如何選擇機器學習演算法？

在這篇文章中，我將解釋機器學習演算法的類型以及應當如何針對你的任務進行選擇。 我認為瞭解機器學習演算法的類型，有助於看清人工智慧技術的全貌，理解在這個領域中大家在做的所有事情的目標是什麼，從而可以幫助你更好地分析現實問題並設計出一個機器學習系統。

本文會使用如下術語：

* 標注過的數據 （labeled data）：由一組訓練樣本組成的資料，其中每個樣本包含輸入和期望的輸出（也稱為監督信號、標籤等 ）
* 分類（classification）：分類的目標是預測（有限的）離散值，例如{1,0}，{True，False}，{垃圾郵件，非垃圾郵件}。
* 回歸（regression）：回歸的目標是預測（無限的）連續值，例如房價。

機器學習演算法的類別

在如何定義機器學習演算法的類別方面，存在一些不同的觀點，但是通常可以根據使用目的將演算法劃分成不同類別，主要類別如下：

* 監督學習：Supervised Learning
* 無監督學習：Unsupervised Learning
* 半監督學習： Semi-supervised Learning
* 強化學習 ：Reinforcement Learning

監督學習

我喜歡用函數逼近（function approximation）的概念來看待監督學習，基本上我們訓練演算法並在最終得到一個能夠最好地描述輸入資料的函數，即對於一個給定的X該函數能夠做出最好的估計y：X -> y。 之所以視之為一種函數逼近，一個原因在於大多數時候我們無法寫出能夠做出正確預測的真實函數的數學封閉形式，另一原因在於這些演算法依賴於我們對機器如何進行學習所作出的一些假設，而這些假設會引入偏差（bias），關於偏差這個話題我會在另一篇文章中進行解釋。

在這裡，人類專家扮演著老師的角色，我們向電腦提供包含輸入（input）和正確輸出（label）的訓練資料，利用監督學習演算法，電腦應該能夠從訓練資料中學習到輸入-輸出的關係模式（pattern）。

監督學習演算法試圖對目標預測輸出和輸入特徵之間的關係和依賴進行建模，這樣我們就可以基於學習到的模型（輸出-輸出之間的關係）來預測新的輸入資料的輸出值。

監督學習類型的演算法，具有以下特徵：

* 屬於預測式模型
* 使用標注過的資料
* 包括回歸（連續值預測）和分類 （離散值預測）

常用的監督學習類型的演算法包括：

* 最近鄰演算法：nearest neighbour
* 樸素貝葉斯：naive bayes
* 決策樹：decision tree
* 線性回歸：linear regression
* 支持向量機：support vector machine
* 神經網路 ：neural network

無監督學習

在無監督學習中，電腦使用未標注過的資料進行訓練。

無監督學習不需要老師，電腦能夠在學習到訓練資料中隱含的模式之後教給你一些新的洞察，這些演算法在人類專家不知道如何處理資料時特別有用。

無監督學習主要用於模式檢測（detection）和描述性（descriptive）建模 。 但是， 在無監督學習中沒有輸出類別或標籤來説明演算法建模。 這些演算法總是嘗試直接在輸入資料中挖掘規則 、檢測模式 、對資料進行總結和分組，從而獲得對資料的洞察，並更好地向使用者描述資料。

無監督學習演算法的主要特點是：

* 屬於描述性模型
* 包括聚類（clustering）演算法和關聯規則（association rule）學習演算法。

無監督學習類型的演算法包括：

* k均值聚類：k-means clustering
* 關聯規則：association rules

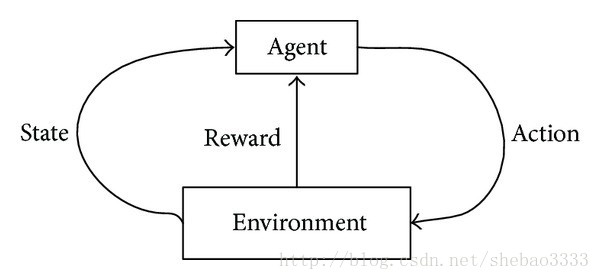
半監督學習

在前兩種類型中，要麼沒有資料集中所有觀測樣本的標籤，要麼所有的觀測值都存在標籤。 半監督學習處於這兩者之間。 在許多實際情況下，標注的成本相當高，因為它需要熟練的人類專家來完成。 因此，在大多數觀測樣本中沒有標籤的情況下，半監督演算法是模型構建的最佳候選者。 半監督學習基於這一思想：即使未標注資料的分組（group）是未知的，這些資料依然攜帶有關於其分組的重要資訊。

強化學習

強化學習方法的目的是利用從與環境（action）的交互中採集到的觀測樣本（observation），來採取可以最大化獎勵（reward）或最小化風險的行動（action）。 強化學習演算法（稱為代理：agent）以反覆運算的方式不斷地從環境中學習。 在這個過程中，代理從它與環境的交互中學習，直到探索完所有可能的狀態（state）。

強化學習是一種機器學習 ，因此也是人工智慧的一個分支。 它允許機器和軟體代理自動地確定特定環境下的理想行為，以最大限度地提高性能。 代理需要簡單的獎勵回饋來學習其行為; 這被稱為強化信號：



有很多不同的演算法來解決這個問題。 事實上，強化學習是由一種特定類型的問題所定義的，所有針對此問題的解決方案都被歸類為強化學習演算法。 在這個問題中，代理應該根據其當前狀態決定最佳的動做。 當這一步驟不斷重複時，這個問題被稱為瑪律科夫決策過程 （MDP）。

為了生成智慧代理，強化學習需要經過以下步驟：

1. 代理觀察輸入狀態。
2. 代理使用決策函數來選擇並執行一個動作。
3. 行動結束後，代理從環境中獲得獎勵或強化信號。
4. 代理存儲狀態-動作對應的回報資訊。

常用的強化學習演算法清單如下：

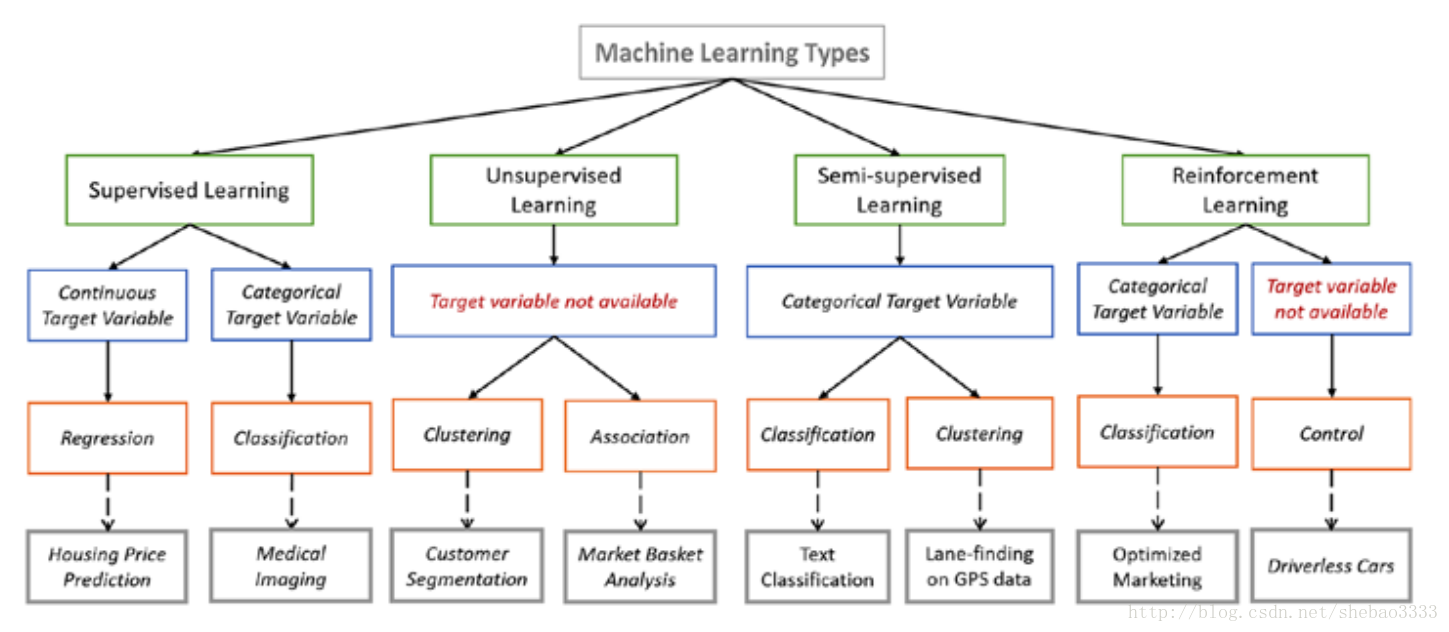
* Q學習：Q-learning
* 時間差分演算法：temporal difference
* 深度對抗網路 ：deep adversarial network

強化學習演算法的典型應用包括機器下棋（國際象棋，圍棋）、機器手和自動駕駛。

結語

有可能存在不同的標準來對機器學習演算法的類型進行分類，但是我認為以要解決的任務作為分類依據，可以很好地視覺化機器學習演算法的類型，我相信根據你面對的問題和所掌握的資料，可以很容易地決定是否將使用監督、無監督或強化學習。 在後續的文章中，我將給出更多關於每種機器學習演算法的例子。

下圖來自en.proft.me，可能會對你有所幫助：



原文：[Types of Machine Learning Algorithms You Should Know](https://towardsdatascience.com/types-of-machine-learning-algorithms-you-should-know-953a08248861)

————————————————