

# 原 Deep Learning (深度学习) 学习笔记整理系列之 (四)

分类: 机器学习 Deep Learning 机器视觉

2013-04-09 00:46 1186人阅读

## Deep Learning (深度学习) 学习笔记整理系列

zouxy09@qq.com

<http://blog.csdn.net/zouxy09>

作者: **Zouxy**

**version 1.0 2013-04-08**

声明:

- 1) 该Deep Learning的学习系列是整理自网上很大牛和机器学习专家所无私奉献的资料的。具体引用的资料请看参考文献。具体的版本声明也参考原文献。
- 2) 本文仅供学术交流, 非商用。所以每一部分具体的参考资料并没有详细对应。如果某部分不小心侵犯了大家的利益, 还望海涵, 并联系博主删除。
- 3) 本人才疏学浅, 整理总结的时候难免出错, 还望各位前辈不吝指正, 谢谢。
- 4) 阅读本文需要机器学习、计算机视觉、神经网络等等基础(如果没有也没关系了, 没有就看看, 能不能看懂, 呵呵)。
- 5) 此属于第一版本, 若有错误, 还需继续修正与增删。还望大家多多指点。大家都共享一点点, 一起为祖国科研的推进添砖加瓦(呵呵, 好高尚的目标啊)。请联系: [zouxy09@qq.com](mailto:zouxy09@qq.com)

目录:

一、概述

二、背景

三、人脑视觉机理

四、关于特征

4.1、特征表示的粒度

4.2、初级(浅层)特征表示

4.3、结构性特征表示

#### 4.4、需要有多少个特征？

### 五、Deep Learning的基本思想

### 六、浅层学习（Shallow Learning）和深度学习（Deep Learning）

### 七、Deep learning与Neural Network

### 八、Deep learning训练过程

#### 8.1、传统神经网络的训练方法

#### 8.2、deep learning训练过程

### 九、Deep Learning的常用模型或者方法

#### 9.1、AutoEncoder自动编码器

#### 9.2、Sparse Coding稀疏编码

#### 9.3、Restricted Boltzmann Machine(RBM)限制波尔兹曼机

#### 9.4、Deep Belief Networks深信度网络

#### 9.5、Convolutional Neural Networks卷积神经网络

### 十、总结与展望

### 十一、参考文献和Deep Learning学习资源

接上

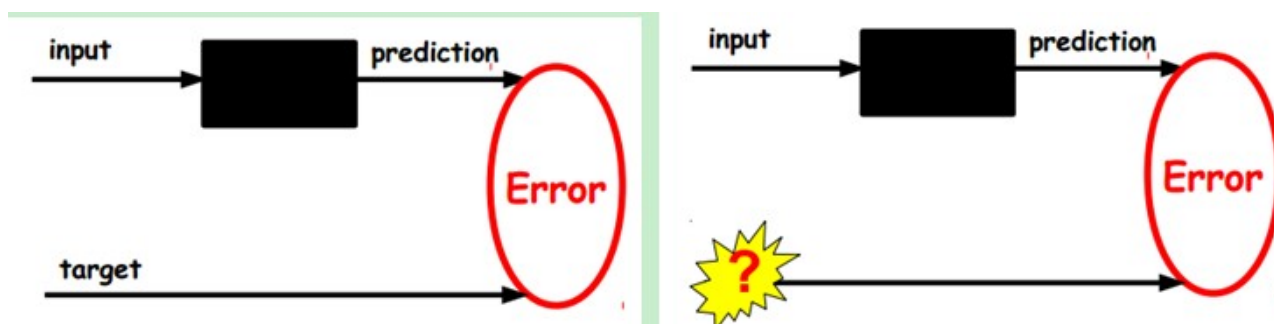
### 九、Deep Learning的常用模型或者方法

#### 9.1、AutoEncoder自动编码器

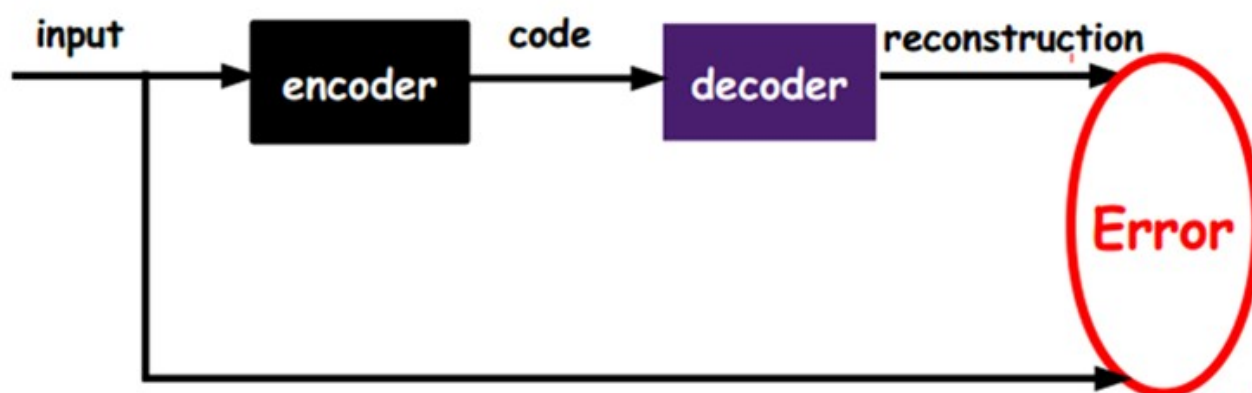
Deep Learning最简单的一种方法是利用人工神经网络的特点，人工神经网络（ANN）本身就是具有层次结构的系统，如果给定一个神经网络，我们假设其输出与输入是相同的，然后训练调整其参数，得到每一层中的权重。自然地，我们就得到了输入I的几种不同表示（每一层代表一种表示），这些表示就是特征。自动编码器就是一种尽可能复现输入信号的神经网络。为了实现这种复现，自动编码器就必须捕捉可以代表输入数据的最重要的因素，就像PCA那样，找到可以代表原信息的主要成分。

具体过程简单的说明如下：

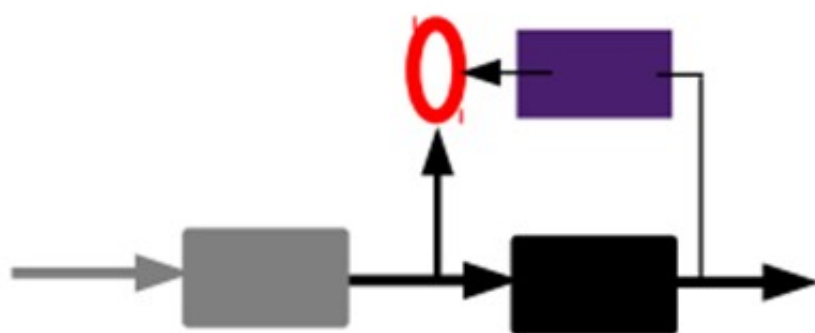
1) 给定无标签数据，用非监督学习学习特征：



在我们之前的神经网络中，如第一个图，我们输入的样本是有标签的，即（input, target），这样我们根据当前输出和target（label）之间的差去改变前面各层的参数，直到收敛。但现在我们只有无标签数据，也就是右边的图。那么这个误差怎么得到呢？

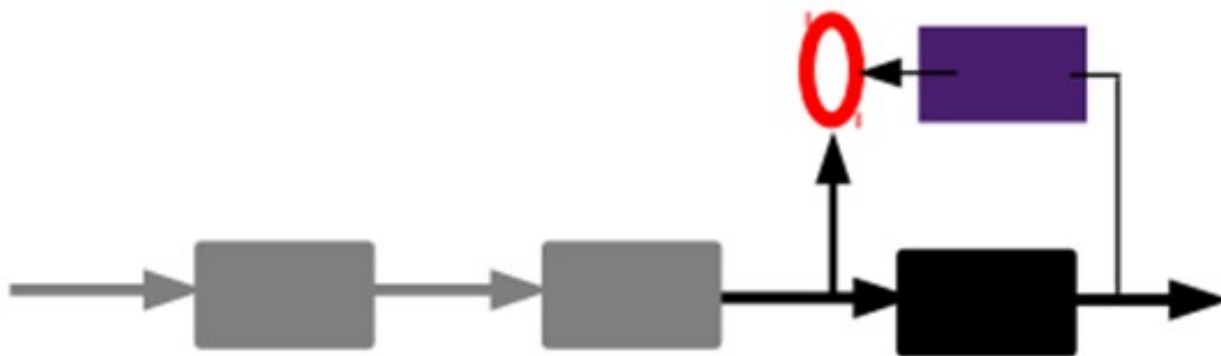


如上图，我们将input输入一个encoder编码器，就会得到一个code，这个code也就是输入的一个表示，那么我们怎么知道这个code表示的就是input呢？我们加一个decoder解码器，这时候decoder就会输出一个信息，那么如果输出的这个信息和一开始的输入信号input是很像的（理想情况下就是一样的），那很明显，我们就有理由相信这个code是靠谱的。所以，我们就通过调整encoder和decoder的参数，使得重构误差最小，这时候我们就得到了输入input信号的第一个表示了，也就是编码code了。因为是无标签数据，所以误差的来源就是直接重构后与原输入相比得到。



2) 通过编码器产生特征，然后训练下一层。这样逐层训练：

那上面我们就得到第一层的**code**，我们的重构误差最小让我们相信这个**code**就是原输入信号的良好表达了，或者牵强点说，它和原信号是一模一样的（表达不一样，反映的是一个东西）。那第二层和第一层的训练方式就没有差别了，我们将第一层输出的**code**当成第二层的输入信号，同样最小化重构误差，就会得到第二层的参数，并且得到第二层输入的**code**，也就是原输入信息的第二个表达了。其他层就同样的方法炮制就行了（训练这一层，前面层的参数都是固定的，并且他们的**decoder**已经没用了，都不需要了）。



### 3) 有监督微调:

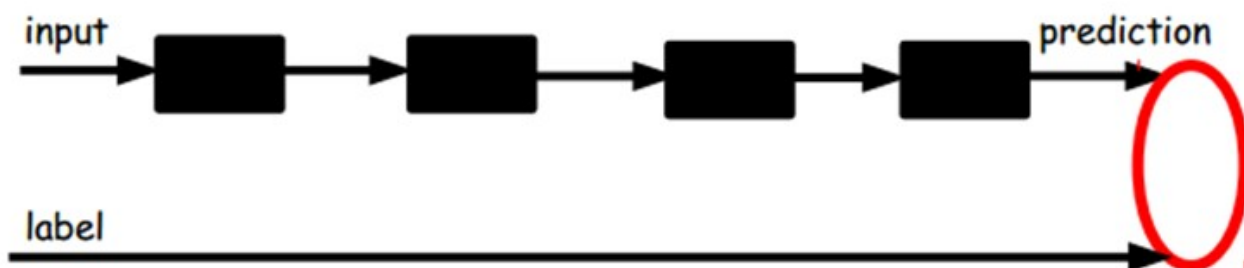
经过上面的方法，我们就可以得到很多层了。至于需要多少层（或者深度需要多少，这个目前本身就没有一个科学的评价方法）需要自己试验调了。每一层都会得到原始输入的不同的表达。当然了，我们觉得它是越抽象越好了，就像人的视觉系统一样。

到这里，这个**AutoEncoder**还不能用来分类数据，因为它还没有学习如何去连结一个输入和一个类。它只是学会了如何去重构或者复现它的输入而已。或者说，它只是学习获得了一个可以良好代表输入的特征，这个特征可以最大程度上代表原输入信号。那么，为了实现分类，我们就可以在**AutoEncoder**的最顶的编码层添加一个分类器（例如罗杰斯特回归、**SVM**等），然后通过标准的多层神经网络的监督训练方法（梯度下降法）去训练。

也就是说，这时候，我们需要将最后层的特征**code**输入到最后的分类器，通过有标签样本，通过监督学习进行微调，这也分两种，一个是只调整分类器（黑色部分）：



另一种：通过有标签样本，微调整个系统：（如果有足够多的数据，这个是最好的。end-to-end learning端对端学习）



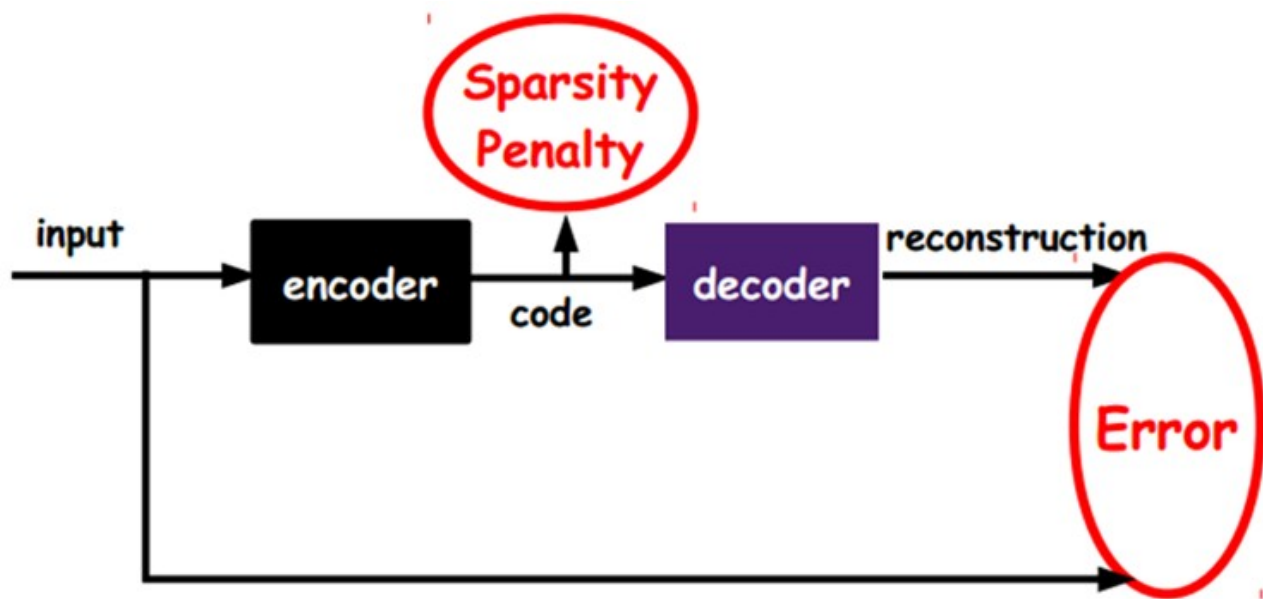
一旦监督训练完成，这个网络就可以用来分类了。神经网络的最顶层可以作为一个线性分类器，然后我们可以用一个更好性能的分类器去取代它。

在研究中可以发现，如果在原有的特征中加入这些自动学习得到的特征可以大大提高精确度，甚至在分类问题中比目前最好的分类算法效果还要好！

AutoEncoder存在一些变体，这里简要介绍下两个：

### **Sparse AutoEncoder**稀疏自动编码器：

当然，我们还可以继续加上一些约束条件得到新的Deep Learning方法，如：如果在AutoEncoder的基础上加上L1的Regularity限制（L1主要是约束每一层中的节点中大部分都要为0，只有少数不为0，这就是Sparse名字的来源），我们就可以得到Sparse AutoEncoder法。



- input:  $X$  code:  $h = W^T X$

- loss:  $L(X; W) = \|W h - X\|^2 + \lambda \sum_j |h_j|$

如上图，其实就是限制每次得到的表达code尽量稀疏。因为稀疏的表达往往比其他的表达要有效（人脑好像也是这样的，某个输入只是刺激某些神经元，其他的大部分神经元是受到抑制的）。

## Denoising AutoEncoders 降噪自动编码器：

降噪自动编码器DA是在自动编码器的基础上，训练数据加入噪声，所以自动编码器必须学习去去除这种噪声而获得真正的没有被噪声污染过的输入。因此，这就迫使编码器去学习输入信号的更加鲁棒的表达，这也是它的泛化能力比一般编码器强的原因。DA可以通过梯度下降算法去训练。

