机器学习公开课笔记

近来，在看机器学习的公开课，Andrew Ng 大牛[1]在 CMU 的上课视频，网易上有中英字幕的视频[2]。Andrew Ng 是机器学习方面的大牛，署名的论文有 100 多篇，在 LDA 和 DL 方面贡献显著，主要工作在人工智能方面，参与斯坦福自主直升机项目与 STAIR 项目，等等[3]。

最近愈发觉得学习是一件长久的事情，要学习的东西很多，所以看过了什么东西就要理解并记忆就显得很重要，否则就就需要翻来覆去的看，这是高中的学习方法，非大学的学习方法，亦非有效的学习方法。而快速的理解并记忆的方法莫过于时常总结。

公开课上有 20 个视频，我的笔记就是按照视频来划分吧。

机器学习动机与应用

第一个视频主要讲述了课程的安排与内容，举了一些例子来说明机器学习的应用。课程主页及资源还有课程安排等可在 http://cs229.stanford.edu/找到。

内容方面，分为两部分。

第一部分是机器学习的定义；分别讲了 Arthur Samuel 与 Tom Mitchell 的定义。Arthur Samuel[4]被称为“pioneer of artificial intelligence research”，他的西洋棋程序是第一个自学习的程序，他对机器学习的定义是“Field of study that gives computers the ability to learn without being explicitly programmed”。Tom Mitchell 则是《Machine Learning》的作者，他给出了更加形式化的定义，“对于某类任务 T 和性能度量 P，如果一个计算机程序在 T 上以 P 衡量的性能随着经验 E 而自我完善，那么我们称这个程序在经验 E 中学习”[5]。值得一提的是，Tom Mitchell 的 ML 书中的第一章即举了西洋棋的例子，足见 Arthur 的 pioneer 地位。

第二部分是内容大纲，包括监督学习、无监督学习、学习理论、加强学习四个方面。

所谓监督学习（Supervised Learning），就是基于标记数据的学习。本问题举

[1] http://ai.stanford.edu/~ang/index.html

[2] http://v.163.com/special/opencourse/machinelearning.html

[3] http://en.wikipedia.org/wiki/Andrew_Ng

[4] http://infolab.stanford.edu/pub/voy/museum/samuel.html

[5] 《Machine Learning》, Tom Mitchell

了两个例子，一个回归问题（Regression），房屋的价格与面积的关系，是连续数据上的模型构建问题；另一个分类问题（Classification），肿瘤是否恶性与肿瘤大小的关系，是离散数据上的问题。

学习理论（Learning Theory），主要是如何选择算法，如何验证算法的有效性，算法需要的数据量等等。

无监督学习（Unsupervised Learning），与监督学习相对，其数据是没有被标记的。这方面的经典案例是聚类；另外 Ng 举了一个图片的例子，是他的学生的项目，通过对图片聚类来构建 3D 模型；还有一个例子是“鸡尾酒会问题”，当很多人在说话时，如何在嘈杂的背景音中提取目标声音，Ng 举了一个两个人的实例，效果很好，据称一行 matlab 代码就能搞定这个问题，对我震撼颇大。

加强学习（Reinforcement Learning），这个问题的基本概念是回报函数，通过定义好的行为和坏的行为，加上趋好避坏的学习型算法，让程序作出一系列正确的决策。Ng 举的例子是倒飞（上下方向）的直升飞机，爬障碍物的机器狗机器蛇，避开障碍物的机器车等。

线性回归、梯度下降、正规方程组

本节视频是对监督学习的讲解。首先，Ng 以自动驾车的视频举例，说明“汽车对方向的预测是连续值，因而是回归问题”，那么到底什么是回归呢？百度百科中这样解释回归分析，“回归分析(regression analysis)是研究一个变量（被解释变量）关于另一个（些）变量（解释变量）的具体依赖关系的计算方法和理论”。

仍然是以房价与房屋面积的例子引出线性回归问题的解答。首先定义一些符号：

m：训练数据的大小

x：输入变量，是向量

y：输出变量，是实数

(x,y)：一个训练实例

$(x^{(i)},y^{(i)})$：第 i 个训练实例，i 是上标而不是指数

在这里，为了方便说明，又添加了一个变量，问题变为房屋面积和卧室数目与房屋价格的关系。

如果假设训练集中的数据使用线性回归解决的话，假设函数如下：

$$\mathrm{h(x)} = \theta_0 + \theta_1 x_1 + \theta_2 x_2 = \sum_{i=0}^{2} \theta_i x_i = h_\theta(x) \tag{1}$$

其中，$h_\theta(x)$表示以θ为参数。对于一般问题，公式如下：

$$h_\theta(x) = \sum_{i=0}^{n} \theta_i x_i = \theta^T x \tag{2}$$

这里的 x 是向量，n 是 x 的长度。从而，我们可以定义目标函数，即要优化的函数：

$$\mathrm{J(\theta)} = \frac{1}{2}\sum_{i=1}^{m}(h_\theta\left(x^{(i)}\right) - y^{(i)})^2 \tag{3}$$

m 即为样例的数目。我们找出使这个函数最小的参数值，就得到了拟合训练集的最佳参数，至于为什么使用该函数会取得这么好的效果，后面会有解释。

使用梯度下降法（gradient descent）来求参数，更新规则为：

$$\theta_j \coloneqq \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta) \tag{4}$$

这里需要求的只有等式右边的偏导数，:=表示赋值。当只有一个训练样例时，偏导数的计算公式如下：

$$\frac{\partial}{\partial \theta_j} J(\theta) = \frac{\partial}{\partial \theta_j}\frac{1}{2}(h_\theta(x) - y)^2 = (h_\theta(x) - y)\frac{\partial}{\partial \theta_j}(h_\theta(x) - y)$$

$$= (h_\theta(x) - y)\frac{\partial}{\partial \theta_j}\sum_{i=0}^{n}\theta_i x_i = (h_\theta(x) - y)x_j \tag{5}$$

将公式 5 的结果代入到公式 4，得到：

$$\theta_j \coloneqq \theta_j - \alpha(h_\theta(x) - y)x_j \tag{6}$$

当然，公式 6 是针对只有一个训练实例时的情况。这也被称为最小二乘法（LMS，least mean squares），也被称为 Widrow-Hoff 学习规则。

考虑到所有 m 个训练实例，更新规则变为：

$$\theta_j \coloneqq \theta_j - \alpha \sum_{i=0}^{m}(h_\theta\left(x^{(i)}\right) - y^{(i)})\, x_j^{(i)} \tag{7}$$

运用这个规则直到收敛，就是批梯度下降算法（batch gradient descent）。其中，收敛的判断有两种规则，一是判断两次迭代后参数的变化，而是判断两次迭

代后目标函数的变化；规则中的α是学习速率，这个需要在实践中进行调整，其值过小会导致迭代多次才能收敛，其值过大会导致越过最优点发生震荡现象。

梯度下降算法会导致局部极值点的产生，解决这个的方法是随机进行初始化，寻找多个最优点结果，在这些最优点中找最终结果。对于本线性回归问题，不会发生局部极值点的问题，因为实际上，本问题的目标函数是凸二次函数（convex quadratic function）。

对于公式 7 的解法，当数据量较大时，每迭代一次就要遍历全部数据一次，这样会使得运行变成龟速。为了解决这个问题，一般采用如下的方法：

Repeat Until Converge{

For i=1 to m{

$$\theta_j := \theta_j - \alpha\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)x_j^{(i)} \qquad \text{(for every j)}$$

}

}

意为更新参数时，不必遍历整个数据集，只需要一个实例便足够了。该算法可以达到很高的效果，但是会导致遍历次数增多，不能精确收敛到最优值等问题。该方法被称为增量梯度下降（incremental gradient descent）或随机梯度下降（stochastic gradient descent）。

梯度下降算法是求目标函数最优值的一种解法，对于本问题，我们可以直接求出参数值而不用迭代的方法。这种方法称为正规方程法。

首先，定义一些符号和概念。定义梯度符号为∇，则 J 的梯度表示为：

$$\nabla_\theta J = \begin{bmatrix} \frac{\partial J}{\partial \theta_0} & \cdots & \frac{\partial J}{\partial \theta_n} \end{bmatrix}^T \in \mathbb{R}^{n+1} \tag{8}$$

再比如，对于一个函数映射（m*n 的矩阵到实数的映射）：

$$f: \mathbb{R}^{m\times n} \longrightarrow \mathbb{R}$$

则 f 的梯度表示为：

$$\nabla_A f(A) = \begin{bmatrix} \frac{\partial f}{\partial A_{11}} & \cdots & \frac{\partial f}{\partial A_{1n}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f}{\partial A_{m1}} & \cdots & \frac{\partial f}{\partial A_{mn}} \end{bmatrix} \tag{9}$$

其中 A 是 m*n 的矩阵。比如对于一个 2*2 矩阵 A，有函数 f，其定义为：

$$\mathrm{f(A)}=\frac{3}{2}A_{11}+5A_{12}^2+A_{21}A_{22}$$

则得到：

$$\nabla_A f(A)=\begin{bmatrix}\frac{3}{2} & 10A_{12}\\ A_{22} & A_{21}\end{bmatrix}$$

对于一个 n*n 的矩阵，我们再定义矩阵的迹为：

$$\mathrm{trA}=\sum_{\mathrm{i=1}}^{n}A_{ii} \tag{10}$$

把梯度和迹组合在一起，我们可以得到如下性质：

$$\mathrm{trAB}=\mathrm{trBA} \qquad \text{(性质 1)}$$

$$\mathrm{trABC}=\mathrm{trCAB}=\mathrm{trBCA} \qquad \text{(性质 2)}$$

$$\mathrm{trA}=\mathrm{tr}A^T \qquad \text{(性质 3)}$$

$$\mathrm{tr(A+B)}=\mathrm{tr(A)}+\mathrm{tr(B)} \qquad \text{(性质 4)}$$

$$\mathrm{traA}=\mathrm{a}*\mathrm{trA} \qquad \text{(性质 5)}$$

$$\mathrm{tra}=\mathrm{a} \qquad \text{(性质 6)}$$

其中，a 是一个实数，A、B、C 均为 n*n 的矩阵。

$$\nabla_A\mathrm{trAB}=B^T \qquad \text{(性质 7)}$$

$$\nabla_{A^T}\mathrm{f(A)}=(\nabla_A f(A))^T \qquad \text{(性质 8)}$$

$$\nabla_A trABA^TC=CAB+C^TAB^T \qquad \text{(性质 9)}$$

$$\nabla_A|A|=|A|(A^{-1})^T \qquad \text{(性质 10)}$$

对于性质 7，要求 AB 是 n*n 矩阵；对于性质 9，要求 ABA^TC 是 n*n 矩阵；对于性质 10，要求矩阵 A 可逆，即 A 为非奇异矩阵。

说完这些定义和性质之后，我们再看看如何用矩阵表示目标函数：

训练数据集合实际上是 m*n 的矩阵，m 是样本个数，n 是每个样本的维度。对于每个样本的目标值，按照顺序排列为 m*1 的向量。因而，数据的矩阵表示如下：

$$\mathrm{X}=\begin{bmatrix}(x^{(1)})^T\\ \vdots\\ (x^{(m)})^T\end{bmatrix} \tag{11}$$

$$\mathrm{Y}=[y^{(1)} \quad \dots \quad y^{(m)}]^T \tag{12}$$

那么，我们可以得到：

$$\mathrm{X\theta - Y} = \begin{bmatrix} \left(x^{(1)}\right)^T\theta \\ \vdots \\ \left(x^{(m)}\right)^T\theta \end{bmatrix} - \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(m)} \end{bmatrix} = \begin{bmatrix} h_\theta\left(x^{(1)}\right) - y^{(1)} \\ \vdots \\ h_\theta\left(x^{(m)}\right) - y^{m} \end{bmatrix} \tag{13}$$

所以，我们得到目标函数 J 的向量表达：

$$\mathrm{J(\theta)} = \frac{1}{2}(X\theta - Y)^T(X\theta - Y) = \frac{1}{2}\sum_{i=1}^{m}(h_\theta\left(x^{(i)}\right) - y^{(i)})^2 \tag{14}$$

所以，我们可以得到计算 J 的梯度的公式推导：

$$\begin{aligned} \nabla_\theta J(\theta) &= \frac{1}{2}\nabla_\theta(\theta^T X^T - Y^T)(X\theta - Y) = \frac{1}{2}\nabla_\theta(\theta^T X^T X\theta - Y^T X\theta - \theta^T X^T Y + Y^T Y) \\ &= \frac{1}{2}\nabla_\theta \mathrm{tr}(\theta^T X^T X\theta - Y^T X\theta - \theta^T X^T Y + Y^T Y) \\ &= \frac{1}{2}[\nabla_\theta \mathrm{tr}(\theta^T X^T X\theta) - \nabla_\theta tr(Y^T X\theta) - \nabla_\theta(\theta^T X^T Y)] \\ &= \frac{1}{2}\nabla_\theta \mathrm{tr}(\theta^T X^T X\theta) - \nabla_\theta tr(Y^T X\theta) \\ &= \frac{1}{2}\nabla_\theta \mathrm{tr}(\theta\theta^T X^T X) - \nabla_\theta tr(Y^T X\theta) \\ &= \frac{1}{2}\nabla_\theta \mathrm{tr}(\theta I\theta^T X^T X) - \nabla_\theta tr(Y^T X\theta) = X^T X\theta - \nabla_\theta tr(Y^T X\theta) \\ &= X^T X\theta - X^T Y \end{aligned} \tag{15}$$

推导说明，公式 15 中，第一行展开；第二行应用性质 6；第三行应用性质 4，且 Y^TY 是常数；第四行应用性质 3；第五行应用性质 1；第六行的 I 是单位矩阵，并应用性质 9 和性质 7。

得到结果后，我们令导数为 0，得到：

$$X^T X\theta = X^T Y \Rightarrow \theta = (X^T X)^{-1}X^T Y \tag{16}$$

从而，我们求出了参数。这种方法就被称为正规方程组。

附录-重要性质的证明

性质 1 的证明：

假设 A 是 n*m 矩阵，B 是 m*n 矩阵，则：

$$\mathrm{trAB} = \sum_{i=1}^{n}\sum_{j=1}^{m} A_{ij}B_{ji} = \sum_{j=1}^{m}\sum_{i=1}^{n} B_{ji}A_{ij} = trBA$$

性质 7 的证明：

假设 A 是 n*m 矩阵，B 是 m*n 矩阵，则：

$$\nabla_A \mathrm{trAB} = \nabla_A \sum_{i=1}^{n} \sum_{j=1}^{m} A_{ij} B_{ji} = \begin{bmatrix} B_{11} & \dots & B_{m1} \\ \vdots & \ddots & \vdots \\ B_{1n} & \dots & Bmn \end{bmatrix} = B^T$$

性质 9 的证明：

$$\begin{aligned} \nabla_A trABA^T C &= \nabla_A trf(\mathrm{A}) A^T C = \nabla_* trf(*) A^T C + \nabla_* trf(A) *^T C \\ &= (A^T C)^T \nabla_A trf(A) + \nabla_* trf(A) *^T C = (A^T C)^T B^T + \nabla_* trf(A) *^T C \\ &= C^T A B^T + \nabla_* trf(A) *^T C = C^T A B^T + (\nabla_{*^T} trf(A) *^T C)^T \\ &= C^T A B^T + ((Cf(A))^T)^T = C^T A B^T + Cf(A) = CAB + C^T A B^T \end{aligned}$$

我们按照=对每个步骤进行讲解，第一步将 AB 作为一个函数，第二步按照乘法求导原则；第三步应用性质 7 和函数求导原则，第四步应用性质 7；第五步是两个相乘矩阵的转置；第六步应用性质 8，第九步应用性质 7，第十步是矩阵的转置的转置就是本身。

性质 10 的证明：

性质 10 的等号左右两端其实都是矩阵 A 的伴随矩阵。