 损失函数

D. 优化算法

4. 欠拟合的直观表现是_____。 答案：A 知识点：【38040106】 知识考核要求：【3】 能力考核要求：【1】 难度系数：【B】

A. 在训练集上精度较低

B. 在训练集上精度较高

C. 在验证集上精度较低

D. 在验证集上精度较高

1. 从以下候选答案集合中为每空**选择一个正确的答案，将其字母编号**填入相应空格。候选答案集合如下：

A. LogisticRegression( )	B. LinearRegression( )
C. 5	D. 2.5
E. m.score(x,y)	F. m.score(array)

现有学习驾驶技术投入的练习时间（月）与通过考试成绩（是否通过）的历史数据，现使用Logistic回归预测训练时间为两个半月的考试成绩。

```
import numpy as np
x=np.array([0.50,0.75,1.00,1.25,1.50,1.75,1.75,2.00,2.25,2.50,2.75,3.00,3.25,3.50,4.00,4.25,4.50
    ,4.75,5.00,5.50])                                    #训练时间（月）
y=np.array([0,0,0,0,0,0,1,0,1,0,1,0,1,0,1,1,1,1,1,1])    #通过考试为1，不通过考试为0

from sklearn.linear_model import LogisticRegression
m=_____（1）_____                                        #逻辑回归模型定义
x=x.reshape(-1,1)                                        #如果只有一个特征，重塑为一列数组
y=y.reshape(-1,1)
m.fit(x,y)                                               #模型学习
p=m.predict(____（2）____)                               #预测
print(p)
print(_____（3）_____)                                   #评估模型准确率
```

1)：【A】
2)：【D】
3)：【E】
知识点：【38040404】【38040404】【38040404】 知识考核要求：【3】【3】【3】 能力考核要求：【1】【1】【1】 难度系数：【C】【C】【C】

A. KMeans(n_clusters=3)	B. KMeans(n_clusters=4)
C. kmeans.labels	D. kmeans.tags
E. drawing	F. plotting

现给定iris.data鸢尾花测量数据集，包括花瓣的长度和宽度、花萼的长度和宽度4类特征。鸢尾花有三个品种：setosa，versicolor，virginnica。使用K-Means算法对数据集样本进行品种聚类。

```
import pandas as pd
from sklearn.cluster import KMeans

filename = 'iris.data'
data = pd.read_csv(filename, header = None)
#生成k-means模型
X = data.iloc[:,0:4].values.astype(float)          #提取数据集中的四个特征数据
kmeans =_____（1）_____                            #模型定义
kmeans.fit(X)                                      #模型学习
print('means.labels_:\n',_____（2）_____)          #输出聚类结果
pd._（3）_.scatter_matrix(data, c=kmeans.labels_)  #生成散点矩阵图观察聚类效果
plt.show( )
```

1)：【A】
2)：【C】
3)：【F】
知识点：【38040408】【38040408】【38040408】 知识考核要求：【3】【3】【3】 能力考核要求：【1】【1】【1】 难度系数：【D】【D】【D】

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

人工智能研究院
Artificial Intelligence Institute

谢谢！

饮水思源 爱国荣校

http://humnetlab.berkeley.edu/~yxu/