我的机器学习/数据挖掘的书单

李航的《统计学习方法》

这本书开篇第一章写得特别好，各个模型的算法推导也比较全，基本涵盖了比较经典的判别模型和生成模型。

《机器学习实战》

这本书代码和应用特别多，了解python用法和机器学习算法的代码实现非常方便。

项亮的《推荐系统实践》

这本书个人感觉偏理论一点，伪代码看着都实现不了，不过关于推荐系统的整个架构讲得挺清楚的，各种推荐引擎的算法设计，基于内容的，基于邻域，基于便签，基于地点和时间（context）等等，最后将这些推荐引擎融合，排序，过滤等等就构成了推荐系统，从各种推荐指标中比较了推荐引擎的优劣。最后还讲了如何预测评分。

《spark大数据处理技术》

这本书算是我的spark启蒙书籍，尤其是介绍spark的核心概念，比如RDD，运行架构和任务架构都讲得比较好。还有不少例子和API的讲解

吴岸城的《神经网络与深度学习》

这本书个人感觉挺好的，极贴近现实，从实际生活讲到了（第一个）机器学习模型（感知机），从浅层学习讲到了深层学习，之后介绍了自编码，RBM,DBN,CNN,RNN.以及后面的AlphaGo,迁移学习，概率图等。有时间会把书中自己感觉重要的做一下笔记

Hinton(辛顿)是反向传播算法和对比散度算法的发明人之一

LeCun(勒丘恩)对卷积神经网络(CNN)改进再应用

Jürgen Schmidhuber是LSTM的发明人，梯度消失的贡献人和递归结构的推动者

人脑的工作原理：

大脑能做的每件事，能记住的每件事，都是很多细胞连接起来以后发挥的功能。大脑皮层的每一块区域负责处理图像的一方面，比如皮毛、尾巴、面部特征和动作等，这些综合起来形成了一个完整的形象，

神经元的工作原理：细胞轴突是一条长长的纤维，它把细胞体的输出信息传到到其他神经元。树突用来接收其他神经元的输入信号。

神经元具有两个最主要的特性，即兴奋性和传导性。

神经元可分为传入神经元、中间神经元、传出神经元

神经元之间相互连接的突触随着动作电位脉冲激励方式与强度的变化，其传递电位的作用可增加或减弱。

人工神经网络：

常用的传递函数：机器学习中常用的传递函数总结

最简单的神经网络：机器学习-感知机

网络的输出根据网络的连接方式，权重和激励函数的不同而不同。

由输入的信号源、隐层及输出组成的层叫做多层神经网络，多层神经网络的一个重要特征是上一层输出只能是下一层输入，不可跨层连接。

输入层和输出层一般按照数据集和需求确定，隐层神经元个数需大于输入层。

深度学习：

人的视觉系统的信息处理是分级的。从视网膜出发，经过低级的V1区提取边缘特征，到V2区的基本形状或目标的局部，再到高层的整个目标，以及到更高层进行分类判断等

神经-中枢-大脑的工作过程或许是一个不断迭代、不断抽象的过程

任何事物都可以划分成粒度合适的浅层特征，而这个浅层特征一般就是我们的第二层输入。

结构性特征具有明显的层级概念，从较小粒度划分，再用划分的基本特征组成上层特征，以此类推，可以展现特征的结

构性。

传统神经网络一半只有两到三层的神经网络，参数和计算单元有限，对复杂函数的表示能力有限，学习能力也有限；而深度学习具有五层以上的神经网络，并且引入了更有效的算法。

误差反向传播算法（BP）的基本思想：（1）信号正向传播，若输出层的实际输出和期望的输出不符，则进入误差的反向传播阶段（2）误差反向传播：从输出到输入，将误差分摊给各层的所有单元，从而获得各层单元的误差信号，此信号作为修正各单元权值的依据。

练习：70行Java代码实现BP神经网络

BP算法的问题：（1）梯度越来越稀疏：从上往下，误差矫正信号越来越小。（2）收敛到局部最小值：尤其是从最优区域开始的时候（3）大部分数据是没有标记的，而BP算法是有监督算法。

自动学习特征的方法统称为深度学习。深度学习首先利用无监督学习对每一层进行逐层训练去学习特征；每次单独训练一层，并将训练结果作为更高一层的输入；然后到最上层改用监督学习从上到下进行微调取学习模型。

深度学习的常用算法：

自动编码器（AutoEncoder）：

稀疏编码：理论上为了确保信号重建的准确度，需要令所采用的取样矩阵各行列之间相干性尽量低，且需矩阵元素取值随机性尽量调高。目前被提出的简化取样矩阵主要包括两种：结构化采样矩阵与数值简化采样矩阵。如果在自动编码器的基础上加L1的规则限制，我们就可以得到SparseAutoEncoder法。

栈式自编码器：栈式自编码器是一个有多层稀疏自编码器组成的神经网络。对于一个n层栈式自编码器的编码过程就是，按照从前向后的顺序执行每一层自编码器的编码步骤：

（1）通过反向传播的方法，利用所有数据对第一层的自编码器进行训练

（2）移除前面自编码器的输出层，用另一个自编码器替代，再用反向传播进行训练，这个步骤称为预训练

（3）在网络的最后一层后面再加上一个或多个连接层。整个网络可以被看作是一个多层感知机，并使用反向传播算法进行训练，这步也称之为微调

有限制玻尔兹曼机（Restricted Boltzmann Machine）

玻尔兹曼机是源于物理学的能量函数的建模方法，能够描述变量的高层互作用。

有限制玻尔兹曼机（Restricted Boltzmann Machine），以下简称RBM，包括隐层和偏置层，其中可见层和隐层的链

接是方向不定的（值可以双向传播，也就是隐层->可见层和可见层->隐层）和完全连接的。

RBM网络是一种随机网络，一个随机网络（比如BNN）包括两个主要结构：概率分布函数（联合概率分布，边缘概率分布，条件概率分布），能量函数。其中能量函数（统计力学）是描述整个系统状态的一个测度。系统越有序或者概率分布越集中，系统的能量越小。反之，系统越无序或者概率分布越趋于均匀分布，则系统的能量越大。能量函数的最小值，就是我们要找的系统最稳定的状态。

RBM网络有几个参数，一个是可视化层与隐藏层之间的权重矩阵W,一个是可视节点的偏移量b,一个是隐藏节点的偏移量c,这几个参数决定了一个n维（可视化层结点数）样本如何编码成一个m维（隐层结点数）样本

RBM的用途：

（1）对数据进行编码（降维）

（2）得到权重矩阵和偏移量，供BP神经网络初始化训练

（3）可以估计联合概率（生成模型）

（4）可以计算条件概率（判别模型）

深度信念网络（Deep Belief Network）

杰弗里.辛顿在2006年提出了深度信念网络（Deep

Belief Network）,DBN是由多个RBM层构成的。

DBN在训练模型的过程中主要分为以下两步：

（1）分别单独无监督的训练每一层RBM网络，确保特征向量映射到不同特征空间是，都尽可能多地保留特征信息（预处理）。

（2）在DBN的最后一层设置BP网络，接收RBM的输出特征向量作为它的输入特征向量，有监督第训练实体关系分类器（微调）。

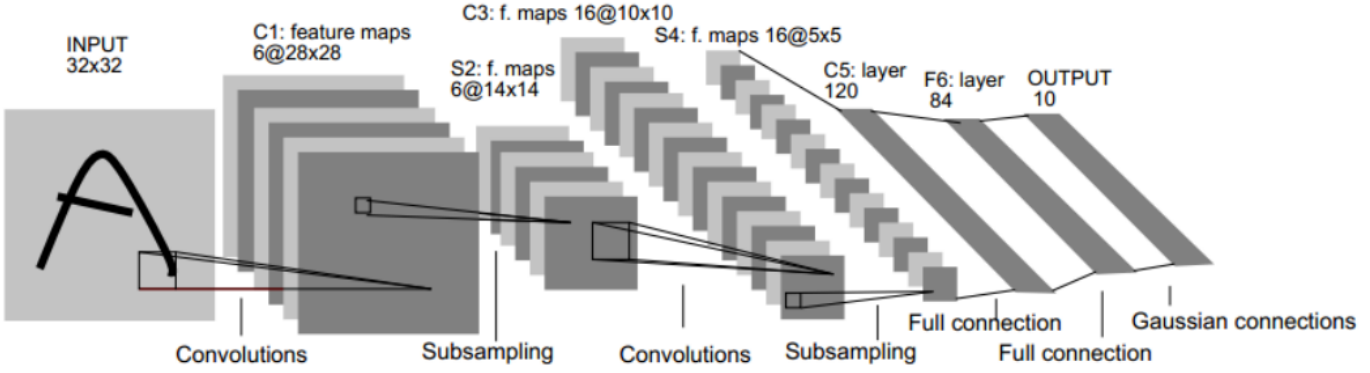
卷积神经网络（Convolutional Neural Network）：

CNN最初诞生是为了识别二维图形；现在，卷积神经网络也可应用于人脸识别，文字识别等方向。

卷积网络的每一层都将三维输入（长，宽，深度/颜色）转换成三维输出值，这个转换过程是由一个图像过滤器（或者称为一个有关权重的方阵）来完成的，这个过程其实就是特征提取，而这个图像过滤器是需要人为设定的。

卷积神经网络的各个组成部分：

如下这是一个**LeNet-5网络**（用于图像内容识别），便于我们更直观的理解CNN的各个层次结构：



卷积层（Convolutional Layer）:对输入数据应用若干过滤器，一个输入参数被用来做了很多类型的特征提取（比如**LeNet-5网络**的第一卷积层(C1)使用了6个5*5的过滤器），对图像应用一个过滤器之后得到的结果被称为特征图谱（FeatureMap）,特征图谱的数目和过滤器的数目相等（C1层是6个FeatureMaps,每个FM大小为：（32-6）*（32-6））。从直觉上讲，如果将一个权重分布在整个图像上，那么这个特征就和位置无关了，同时多个过滤器可以分别探测出不同的特征。比如，像青蛙捕虫一样，有的脑部区域负责提取(判定蚊子的）图像特征，有的脑部区域负责提取物体的运动轨迹；一旦这两个特征提取出来，就向特征重合的地方吐舌头。

子采样层（Subsamping Layer）:又叫池化（Pooling）层,缩减输入数据的规模（比如S2层应用一个2*2的采样区域后，每个FM大小为：（28/2）*(28/2)），常用的子采样方法有最大值合并，平均值合并和随机合并。

全连接层（Full Connection Layer）:F6层与C5层全相连，F6层计算输入向量和**权重向量**之间的点积，再加上一个偏置，然后将其传给sigmoid函数产生单元i的一个状态

输出层：输出层由欧式径向基函数（Euclidean Radial Basis Function）单元组层。每类一个单元，换句话说，每个输出ERBF单元计算输入向量和函数向量之间的欧氏距离。一个ERBF输出可以被理解为输入模式和与EFBF相关联类的一个模型的匹配程度的惩罚项。

卷积神经网络的训练过程通过改进的反向传播实现，将子采样层作为考虑的因素并基于所有值来更新卷积过滤器的权重。实操中也可以设定前向反馈一些数据，以便调整。

附：卷积神经网络 LeNet-5各层参数详解

理解卷积神经网络的一个在线例子：http://cs.stanford.edu/people/karpathy/convnetjs/demo/mnist.html

深入：CNN特性/优点待续。。。

张重生的《深度学习原理与实践》

以Caffe作为工具讲解，这本书代码太多，不过例子也挺多，因为本人涉及DL较少，所以感觉收获较少，同时很多扩展阅读和链接也没读。BP和CNN理论，都举了一个例子。之后的章节就都是实验了，各种贴代码的套路。最后作者也说DL很多缺乏理论解释，大部分处理监督，调参困难，软硬件的门槛较高等。

吴军的《数学之美》

这本书涉及到语言检索，语言处理，搜索引擎，机器学习算法和数据结构算法很多范围，还有几章是介绍名人或者思想的；每一章都不太多，我每天做地铁就可以读两章。而且有的还从网上找了例子，尤其是信息度量和信息指纹，以及输入法的例子

下面贴几个我练手的例子，加深书中内容的理解：

处理海量数据查找的BloomFilter

中文分词算法（HMM,维特比算法）

衡量query和网页的相关性/主题模型

信息指纹/判断字符串数组内容相同——google的simHash

《python自然语言处理》