Machine or Deep Learning

冯迁

2017 年 9 月 25 日

目录

前言

本文档是我的一个学习过程，学习时间开始于 2016 年末，打算将其写出来于 2017 年中。本着不是科班出生，很多方面都不太扎实，包括程序设计，对算法的性能分析，以及囿于匮乏的经验对市场的状况认识还不太合格。将其写出来一来梳理自己，二来也可以利用自己的数学优势以及先行几步减少部分初学者学习上的困惑，最后也能知道自己的不足之处以及理解有误的地方。

首先以词条人工智能开始维基百科之旅: 人工智能。

AI 的核心问题包括推理, 知识, 规划, 学习, 交流, 感知, 移动和操作物体等。目前比较流行的方法包括统计方法，计算智能和传统意义的 AI。大量应用的人工智能包括搜索和数学优化, 逻辑推演。

接着自由选择进入机器学习页面：机器学习是人工智能的一个分支。在 30 多年的发展中，已成为一门多领域的交叉学科，涉及概率论，统计学，逼近论，凸分析，计算复杂性理论等多门学科。其算法主要实现从数据中自动分析获得规律，并利用规律对未知数据进行预测。主要分为四类：监督学习 (回归分析和统计分类)，无监督学习 (聚类)，半监督学习和增强学习 (基于环境行动，以取得最大化的预期利益)。

具体的机器学习算法:

1. 构造间隔理论分布: 聚类分析和模式识别
   (a) 人工神经网络
   (b) 决策树
   (c) 感知器
   (d) 支持向量机
   (e) 集成学习 AdaBoost
   (f) 降维与度量学习
   (g) 聚类
   (h) 贝叶斯分类器
2. 构造条件概率: 回归分析和统计分类
   (a) 高斯过程回归

   (b) 线性判别分析
   (c) 最近邻法
   (d) 径向基函数核
3. 通过再生模型构造概率密度函数:
   (a) 最大期望算法
   (b) 概率图模型: 贝叶斯网和 Markov 随机场
   (c) Generative Topographic Mapping
4. 近似推断技术:
   (a) 马尔科夫链
   (b) 蒙特卡洛方法
   (c) 变分法
5. 最优化: 大多数以上方法，直接或间接使用最优化算法。

以上是中文版的粗略分类，关于更详尽的分类可参考:Outline_of_ml. 假设我们已经浏览了此页面上所有内容，包括其链接内容。这样我们对人工智能整体有了个一个简单的认识: 算法庞杂，理论繁多，近几年活跃度很大。一方面方法可以很简单，比如 KNN,PCA,BP..., 另一方面也可以很复杂, 比如变分法，最优传输等。以现有的目光来看，很多基本的问题还需要学者们去解答。包括一个统一的理论框架，对深度黑箱的解释等。另外如此多的算法，在面对实际问题时，往往局限于模型的理想化，以及问题的类型，需要根据实际选择并改装。这也就促使我们选择自己感兴趣的分支，并掌握所需算法的精髓。似乎一下就变成了生命能承受之重了，... 似轻非轻..., 道路很长。

* 选择：CV
* 工具:TensorFlow,Pytorch,sklearn,Numpy,Opencv...
* 理论: 相似与度量学习, 凸优化, 概率统计, 最优传输
* 问题: 怎样学到最少的东西, 解决更多的问题?

关于工具的问题，纯粹学习工具本身，是一个挺无聊的过程。带着问题或者项目学习，在一定程度上能减少一些莫名的痛苦。学习工具之前要有一定的理论基础，单单学习框架是无意义的，假设不从长远来看的话，那就没问题了。理论学习若能结合具体问题进行比较，在面向工程时，感觉会过渡得更加自然，反过来，也可能会给理论研究注入新鲜东西。

第一章　数据处理

数据处理是机器学习的关键一步，不论是在训练前还是在训练当中，都存在对数据的各种处理。训练前，数据收集，数据质量评判，数据表示，数据特征抽取，数据降维，数据归一化，训练中，数据批归一化，数据重构...。

接下来假设我们已经拥有了比较完整均匀的数据。

数据预处理

Multivariate Statistical Analysis

多变量分析主要用于分析拥有多个变数的资料，探讨资料彼此之间的关联性或是厘清资料的结构，而有别于传统统计方法所着重的参数估计以及假设检定。常见的分析方法有 PCA,CCA,MDS,SEM 等。

PCA

主成分分析:PCA

PCA 分析计算的核心就是矩阵的奇异值分解，奇异值分解属于谱定理的一小部分，数学上谱定理是个很精彩的定理，但这里我们只能介绍 SVD。

假设 M 是一个 $m \times n$ 阶矩阵，其中的元素全部属于域 K，也就是实数域或复数域。如此则存在一个分解使得

$$M = U\Sigma V^*$$

其中 U 是 $m \times m$ 阶酉矩阵； 是 $m \times n$ 阶非负实数对角矩阵；而 V^*，即 V 的共轭转置，是 $n \times n$ 阶酉矩阵。这样的分解就称作 M 的奇异值分解。Σ 对角线上的元素 Σ_i, i 即为 M 的奇异值。

对于 PCA，我们要分解的就是数据的经验协方差阵，因为协方差阵是对称的，在线性代数里，我们知道每个正规矩阵都可以被一组特征向量对角化。即:

$$M = U\Sigma U^*.$$

实际上对于对称矩阵我们还可以做到

$$M = U\Sigma U^{-1}.$$

意义自明，U 的第 i 列表示 M 的第 i 个特征值对应的特征向量 (这里假设特征值是按顺序排列了)。现在我们需要多大比例的保持方差极大信息，选择一定数量的特征值及其特征向量即可。

“PCA 具有保持子空间拥有最大方差的最优正交变换的特性。然而，当与离散余弦变换相比时，它需要更大的计算需求代价。非线性降维技术相对于 PCA 来说则需要更高的计算要求。”

CCA

典型相关分析:CCA

CCA 寻找两个具有相互关系的随机变量的特征的线性组合，使其表示成的新特征之间具有最大的相关性。可以说是一种保持特征相关性的特征重构。具有降维的作用。其计算过程和 PCA 差不多，首先根据两随机向量 X, Y 计算其互协方差矩阵，然后求解向量 a, b 使得 $\rho = corr(a^{'}X, b^{'}Y)$ 最大，其中 $U = a^{'}X, V = b^{'}Y$ 是第一对典型变量，然后依次求得不相关的典型变量对。而这个问题最后被转化成一个求由协方差阵组合成的某对称矩阵的特征向量问题。

相关代码参考:PyCCA

Multidimensional scaling

多维标度:MDS

代码参考:PyMDS

AutoEncoder

AutoEncoder

从维基上我们看到，自编码是一种无监督式的数据重构方法，其理论比较简单，相应的利用 Tensorflow 或者 Pytorch 实现它也很简单，其扩展方式很多。

现在我们来看看采用概率图模型的自编码方法:Variational autoencoder。这里算了提前进入机器学习概率这一板块了，讲道理，这块是我的弱项。算是提前在这里熟悉概率的一些基本的东西吧。

通俗 VAE此讲解作为第一次阅读，以及后面的彩蛋，都不错。结合入门 VAE该文章小错误比较多，作为入门理解，还是不错的，且不可关注过多细节。入门 2AVE

基础阅读材料:TutorialVAE以及简短的变分推理 Blei, David M. ”Variational Inference.” Lecture from Princeton。

传统图像预处理

Pycode

第二章　机器学习

本章包含了机器学习的经典算法. 经典的机器学习算法有很多库都已经实现，我们没必要所有都去造轮子，我的选择是理解其数学部分，使用现有的库 sklearn，并在实践中分析理论与和实际的差距。假设我们对理论和库的调用都不太熟悉，实际上 sklearn 的 document 本身就是一个很好的学习地方，那里包含了算法的相关参考文献。后面的章节我们首先以这种方式来学习经典机器学习。

开始页面：sklearn-user-guide

4.6 核近似

4.7 Pairwise metrics, Affinities and Kernels

4.8 变换目标值

5 数据导入

6 大数据

以上是 sklearn 指导文档首页的部分目录，现在我们随机选择一些东西学习，比如我这里选择了接下来的四节内容。这只是一个初步的学习，剩下的就是在实践中不断的深化理解实际和理论上的差别，然后再反过来思考理论上的问题。这部分的理论相对简单，但这些优化方法却是人工智能的基础。

Regression

进入一般线性模型，玲琅满目，眼花缭乱。

2.0.1 Logistic regression

逻辑回归是一个二分类概率模型，其很容易扩展到多元情形，它将特征向量映射为一个概率向量，其每个分量表示特征属于其对应标签的概率。模型可表示如下：

$$P^{LR}(W) = C\sum_{i=1}^{n} log(1 + e^{-y_i W^T \mathrm{x_i}}) + 1/2W^T W \tag{2.1}$$

其中 $\{\mathrm{x_i}, y_i\}_{i=1}^{n}$ 表示数据以及其标签，$\mathrm{x_i} \in R^m, y_i \in \{1, -1\}.C > 0, W$ 是要学习的参数。

给定数据及其标签，我们可以用如下公式来表示条件概率。

$$P_W(y = \pm 1|\mathrm{x}) = \frac{1}{1 + e^{-yW^T\mathrm{x}}} \tag{2.2}$$

根据极大似然原理，我们很容易由 (2.2) 得到 (2.1)，如果我们去掉 (2.1) 的 $1/2W^TW$，这个多余的东西其实就是正则项，用来限制参数 W，防止过拟合的技巧。这个后面详说。在实际情况中，往往需要很多额外起脚来使模型更加实用。

现在的问题是如何得到模型参数 W, C?

答案：Coordinate descent approch, quasi-Newton method,iterative scaling method, exponetiatel gradient ... 如果你学过数值分析的话，你会觉得很多似曾相识，如果你学过凸分析的话，你会觉得很亲切，随着学习的深入，我们会逐渐建立更清晰的理论框架。现在不妨将视角转向 SVM。

Linear regression Logisticregression

支持向量机

SVM，一个二分类线性模型，简单的说，就是我们高中遇见的线性规划问题的推广。我们知道直线 $y = kx + b$ 将平面 xy 分成两部分，其实也就是两类，一类在“上面”，一类在“下面”。现在我们的情况只是在维度上增加了，也就是寻找一个超平面 $y = Wx + b$ 能将数据分类出来。模型可表示如下：

$$P^{SVM}(W) = C\sum_{i=1}^{n} max(1 - y_i W^T \mathrm{x_i}, 0) + 1/2 W^T W \tag{2.3}$$

很明显超平面是依赖训练数据的。从文档上看，该分类其的好处有：

- 高维空间上比较高效。
- 对特征维度大于样本量时，仍然有效 (大过多时，需要适当的选择核函数和正则项)。
- 用部分数据来得到决策函数，也即支持向量，能减少内存。

上面只是最简单的情形，多分类，多元回归呢?

我们先来看看文档里的情况，对于多分类，从文档里我们了解到两种方法：SVC 的“one-against-one”，n 类标签构建 $\binom{n}{2}$ 个分类器;LinearSVC 的“one-vs-the-rest”，训练 n 个模型。

对于回归问题，其对应的模块名叫 SVR。自行查看即可。现在的问题是：怎么从分类模型过度到回归模型呢？完整想出来，似乎还是有点难度，但是我们看到网页所给参看文献SVR，好了，又到了真正学习的时候了。

此段讲理论，以及代码分析。

然而实际上我们对多分类问题的处理方式还是失望的，我们并不想重复二分类模型，针对这个问题，表明我们该看看 1.12 节Multiclass-Multilabel了。

Maximum Entropy

蹦，问题又来了，所谓超平面，直观上看，毕竟是个“平”的。实际问题中，数据往往需要用一个弯曲的面才能将其较好的分类出来，这时怎么办呢？自然的我们有两种想法，一种直接把超曲面算出来，但这样不太好，考虑到曲面的表示方式，能控制的范围太小了，而实际变化范围太大; 另一种方法就是保持超平面不变，直接映射输入数据，使其能被平面分割。这就是所谓的核技巧。

核技巧讲完了。

现在我们来看看以上模型该如何学习参数。

随机选择一个 Reference:Dual coordinate descent for LR and ME

Adaboost

高斯混合模型

第三章　理论知识

列表

第四章　深度学习

首先推荐的书籍:neuralnetworks
DL

神经网络

卷积神经网络

卷积神经网络都被说烂了。

RNN

GAN

各种变体。

DQN

A3C

DDPG

第五章　传统智能

包含了传统智能算法.

AG

这里主要以实际问题为主, 对 ag 进行一个梳理.

第六章　论文阅读

其实也没什么。

GAN