

# Introduction

define a set of function(model) -> goodness of function -> pick the best function

## Learning Map

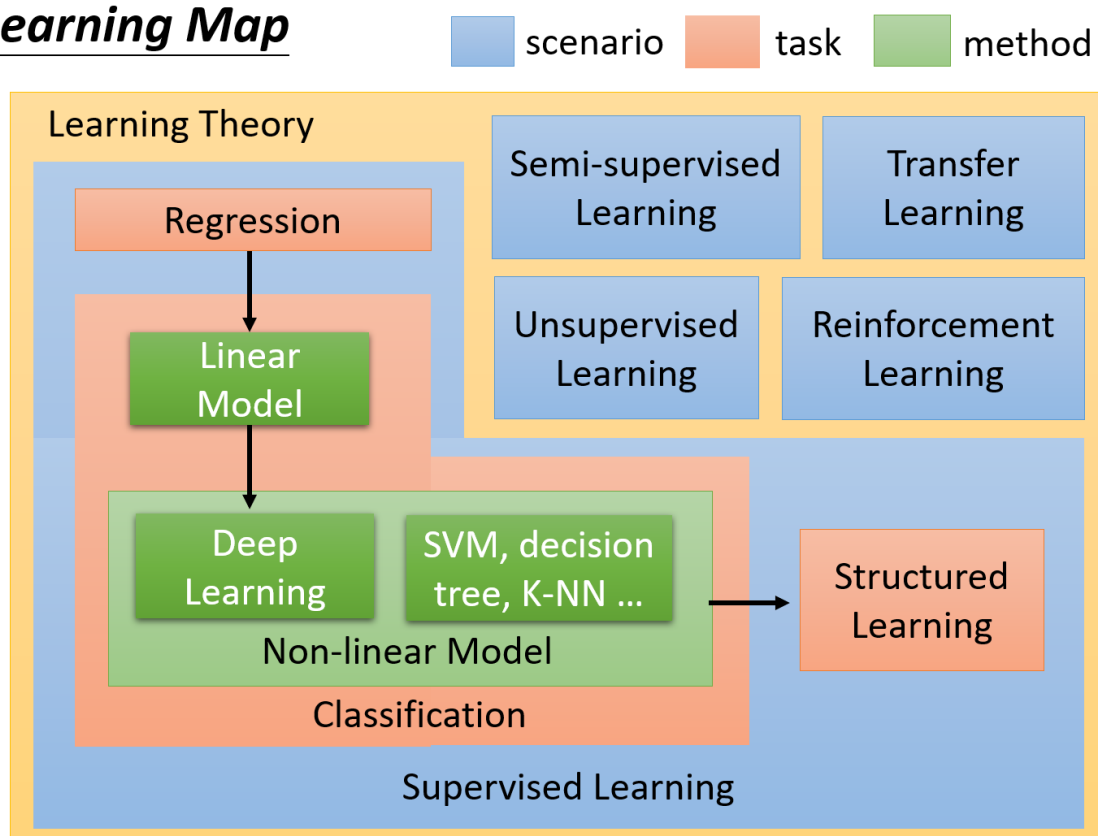
下图中，同样的颜色指的是同一个类型的事情

蓝色方块指的是scenario，即学习的情境。通常学习的情境是我们没有办法控制的，比如做 reinforcement Learning是因为我们没有data、没有办法来做supervised Learning的情况下才去做的。如果有data，supervised Learning当然比reinforcement Learning要好；因此手上有什么样的data，就决定你使用什么样的scenario

红色方块指的是task，即要解决的问题。你要解的问题，随着你要找的function的输出不同，有输出scalar的regression、有输出options的classification、有输出structured object的structured Learning...

绿色的方块指的是model，即用来解决问题的模型(function set)。在这些task里面有不同的model，也就是说，同样的task，我们可以用不同的方法来解它，比如linear model、Non-linear model(deep Learning、SVM、decision tree、K-NN...)

## Learning Map



## Supervised Learning(监督学习)

supervised learning 需要大量的training data，这些training data告诉我们说，一个我们要找的function，它的input和output之间有什么样的关系

而这种function的输出，通常被叫做label(标签)，也就是说，我们要使用supervised learning这样一种技术，我们需要告诉机器，function的input和output分别是什么，而这种output通常是通过人工的方式标注出来的，因此称为人工标注的label，它的缺点是需要大量的人工effort

## Regression(回归)

regression是machine learning的一个task，特点是**通过regression找到的function，它的输出是一个scalar数值**

比如PM2.5的预测，给machine的training data是过去的PM2.5资料，而输出的是对未来PM2.5的预测数值，这就是一个典型的regression的问题

## Classification(分类)

regression和classification的区别是，我们要机器输出的东西的类型是不一样的，在regression里机器输出的是scalar，而classification又分为两类：

### Binary Classification(二元分类)

在binary classification里，我们要机器输出的是yes or no，是或否

比如G-mail的spam filtering(垃圾邮件过滤器)，输入是邮件，输出是该邮件是否是垃圾邮件

### Multi-class classification(多元分类)

在multi-class classification里，机器要做的是选择题，等于给他数个选项，每一个选项就是一个类别，它要从数个类别里面选择正确的类别

比如document classification(新闻文章分类)，输入是一则新闻，输出是这个新闻属于哪一个类别(选项)

## model(function set) 选择模型

在解任务的过程中，第一步是要选一个function的set，选不同的function set，会得到不同的结果；而选不同的function set就是选不同的model，model又分为很多种：

- Linear Model(线性模型)：最简单的模型
- Non-linear Model(非线性模型)：最常用的模型，包括：
  - **deep learning**  
如alpha-go下围棋，输入是当前的棋盘格局，输出是下一步要落子的位置；由于棋盘是19\*19的，因此可以把它看成是一个有19\*19个选项的选择题
  - **SVM**
  - **decision tree**
  - **K-NN**

## Semi-supervised Learning(半监督学习)

举例：如果想要做一个区分猫和狗的function

手头上有少量的labeled data，它们标注了图片上哪只是猫哪只是狗；同时又有大量的unlabeled data，它们仅仅只有猫和狗的图片，但没有标注去告诉机器哪只是猫哪只是狗

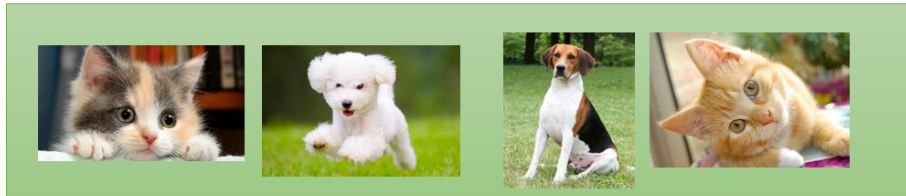
在Semi-supervised Learning的技术里面，这些没有labeled的data，对机器学习也是有帮助的

For example, recognizing cats and dogs

Labelled  
data



Unlabeled  
data



(Images of cats and dogs)

### Transfer Learning(迁移学习)

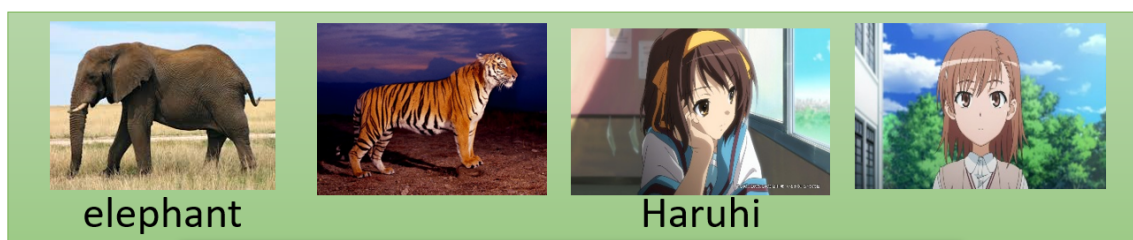
假设一样我们要做猫和狗的分类问题

我们也一样只有少量的有labeled的data; 但是我们现在有大量的不相干的data(不是猫和狗的图片, 而是一些其他不相干的图片), 在这些大量的data里面, 它可能有label也可能没有label

Transfer Learning要解决的问题是, 这一堆不相干的data可以对结果带来什么样的帮助

For example, recognizing cats and dogs

Labelled  
data



elephant

Haruhi

Data not related to the task considered  
(can be either labeled or unlabeled)

### Unsupervised Learning(无监督学习)

区别于supervised learning, unsupervised learning希望机器学到无师自通, 在完全没有任何label的情况下, 机器到底能学到什么样的知识

举例来说, 如果我们给机器看大量的文章, 机器看过大量的文章之后, 它到底能够学到什么事情? 它能不能学会每个词汇的意思?

学会每个词汇的意思可以理解为：我们要找一个function，然后把一个词汇丢进去，机器要输出告诉你这个词是什么意思，也许他用一个向量来表示这个词汇的不同的特性，不同的attribute

又比如，我们带机器去逛动物园，给他看大量的动物的图片，对于unsupervised learning来说，我们的data中只有给function的输入的大量图片，没有任何的输出标注；在这种情况下，机器该怎么学会根据testing data的输入来自己生成新的图片？

- Machine Drawing



## Structured Learning(结构化学习)

在structured Learning里，我们要机器输出的是，一个有结构性的东西

在分类的问题中，机器输出的只是一个选项；在structured类的problem里面，机器要输出的是一个复杂的物件

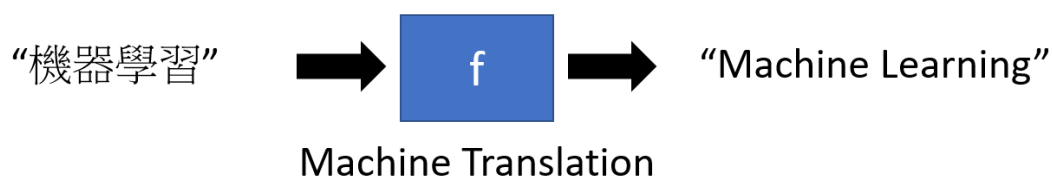
举例来说，在语音识别的情境下，机器的输入是一个声音信号，输出是一个句子；句子是由许多词汇拼凑而成，它是一个有结构性的object

或者说机器翻译、人脸识别(标出不同的人的名称)

比如GAN也是structured Learning的一种方法

# Structured Learning

## - Beyond Classification



## Reinforcement Learning(强化学习)

**Supervised Learning:** 我们会告诉机器正确的答案是什么，其特点是**Learning from teacher**

- 比如训练一个聊天机器人，告诉他如果使用者说了“Hello”，你就说“Hi”；如果使用者说了“Bye bye”，你就说“Good bye”；就好像有一个家教在它的旁边手把手地教他每一件事情

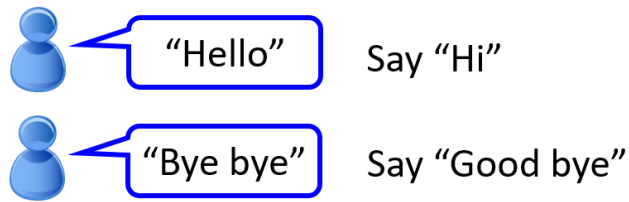
**Reinforcement Learning:** 我们没有告诉机器正确的答案是什么，机器最终得到的只有一个分数，就是它做的好还是不好，但他不知道自己到底哪里做的不好，他也没有正确的答案；很像真实社会中的学习，你没有一个正确的答案，你只知道自己是做得好还是不好。其特点是**Learning from critics**

- 比如训练一个聊天机器人，让它跟客人直接对话；如果客人勃然大怒把电话挂掉了，那机器就学到一件事情，刚才做错了，它不知道自己哪里做错了，必须自己回去反省检讨到底要如何改进，比如一开始不应该打招呼吗？还是中间不能骂脏话之类的

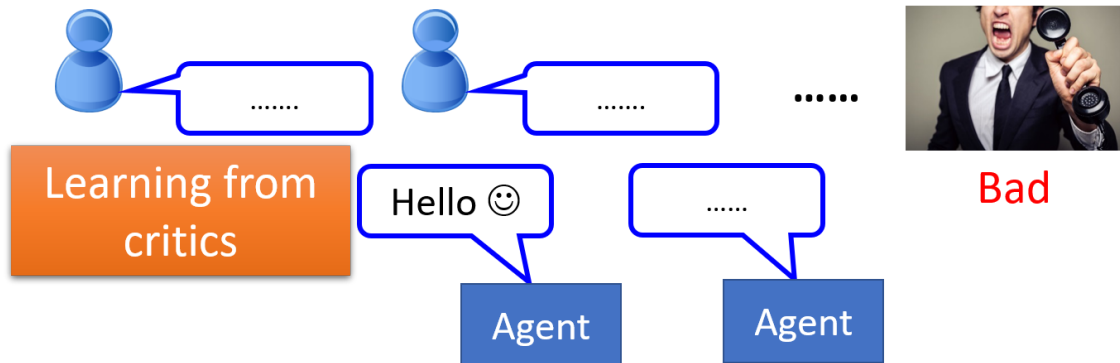
# Supervised v.s. Reinforcement

- Supervised

## Learning from teacher



- Reinforcement



再拿下棋这件事举例，supervised Learning是说看到眼前这个棋盘，告诉机器下一步要走什么位置；而reinforcement Learning是说让机器和对手互弈，下了好几手之后赢了，机器就知道这一局棋下的不错，但是到底哪一步是赢的关键，机器是不知道的，他只知道自己是赢了还是输了

其实Alpha Go是用supervised Learning+reinforcement Learning的方式去学习的，机器先是从棋谱学习，有棋谱就可以做supervised的学习；之后再做reinforcement Learning，机器的对手是另外一台机器，Alpha Go就是和自己下棋，然后不断的进步