# 表示学习:综述与新视角

Yoshua Bengio†, Aaron Courville, and Pascal Vincent†
Department of computer science and operations research, U. Montreal
† also, Canadian Institute for Advanced Research (CIFAR)

**Abstract**—
机器学习算法的成功通常取决于数据表示，我们假设这是因为不同的表示可以或多或少地纠缠和隐藏数据背后不同的解释变量因素。虽然特定的领域知识可以用来帮助设计表示，但也可以使用通用先验进行学习，对人工智能的追求正在激励设计更强大的表示学习算法来实现这些先验。本文综述了近年来在无监督特征学习和深度学习领域的工作，涵盖了概率模型、自编码器、流形学习和深度网络等方面的进展。这激发了有关学习良好表示、计算表示(即推理)的适当目标，以及表示学习、密度估计和流形学习之间的几何联系的长期悬而未决的问题。

**Index Terms**—Deep learning, representation learning, feature learning, unsupervised learning, Boltzmann Machine, autoencoder, neural nets

## 1 简介

机器学习方法的性能在很大程度上依赖于选择应用它们的数据表示(或特征)。为了因此，部署机器学习的大部分实际工作都是出于这个原因算法涉及到预处理管道和数据的设计转换的结果可以表示数据支持有效的机器学习。这种特征工程很重要但劳动密集，突出了当前学习的弱点算法:无法提取和组织判别式数据中的信息。特征工程是一种利用优势的方法人类的聪明才智和先验知识来弥补这一弱点。在为了扩大机器学习的适用范围和易用性，提出了一种新的机器学习方法降低学习算法的依赖性是非常可取的吗特征工程，以构建新颖的应用更快，更重要的是，要向人工进步智能(AI)。人工智能必须从根本上理解世界在我们周围，我们认为这只有在它可以学习的情况下才能实现找出并解开隐藏在其中的潜在解释因素低级感官数据的观察环境。

这篇论文是关于表示学习的，即，学习数据的表示，使其更容易提取构建分类器或其他预测器时的有用信息。在对于概率模型，一个好的表示通常是这样的捕捉潜在解释因素的后验分布用于观测输入。良好的表征也可以作为监督预测器的输入。在各种学习方法中表示，本文主要关注深度学习方法:那些由多个非线性变换组成，目标是产生更抽象的——最终更有用的东西——表示。在这里，我们对这个快速发展的地区进行了调查，并特别强调了最近的发展进步。我们考虑了一些基本的问题推动这一领域的研究。具体来说，是什么构成了一个表示比另一个更好?给定一个例子，我们应该如何计算它表示，即进行特征提取?还有，什么是合适的学习良好表示的目标是什么?

## 2 我们为什么要关心表示学习?

表示学习本身已经成为机器中的一个领域学习社区，定期在会议上举办讲习班作为NIPS和ICML，以及一个致力于它的新会议，ICLR [1]，有时在深度学习或特征学习。虽然深度是故事的重要组成部分，但很多其他先验是有趣的，可以很方便地捕获当问题被视为学习表示的问题之一时，作为下一节将讨论。科学活动的迅速增加对表示学习的研究一直伴随着和滋养非凡的一连串的经验成功，无论是在学术界还是在工业。下面，我们简要介绍其中的一些要点。

语音识别与信号处理

语音是神经网络的早期应用之一，特别是卷积(或时延)神经网络[2]。最近人们对神经网络的兴趣重新燃起，深度学习和表示学习已经在语音识别领域产生了巨大影响，突破性地结果(Dahl *et al.*, 2010; Deng *et al.*, 2010; Seide *et al.*, 2011a; Mohamed *et al.*, 2012; Dahl *et al.*, 2012; Hinton *et al.*, 2012) 由几位学者和研究人员在工业实验室带来了这些将算法扩展到更大的规模并转化为产品。例如，微软在2012年发布了一个新的版本的MAVIS(微软音频视频索引服务) 基于深度学习的语音系统(Seide *et al.*, 2011a)。这些作者设法减少了单词的错误率四个主要基准提高约30%(例如，从27.4%在rt03上达到18.5%)，而不是最先进的模型基于高斯混合进行声学建模并在相同的数据量(309小时的语音)上进行训练。得到错误率的相对改善Dahl *et al.* (2012)对较小的大词汇量语音识别基准测试(必应移动商业搜索数据集，包含40小时的语音) 在16%到23%之间。

表示学习算法也被应用到音乐中，大大超过了最先进的复调转录(Boulanger-Lewandowski *et al.*, 2012)，具有相对的误差改善在4个数据集的标准基准上在5%到30%之间。深度学习也帮助赢得了MIREX(音乐信息检索) 例如，2011年关于音频标记的比赛(Hamel *et al.*, 2011)。

目标识别

2006年，深度学习开始专注于MNIST数字图像分类问题(Hinton *et al.*, 2006; Bengio *et al.*, 2007)，打破svm在这个数据集上的霸主地位(1.4%的误差)[3]。最新记录仍然由深度网络控制: Ciresan *et al.* (2012)目前声称该任务的非受限版本是最先进的(例如，使用卷积架构)，误差为0.27%，Rifai *et al.* (2011c)是最先进的MNIST的无知识版本，误差为0.81%。

在过去的几年中，深度学习已经从数字发展到物体识别自然图像，并取得了最新的突破在ImageNet数据集[4]上将最先进的错误率从26.1%降低到15.3%(Krizhevsky *et al.*, 2012)。

自然语言处理

除了语音识别，还有许多其他的自然语言处理(NLP)。表示学习的应用。分布式符号数据的表示由Hinton (1986)引入，首先是在统计语言建模的背景下发展起来的通过所谓的神经网络语言Bengio *et al.* (2003) 模型(Bengio, 2008)。它们都是基于学习每个单词的分布式表示，称为单词嵌入。添加了卷积架构，Collobert *et al.* (2011)开发了SENNA系统[5]下载在

1. 国际学习表示会议

2. 有关早期工作的回顾，请参见Bengio (1993) 面积。

3. 为知识免费任务的版本，其中没有使用特定于图像的先验信息，如image 变形或卷积

4. 1000类ImageNet基准，其结果详细如下:
http://www.image-net.org/challenges/LSVRC/2012/results.html

5. 可从://ml.nec-labs.com/senna/

语言建模任务中共享表示，词性标注，组块分析，命名实体识别，语义角色标注，句法分析解析。塞纳接近或超过了最先进的这些任务比传统的预测器更简单、更快。单词嵌入的学习可以与图像表示的学习相结合以一种允许关联文本和图像的方式。这种方法已成功地用于构建谷歌图像搜索，利用大量的数据来映射图像和查询在同一空间(Weston *et al.*, 2010)最近，它已经扩展到更深的层次多模态表示(Srivastava and Salakhutdinov, 2012)。

通过在中加入递归，对神经网络语言模型进行了改进隐藏层(Mikolov *et al.*, 2011)，使它不仅在困惑度(平均负指数)方面击败了最先进的(平滑n-gram模型) 预测下一个正确单词的对数可能性，从140下降到102) 也包括单词错误由于语言模型是语音识别的重要组成部分)，从17.2%(KN5基线)降低。在《华尔街日报》上从16.9%(判别式语言模型)到14.4%基准任务。类似的模型也应用于统计学中机器翻译(Schwenk *et al.*, 2012; Le *et al.*, 2013)，提高困惑度以及BLEU分数。递归自动编码器(泛化递归网络)也被用来击败最先进的全句复述检测(Socher *et al.*, 2011a) 复述检测的F1分数几乎翻倍。表示学习也可以用于执行词义消歧(Bordes *et al.*, 2012)，提出在Senseval-3子集上的准确率从67.8%到70.2%其中该系统可以应用于(主谓宾句)。最后，它也被成功地用来超越情感分析的最新进展(Glorot *et al.*, 2011b; Socher *et al.*, 2011b)。

多任务与迁移学习，领域自适应

迁移学习是学习算法可以利用的能力不同学习任务之间的共性，以共享统计加强，并跨任务转移知识。如下所述，本文假设表示学习算法具有优势因为他们学习捕获基本因素的表示，其中的一个子集可能与每个特定的任务相关，如图所示在图1中。这个假设似乎得到了证实通过一些实证结果显示了代表性的力量迁移学习场景下的学习算法。

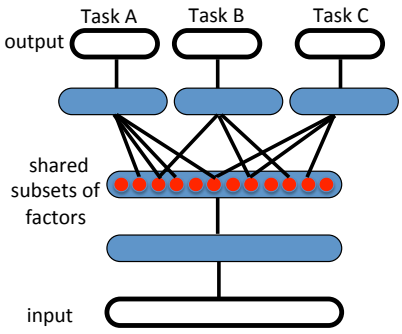


Fig. 1. 表示学习图解解释性发现因素(中间隐藏层，红色)，一些解释输入(半监督设置)和一些解释为每项任务设定目标。因为这些子集重叠，共享统计强度有助于泛化。

最令人印象深刻的是2011年举行的两次迁移学习挑战由表示学习算法获胜。首先是迁移学习在ICML 2011的同名研讨会上提出的挑战获得了冠军使用无监督分层预训练(Bengio, 2011; Mesnil *et al.*, 2011)。a 同年举办第二届迁移学习挑战赛并获胜通过Goodfellow *et al.* (2011)。结果在NIPS 2011上公布学习层次模型的挑战工作坊。在相关的域适应设置中，目标保持不变，但输入分布变化(Glorot *et al.*, 2011b; Chen *et al.*, 2012)。在多任务学习环境下，也发现了表示学习优势Krizhevsky *et al.* (2012); Collobert *et al.* (2011)，因为共享因素跨任务。

## 3 是什么造就了优秀的表现?

### 3.1 人工智能表示学习的先验知识

在Bengio and LeCun (2007)中，我们中的一位介绍了人工智能任务的概念，当前的机器学习算法具有哪些挑战性复杂但高度结构化的依赖关系。显式处理表示的一个原因很有趣是因为它们可以方便地表达许多一般的先验我们周围的世界，即不是特定于任务但会的先验可能对学习机解决人工智能任务有用。下面是这种通用先验的例子。

- 平滑性:假设要学习的函数$f$是s.t. $x \approx y$一般意味着$f(x) \approx f(y)$。这个最基本的先验知识存在于大多数机器学习中，但不足以解决维数灾难，请参见3.2部分。
- 多重解释因素:数据生成分布由不同的潜在因素，以及在很大程度上人们对一个人的了解Factor可以推广到其他因子的许多配置。目标恢复或至少解开这些潜在的变异因素在3.5部分讨论。这个假设是分布式表示的思想，将在下面的3.3部分讨论。
- 解释因素的层次组织:有用的概念用于描述我们周围的世界可以用其他概念来定义，在一个层级中，层次越高，抽象的概念越高，以不那么抽象的方式定义。这一假设是通过深度表示进行开发的在下面的3.4 部分。
- 半监督学习: 输入$X$和目标$Y$进行预测，解释$X$分布的部分因素可以说明很多问题的$Y$，给出$X$。因此表示对$P(X)$很有用在学习$P(Y|X)$时往往很有用，允许共享统计无监督学习和有监督学习任务之间的强度，见4 部分。
- 跨任务共享因素:与许多$Y$的兴趣或学习任务多一般，任务多(例如，相应的$P(Y|X, \text{task})$) 是由与其他任务共享的因素解释的，允许共享如前所述，跨任务的统计强度章节(多任务和迁移学习，域适应)。
- 流形:概率质量集中在具有维数比数据所在的原始空间小得多。这在一些自动编码器算法中得到了明确的利用其他流形启发的算法将在章节中分别描述7.2和8。
- 自然聚类:的不同值像对象类这样的分类变量是与独立关联的流形。更准确地说，流形上的局部变化趋向于保留类别的值，并在两者之间进行线性插值不同类别的例子通常涉及到低密度区域，即$P(X|Y = i)$对应不同的$i$，往往会被很好地分开而且没有太多重叠。例如，这在流形中被利用切线分类器在8.3部分讨论。这个假设与人类命名的想法是一致的类别和类因为这样的统计结构(发现通过他们的大脑和文化传播)，以及机器学习任务通常涉及预测此类分类变量。
- 时间和空间的一致性:连续的(从一个序列中)或空间的附近的观测值往往是与相关范畴概念的相同值相关联的，或导致高密度歧管表面的小移动。更一般地说，不同的因素在不同的时间和空间尺度上发生变化，而且有许多是类别型的兴趣的概念变化缓慢。当试图捕捉这样的分类变量，这种先验可以通过进行关联表示来强制执行缓慢变化，即随着时间或空间的变化而惩罚值的变化。这个先验在Becker and Hinton (1992)中介绍，在11.3节中讨论。
- 稀疏性:对于任何给定的观察$x$，只有一小部分可能的因素是相关的。在表示方面，这个可以用通常为0的特征表示(如最初由Olshausen and Field (1996) 提出)，或由大多数的事实所提取的特征对$x$的微小变化不敏感。这可以通过隐变量的某些先验形式来实现(峰值为0)，或者使用非线性函数，其值通常是在0处平坦(即0且导数为0)，或者简单地通过惩罚函数映射的雅可比矩阵(关于导数)的大小表示的输入。这在部分6.1.1和7.2。
- 因素依赖关系的简单性:在良好的高级表示中，因素之间通过简单的、典型的线性依赖关系相互联系。这可以在许多物理定律中看到，并且在代入线性时是假设的在学习到的表示之上的预测器。

我们可以将上述许多先验看作是帮助学习者发现的方法并解开一些潜在的(和先验未知的)变异因素数据可能揭示了这一点。我们将进一步探讨这个想法章节3.5 和11.4。

### 3.2 平滑性和维度诅咒

对于人工智能任务，如视觉和NLP，仅仅依靠简单似乎是没有希望的参数模型(如线性模型)，因为它们无法捕捉到足够

的复杂性的兴趣，除非提供合适的特征空间。相反，机器学习研究人员一直在寻求灵活性本地[6] 非参数 像核机器这样的学习者具有固定的通用局部响应核(如高斯核)。不幸的是，正如Bengio and Monperrus (2005); Bengio *et al.* (2006a); Bengio and LeCun (2007); Bengio (2009); Bengio *et al.* (2010)，这些算法大多仅利用局部搜索的原理泛化，即假设目标函数(为( Learned)足够平滑，因此它们依赖于示例来明确映射消除目标函数的皱纹。基本实现了泛化通过一种相邻训练样本之间的局部插值形式。虽然平滑是有用的假设，它不足以处理维数灾难，因为这样的皱纹的数量(目标函数的起伏) 可能随相关交互因素的数量呈指数增长，何时数据在原始输入空间中表示。我们提倡灵活的学习算法非参数[7] 但不要完全依赖于平滑假设。相反，我们建议合并那些枚举的通用先验以上转化为表示学习算法。基于平滑度的学习者(例如内核机器)和线性模型仍然可以很有用这样的学习表征。事实上，学习表示和核机器的组合是等效的学习核函数，即特征空间。内核机器是有用的，但它们依赖于预先的定义一个合适的相似性度量，或者一个简单的相似性度量就足够的特征空间。我们想使用这些数据，以及非常通用的先验来发现这些特征，或者等价地说，一个相似度函数。

### 3.3 分布式表示

好的表示法具有表现力，这意味着一个合理大小的学习表示可以捕获大量可能的输入配置。一个简单的计数参数有助于我们评估模型的表达能力生成表示:需要与多少参数进行比较它能区分多少个输入区域(或配置)? 独热表示的学习者，如传统的聚类算法，高斯混合，最近邻算法，决策树，或高斯svm都需要$O(N)$参数(和/或$O(N)$示例) 区分$O(N)$输入区域。人们可以天真地相信这一点没有比这更好的了。然而，RBMs，稀疏编码，自动编码器或多层神经网络都可以表示到$O(2^k)$ 输入区域仅使用$O(N)$参数($k$为非零的数量元素在稀疏表示中，而$k = N$在非稀疏RBMs和其他稠密表示中表示)。这些都是分发[8] 或稀疏[9]表示。聚类到分布式表示的推广是多聚类(multi-clustering)吗并行进行，或者同一个集群应用于不同的部分的输入，例如非常流行的分层特征提取用于基于在不同块中检测到的聚类类别的直方图进行目标识别图片(Lazebnik *et al.*, 2006; Coates and Ng, 2011a)。分布或稀疏的指数增益表示法将在Bengio (2009)的3.2节(和图3.2)进一步讨论。它出现的原因是每个参数(例如，稀疏中的一个单元的参数代码，或受限玻尔兹曼机中的一个单元)可以重复使用在许多例子中，它们不是简单的近邻，然而在局部泛化中，输入空间中的不同区域基本与其关联拥有自己的私有参数集，例如决策树、最近邻、高斯svm等。在分布式表示中，指数大量可能的特征子集或隐藏单元可以被激活到给定的输入。在单层模型中，每个特征通常与首选输入方向相关联，对应于输入空间中的超平面，以及代码或表示与该输入相关联的正是激活模式(哪些特征对输入做出响应，以及响应程度)。这与非分布式形成对比表示，如大多数聚类算法学习的表示，例如k-means，其中，给定输入向量的表示是one-hot代码识别少数几个簇的质心中哪一个最能代表输入[10]。

### 3.4 深度与抽象

深度是我们考虑的表示学习策略的一个关键方面这篇论文。正如我们将要讨论的，深度架构通常具有挑战性有效地训练，这是最近许多研究的主题进步。然而，尽管有这些挑战，他们携带两个这些显著的优势激发了我们对发现的长期兴趣深度架构的成功训练策略。这些优点是:(1)深度架构促进了功能的重用; 深度架构可能导致逐渐变得更加抽象 表示层更高层的特征(从数据中删除更多)。

功能重用。重用的概念，这解释了分布式的力量表示，也是理论优势的核心深度学习的背后，即构建多层次的表示或者学习特征的层次结构。电路的深度就是电路的长度从电路的输入节点到输出节点的最长路径。深度电路的关键属性是它的路径数量，即到达重复使用不同的部分，可以随着深度指数增长。形式上，人们可以通过改变每个电路的定义来改变给定电路的深度节点can 计算，但只计算一个常数因子。我们允许的典型计算节点包括:加权和、乘积、人工神经元模型(如单调仿射变换上的非线性)、核的计算、或者逻辑门。理论结果清楚地显示了家庭函数的深度表示可以指数级地提高效率而不是深度不够的(Håstad, 1986; Håstad and Goldmann, 1991; Bengio *et al.*, 2006a; Bengio and LeCun, 2007; Bengio and Delalleau, 2011)。如果相同的函数族可以用更少的参数表示(或者更准确地说，vc维度更小)，学习理论是否表明可以用更少的例子来学习，从而提高两者的性能计算效率(访问的节点更少) 以及统计效率(需要学习的参数更少，以及在许多不同类型的输入上重用这些参数)。

抽象和不变性。深度架构可能导致抽象表示，因为更多抽象的概念通常可以用不那么抽象的方式来构建1。在某些情况下，例如在卷积神经网络中网络(LeCun *et al.*, 1998b)，我们显式地通过池机制(参见11.2节)。更抽象的概念通常对输入的大多数局部变化是不变的。那使得捕获这些概念的表示通常很高原始输入的非线性函数。这显然是真的范畴概念，更抽象的表示检测类别覆盖更多不同的现象(例如，更大的流形有更多的皱纹) 因此，它们可能具有更大的预测能力。抽象可以也出现在高级的连续值属性中对输入中某些非常特定类型的变化敏感。学习这种类型的不变特征一直是模式的目标认可。

### 3.5 变异因子的解缠

除了分布和不变，我们希望我们的解缠变异因子的表示。不同数据的解释因子往往是相互独立变化的在输入分布中，只有少数在某一时刻趋于变化时才会发生变化考虑一系列连续的真实世界输入。

复杂数据是多源数据相互作用的结果。这些因素在复杂的web中进行交互，这可能会使ai相关的任务复杂化，例如目标分类。例如，图像是由交互组成的在一个或多个光源之间，物体形状和材料图像中存在的各种表面的属性。阴影场景中的物体可以以复杂的模式落在彼此身上，创造物体边界的错觉，没有边界，而且很戏剧性影响感知的物体形状。我们如何应对这些复杂的问题互动?我们如何解开对象和它们的阴影? 最终，我们相信我们为克服这些问题所采取的方法挑战必须利用数据本身，使用大量的未标记的示例，以学习将各种解释性的来源。这样做应该会产生一种代表性对复杂和

6. 局部意义上的学问的价值在$x$的功能主要取决于训练示例$x^{(t)}$的接近$x$

7. 我们将非参数理解为包括所有学习算法的容量可以随着数量的增加而适当增加数据及其复杂性要求它，例如混合模型和神经网络其中参数的数量是一个数据选择的超参数。

8. 分布式表示:其中$k$出$N$ 表示元素或特征值可以独立变化，例如，它们不是互斥的。每个概念都通过打开或激活$k$功能来表示，同时每个特征都涉及到许多概念的表示。

9. 稀疏表示:仅在其中的分布式表示其中一些元素可以同时改变，即$k < N$。

10. 如(Bengio, 2009)所述，当使用连续值时，情况只会稍微好一点隶属度值，例如，在普通的混合模型中(具有单独的参数对于每种混合物成分)，但差异代表性的力量仍然是指数级的(Montufar and Morton, 2012)。对于每个给定的输入，使用决策树似乎更好与叶子上的one-hot代码相关联，哪些是确定的选择关联的祖先节点(从根节点到节点的路径)。不幸的是，表示不同区域的数量(等于叶子的数量树的)仍然只是线性增长用于指定它的参数数量(Bengio and Delalleau, 2011)。

结构丰富的变化更加稳健存在于人工智能相关任务的自然数据源中。

重要的是要区分相关但明确的目标学习不变特征和学习解缠解释因素。最主要的区别是保存信息。根据定义，不变特征的敏感性降低了不变性的方向。这是构建具有这些特性的功能的目标对对任务没有信息的数据变化不敏感手。不幸的是，通常很难确定先验 哪一组特征和变化最终将与at的任务相关手。此外，就像深度学习方法中经常出现的情况一样，被训练的特征集可能注定是这样的用于多个任务，这些任务可能具有不同的相关特征子集。考虑到这些因素，我们得出结论，最稳健的特征学习的方法是解缠尽可能多的因素这是可能的，尽可能少地丢弃有关数据的信息实用。如果需要某种形式的降维，那么我们假设局部变化方向在首先应该对训练数据进行剪枝(例如，在PCA中，这样做是全局的，而不是围绕每个示例)。

### 3.6 学习表示的好标准?

表征学习的挑战之一，区别于它分类等其他机器学习任务是其难点为培训建立一个明确的目标。在...的情况下分类，目标(至少在概念上)是显而易见的，我们想要最小化训练数据集上的错误分类数量。在就表示学习而言，我们的目标离最终目标，通常是学习一个分类器或其他东西预测器。我们的问题让人想起学分分配问题在强化学习中遇到的。我们提议一个好的表示法是能够解开潜在的因素但是我们如何将其转化为合适的训练标准呢? 在一个好的模型下，除了最大化似然之外，还有必要做其他事情吗或者我们可以引入上述的先验知识(可能是依赖于数据的) 表示法更好地进行这种解缠?这个问题仍然存在显然是开放的，但在3.5节中有更详细的讨论和11.4。

## 4 构建深度表示

2006年，特征学习和深度研究取得突破学习是由Geoff Hinton发起的，并很快跟进同年(Hinton *et al.*, 2006; Bengio *et al.*, 2007; Ranzato *et al.*, 2007)，不久之后是Lee *et al.* (2008)，之后更多。一直是在Bengio (2009)上进行了广泛的回顾和讨论。一个中心思想，称为贪婪的逐层无监督预训练，是为了学习一次一层的特征层次结构，使用无监督特征学习在每个级别上学习新的转换，以与之前学习的转换相结合变换;本质上，无监督特征学习的每次迭代都会向深度学习添加一层权重神经网络。最后，可以合并层集进行初始化深度监督预测器，如神经网络分类器或深度监督预测器生成模型，如深度玻尔兹曼机器(Salakhutdinov and Hinton, 2009)。

本文主要介绍可以使用的特征学习算法形成深度架构。特别是，它是根据经验观察到的这种逐层堆叠的特征提取通常效果更好表示，例如，根据分类错误(Larochelle *et al.*, 2009; Erhan *et al.*, 2010b)，质量由概率模型生成的样本(Salakhutdinov and Hinton, 2009) 或者根据所学特征的不变性属性(Goodfellow *et al.*, 2009)。虽然本节关注的是堆叠单层模型的想法，10 部分接着讨论了联合培训所有的层。

在贪婪的逐层无监督预训练之后，由此产生的深度特征既可以用作标准监督的输入机器学习预测器(如SVM)或深度监督神经网络的初始化网络(例如，通过添加逻辑回归层或纯监督层多层神经网络)。分层过程也可以应用在纯监督的情况下，称为逐层监督的贪婪预训练(Bengio *et al.*, 2007)。例如，在第一个隐藏层之后训练MLP，丢弃其输出层，另一个单隐藏层MLP可以堆叠在上面，等等。虽然结果报告在Bengio *et al.* (2007) 虽然不如无监督的预训练，但它们还是更好比完全不进行预训练要好。或者，前一个的输出 层可以作为下一层的额外输入(除了原始输入)，成功地在Yu *et al.* (2010)上完成。另一个变种(Seide *et al.*, 2011b)预训练在监督的方式下，所有之前在迭代的每个步骤中添加的层实验这种判别变体比无监督预训练产生了更好的结果。

虽然将单个层组合成监督模型很简单，目前还不清楚如何将无监督学习预训练的层组合起来形成更好的无监督模型。我们在这里介绍一些这样做的方法，但是没有明确的赢家出现，必须做很多工作来验证现有的建议或改进它们。

第一个建议是将预训练的rbm堆叠到一个深度信念网络(Hinton *et al.*, 2006)或DBN，其中顶层被解释为RBM，下层被解释为有向sigmoid置信网络。然而，目前还不清楚如何近似最大似然训练来进一步优化这个生成模型。一种选择是唤醒-休眠算法(Hinton *et al.*, 2006)但工作更多是否应该评估这一过程的效率以提高生成能力模型。

提出的第二种方法是将RBM参数进行组合转化为深度玻尔兹曼机(DBM)，基本上通过将RBM权重减半来获得DBM权重(Salakhutdinov and Hinton, 2009)。然后可以通过近似的最大似然来训练DBM 稍后将详细讨论(10.2节)。这次联合训练在这两方面都带来了实质性的改善似然度和由此产生的深度特征的分类性能学习者(Salakhutdinov and Hinton, 2009)。

另一种早期的方法是将RBMs或自动编码器堆叠到深的自动编码器 (Hinton and Salakhutdinov, 2006)。如果我们有一系列编码器-解码器对$(f^{(i)}(\cdot), g^{(i)}(\cdot))$，然后是整体编码器是编码器的组成部分，$f^{(N)}(\dots f^{(2)}(f^{(1)}(\cdot)))$，整个解码器就是它的转置。(通常也使用转置的权重矩阵)，$g^{(1)}(g^{(2)}(\dots f^{(N)}(\cdot)))$。深度自动编码器(或其正则化版本，如章节讨论7.2)就可以与所有人进行联合训练根据全局重构误差准则优化参数。更多显然，这条大道上的工作是需要做的，但它可能被避免了由于害怕训练深度前馈网络的挑战，讨论在章节中10 以及最近非常令人鼓舞的结果。

这是最近提出的另一种训练深度架构的方法(Ngiam *et al.*, 2011)是为了考虑自由能函数的迭代构造(即，没有显式的隐变量，除了可能是隐藏单元的顶层)用于深层架构作为组合变换与较低的层相关联，随之而来通过顶级隐藏单元。问题是然后如何训练一个由任意参数化(自由)能量函数定义的模型。Ngiam *et al.* (2011)有使用混合蒙特卡洛(Neal, 1993)，但其他选项包括对比差异(Hinton, 1999; Hinton *et al.*, 2006)，分数匹配(Hyvärinen, 2005; Hyvärinen, 2008)，去噪分数匹配(Kingma and LeCun, 2010; Vincent, 2011)，比率匹配(Hyvärinen, 2007) 噪声对比估计(Gutmann and Hyvarinen, 2010)。

## 5 单层学习模块

在对表示学习感兴趣的研究人员社区中，已经形成了两种大致平行的调查方式:一种是根深蒂固的概率图模型和基于神经网络的模型。从根本上说，这两种范式的区别在于深度学习模型的分层架构可以解释为描述概率图模型的或描述计算的图形。简而言之，被认为是潜在随机变量的隐藏单元或作为计算节点?

到目前为止，这两种范式之间的两分法也许还没有引起人们的注意因为它们似乎有更多的共同特征，而不是将它们分开。我们认为，这可能是由于两者最近取得的进展这些领域的重点是单层贪婪学习模块及其相似性在已探索的单层模型类型之间:主要是受限制的概率方面的玻尔兹曼机(RBM)，以及神经网络方面的自编码器变体网络端。事实上，正如我们中的一个(Vincent, 2011)和其他人(Swersky *et al.*, 2011)所表明的那样，在受限玻尔兹曼机的情况下，通过归纳训练模型被称为分数匹配的原则(Hyvärinen, 2005)(将在6.4.3节中讨论)本质上与将正则化重建目标应用于自编码器相同。另一个两组模型之间存在着很强的联系是什么时候计算图的用于计算神经网络模型中的表示完全对应于计算图这对应于概率模型中的推理，这也和结构相对应图形模型本身的(例如，在RBM中)。

当我们考虑更深层次的模型时，这两种范式之间的联系变得更加薄弱，在概率模型的情况下，精确推理通常是难以

解决的。在深度模型的情况下，计算图偏离模型的结构。例如，在深度玻尔兹曼机的情况下，展开变分(近似) 对计算图的推理结果是递归图结构。我们已经进行了初步探索(Savard, 2011) 深度自编码器的确定性变体它的计算图类似于深度玻尔兹曼机(实际上非常接近与玻尔兹曼机相关的平均场变分近似)，这是一个值得探索的有趣的中间点(在确定性方法和图形模型方法)。

在接下来的几节中，我们将回顾单层训练模块用于支持特征学习和尤其是深度学习。我们将这些部分分为(6部分) 概率模型，具有推理和训练方案直接参数化生成——或解码——路径和(7节) 典型的基于神经网络的模型直接参数化编码通路。有趣的是，有些模型，比如预测稀疏分解(PSD) (Kavukcuoglu *et al.*, 2008)继承这两个属性，也将继承讨论(7.2.4部分)。然后我们提出一个不同的观点的表示学习，基于相关的几何和流形假设，在8节。

首先，让我们考虑一个无监督的单层表示学习算法涵盖所有三个视图:概率、自动编码器和流形学习。

主成分分析

我们将使用可能是最古老的特征提取算法principal 成分分析(PCA)，到说明概率，自动编码器和流形的观点表示学习。PCA学习输入$x \in \mathbb{R}^{d_x}$的线性变换$h = f(x) = W^T x + b$，其中的列$d_x \times d_h$ 矩阵$W$构成的正交基为$d_h$正交方向训练数据的最大方差。结果是$d_h$特征(表示的组件$h$) 去相关。PCA的三种解释如下。A )它与概率模型有关 (6节)如概率PCA、因子分析以及传统的多元高斯分布(协方差矩阵的主要特征向量为主成分);B)它学习的表示本质上是与基本的线性自编码器学习的相同(节7.2);c)它可以被看作是一种简单的线性形式线性流形学习(8节)，即特征化输入空间中数据密度接近的低维区域达到顶峰。因此，PCA可能是在读者的脑海中作为一个共同的线索联系着这些不同的观点。不幸的是，线性特征的表达能力非常有限:它们不能堆叠以形成更深、更抽象的表示，因为线性运算的组合会产生另一个线性运算。这里，我们专注于最近开发的用于提取非线性的算法特征可以堆叠在深度网络的构建中，尽管有些作者只需插入一个学习到的单层线性之间的非线性预测(Le *et al.*, 2011c; Chen *et al.*, 2012)。

又一个富裕家庭本文没有介绍任何特征提取技术细节部分由于空间限制采用独立分量分析或ICA (Jutten and Herault, 1991; Bell and Sejnowski, 1997)。相反，我们引用读者来信Hyvärinen *et al.* (2001a); Hyvärinen *et al.* (2009)。注意，while在最简单情况(完全，无噪声)ICA产生线性特征，在更一般的情况下，它可以等同于线性生成模型 具有非高斯独立隐变量，类似于稀疏编码(6.1.1节)，结果非线性特征。因此，ICA及其变种喜欢独立的而地形ICA (Hyvärinen *et al.*, 2001b)可以并且已经被用来构建深度网络(Le *et al.*, 2010, 2011c):参见11.2 部分。获得独立分量的概念这看起来也类似于我们阐明的解缠底层的目标深度网络中的解释因子。然而，对于复杂的现实世界分布，真正独立的潜在因素和观测到的高维数据之间的关系能否用线性变换充分地表征是值得怀疑的。

## 6 概率模型

从概率建模的角度，特征问题学习可以被解释为尝试恢复一个简约的集合描述观测数据上分布的潜在随机变量。我们可以表示为$p(x, h)$在潜在的关节空间上的概率模型变量$h$和观测数据或可见变量$x$。特征值被认为是推理过程的结果确定给定的潜变量的概率分布数据，即$p(h \mid x)$，通常称为后验概率。学习的概念是估计一组模型参数，(局部)最大化训练数据的正则化似然。概率图模型形式化为我们提供了两种可能的建模我们可以考虑推断潜在问题的范式变量，有向和无向的图形模型，它们是不同的在其参数化的联合分布$p(x, h)$，产生对推理和计算成本的主要影响学习。

### 6.1 有向图模型

有向隐语义模型分别参数化条件似然$p(x \mid h)$和先验 $p(h)$构建联合分销，$p(x, h) = p(x \mid h)p(h)$。这种分解的例子包括:Principal 成分分析(PCA) (Roweis, 1997; Tipping and Bishop, 1999)，稀疏编码(Olshausen and Field, 1996), sigmoid信念网络(Neal, 1992) 以及新引入的spike- slab稀疏编码模型(Goodfellow *et al.*, 2011)。

#### *6.1.1* 解释

有向模型通常会导致一个重要的属性:解释，即: 一个事件的先验独立原因可以成为对事件的观察是非独立的。潜在因子模型通常可以解释为潜在原因模型，其中$h$激活导致观察到的$x$。这个呈现出先验的独立性$h$ 非独立的。因此，恢复后验分布$h$, $p(h \mid x)$(我们使用它作为特征表示的基础)是通常在计算上具有挑战性，也可以完全如此棘手，尤其是当$h$是离散的时候。

说明这种现象的一个经典例子是想象你在上离开家度假时，你会接到保安的电话系统公司，告诉你报警已激活。你开始担心家里被盗了，但是然后你在收音机里听到有一个小地震的报道你家的区域。如果你从之前的经验中知道地震有时会使你的家庭警报系统启动突然间，你放松下来，确信你的家很可能已经没有了入室盗窃。

这个例子说明了如何渲染报警激活 两个完全独立的原因，盗窃和地震，变得依赖——在这种情况下，依赖是互斥的:互斥的因为他们都盗窃了地震是非常罕见的事件，两者都可能引起警报激活，一种现象解释了另一种现象。尽管有计算障碍，我们脸时试图恢复臀部过$h$，解释承诺提供一个简约的$p(h \mid x)$，这可以是一个极其特征编码方案的有用特性。如果一个人认为表征是存在的由各种特征检测器和估计属性组成观察输入，它是有用的，允许不同的特征相互竞争和协作来解释输入。这是自然地通过有向图形模型实现，但也可以可以用无向模型实现(参见6.2部分) 比如玻尔兹曼机如果对应单元之间有横向连接 或者能量函数中相应的相互作用项 定义概率模型。

**PCA**的概率解释。 PCA可以给出一个自然的概率解释(Roweis, 1997; Tipping and Bishop, 1999) As因子分析:

$$\begin{aligned} p(h) &= \mathcal{N}(h; 0, \sigma_h^2 \mathbf{I}) \\ p(x \mid h) &= \mathcal{N}(x; Wh + \mu_x, \sigma_x^2 \mathbf{I}), \end{aligned} \tag{1}$$

$x \in \mathbb{R}^{d_x}$, $h \in \mathbb{R}^{d_h}$, $\mathcal{N}(v; \mu, \Sigma)$是多元的吗正态密度$v$，均值$\mu$，协方差$\Sigma$，和$W$的列与前面的$d_h$主成分跨越相同的空间，但不是被约束为标准正交的。

稀疏编码。 与PCA 一样，稀疏编码同时具有概率和非概率解释。还有稀疏编码关联潜在表示$h$(随机变量的向量或一个特征向量，取决于解释)到数据$x$通过一个线性映射$W$，我们称之为字典。稀疏编码和PCA之间的区别在于稀疏编码包含一个惩罚项来确保a $h$的激活用于编码每个输入$x$。从非概率的角度来看，稀疏编码可以看作是恢复与新关联的代码或特征向量通过以下方式输入$x$:

$$h^* = f(x) = \operatorname*{argmin}_h \|x - Wh\|_2^2 + \lambda \|h\|_1, \tag{2}$$

学习字典$W$可以通过优化关于$W$的培训标准如下:

$$\mathcal{J}_{\text{SC}} = \sum_t \|x^{(t)} - Wh^{*(t)}\|_2^2, \tag{3}$$

其中$x^{(t)}$是$t$ -th示例，$h^{*(t)}$是相应的稀疏代码确定通过Eq. 2。$W$通常是约束具有单位范数列(因为可以任意交换列$i$的标度与$h_i^{(t)}$，这样的约束对于L1惩罚项产生任何影响是必要的)。

稀疏编码的概率解释与此不同而不是高斯先验对于潜在随机变量$h$，我们使用稀疏诱导拉普拉斯先验(对应于L1惩罚):

$$\begin{aligned} p(h) &= \prod_i^{d_h} \frac{\lambda}{2} \exp(-\lambda|h_i|) \\ p(x \mid h) &= \mathcal{N}(x; Wh + \mu_x, \sigma_x^2 \mathbf{I}). \end{aligned} \tag{4}$$

在稀疏编码的情况下，因为我们会寻求稀疏表示形式(即，一个具有许多特征设置为零的特征)，我们将对恢复MAP(后验概率的最大值) 感兴趣$h$:即$h^* = \operatorname{argmax}_h p(h \mid x)$而不是它的预期价值$\mathbb{E}[h|x]$。在这种解释下，就是字典学习在给定这些地图的情况下，最大化数据的可能性$h^*$: $\operatorname{argmax}_W \prod_t p(x^{(t)} \mid h^{*(t)})$ 受$W$规范约束。注意，这个参数是学习的方案，受制于MAP的潜值$h$，是不标准的概率图模型文献中的实践。一般来说数据$p(x) = \sum_h p(x \mid h)p(h)$的可能性是最大化的直接。在潜变量存在的情况下，期望在参数存在的地方采用最大化相对于边际似然优化，即求和或对所有潜在值的联合对数似然进行积分其后验下的变量$P(h \mid x)$，而不是只考虑$h$的单个MAP值。这种形式的参数的理论性质学习还没有被很好地理解，但似乎在实践中很有效(例如，k均值vs高斯混合模型和hmm的Viterbi训练)。还需要注意的是，将稀疏编码解释为MAP估计可以被质疑(Gribonval, 2011)，因为即使将L1惩罚解释为对数先验是可能的解释，可以有其他贝叶斯解释兼容培训标准。

稀疏编码是解释能力的一个很好的例子。即使有一个非常完整的字典[11]，稀疏编码中用到的MAP推理过程$h^*$可以选出最合适的基数，其他的为零，尽管它们与输入有高度的相关性。这个属性在有向图模型中自然产生，如稀疏编码和完全是由于解释效应。它在常用中不常见无向概率模型，如RBM, nor 它是否出现在参数化特征编码方法中，例如自动编码器。需要权衡的是，与RBMs和自动编码器，稀疏编码中的推理涉及到一个额外的内部循环优化找到$h^*$与之相应的增加特征提取的计算代价。与自动编码器相比RBMs，稀疏编码中的代码是每个示例的自由变量，并且从这个意义上说，隐式编码器是非参数的。

人们可能会期望稀疏编码的简约性表象和它的解释作用将是有利的，确实如此似乎是这样。Coates and Ng (2011a)在CIFAR-10上演示目标分类任务(Krizhevsky and Hinton, 2009)具有基于块的特征提取管道，在很少($< 1000$)标记的训练示例的政权中类，稀疏编码表示的性能明显优于其他方法高度竞争的编码方案。可能是因为这些性质，也可能是因为计算效率很高已经提出的算法(与一般的在解释的情况下进行推理)，稀疏编码的优点是作为特征学习和编码范式的流行度。有很多它作为一种特征表示方案的成功应用示例: 包括自然图像建模(Raina *et al.*, 2007; Kavukcuoglu *et al.*, 2008; Coates and Ng, 2011a; Yu *et al.*, 2011)，音频分类(Grosse *et al.*, 2007)，NLP (Bagnell and Bradley, 2009)，也非常成功早期视皮层模型(Olshausen and Field, 1996)。稀疏性准则也可以成功地推广到组倾向于全部为0的特征，但前提是其中一个或几个特征是活跃的那么激活群体中的其他人的惩罚是很小的。不同的组稀疏模式可以包含不同的组先验知识的形式(Kavukcuoglu *et al.*, 2009; Jenatton *et al.*, 2009; Bach *et al.*, 2011; Gregor *et al.*, 2011)。

**spike - slab**稀疏编码。Spike-and-slab稀疏编码(S3C)是一个很有前途的特征稀疏编码变体的例子学习(Goodfellow *et al.*, 2012)。S3C模型具有一组潜变量二元脉冲变量和一组潜在实值变量。脉冲变量的激活决定了稀疏模式。S3C已应用于CIFAR-10和CIFAR-100 对象分类任务(Krizhevsky and Hinton, 2009)，并显示与稀疏编码相同的模式在的范围内具有优越的性能相对较少($< 1000$)标记的例子per (Goodfellow *et al.*, 2012) 班。事实上，在CIFAR-100数据集(with 每个类别500个样本)和CIFAR-10数据集(当示例减少到类似的范围)，实际上是S3C表示优于稀疏编码表示。这种优势显露出来了显然，S3C赢得了NIPS 2011年的迁移学习挑战(Goodfellow *et al.*, 2011)。

11. Overcomplete: $h$的维度比$x$的维度多。

### 6.2 无向图模型

无向图模型，也称为马尔可夫随机场(MRFs)，可以参数化这个关节$p(x, h)$通过一个未归一化的乘积非负派系势:

$$p(x,h) = \frac{1}{Z_\theta} \prod_i \psi_i(x) \prod_j \eta_j(h) \prod_k \nu_k(x,h) \tag{5}$$

哪里$\psi_i(x)$, $\eta_j(h)$和$\nu_k(x,h)$是小集团的潜力描述可见元素之间的交互隐藏变量，以及可见变量和隐藏变量之间的交互变量。分区函数$Z_\theta$确保分布是规范化的。在无监督特征的上下文中学习时，我们一般把一种特殊形式的马尔可夫随机场称为团势约束为正的Boltzmann分布:

$$p(x,h) = \frac{1}{Z_\theta} \exp\left(-\mathcal{E}_\theta(x,h)\right), \tag{6}$$

其中$\mathcal{E}_\theta(x,h)$是能量函数并包含相互作用模型由MRF团势描述，$\theta$为模型表征这些相互作用的参数。

玻尔兹曼机最初被定义为对称耦合的二元随机变量或单位。这些随机单元可分为两组:(1)可见的 单位$x \in \{0,1\}^{d_x}$表示数据，以及(2) 隐藏的或潜在的单元$h \in \{0,1\}^{d_h}$中介可见单元之间的依赖关系通过他们的相互作用。交互的模式被指定通过能量函数:

$$\mathcal{E}_\theta^{\mathrm{BM}}(x,h) = -\frac{1}{2}x^T U x - \frac{1}{2}h^T V h - x^T W h - b^T x - d^T h, \tag{7}$$

其中$\theta = \{U, V, W, b, d\}$分别是哪些模型参数编码可见到可见的交互，隐藏到隐藏的交互交互，从可见到隐藏的交互，可见的自我联系和隐藏的自我联系(称为偏差)。To 为了避免过度参数化，$U$和$V$的对角线设置为零。

玻尔兹曼机能量函数指定$[x, h]$上的概率分布，通过玻尔兹曼分布，如6，与分区函数$Z_\theta$由:

$$Z_\theta = \sum_{x_1=0}^{x_1=1} \cdots \sum_{x_{d_x}=0}^{x_{d_x}=1} \sum_{h_1=0}^{h_1=1} \cdots \sum_{h_{d_h}=0}^{h_{d_h}=1} \exp\left(-\mathcal{E}_\theta^{\mathrm{BM}}(x,h;\theta)\right). \tag{8}$$

这个联合概率分布产生了条件集合形式的分布:

$$P(h_i \mid x, h_{\setminus i}) = \operatorname{sigmoid}\left(\sum_j W_{ji}x_j + \sum_{i' \neq i} V_{ii'}h_{i'} + d_i\right) \tag{9}$$

$$P(x_j \mid h, x_{\setminus j}) = \operatorname{sigmoid}\left(\sum_i W_{ji}x_j + \sum_{j' \neq j} U_{jj'}x_{j'} + b_j\right). \tag{10}$$

一般来说，玻尔兹曼机的推理是难以解决的。例如，在给定可见项$P(h_i \mid x)$的情况下，计算$h_i$的条件概率需要将其余隐藏项边缘化，这意味着用$2^{d_h-1}$项计算总和:

$$P(h_i \mid x) = \sum_{h_1=0}^{h_1=1} \cdots \sum_{h_{i-1}=0}^{h_{i-1}=1} \sum_{h_{i+1}=0}^{h_{i+1}=1} \cdots \sum_{h_{d_h}=0}^{h_{d_h}=1} P(h \mid x) \tag{11}$$

然而，有一些明智的选择在可见单元和隐藏单元之间的交互模式中，更多接下来我们将讨论模型族中可能存在的可处理子集。

受限玻尔兹曼机**(RBMs)**。 R BM可能是最受欢迎的子类玻尔兹曼机(Smolensky, 1986)。它的定义是通过限制的交互Bol tzmann能量函数，在等式7中，只有那些在$h$和$x$ 之间，即$\mathcal{E}_\theta^{\mathrm{RBM}}$是$\mathcal{E}_\theta^{\mathrm{BM}}$ 通过$U = \mathbf{0}$和$V = \mathbf{0}$。因此，R BM可以说是由可见图和在图中形成两层顶点的隐藏点(并且之间没有连接

同层单位)。有了这个约束条件下，RBM具有条件隐单元的分布可以在给定可见项的情况下分解:

$$P(h \mid x) = \prod_i P(h_i \mid x)$$
$$P(h_i = 1 \mid x) = \text{sigmoid}\left(\sum_j W_{ji} x_j + d_i\right). \quad (12)$$

同样，给定的可见单位上的条件分布Hiddens还可以分解:

$$P(x \mid h) = \prod_j P(x_j \mid h)$$
$$P(x_j = 1 \mid h) = \text{sigmoid}\left(\sum_i W_{ji} h_i + b_j\right). \quad (13)$$

这使得RBMs中的推理很容易处理。例如，RBM特征表示被认为是后验边缘的集合$P(h_i \mid x)$中描述的条件独立性Eq. 12，立即可用。注意，这是在与流行的有向图形模型的情况形成了鲜明的对比无监督特征提取，计算后验概率是棘手的。

重要的是，RBM的易处理性并没有扩展到它的分区函数，它仍然涉及指数项的求和。然而，它确实意味着我们可以将词条的数量限制为$\min\{2^{d_x}, 2^{d_h}\}$。通常这仍然是一个难以管理的术语数量因此，我们必须求助于近似方法来处理它估计。

RBM对.....领域的影响怎么说也不为过无监督特征学习与深度学习。它已经被用于令人印象深刻的各种应用，包括fMRI图像分类(Schmah *et al.*, 2009)，运动和空间转变(Taylor and Hinton, 2009; Memisevic and Hinton, 2010)，协同过滤(Salakhutdinov *et al.*, 2007)和自然图像建模(Ranzato and Hinton, 2010; Courville *et al.*, 2011b)。

### 6.3 RBM在实值数据上的推广

过去几年在定义方面取得了重要进展更好地捕获实值数据的RBM的推广，特别是实值图像数据，通过更好地建模输入像素的条件协方差。如前所述，标准的RBM定义都是二进制可见的变量$v \in \{0,1\}$和二元潜在变量$h \in \{0,1\}$。The RBM中推理和学习的可处理性启发了许多作者通过修改它的能量函数来扩展它，以模拟其他各种数据分布。特别是，有多个试图开发实值数据的rbm类型模型，其中$x \in \mathbb{R}^{d_x}$。这是实数建模最直接的方法RBM框架中的观测值是所谓的高斯*RBM* (GRBM)，其中RBM能量函数的唯一变化是可见的单位偏置，通过在可见单位中添加一个二次偏置项$x$。虽然它可能仍然是最流行的实值建模方法RBM框架内的数据Ranzato and Hinton (2010)表明GRBM已被证明是一种不太令人满意的自然模式图像。经过训练的特征通常不能表示尖锐的边缘发生在对象边界，并导致不存在的潜在表示特别有用的分类特征任务。Ranzato and Hinton (2010)认为GRBM的失败充分捕捉自然图像的统计结构源于独家使用模型容量来捕获at的条件均值条件协方差的开销。他们认为，自然图像就是如此主要特征是像素值的协方差，而不是它们绝对值。常用的预处理也支持这一点将像素值的全局缩放标准化的方法数据集中的图像或每个图像中的像素值。

这些都是关于GRBM建模能力的问题自然图像数据导致了替代基于rbm的发展每个模型都试图承担这个更好的建模目标非对角条件协方差。(Ranzato and Hinton, 2010)介绍均值和协方差*RBM* (mcRBM)。和GRBM一样，mcRBM也是2层玻尔兹曼机显式地将可见单元建模为高斯分布量。然而与GRBM不同的是mcRBM 使用其隐藏层独立参数化均值和通过两组隐藏单元的数据的协方差。mcRBM是一个组合协方差RBM (cRBM) (Ranzato *et al.*, 2010a)，它使用捕获的GRBM对条件协方差进行建模条件均值。而GRBM已经显示出相当大的作用潜力作为一个高度成功的音素识别系统的基础(Dahl *et al.*, 2010)，似乎由于困难在在训练mcRBM时，该模型在很大程度上已经被mPoT取代模型。mPoT模型(学生$t$分布的均值积) 模型)(Ranzato *et al.*, 2010b)是一个组合GRBM和学生t分布的乘积模型(Welling *et al.*, 2003)。这是一种基于能量的模型以隐藏单元为条件的可见单元的条件分布变量是多元高斯分布(非对角协方差) 给定隐变量上的互补条件分布可见项是一组独立的伽马分布。

壶模型最近已经泛化到mPoT模型(Ranzato *et al.*, 2010b)来包含非零高斯意味着通过添加类似grbm的隐藏单元，类似于mcRBM对cRBM的推广。mPoT模型已被用于合成大规模自然图像(Ranzato *et al.*, 2010b)显示大规模特征和阴影结构。它被用来模拟自然配置中的纹理(Kivinen and Williams, 2012) (参见11.2部分)。

另一个最近引入的基于rbm的模型，目的是拥有隐藏单元编码的均值和协方差信息是钉板限制玻尔兹曼机(ssRBM) (Courville *et al.*, 2011a,b)。ssRBM是定义为同时具有实数的'slab '变量和二进制变量' spike '变量与隐藏层中的每个单元相关联。ssRBM已被证明是一种特征学习和提取方案在CIFAR-10的背景下，从自然图像中进行对象分类(Krizhevsky and Hinton, 2009) 在角色(Courville *et al.*, 2011a,b)。训练时在完整的*CIFAR-10*自然图像上，该模型通过卷积(参见11.2部分)演示了能够生成自然图像这些样本似乎捕获了自然图像的广泛统计结构比之前的参数生成模型更好，如图2中的示例所示。

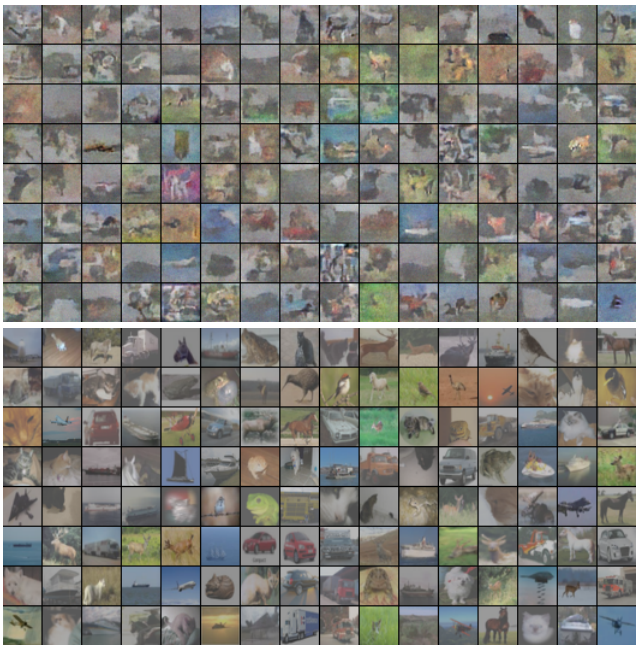

Fig. 2. (上图)卷积训练的样本$\mu$-ssRBM来自Courville *et al.* (2011b)。(底部)CIFAR-10中的图像与对应的训练集最近(具有对比度归一化训练图像的L2距离) 模型样本在顶部。这个模型似乎不是过拟合特定的训练示例。

mcRBM、mPoT和ssRBM都是为了对真实值数据进行建模隐藏单元不仅编码了数据的条件均值，还编码了数据的条件均值它的条件协方差。除了培训计划上的差异，这些模型之间最显著的区别是它们如何对条件协方差进行编码。而mcRBM和mPoT使用激活隐藏单元来强制约束$x$的协方差，ssRBM利用隐藏单元对精度矩阵进行压缩由相应的权重向量指定的方向。这两种建模条件协方差的方法在隐藏层的维数与输入层的维数有显著不同。在过完备设置中，使用ssRBM进行稀疏激活参数化只允许在稀疏的选定方向上发生变化激活隐藏单位。这是ssRBM与稀疏编码共享的属性模型(Olshausen and Field, 1996; Grosse *et al.*, 2007)。另一方面，在mPoT或mcRBM的情况下，对协方差意味着捕获任意协方差输入的特定方向可能需要减少all 在该方向上具有正投影的

约束。从这个角度来看，mPoT和mcRBM似乎并不适合在中提供稀疏表示过于完整的设置。

### 6.4 RBM参数估计

我们这里讨论的许多RBM训练方法如下适用于更一般的无向图形模型，但是特别适用于RBM环境。Freund and Haussler (1994) 提出了一种基于投影寻踪。对比Divergence (Hinton, 1999; Hinton *et al.*, 2006)最常用于训练RBMs，最近的许多论文使用随机最大似然(Younes, 1999; Tieleman, 2008)。

如6.1节所述，在训练概率模型时，参数通常是自适应的为了最大化训练数据的似然度(或等价于对数似然，或者它的惩罚版本，增加了正则化项)。通过$T$ 在训练的例子中，对数似然函数如下:

$$\sum_{t=1}^{T} \log P(x^{(t)};\theta) = \sum_{t=1}^{T} \log \sum_{h\in\{0,1\}^{d_h}} P(x^{(t)},h;\theta). \quad (14)$$

基于梯度的优化需要梯度，对于玻尔兹曼机，由:

$$\begin{aligned}\frac{\partial}{\partial\theta_i}\sum_{t=1}^{T}\log p(x^{(t)}) &= -\sum_{t=1}^{T}\mathbb{E}_{p(h|x^{(t)})}\left[\frac{\partial}{\partial\theta_i}\mathcal{E}_\theta^{\mathrm{BM}}(x^{(t)},h)\right] \\ &\quad +\sum_{t=1}^{T}\mathbb{E}_{p(x,h)}\left[\frac{\partial}{\partial\theta_i}\mathcal{E}_\theta^{\mathrm{BM}}(x,h)\right], \quad (15)\end{aligned}$$

我们对$p(h^{(t)} \mid x^{(t)})$有什么期望在“夹紧”的情况下(也称为正相)，并超过全关节$p(x,h)$ 在‘unclamped’条件下(也称为否定阶段)。直观地说，梯度的作用是局部移动模型朝向数据分布的分布(负相位分布) (正相位分布)，通过向下推能量$(h,x^{(t)})$成对($h \sim P(h|x^{(t)})$)，同时推高$(h,x)$的能量成对($(h,x) \sim P(h,x)$)，直到两个力处于平衡状态，此时足够的统计量(能量函数梯度) 与从训练分布中采样的$x$或与$x$从模型中采样。

RBM的条件独立性表明，在的正相位Eq. 15是容易处理的。负相项，由配分函数对对数似然梯度的贡献——更多问题在于关节上的期望计算不是容易处理。处理配分函数贡献的各种方法梯度带来了许多不同的训练算法，很多人试图近似对数似然梯度。

来近似联合分布的期望在对梯度的负阶段，自然需要再次考虑利用RBM的条件独立性来指定蒙特卡洛对关节的期望近似:

$$\mathbb{E}_{p(x,h)}\left[\frac{\partial}{\partial\theta_i}\mathcal{E}_\theta^{\mathrm{RBM}}(x,h)\right] \approx \frac{1}{L}\sum_{l=1}^{L}\frac{\partial}{\partial\theta_i}\mathcal{E}_\theta^{\mathrm{RBM}}(\tilde{x}^{(l)},\tilde{h}^{(l)}), \quad (16)$$

用a绘制的样本$(\tilde{x}^{(l)},\tilde{h}^{(l)})$ block Gibbs MCMC(马尔可夫链蒙特卡洛)采样程序:

$$\begin{aligned}\tilde{x}^{(l)} &\sim P(x \mid \tilde{h}^{(l-1)}) \\ \tilde{h}^{(l)} &\sim P(h \mid \tilde{x}^{(l)}).\end{aligned}$$

简单地说，对于每个梯度更新步骤，我们都会进行吉布斯采样链，等到链收敛到平衡分布和然后绘制足够数量的样本来近似期望的梯度关于Eq. 16中的模型(联合)分布。然后重新开始下一步的近似梯度上升过程对数似然。这个程序有一个明显的缺陷，就是等待吉布斯链到“老化”，每个梯度重新达到平衡更新不能构成实际训练算法的基础。对比散度(Hinton, 1999; Hinton *et al.*, 2006)，随机最大似然(Younes, 1999; Tieleman, 2008) 快速权重持续存在对比发散或FPCD (Tieleman and Hinton, 2009)都是方法避免或减少对老化的需要

#### *6.4.1* 对比散度

对比散度(CD)估计(Hinton, 1999; Hinton *et al.*, 2006) 估计负相位期望(Eq. 15) 吉布斯链很短(通常只有一个Step)在正训练数据上进行初始化阶段。这减少了梯度估计器的方差向负链样本的方向移动相关的正链样本。关于属性和其他解释已经有很多文章了CD 及其与自动编码器训练的相似性，例如Carreira-Perpiñan and Hinton (2005); Yuille (2005); Bengio and Delalleau (2009); Sutskever and Tieleman (2010)。

#### *6.4.2* 随机最大似然

随机最大似然(Stochastic Maximum Likelihood, SML)算法持续对比散度或PCD) (Younes, 1999; Tieleman, 2008)是另一种避免负相位老化的方法吉布斯采样器。在每次梯度更新时，而不是初始化吉布斯在CD中，SML初始化链在链的最后一个状态用于上一次更新。换句话说，SML使用a 连续运行的吉布斯链(或者经常是多个吉布斯链同时运行) 绘制哪些样本来估计负相位期望。尽管如此模型参数在更新之间发生变化，这些变化应该很小足以说明吉布斯只有几步(实际上往往是一步)所用的)均需保持样品的平衡分布吉布斯链，即模型分布。

SML的一个麻烦之处在于它依赖于吉布斯链来很好地混合(特别是在模式之间) 学会成功。通常，随着学习的进展和权重RBM增大，Gibbs样本的遍历性开始分解[12]。如果学习率$\epsilon$与梯度上升$\theta \leftarrow \theta + \epsilon\hat{g}$ (附$E[\hat{g}] \approx \frac{\partial \log p_\theta(x)}{\partial\theta}$) 并不是为了补偿，吉布斯采样器将偏离模型分布和学习就会失败。Desjardins *et al.* (2010); Cho *et al.* (2010); Salakhutdinov (2010b,a)拥有一切考虑到各种形式的缓和过渡到地址吉布斯链混合的失败，以及令人信服的解决方案还没有得到明确的证明。这是最近推出的一个有前途的途径依赖于深度本身，显示模式之间的混合要容易得多关于更深层次(Bengio *et al.*, 2013) (9.4节)。

Tieleman and Hinton (2009) 提出这是解决潜在混合问题的不同方法SML的快速权重持久性对比散度(FPCD)，它也被用于训练深度玻尔兹曼机(Salakhutdinov, 2010a) 并构造了RBMs (Breuleux *et al.*, 2011)的纯采样算法。FPCD建立在吉布斯令人惊讶但稳健的趋势之上在SML学习过程中，链比模型参数时更好地混合固定。这种现象的根源在于似然梯度的形式本身(Eq. 15)。从SML吉布斯链中提取样品都是在梯度的负阶段使用的，这意味着学习更新将略微增加能量(降低概率) 这些样本，使得这些样本的邻域更小可能会被重新采样，因此使样本更有可能将移动到其他地方(通常是靠近另一种模式)。而不是从当前模型的分布中抽取样本(带参数$\theta$)，FPCD夸大了这种效果是通过从一个局部扰动的模型中抽取样本来实现的参数$\theta^*$和更新

$$\theta^*_{t+1} = (1-\eta)\theta_{t+1} + \eta\theta^*_t + \epsilon^*\frac{\partial}{\partial\theta_i}\left(\sum_{t=1}^{T}\log p(x^{(t)})\right), \quad (17)$$

哪里$\epsilon^*$是相对较大的快速权重学习率($\epsilon^* > \epsilon$)和$0 < \eta < 1$(但接近1)是一个遗忘因素这使被扰动的模型与当前模型保持接近。不一样经过回火，FPCD不收敛于$\epsilon$那样的模型分布和$\epsilon^*$去0，和进一步的工作是必要的，以表征的性质它是模型分布的近似。然而，FPCD是一种这是一种流行且明显有效的从中提取近似样本的方法忠实地代表其多样性的模型分布有时会在两种模式之间产生虚假样本(因为快速权重大致对应于当前模型的平滑视图能量函数)。它已经在各种应用中得到了应用(Tieleman and Hinton, 2009; Ranzato *et al.*, 2011; Kivinen and Williams, 2012) 并转化为采样算法(Breuleux *et al.*, 2011) 它也和牧群分享这种快速混合的特性(Welling, 2009)，出于同样的原因，即引入负相关 连续的样品链以促进更快的混合。

#### *6.4.3* 伪似然，比率匹配等等

而CD、SML和FPCD是迄今为止最流行的训练方法rbm和基于rbm的模型，所有这些方法可能都是最自然的描述为提供最大可能性的不同近似值培训。还有其他的归纳原则可以替代最大似然函数也可用于训练RBMs。特别是，这些包括伪似然(Besag, 1975)和比率匹配(Hyvärinen, 2007)。这两个归纳原则尽量避免显式地处理分区函数，并分析了它们的渐近效率(Marlin and de Freitas, 2011)。伪似然性寻求最大化所有的

12. 当权重变大，估计的分布更加尖峰，并且Chain需要很长的时间来混合，从一个模式移动到另一个模式，所以实际上梯度估计器可能非常差。这是一个严重的鸡和蛋的问题因为如果采样不有效，训练过程也不有效，这可能似乎停滞不前，并产生更大的重量。

乘积一维条件分布形式$P(x_d|x_{\setminus d})$，而比率匹配可以解释为分数匹配的扩展(Hyvärinen, 2005) to 离散数据类型。这两种方法都是加权的RBM自由能梯度的差异[13]在一个数据点和在邻近点。这些方法的一个潜在缺点是，依赖于的参数化能量函数，其计算需求可扩展到$O(n_d)$比差CD、SML、FPCD或去噪分数匹配(Kingma and LeCun, 2010; Vincent, 2011)，下面讨论。Marlin *et al.* (2010)根据经验比较了所有这些方法(除去噪得分匹配) 在一系列的分类，重建和密度建模任务和发现，一般来说，SML提供了最佳的组合总体性能和计算可操作性。然而，在后来研究中，同样的作者(Swersky *et al.*, 2011)发现了去噪分数匹配互为竞争的归纳原则在分类性能方面(相对于SML)和在计算效率方面( 关于分析获得分数匹配)。去噪得分匹配是去噪自编码器训练的特例吗判据(7.2.2节)当重建误差残差等于梯度，即相关的得分函数与能量函数，如(Vincent, 2011)所示。

遵循玻尔兹曼机梯度的精神(Eq. 15) 已经提出了几种基于能量的训练方法模型。一种是噪声对比估计(Gutmann and Hyvarinen, 2010)，其中训练准则转化为概率分类问题: 区分(正的)训练样本和(负的)噪声由宽分布(如高斯分布)生成的样本。另一个家族的方法，更多的是在精神上的对比分歧，依赖关于区分(训练分布的)正例和通过对正例的扰动得到负例示例(Collobert and Weston, 2008; Bordes *et al.*, 2012; Weston *et al.*, 2010)。

## 7 直接学习从输入到表示的参数映射

在采用的概率模型框架内6 部分，学习到的表示总是与潜在的相关联变量，特别是给定观测值的后验分布输入$x$。不幸的是，这是后验分布往往会变得非常复杂和棘手，如果模型有多个相互连接的层，无论是定向的还是无向图模型框架。然后就有必要求助了以抽样或近似推理技术，与支付相关联计算和近似误差代价。若真后验有大重要模式的数量那么目前的推理技术可能面临一个无法克服的问题挑战或忍受一个潜在的严重近似。这是除了无向的难解配分函数所带来的困难图形模型。此外，潜在的后验分布 变量还不是一个简单可用的特征向量 将示例提供给分类器。所以实际的特征值从这个分布中推导出来的，取潜变量期望(RBMs通常这样做)，它们的边际概率，或者寻找它们最可能的值(如稀疏编码)。如果我们要最后提取稳定的确定性数值特征值另一种(显然)非概率特征学习范式重点是高效地执行这部分计算自动编码器和其他直接参数化的特征或表示函数。这些方法的共同点是它们学习一种直接编码，即从输入到参数映射他们的代表。

接下来讨论的正则化自编码器也涉及学习将表示映射回输入空间的解码函数。章节8.1 和11.3讨论不需要的直接编码方法解码器，如半监督嵌入(Weston *et al.*, 2008)和慢特征分析(Wiskott and Sejnowski, 2002)。

### 7.1 自动编码器

在自动编码器框架(LeCun, 1987; Bourlard and Kamp, 1988; Hinton and Zemel, 1994)中，首先显式地定义a 特定参数化封闭形式的特征提取函数。这个函数，我们将其表示为$f_\theta$，称为编码器 并允许进行简单而高效的计算从输入$x$得到的特征向量$h = f_\theta(x)$。对于来自数据集$\{x^{(1)}, \ldots, x^{(T)}\}$的每个示例$x^{(t)}$，我们定义

$$h^{(t)} = f_\theta(x^{(t)}) \tag{18}$$

$h^{(t)}$是特征向量还是从$x^{(t)}$计算的表示或代码。另一个封闭形式的参数化函数$g_\theta$，称为解码器，将特征空间映射回输入空间，产生a 重建$r = g_\theta(h)$。而概率模型是从显式概率函数定义，并训练为最大化(通常近似)数据可能性(或代理)，自动编码器是通过它们的编码器和解码器进行参数化，并使用不同的训练原则。参数的设置$\theta$ 编码器和解码器在任务中同时学习尽可能重建原始输入，即尝试招致尽可能低的重建误差$L(x, r)$——一个措施$x$与重建之间的差异$r$——结束训练示例。良好的泛化能力意味着较低的重构误差在测试示例，而对于大多数其他$x$配置具有高重构误差。的结构数据生成分布，因此重要的是在训练准则或参数化阻碍了自动编码器从具有零重构性的恒等函数学习得到错误无处不在。这是通过各种不同的手段来实现的自动编码器的形式，如下所述，我们称之为这些正则化的自动编码器。正则化的一种特殊形式在于约束代码具有低维度，这是经典的自动编码器或PCA做什么。

总而言之，基本的自动编码器训练包括找到的值参数向量$\theta$ 最小化重构误差

$$\mathcal{J}_{\text{AE}}(\theta) = \sum_t L(x^{(t)}, g_\theta(f_\theta(x^{(t)}))) \tag{19}$$

其中$x^{(t)}$是一个训练示例。这种最小化通常通过随机梯度下降来实现多层感知器(mlp)的训练。因为自动编码器主要是作为mlp开发的预测他们的输入最常用的编码器和解码器形式是仿射映射，可选地，后面跟着一个非线性函数:

$$f_\theta(x) = s_f(b + Wx) \tag{20}$$
$$g_\theta(h) = s_g(d + W'h) \tag{21}$$

其中$s_f$和$s_g$是编码器和解码器的激活函数(通常是sigmoid函数或双曲正切非线性函数，或者恒等函数if 保持线性)。这种模型的参数集是$\theta = \{W, b, W', d\}$，其中$b$和$d$称为编码器和解码器偏差向量，$W$和$W'$是编码器和解码器权重矩阵。

$s_g$和$L$的选择很大程度上取决于输入域的范围和自然，通常选择，以便$L$返回负对数似然为$x$的观测值。无界区域的一个自然选择是A的平方的线性解码器重构错误，即$s_g(a) = a$和$L(x, r) = \|x - r\|^2$。然而，如果输入在0和1之间有界，则确保a 使用$s_g = \text{sigmoid}$可以实现类似的边界重建。此外，如果输入具有二进制性质，有时使用二进制交叉熵损失[14]。

如果编码器和解码器都使用sigmoid非线性，则$f_\theta(x)$ 而且$g_\theta(h)$ 有精确的相同形式 作为条件语句$P(h \mid v)$ 而且$P(v \mid h)$ 二元RBMs (见章节) 6.2)。这种相似性促使了一项初步研究(Bengio *et al.*, 2007) 的替换rbm的可能性自动编码器作为构建深度的基本预训练策略网络以及自编码器重构的对比分析误差梯度和对比散度更新(Bengio and Delalleau, 2009)。

参数化的一个显著区别是RBMs使用一个权重矩阵，如下所示自然地从他们的能量函数，而自动编码器框架允许一个编码器和解码器中的矩阵不同。然而，在实践中，定义$W' = W^T$的权重可以(也是最常用的)使用，使参数化相同。然而，两者之间通常的训练程序有很大的不同方法。训练的实际优势自编码器的变种是它们定义了一个简单的可处理的可用于监控进度的优化目标。

对于具有平方重构误差的线性自编码器(线性编码器和解码器)，最小化Eq. 19学到了同样的东西子空间[15] as pca。当使用sigmoid非线性时也是如此在编码器(Bourlard and Kamp, 1988)中，但不是如果权重$W$和$W'$相等($W' = W^T$)，因为$W$不能被迫变得小而$W'$大来实现线性编码器。

类似地，Le *et al.* (2011b)最近显示了添加正则化形式$\sum_t \sum_j s_3(W_j x^{(t)})$的项线性自动编码器与捆绑权重，其中$s_3$是一个非线性凸函数，得到an 一种学习线性*ICA*的有效算法

13. The 自由能$\mathcal{F}(x;\theta)$是与数据边际概率相关联的能量，$\mathcal{F}(x;\theta) = -\log P(x) - \log Z_\theta$和对RBM来说是可操作的。

14. $L(x, r) = -\sum_{i=1}^{d_x} x_i \log(r_i) + (1 - x_i)\log(1 - r_i)$

15. 与传统的PCA载荷因子相反，但是与概率PCA学习到的参数类似，线性自编码器学习到的权重向量不受约束形成标准正交基，也不受约束有一个有意义的排序。但是它们会张成相同的子空间。

### 7.2 正则化自编码器

与PCA一样，自动编码器最初被视为一个维度因而使用了一个瓶颈，即$d_h < d_x$。另一方面，稀疏编码和RBM方法的成功使用则更受青睐过度完整的表示，即$d_h > d_x$。这可以允许自动编码器简单地复制输入中的特征，具有perfect 在没有提取更有意义特征的情况下进行重建。最近的研究表明非常成功另一种称为正则化自动编码器的方法来“约束” 代表，即使它是过度完整的。瓶颈或正则化的影响自动编码器不能很好地重建一切，经过训练，可以很好地重建训练样本和泛化能力意味着测试样本的重构误差也很小。对稀疏性惩罚的一个有趣的理由(Ranzato *et al.*, 2008) (或者任何以软方式限制hidden体积的惩罚学习者容易获得的配置)是它在精神上行动就像RBMs的配分函数，通过确保只有少量的输入配置可以有较低的重构误差。

或者，可以将正则化应用于自编码器的目标视为在尊重的情况下，使表示尽可能为“常量‘(不敏感) 输入的变化。这种观点立即证明了正则化的两种变体是正确的自动编码器描述如下:收缩自动编码器减少表示的有效自由度(围绕每个点)通过使编码器收缩，即使编码器的导数小(从而使隐藏单元饱和)，而去噪自编码器则是整体映射“鲁棒”，即对小的随机扰动不敏感，或者收缩，确保当在大多数方向上移动时，重建不能保持良好一个训练示例。

#### *7.2.1* 稀疏自编码器

单层自动编码器最早用于构建深度架构通过堆叠(Bengio *et al.*, 2007)考虑了捆绑的想法编码器权重和解码器权重以限制容量以及引入一种形式的稀疏正则化的想法(Ranzato *et al.*, 2007)。表示中的稀疏性通过惩罚隐藏的可以实现吗单位偏差(使这些加性偏移参数更负面)(Ranzato *et al.*, 2007; Lee *et al.*, 2008; Goodfellow *et al.*, 2009; Larochelle and Bengio, 2008) 或者直接惩罚隐藏单元激活的输出(使它们更接近它们的饱和值在0)(Ranzato *et al.*, 2008; Le *et al.*, 2011a; Zou *et al.*, 2011)。惩罚偏差可能会导致权重补偿偏差，这可能会损害数值优化。当直接惩罚隐藏单元输出时，有几个变体可以在文献中找到，但清晰的对比分析仍然缺乏。尽管L1惩罚项(即，在非线性sigmoid的情况下，仅仅是输出元素的和$h_j$) 看起来是最自然的(因为它用于稀疏编码)，不是吗在涉及稀疏自编码器的少数论文中使用。的近亲L1惩罚是学生-t惩罚($\log(1+h_j^2)$)，最初提出是为了稀疏编码(Olshausen and Field, 1996)。几篇论文惩罚平均输出$\bar{h}_j$(例如，在一个小批量上)，相反把它推到0，鼓励它接近一个固定的目标，要么通过均方误差惩罚，或者更明智的做法(因为$h_j$的行为类似于概率)，关于的二项分布的kullback - libler散度概率$\rho$: $-\rho\log\bar{h}_j-(1-\rho)\log(1-\bar{h}_j)$+常数，例如，用$\rho=0.05$。

#### *7.2.2* 降噪自动编码器

Vincent *et al.* (2008, 2010) 提出改变培养目标在Eq. 19来自mere重构为去噪人为损坏的输入，例如，学习从损坏的版本中重建干净的输入。学习身份不再是足够的:学习者必须捕获结构的影响，以最佳地撤销的影响腐败过程，重建基本上是在附近，但密度更高比损坏的输入点更重要。图3说明了这一点降噪自编码器(DAE)正在学习一个重构函数对应指向高密度区域的向量场(流形例子集中的地方)。

形式上，DAE优化的目标为:

$$\mathcal{J}_{\text{DAE}} = \sum_t \mathbb{E}_{q(\tilde{x}|x^{(t)})}\left[L(x^{(t)}, g_\theta(f_\theta(\tilde{x})))\right] \tag{22}$$

$\mathbb{E}_{q(\tilde{x}|x^{(t)})}[\cdot]$平均超过腐败的例子$\tilde{x}$来自腐败的过程$q(\tilde{x}|x^{(t)})$。在实践中，这是随机优化的梯度下降，其中随机梯度通过画一个来估计或者每次考虑$x^{(t)}$时$x^{(t)}$的几个损坏版本。在Vincent *et al.* (2010)中考虑的腐败包括加性各向同性高斯噪声，灰度图像的椒盐噪声和掩蔽噪声例如，将一些随机选择的输入设置为0(独立设置) 每个例子)。大多数仿真都使用了掩码噪声。去噪后得到了质量更好的特征在改进的分类器中，使用DAE进行特征提取与RBM特征相似或更好。Chen *et al.* (2012) 展示一个更简单的闭式解决方案可以限定为线性时得到的并成功地将其应用于领域自适应。

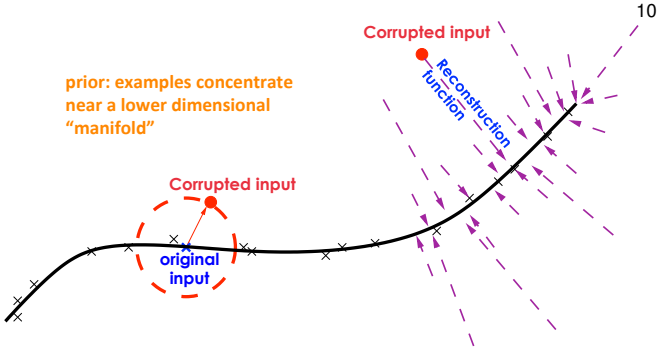


Fig. 3. 当数据集中在低维流形附近时，腐蚀向量通常几乎与流形正交，重建函数学习去噪，从低概率进行映射配置(损坏的输入)到高概率的(原始输入)，创建一个与得分对齐的向量场(导数估计密度)。

Vincent (2011) 关联DAEs 到基于能量的概率模型: DAEs基本上在$r(\tilde{x})-\tilde{x}$学习指向估计分数方向的向量$\frac{\partial \log p(\tilde{x})}{\partial \tilde{x}}$ (图3)。在线性重构和平方误差的特殊情况下，Vincent (2011)显示了训练仿射-sigmoid-仿射DAE 相当于学习该模型的能量函数与a的能量函数非常接近grbm。训练使用分数匹配参数的正则化变体估计技术(Hyvärinen, 2005; Hyvärinen, 2008; Kingma and LeCun, 2010) 去噪分数匹配(Vincent, 2011)。Swersky (2010)已经表明训练GRBMs使用分数匹配等同于训练一个普通的带有额外正则化项的自动编码器在Vincent (2011), Swersky *et al.* (2011)上的理论结果展示了去噪实现分数匹配的实际优势高效。最后Alain and Bengio (2012)概括Vincent (2011) 并证明了任意参数化的DAEs 小的高斯腐败噪声是分数的一般估计器。

#### *7.2.3* 收缩式自动编码器

收缩自动编码器(Contractive Auto-Encoders, CAE)，由Rifai *et al.* (2011a)提出，后续降噪自编码器(DAE)和学习鲁棒性的动机相似表示。CAEs通过添加分析的收缩惩罚来实现这一点 致Eq. 19: the Frobenius norm of the 编码器的雅可比矩阵，并导致惩罚的敏感性习得特征到无穷小的输入变化。让$J(x)=\frac{\partial f_\theta}{\partial x}(x)$的雅可比矩阵编码器在$x$。中国工程院的培训目标是

$$\mathcal{J}_{\text{CAE}} = \sum_t L(x^{(t)}, g_\theta(f_\theta(x^{(t)}))) + \lambda\left\|J(x^{(t)})\right\|_F^2 \tag{23}$$

其中$\lambda$是控制正则化强度的超参数。对于仿射sigmoid编码器，收缩惩罚项很容易计算:

$$\begin{aligned} J_j(x) &= f_\theta(x)_j(1-f_\theta(x)_j)W_j \\ \|J(x)\|_F^2 &= \sum_j (f_\theta(x)_j(1-f_\theta(x)_j))^2\|W_j\|^2 \end{aligned} \tag{24}$$

与DAEs至少有三个显著的差异，这可能是部分原因CAE特征似乎在经验上证明了更好的性能:(a)敏感性的功能受到惩罚[16]而不是重建;(b)惩罚是分析的而不是随机的:这是一种有效可计算的方法表达式替换可能需要$d_x$损坏的样本大小(即在$d_x$方向上的灵敏度);(c)超参数$\lambda$允许对重构和鲁棒性之间的权衡进行精细控制(而两者在DAE中混合)。但是请注意，DAE和CAE之间有紧密的联系:如图所示在(Alain and Bengio, 2012) a 噪音腐蚀小可以(通过泰勒展开)看到作为一种收缩式自动编码器，其中收缩式惩罚是对整个重建功能而不仅仅是对编码器[17]。

16. 即，的鲁棒性鼓励代表。
17. 但是注意，在CAE中，为了避免，解码器权重与编码器权重绑定在一起退化解，这也应该使解码器收缩。

CAE的分析惩罚的一个潜在缺点是它相当于只鼓励对无穷小输入变化的鲁棒性。这在Rifai *et al.* (2011b)中得到了补救CAE+H，以一种有效的随机方式惩罚了所有的高阶导数，通过添加一个鼓励$J(x)$和$J(x+\epsilon)$接近的术语:

$$\mathcal{J}_{\text{CAE+H}} = \sum_t L(x^{(t)}, g_\theta(x^{(t)})) + \lambda \left\| J(x^{(t)}) \right\|_F^2 + \gamma \mathbb{E}_\epsilon \left[ \|J(x) - J(x+\epsilon)\|_F^2 \right] \quad (25)$$

$\epsilon \sim \mathcal{N}(0, \sigma^2 I)$和$\gamma$在哪里关联正则化强度超参数。

DAE和CAE已成功应用于无监督和转移的最后阶段学习挑战(Mesnil *et al.*, 2011)。CAE学习到的表示趋于饱和而不是稀疏，即大多数隐藏单元都接近极端它们的范围(例如0或1)，以及它们的导数$\frac{\partial h_i(x)}{\partial x}$ 接近0。非饱和单元很少，对输入很敏感相关的滤波器(权重向量)一起形成了一个基础，解释了$x$周围的局部变化，如8.2节所述。另一种方法饱和(几乎二元)单位是语义哈希(Salakhutdinov and Hinton, 2007)。

### *7.2.4* 预测稀疏分解

稀疏编码(Olshausen and Field, 1996)可以看作是一种使用平方重构线性解码器的自动编码器错误，但其非参数编码器$f_\theta$执行相对不平凡和相对昂贵迭代最小化Eq. 2。是稀疏编码和自动编码器的一个非常成功的变体，名为预测性稀疏分解 或PSD (Kavukcuoglu *et al.*, 2008)用快速的非迭代近似取代了昂贵的和高度非线性的编码步骤识别过程中(计算学习到的特征)。PSD已被应用于图像中的物体识别视频(Kavukcuoglu *et al.*, 2009, 2010; Jarrett *et al.*, 2009)，而且还以音频(Henaff *et al.*, 2011)为主，大多在里面多阶段卷积深度架构框架(11.2部分)。主要思想可以概括由下面的公式为训练准则，即同时优化了关于隐藏代码(表示)$h^{(t)}$和with 尊重参数$(W, \alpha)$:

$$\mathcal{J}_{\text{PSD}} = \sum_t \lambda \|h^{(t)}\|_1 + \|x^{(t)} - W h^{(t)}\|_2^2 + \|h^{(t)} - f_\alpha(x^{(t)})\|_2^2 \quad (26)$$

其中$x^{(t)}$是输入向量，例如$t$, $h^{(t)}$是该示例的优化隐藏代码，$f_\alpha(\cdot)$是编码函数，最简单的变体是

$$f_\alpha(x^{(t)}) = \tanh(b + W^T x^{(t)}) \quad (27)$$

其中编码权重是解码的转置重量。已经提出了许多变体，包括使用收缩操作代替双曲正切(Kavukcuoglu *et al.*, 2010)。请注意，$h$上的L1惩罚往往使它们变得稀疏这和稀疏编码的标准是一样的吗字典学习(Eq. 3)除了额外的约束人们应该能够近似的稀疏代码$h$与参数化编码器$f_\alpha(x)$。因此，我们可以将PSD看作是稀疏编码的一种近似，我们得到一个快速的近似编码器。一旦PSD被训练，对象表示$f_\alpha(x)$用于提供分类器。它们是计算出来的快速且可以进一步微调: 编码器可以看作是一个阶段或一层可训练的多阶段系统，如前馈神经网络。

PSD也可以被视为一种自动编码器在哪里，代码$h$被赋予了一些自由，可以提供帮助进一步完善重建工作。还可以查看编码在稀疏编码的基础上增加了惩罚，作为一种正则化强制稀疏代码几乎可计算的平滑和高效编码器。这与获得的代码形成了对比通过对稀疏编码准则的完全优化，得到是非光滑的，甚至是不可微的，有问题吗有动机其他平滑稀疏编码的推断代码的方法(Bagnell and Bradley, 2009)，因此，稀疏编码阶段可以与深度架构的以下阶段。

## 8 表示学习作为流形学习

表示学习的另一个重要观点是基于几何概念多方面的。它的前提是歧管假设，根据哪个呈现在高维空间中的真实数据被期望集中在低得多的歧管$\mathcal{M}$附近维度$d_{\mathcal{M}}$，嵌入在高维输入中太空$\mathbb{R}^{d_x}$。这一先验似乎特别适合对于那些涉及图像、声音或文本的人工智能任务，for 哪些最均匀采样的输入配置与自然配置不同刺激。一旦有了'表示'的概念"，那么我们可以通过考虑输入空间的变化来考虑流形，被捕获或反映(通过相应的变化) 在学习到的表示中。粗略地说，一些方向被很好地保留了下来(切线方向) 流形)，而其他不是(与流形正交的方向)。从这个角度来看，主要的无监督学习任务被视为对数据支持流形结构[18]。可以将正在学习的关联表示与关联嵌入式流形上的内在坐标系统。毫无疑问，原型流形建模算法是: 还有原型低维表示学习算法: 主成分分析，对a进行建模线性流形。它最初的设计目标是寻找与数据点云最近的线性流形。主成分，即表示$f_\theta(x)$, PCA为输入点$x$生成的表示，唯一地定位它在流形上的投影:它对应流形上的内在坐标。然而，现实世界复杂领域的数据流形是预期的强非线性。他们的建模有时接近于局部线性的拼接切空间(Vincent and Bengio, 2003; Brand, 2003)。绝大多数算法都建立在这种几何角度上采用一种基于训练集最近邻的非参数方法图(Schölkopf *et al.*, 1998; Roweis and Saul, 2000; Tenenbaum *et al.*, 2000; Brand, 2003; Belkin and Niyogi, 2003; Donoho and Grimes, 2003; Weinberger and Saul, 2004; Hinton and Roweis, 2003; van der Maaten and Hinton, 2008)。在这些非参数方法，每个高维训练Point有自己的自由低维集合嵌入坐标，这是优化的，使某些属性的邻域在原始高维输入空间中计算的图被最好地保留。然而，这些方法不能直接学习参数化特征提取功能$f_\theta(x)$适用于新测试点[19]，其中严重限制了它们作为特征提取器的使用，除非在转导设置中。相对较少的非线性流形学习方法被提出参数映射，可以直接计算新点的表示;我们将关注这些。

### 8.1 基于邻域图的参数化映射学习

上面的一些非参数流形学习算法可以修改为学习一个参数地图$f_\theta$，即适用于新的点: 而不是自由的低维空间每个训练点的嵌入坐标"参数"，这些坐标通过显式参数化函数获得，与参数变体一样(van der Maaten, 2009) t -SNE (van der Maaten and Hinton, 2008)。

相反，半监督嵌入(Weston *et al.*, 2008) 学习一种直接编码通过邻域图考虑流形假设。参数化神经网络架构同时学习流形嵌入和分类器。培训标准鼓励训练集的邻居具有相似的表示。

该参数中自由参数的减少和严格控制的数量方法，与它们的纯非参数对应，力模型泛化流形形状非本地(Bengio *et al.*, 2006b) 可以转化为更好的功能和最终性能(van der Maaten and Hinton, 2008)。然而，基于训练集邻域对流形进行建模在高维空间中，两性关系可能存在统计学上的风险(由于维度诅咒，人口稀疏)例如，大多数欧几里得最近邻(Euclidean nearest neighbors)在语义上存在共同点太少的风险。最近邻图的密度不够，无法映射令人满意的是，目标的皱纹流形(Bengio and Monperrus, 2005; Bengio *et al.*, 2006b; Bengio and LeCun, 2007)。它计算上也会成问题吗数据点对[20]，它与训练集呈二次增长尺寸。

### 8.2 非线性流形的表示学习

我们可以在不需要最近邻搜索的情况下学习流形吗?是的，为了例如，使用正则化自动编码器或PCA。在PCA中，提取的组件(代码)对输入变化的敏感性无论职位如何$x$。切空间是一样的沿着线性流形到处都是。相比之下，对于非线性流形，流形的切线随着我们的变化而变化移动歧管，如图6所示。在非线性表示学习算法中，这是很方便的将表示中的局部变化考虑为输入$x$在流形上是不同的，即，当我们在其中移动时高概率配置。正如我们下面讨论的，的一阶导数因此，编码

18. 实际上，严格来说，数据点不一定会说谎在"流形"上，但概率密度预计将急剧下降为1 远离它，它实际上可能由几个组成可能是具有不同内在特性的不连通流形维度。

19. 对于几个这些表示新点的技术可以使用Nyström进行计算近似as已被提议作为(Bengio *et al.*, 2004)的扩展，但是这仍然很麻烦，计算成本很高。

20. 即使成对是随机选择的，很多必须在获得一个对优化目标。

器指定了形状流形(其切平面)周围的一个例子$x$躺在上面。如果密度集中在流形上，编码器捕获了这个，我们会找到编码器的导数只有在切平面张成的方向上是非零的。

让我们从这个角度考虑稀疏编码:参数矩阵$W$可能是解释为输入方向的字典，其中将包含不同的子集 选择在流形上的$x$处建模局部切空间。这个子集对应到主动，即非零，特征输入$x$。非零分量$h_i$ 会对输入方向的微小变化敏感吗关联权重向量$W_{:,i}$，而非活动特征更容易被卡住直到输入空间发生显著位移。

局部坐标编码(LCC)算法(Yu *et al.*, 2009) 非常类似于稀疏编码，但显式地源自流形透视。使用与稀疏编码相同的表示法Eq. 2, LCC 取代正则化项$\|h^{(t)}\|_1 = \sum_j |h_j^{(t)}|$产生目标

$$\mathcal{J}_{\text{LCC}} = \sum_t \left( \|x^{(t)} - W h^{(t)}\|_2^2 + \lambda \sum_j |h_j^{(t)}| \|W_{:,j} - x^{(t)}\|^{1+p} \right) \tag{28}$$

这是相同的稀疏编码时$p = -1$，但与更大的$p$它鼓励活动锚点$x^{(t)}$(即码本向量$W_{:,j}$ 与不可忽视的$|h_j^{(t)}|$相结合重构$x^{(t)}$)离$x^{(t)}$不远，因此算法的局部方面。重要的理论贡献Yu *et al.* (2009)就是为了说明这一点任何Lipschitz-smooth函数$\phi : \mathcal{M} \to \mathbb{R}$ 在光滑上定义非线性流形$\mathcal{M}$嵌入$\mathbb{R}^{d_x}$ 可以被一个全局线性函数很好地逼近尊重由此产生的编码方案(即$h$中的线性)，其中的准确性锚点的近似值和所需数量$d_h$取决于$d_{\mathcal{M}}$而不是$d_x$。进一步推广了这一结果局部切线方向的使用(Yu and Zhang, 2010)，以及多层(Lin *et al.*, 2010)。

现在让我们考虑高效的非迭代"前馈"编码器$f_\theta$，由PSD和在中审查的自动编码器使用7.2部分，在格式为Eq. 20或27。$x$的计算表示只对非常敏感与非饱和隐藏相关联的输入空间方向单位(参见例子Eq. 24的sigmoid的雅可比矩阵层)。表示的这些方向如PCA或稀疏编码，显著敏感的情况可以被视为在训练点$x$上跨越流形的切线空间。

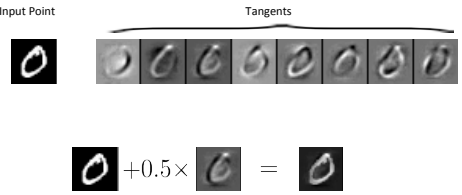


Fig. 4. 高密度流形的切向量据收缩自动编码器估计(Rifai *et al.*, 2011a)。原始输入显示在左上角。每个切向量(图像在第一排右侧) 对应于原始输入的一个合理的加性变形，如第二行所示，其中第三行有点奇异向量与原始图像相加，形成平移和变形的图像。不一样在PCA中，对于不同的输入，切向量是不同的，因为估计流形是高度非线性的。

Rifai *et al.* (2011a) 据此对奇异值进行实证分析雅可比矩阵的谱(表示向量对输入向量) 训练有素的CAE。这里SVD提供了一个有序的最敏感方向的标准正交基。频谱急剧下降，表明相对较少的几个显著敏感方向。这被认为是CAE确实建模的经验证据低维流形的切空间。开头的单数向量构成估计流形的切平面的一组基，如图4所示。CAE标准是相信这要归功于它的两个相反的术语:各向同性契约惩罚，鼓励代表平等对任何输入方向的变化和重建不敏感术语，这将不同的训练点(特别是邻居)推向有不同的表示(所以它们可以被精确地重建)，这样就抵消了各向同性的收缩压力，只在方向上与流形相切。

通过的视角分析学习到的表示雅可比矩阵的谱并将其与流形切空间的概念联系起来是可行的，无论映射是可微的，无论它是怎样的无论是直接编码(如自动编码器变体)，还是学习从隐变量推断(如稀疏编码或RBMs)派生。精确的低维流形模型(如PCA)将产生非零与流形方向相关的奇异值，以及精确零用于与流形正交的方向。但很顺利像CAE或RBM这样的模型我们将取而代之大的和相对小的奇异值(相对于非零和完全没有)。

### 8.3 利用建模的切空间

沿着流形的一点的局部切线空间可以被认为是捕获局部有效的变换，在训练数据。例如Rifai *et al.* (2011c)检查的雅可比矩阵的SVD提取切线方向在数字、图像或文本文档数据上训练的cae:它们似乎对应图像或数字的小平移或旋转，以及替换文档中同一主题内的单词。沿着数据流形的这种非常局部的变换不会改变类身份。为了构建他们的流形切线分类器(MTC)，Rifai *et al.* (2011c)然后应用诸如切线距离(Simard *et al.*, 1993)和切线传播(Simard *et al.*, 1992)，那最初是为了构建而开发的对输入变形不敏感的分类器提供先验领域知识。现在这些技术得到了应用利用提取的局部领先切线方向通过CAE，即不使用任何先验领域知识(除了关于存在歧管的宽泛先验)。这种方法创造了MNIST数字的新记录无先验知识方法的分类[21]。

## 9 概率和直接编码模型之间的联系

概率模型的标准似然框架分解带参数模型的训练准则$\theta$ 分为两部分:对数似然$\log P(x|\theta)$ (或$\log P(x|h,\theta)$与潜变量$h$)，以及先验$\log P(\theta)$(或$\log P(h|\theta) + \log P(\theta)$与潜变量)。

### 9.1 PSD:一种概率解释

在PSD算法的情况下，连接可以是在上述标准概率视图与直接编码计算之间进行了比较图形。PSD的概率模型是相同的定向生成模型模型$P(x|h)$的稀疏编码(章节6.1.1)，其中只负责解码器。编码器被视为一种近似推理机制来猜测$P(h|x)$和初始化一个MAP迭代推理(其中考虑到稀疏先验$P(h)$)。然而，在PSD中，编码器与解码器联合训练，而不是简单地将迭代推理的最终结果作为目标近似地。一个有趣的观点[22]来调和这些事实，编码器是一个变分下界映射解的参数逼近关于联合对数似然。当地图学习被视为一种特殊情况变分学习(其中联合对数似然的近似狄拉克分布位于映射解)，变分菜谱告诉我们要同时提高可能性(减少重建改进变分近似(减少差异在编码器输出和隐变量值之间)。因此，PSD位于概率模型的交叉点(与潜在的变量)和直接编码方法(直接参数化从输入到表示的映射)。十字路口也有RBMs 因为它们的特定参数化包括一个显式的由于连接受限，从输入到表示的映射隐藏单元之间。然而，这个好特性并没有延伸到它们自然的深度归纳，即深度玻尔兹曼机，在10.2部分讨论。

### 9.2 正则化自编码器捕获密度的局部结构

我们还可以谈谈概率解释吗正则化的自动编码器? 他们的训练标准不符合标准的似然框架因为这将涉及到一个依赖于数据的'先验'。最近出现了一个有趣的假设来回答这个问题理论的结果(Vincent, 2011; Alain and Bengio, 2012): 正则化自编码器的训练准则，而不是最大似然的形式，对应于不同的归纳原则，如分数匹配。分数匹配连接在7.2.2 部分，并已显示为一个特定的参数化和等效高斯函数RBM (Vincent, 2011)。工作在Alain and Bengio (2012)中，将这一想法推广到更广泛的领域参数化类(任意编码器和解码器)和显示通过正则化自动编码器使其收缩，得到了重构函数它的导数估计了基础的一阶和二阶导数 数据生成密度。这个视图可以被利用成功地从自动编码器中采样，如下所示在Rifai *et al.* (2012); Bengio *et al.* (2012)。建议的采样算法是MCMC，类似于Langevin MCMC，使用不仅是估计的密度的一阶导数，而且估计的流形切线以便保持接近流形高密度的。

这种解释与引入的几何透视很好地联系在一起在8部分。正则化效应(例如，由于a 稀疏正则化，收缩正则化，或去噪Criterion)要求学习到的表示尽可能不敏感，同时最小化训

21. 结果是0.81%的误差使用完整MNIST训练集的率，没有先验变形，没有卷积。

22. 由Ian Goodfellow提出，个人沟通

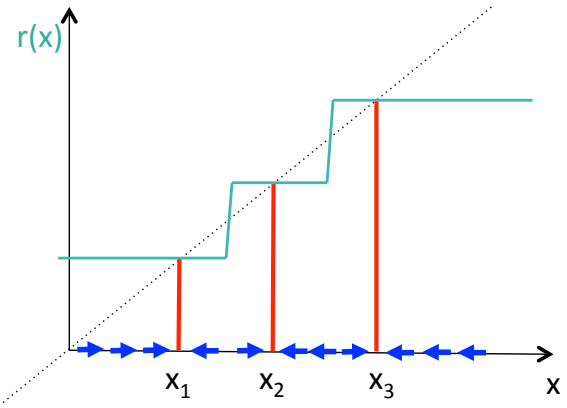


Fig. 5. 重构功能$r(x)$(绿色) 由大容量自动编码器学习1维输入，最小化训练示例的重构误差$x^{(t)}$ (红色的$r(x^{(t)})$ 尽量保持不变。虚线是特征重构(可以在没有正则化项的情况下获得)。蓝色箭头显示$r(x) - x$指向的向量场高密度峰值由模型估计，并估计分数(对数密度导数)。

练上的重建误差示例强制表示包含足够的信息区分它们。解决办法是沿着高密度变化流形被保留，而其他变体被压缩:重构函数应尽可能保持不变重现训练示例，即训练示例附近的点应该映射到训练示例(图5)。重建函数应该将输入映射到最近的点流形，即重建和输入之间的差是一个向量与估计得分(对数密度的导数与对输入的尊重)。在流形上的分数可以为零(其中重建误差也是零)，在对数密度的局部最大值处，但它也可以在局部最小值。这意味着我们不能将低重建误差等同于高重建误差估计概率。对数密度的二阶导数对应于重构函数的一阶导数，在上流形(其中一阶导数为0)，表示的切线方向流形(一阶导数保持在0附近)。

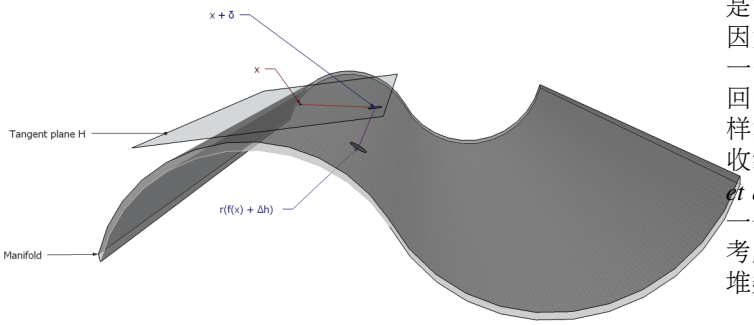


Fig. 6. 正则化采样自动编码器(Rifai *et al.*, 2012; Bengio *et al.*, 2012): 每个MCMC步骤添加到当前状态$x$噪声$\delta$主要在方向上的估计流形切平面$H$和项目返回重建流形(高密度区域) 步骤。

如图6所示，介绍了自编码器采样算法的基本思想在Rifai *et al.* (2012); Bengio *et al.* (2012)就是做MCMC在一个(a)沿着流形移动的地方移动密度梯度(即应用重建)和(b)添加噪声在重建的主奇异向量的方向上(或编码器)雅可比矩阵，对应于与最小相关的那些对数密度的二阶导数。

### 9.3 学习近似推理

现在让我们更仔细地考虑表示是如何计算的迭代推理时具有隐变量的概率模型必填。有一个计算图(可能是随机数在一些节点中生成，在MCMC的情况下)，将输入映射到在确定性推理的情况下(例如，MAP 推理或变分推理)，该函数可以被优化直接。这是最近探索的一种泛化PSD的方法在推理和。的交叉点上研究概率模型学习(Bagnell and Bradley, 2009; Gregor and LeCun, 2010b; Grubb and Bagnell, 2010; Salakhutdinov and Larochelle, 2010; Stoyanov *et al.*, 2011; Eisner, 2012)，哪里有一个中心想法是，与其使用通用的推理机制，一个人可以使用一个学习的，更有效的，利用优势它所应用的数据类型的细节。

### 9.4 采样挑战

这是许多带有潜变量的概率模型的一个麻烦的挑战与大多数玻尔兹曼机变体一样，良好的MCMC采样是必需的作为学习过程的一部分，但是抽样变得非常随着训练的进行，效率低下(或不可靠)，因为学习到的分布模式变得更加尖锐，使得模式之间的混合非常缓慢。然而最初在训练过程中，随着训练的进行，学习者几乎一致地分配质量，其熵减小，接近目标分布的熵为文中给出了更多的实例和计算。根据我们的流形和自然聚类先验部分3.1，目标分布具有尖锐的模态(流形)低密度区域。混合变得更加困难，因为MCMC方法，就其本质而言，倾向于从一小步走向附近的高概率配置。如图7所示。

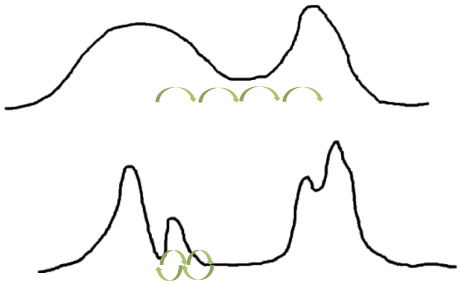

Fig. 7. 上图:在训练初期，MCMC很容易混合模态，因为估计的分布具有较高的熵，并为每个小步骤提供足够的质量移动(MCMC):从一个模式切换到另一个模式。底部:后期，训练依靠在良好的混合可能失速，因为估计模态被广泛的分开低密度的沙漠。

Bengio *et al.* (2013) 表明深度表征可以帮助混合这些分离良好的模式，基于两种理论论证和经验证据。这个想法是如果更高层次因此，表示法可以更好地解开潜在的抽象因素这个抽象空间中的小步骤(例如，从一个类别交换到另一个)可以很容易地由MCMC完成。高级表示可以然后映射回输入空间以获得输入级样本，如深度信念网络(DBN)采样算法(Hinton *et al.*, 2006)。这已经在DBNs和与新提出的从收缩和抽样的算法去噪自动编码器(Rifai *et al.*, 2012; Bengio *et al.*, 2012)。这个仅靠观察不足以解决训练DBN的问题或者一个DBM，但它可能提供一个关键的成分，它使它成为可能考虑从程序训练的深度模型中成功采样不需要MCMC，比如堆叠的正则化自动编码器用于Rifai *et al.* (2012)。

### 9.5 评估和监控性能

对于特征学习算法，总是可以用它对于特定任务(例如对象分类)，其预测器的馈送或初始化为学习特征。在实践中，我们通过保存学到的特征来做到这一点(例如，在训练期间定期进行提前停止) 然后在上面训练一个廉价的分类器(例如线性分类器)。然而，训练最终的分类器可能是一个巨大的计算开销(例如，有监督的微调深度神经网络通常需要更多的训练迭代次数比特征学习本身多)，所以我们可能想要避免为无监督的每次训练迭代训练一个分类器学习者和每个超参数设置。更重要的是，这可能会给出一个不完整的特征评估(会发生什么其他任务?)。所有这些问题促使使用方法进行监测和评估纯粹的无监督性能。这很容易自动编码器变体(在下面列出一些注意事项) 无向图模型如RBM和玻尔兹曼机。

对于自编码器和稀疏编码变体，测试集重构误差可以很容易计算，但本身可能会误导，因为更大的容量(例如，更多的特征，更多的训练时间)往往会系统地导致更低的重构误差，即使在测试中也是如此集合。因此，它不能可靠地用于选择大多数超参数。另一方面，去噪重构误差显然是免疫的这解决了DAEs的问题。基于中发现的DAEs和CAEs之间的联系Bengio *et al.* (2012); Alain and Bengio (2012)，这种免疫可以扩展到DAEs，但不能扩展到超参数控制噪音或收缩的量

对于RBMs和一些(不太深)玻尔兹曼机，一个选择是使用退火重要性抽样(Murray and Salakhutdinov, 2009)，以估计配分函数(以及测试对数似然)。请注意估计量可以具有高方差和变得不可靠(方差变得太大) 随着模型变得越来越有趣，随着更大的权重，更多的非线性，更清晰的模式和更清晰的概率密度函数(见我们之前在9.4部分的讨论)。另一个有趣的和最近提出的选择RBMs是在训练过程中跟踪配分函数(Desjardins *et al.*, 2011)，这可能有助于早期停止和降低普通人工免疫系统的成本。对于玩具RBMs(例如，25个或更少的隐藏单元，或25 输入值或更少)时，也可以计算出精确的对数似然从分析的角度来看，这是调试和验证的好方法感兴趣的属性。

## 10 深度模型的全局训练

深度架构提出的最有趣的挑战之一是:应该如何我们联合培训所有级别?在上一节中在4节中，我们只讨论了如何进行单层模型可以组合形成深度模型。这里我们考虑到各级联合训练的困难这种情况可能会出现。

### 10.1 训练深度架构的挑战

更高层次的抽象意味着更多的非线性。这意味着两个在附近输入配置可能有非常不同的解释因为一些表面细节改变了底层语义，而大多数表面细节的其他变化不会改变底层语义。The 与输入流形相关的表示可能很复杂，因为从输入到表示的映射可能必须展开和扭曲输入流形将复杂的形状转化为分布简单得多的空间，因素之间的关系更简单，甚至可能是线性的或涉及的许多(有条件的)独立。我们的期望是高层抽象和概念之间的联合分布应该是在需要学习的数据少得多的意义上更容易。困难一部分是学习一个好的表示法来完成这个展开和解缠。这可能要付出更大的代价训练问题，可能涉及不良条件和局部最小值。

直到2006年，研究人员才认真研究了如何做到这一点训练深度架构，卷积除外网络(LeCun *et al.*, 1998b)。第一个实现(4节)是无监督或有监督的分层训练更容易，这可以通过堆叠来利用将单层模型转换为深层模型。

有趣的是，为什么分层无监督预训练过程有时帮助一个被监督的学习者(Erhan *et al.*, 2010b)。似乎有一个更普遍的原则在起作用[23] 来指导中间表征的训练，这可能是这比一次性学习要容易得多。这和课程学习理念(Bengio *et al.*, 2009)，即可能首先学习更简单的概念，然后构建更高层次的概念，这要容易得多最简单的。这也与一些深度学习算法的成功相一致为中间表示提供了一些指导，比如半监督嵌入(Weston *et al.*, 2008)。

为什么无监督的预训练可能是有用的被广泛研究(Erhan *et al.*, 2010b)，尝试将答案分解为正则化效应和优化效果。正则化效果从使用堆叠RBMs或降噪自编码器的实验初始化一个监督分类神经网络网络(Erhan *et al.*, 2010b)。它可能只是来自于使用无监督学习对学习动态进行偏差和初始化(训练的)一个"好的"局部最小值的吸引力盆地 Criterion)，其中"good '代表泛化误差。The 这个过程所利用的潜在假设是，一些擅长捕捉主导的特征或潜在因素输入分布的变化也善于捕捉目标输出感兴趣的随机变量的变化(例如: 类)。优化效果更难梳理因为深度神经网络的最上面两层可能会过拟合训练集，无论低层是否计算有用的特征，但有几个迹象表明，优化较低的水平关于监督训练标准可能具有挑战性。

其中一种指示是改变的数值条件优化程序可以对深度架构的联合训练产生了深远的影响，例如，通过改变初始化范围和类型非线性使用(Glorot and Bengio, 2010)，更多而不是浅层的架构。一个假设来解释一下深度优化中的一些难点体系结构以雅可比矩阵的奇异值为中心与从一个层次上的特征进行变换相关联的矩阵进入下一阶段的功能(Glorot and Bengio, 2010)。如果这些奇异值是All小(小于1)，则映射在every中是收缩的当反向传播时，方向和梯度将消失 通过许多层。这是一个已经讨论过的问题循环神经网络(Bengio *et al.*, 1994)，可以被视为每一层都有共享参数的深度网络，及时展开。这种优化困难是有动机的深度架构的二阶方法探索递归网络，特别是无hessian二阶网络方法(Martens, 2010; Martens and Sutskever, 2011)。也有人提出无监督的预训练来帮助解决这个问题训练递归网络和时间RBMs (Sutskever *et al.*, 2009)，也就是说，在每个时间步长都有一个本地信号来指导发现状态变量中需要捕获的好特征:使用当前状态建模(作为隐藏单元) 前一状态和当前输入的联合分布。自然梯度(Amari, 1998)方法，可以应用于网络与数百万个参数(即具有良好的缩放特性)也已经得到了提议(Le Roux *et al.*, 2008b; Pascanu and Bengio, 2013)。Cho *et al.* (2011) 提出使用自适应学习率进行RBM训练，并提出了一种新颖的方法这是一个有趣的梯度估计器说明了该模型对隐藏单元比特翻转和反转的不变性对应权重向量的符号。至少一项研究表示初始化的选择(使雅可比矩阵每个层的所有奇异值都接近1)都可以大幅降低深度网络的训练难度(Glorot and Bengio, 2010) 这与初始化过程的成功是一致的回声国家网络(Jaeger, 2007)，正如Sutskever (2012)最近研究的那样。还有一些实验结果(Glorot and Bengio, 2010; Glorot *et al.*, 2011a; Nair and Hinton, 2010) 表明非线性隐单元的选择可以同时影响训练和泛化性能，并获得了特别有趣的结果稀疏整流单元(Jarrett *et al.*, 2009; Nair and Hinton, 2010; Glorot *et al.*, 2011a; Krizhevsky *et al.*, 2012)。关于神经网络的条件反射问题的一个旧想法这是对称破缺:收敛速度慢的一部分吗可能是由于许多单位一起移动(像羊)和所有的尝试减少相同示例的输出误差。通过初始化用稀疏权重(Martens, 2010)或通过使用常饱和非线性(如整流器作为最大池化单元)，梯度只会流动一些路径，这可能有助于隐藏单元更快地专业化。另一个有前途的想法改善神经网络的训练条件是为了抵消每个函数的平均值和斜率隐藏单元输出(Raiko *et al.*, 2012)，并可能在本地也归一化大小(Jarrett *et al.*, 2009)。关于使用随机等在线方法的争论仍然激烈梯度下降和在大的小批量上使用二阶方法(几千个例子)(Martens, 2010; Le *et al.*, 2011a)，一种随机梯度下降的变体最近赢得了an 优化挑战[24]。

最后，介绍了利用大量的标记数据表明，通过适当的初始化和选择非线性，非常深的纯监督网络可以训练成功地没有任何分层预训练(Ciresan *et al.*, 2010; Glorot *et al.*, 2011a; Seide *et al.*, 2011a; Krizhevsky *et al.*, 2012)。研究人员报告说，在这样的条件下，逐层无监督与纯监督学习相比，预训练带来的改进很少或根本没有从零开始当训练时间足够长。这就加强了假设这种无监督的预训练作为先验，可能没有那么必要当有非常大量的标记数据可用时，但要求为什么没有更早发现这个问题。最新结果在这方面的报道(Krizhevsky *et al.*, 2012)特别有趣因为它们可以大大降低物体识别的错误率在基准测试中(1000类ImageNet任务) 还有更多传统的计算机视觉方法吗经过评估(http://www.image-net.org/challenges/LSVRC/2012/results.html)。使这一成功的主要技术包括以下是高效的*GPU*训练，可以让人训练更长时间 (示例访问超过1亿次)，这方面最先由Lee *et al.* (2009a); Ciresan *et al.* (2010)报道，大量标注样本，人工变换例子(见11.1部分)，大量任务(ImageNet为1000或10000类)，卷积架构 使用最大池化(有关后者，请参见11节两种技巧)，校正非线性(上文讨论)，仔细的初始化(上面讨论过)，仔细的参数更新和自适应学习率启发式算法，逐层特征规范化(跨特征)，以及一种新的基于

23. 首先是由Leon Bottou提出的

24. https://sites.google.com/site/nips2011workshop/optimization-challenges

强二进制乘性注入的技巧隐藏单元的噪声。这个技巧类似于二进制噪声注入用于堆栈去噪的每一层自动编码器。未来的工作有望帮助识别这些元素中哪些最重要，如何概括它们跨越各种各样的任务和体系结构，以及在大多数样本未标记的特定情况下，即，在训练中包含无监督成分标准。

### 10.2 深度玻尔兹曼机的联合训练

我们现在考虑一个特定的所有层的联合训练问题无监督模型，深度玻尔兹曼机(DBM)。尽管进步很大(尽管有许多悬而未决的问题)已经在联合训练上做出了决定使用反向传播梯度的深度架构的所有层(即: 主要是在有监督的情况下)，所做的工作要少得多它们完全是无监督的，例如DBMs [25]。不过要注意人们可以希望前面描述的成功的技术Section可应用于无监督学习算法。

和RBM一样，DBM是另一种玻尔兹曼机模型族的特定子集，其中单位也是分层排列的。然而，与RBM不同的是，DBM拥有多个隐藏单元层，奇数层中的单元是条件独立的给定偶数层，反之亦然。关于Eq. 7的Boltzmann能量函数，DBM 对应于在两者中设置$U = 0$和稀疏连接结构$V$和$W$。我们可以通过指定使DBM的结构更加明确它的能量函数。对于有两层隐藏层的模型，如下所示:

$$\mathcal{E}_\theta^{\mathrm{DBM}}(v, h^{(1)}, h^{(2)}; \theta) = -v^T W h^{(1)} - {h^{(1)}}^T V h^{(2)} - {d^{(1)}}^T h^{(1)} - {d^{(2)}}^T h^{(2)} - b^T v, \quad (29)$$

用$\theta = \{W, V, d^{(1)}, d^{(2)}, b\}$。DBM也可以被描述为两个顶点集之间的二分图，由奇数和偶数组成图层(使用$v := h^{(0)}$)。

#### *10.2.1* 平均场近似推理

背离RBM的一个关键点是隐藏单元的后验分布(给定可见项)不再易处理，由于隐藏单元之间的相互作用。Salakhutdinov and Hinton (2009) resort to a 后验的平均场近似。具体来说，对于a 有两个隐藏层的模型，我们希望用因子来近似$P\left(h^{(1)}, h^{(2)} \mid v\right)$ 分布$Q_v(h^{(1)}, h^{(2)}) = \prod_{j=1}^{N_1} Q_v\left(h_j^{(1)}\right) \prod_{i=1}^{N_2} Q_v\left(h_i^{(2)}\right)$，这样的KL 发散度$\mathrm{KL}\left(P\left(h^{(1)}, h^{(2)} \mid v\right) \| Q_v(h^1, h^2)\right)$是最小的，或相等的，那一个较低与对数似然函数的绑定是最大化的:

$$\log P(v) > \mathcal{L}(Q_v) \equiv \sum_{h^{(1)}} \sum_{h^{(2)}} Q_v(h^{(1)}, h^{(2)}) \log\left(\frac{P(v, h^{(1)}, h^{(2)})}{Q_v(h^{(1)}, h^{(2)})}\right) \quad (30)$$

最大化平均场分布的这个下界$Q_v(h^1, h^2)$(通过将导数设置为零) 得到如下的平均场更新方程:

$$\hat{h}_i^{(1)} \leftarrow \mathrm{sigmoid}\left(\sum_j W_{ji} v_j + \sum_k V_{ik} \hat{h}_k^{(2)} + d_i^{(1)}\right) \quad (31)$$

$$\hat{h}_k^{(2)} \leftarrow \mathrm{sigmoid}\left(\sum_i V_{ik} \hat{h}_i^{(1)} + d_k^{(2)}\right) \quad (32)$$

注意上面的方程表面上看起来像一个固定点循环神经网络，即具有恒定输入。以同样的方式RBM可以与一个简单的自动编码器相关联DBM的平均场更新方程可以与a联系起来循环自编码器。在这种情况下，训练标准涉及在last或at的重构错误连续的时间步长。对这种类型的模型进行了探索Savard (2011)和Seung (1998) 并且在降噪方面比普通的自动编码器做得更好。

25. 接头训练一个深度信念网络的所有层更具挑战性因为涉及到更困难的推理问题。

迭代Eq. (31 - 32)直到收敛产生$Q$的“变分正相位”的参数Eq. 33:

$$\begin{aligned}\mathcal{L}(Q_v) =& \mathbb{E}_{Q_v}\left[\log P(v, h^{(1)}, h^{(2)}) - \log Q_v(h^{(1)}, h^{(2)})\right] \\ =& \mathbb{E}_{Q_v}\left[-\mathcal{E}_\theta^{\mathrm{DBM}}(v, h^{(1)}, h^{(2)}) - \log Q_v(h^{(1)}, h^{(2)})\right] \\ & - \log Z_\theta \\ \frac{\partial \mathcal{L}(Q_v)}{\partial \theta} =& -\mathbb{E}_{Q_v}\left[\frac{\partial \mathcal{E}_\theta^{\mathrm{DBM}}(v, h^{(1)}, h^{(2)})}{\partial \theta}\right] \\ & + \mathbb{E}_P\left[\frac{\partial \mathcal{E}_\theta^{\mathrm{DBM}}(v, h^{(1)}, h^{(2)})}{\partial \theta}\right] \end{aligned} \quad (33)$$

这种变分学习过程留下了“负阶段”因此，可以通过SML或对比散度来估计(Hinton, 2000)和RBM的情况一样。

#### *10.2.2* 深度玻尔兹曼机训练

训练DBM和RBM的主要区别不是直接最大化可能性，而是选择参数来最大化所给出的可能性的下界Eq. 30。基于sml的最大化算法这个下限如下所示:

1) 将可见单元固定到训练示例上。
2) 迭代Eq. (31 - 32)直到收敛。
3) 通过SML生成负相位样本$v^-$, $h^{(1)-}$和$h^{(2)-}$。
4) 计算$\partial \mathcal{L}(Q_v) / \partial \theta$ 使用步骤2-3中的值。
5) 最后，更新模型参数近似随机梯度上升。

而上面的程序似乎是高度的简单延伸训练RBMs的有效SML方案，如Desjardins *et al.* (2012)，这个程序似乎很容易陷入贫困局部最小值，使许多隐藏单元有效地死亡(不显著不同于它的小范数随机初始化)。

指出了SML联合训练策略的失败Salakhutdinov and Hinton (2009)。作为一种选择，他们提出了一种贪婪的逐层训练策略。这个过程包括预训练DBM的层，与深度信念网络的方式大致相同: 通过堆叠RBMs并训练每一层来独立建模的输出前一层。最后的联合“微调”是按照上述基于sml的程序完成的。

## 11 内置不变性

众所周知，结合先验领域知识是有帮助的机器学习。探索好的策略是非常重要的研究大道。然而，如果我们要推进我们对核心机器的理解学习原则，保持比较是很重要的预测公平，保持清晰意识到先验领域知识的不同使用学习算法，特别是在比较它们的性能时基准问题。到目前为止，我们只介绍了仅利用的算法高维的一般归纳偏差问题，从而使它们具有潜在的适用性对于任何高维问题。最普遍的方法是手工设计更好的特征来提供通用分类器，并被广泛使用计算机视觉(例如(Lowe, 1999))。在这里，我们更关注如何基本输入的 领域知识，特别是其拓扑结构(例如具有2D结构的位图图像)，可以用于学习 更好的功能。

### 11.1 生成转换示例

泛化性能通常是通过提供更大数量的代表性数据进行改进。这个罐子通过应用小的随机变形来生成新样本对于原始训练示例，使用变形已知不会改变目标变量我们感兴趣的是，例如object类对small不变图像的变换，如平移、旋转、缩放或剪切。这种古老的方法(Baird, 1990)最近已经被广泛应用在Ciresan *et al.* (2010)的工作中成功使用了高效的GPU 实现(40× speedup)训练一个标准但很大基于变形MNIST数字的深度多层感知器。同时使用仿射和弹性变形(Simard *et al.*, 2003)，与普通的随机梯度下降，它们的分类错误率达到了创纪录的0.32%。

## 11.2 卷积和池化

另一种强大的方法仅基于拓扑的更基本的知识输入维度的结构。这里我们指的是例如，the 图像或音频频谱中的二维像素布局，视频的3D结构，文本或的1D顺序结构一般的时间序列。基于这样的结构，我们可以定义局部接受野(Hubel and Wiesel, 1959)，使每个底层特征将仅从输入的一个子集计算:拓扑中的一个邻域(例如，a 给定位置的子图像)。这种拓扑局部性约束对应于具有一个非常稀疏的非零权重矩阵只允许在拓扑上本地连接。计算相关的矩阵乘积可以当然比处理一个稠密的矩阵更有效，此外，从更少的自由参数中获得的统计增益。在具有这种拓扑结构的领域中，相似的输入模式可能出现在不同的位置和附近的值(例如: 连续的帧或附近的像素)可能会更强依赖关系对数据建模也很重要。事实上这些可以利用依赖关系来发现拓扑(Le Roux *et al.*, 2008a)，即。从一组没有向量的集合中恢复一个规则的像素网格任何顺序信息，例如元素已经任意之后所有的例子都以同样的方式洗牌。因此，相同的局部特征计算是否可能与所有翻译职位的接受相关字段。因此，这种想法是将这样一个局部特征提取器扫过拓扑结构:这对应于a 卷积，并将输入转换为类似形状的特征地图。与扫描相同，这可以被视为静态但位置不同的复制功能共享相同参数的提取器。这是卷积网络的核心(LeCun *et al.*, 1989, 1998b) 哪些应用于目标识别以及图像分割(Turaga *et al.*, 2010)。卷积架构的另一个特征是价值由应用于多个相邻输入的相同特征检测器计算然后通过池操作对位置进行汇总，通常是取它们的最大值或总和。这赋予了最终的池化特征层一些输入翻译的不变性程度，以及这种架构风格(交替选择特征提取和不变性创建池化)一直是卷积网络的基础Neocognitron (Fukushima, 1980) 和HMAX (Riesenhuber and Poggio, 1999)模型，并被认为是哺乳动物大脑的结构用于物体识别(Riesenhuber and Poggio, 1999; Serre *et al.*, 2007; DiCarlo *et al.*, 2012)。无论特定功能位于内部的何处，池化单元的输出都是相同的它的池化区域。从经验上看，池化的使用似乎有所帮助显著提高了物体分类的准确率任务(LeCun *et al.*, 1998b; Boureau *et al.*, 2010, 2011)。a 成功的池化连接变体稀疏编码是L2池化(Hyvärinen *et al.*, 2009; Kavukcuoglu *et al.*, 2009; Le *et al.*, 2010)，其中池的输出是可能加权值的平方根滤波器输出的平方和。理想情况下，我们希望对特征池化进行泛化以进行学习哪些功能应该被整合在一起，例如成功在几篇论文中完成(Hyvärinen and Hoyer, 2000; Kavukcuoglu *et al.*, 2009; Le *et al.*, 2010; Ranzato and Hinton, 2010; Courville *et al.*, 2011b; Coates and Ng, 2011b; Gregor *et al.*, 2011)。这样，池的输出学习到对不变 由汇集的特征跨度捕获的变化。

### 基于块的训练

在an中学习卷积层的最简单方法无监督的方式是基于块的训练: 只需向通用的无监督特征学习算法提供局部补丁在输入的随机位置提取。生成的特征提取器然后可以在输入上滑动以产生卷积特征吗地图。该映射可以用作下一层的新输入，并重复操作从而学习和堆叠几个层。这种方法最近被用于Independent 对三维视频块进行(Le *et al.*, 2011c) 子空间分析，达到Hollywood2, UCF, KTH和YouTube动作识别数据集的最新进展。类似地，(Coates and Ng, 2011a)比较了几种特征学习器基于块的训练，并在几个分类上取得了最先进的结果基准测试。有趣的是，在这项工作中，表现几乎与非常简单一样好k均值聚类与更复杂的特征学习器一样。然而我们推测之所以会出现这种情况，只是因为补丁的维度相当低(与整幅图像的尺寸相比)。大型数据集可能就足够了对空间的覆盖，例如$6 \times 6$补丁中普遍存在的边缘，因此，分布式表示不是绝对必要的。这种成功的另一个合理解释是集群然后将每个图像块中的标识汇集成直方图与较大的子图像相关联的簇计数。而输出直方图(one-hot non-distributed code)是常规聚类的一种它本身是一种分布式表示，而“软”k均值呢(Coates and Ng, 2011a)表示不仅允许最近的过滤器而且它的邻居也很活跃。

### 卷积和平铺卷积训练

可以直接使用无监督准则。早期的方法(Jain and Seung, 2008)训练标准但深入卷积MLP在图像去噪任务中的应用，即作为一个深度的，卷积降噪自编码器。的卷积版本RBM或其扩展也已经开发(Desjardins and Bengio, 2008; Lee *et al.*, 2009a; Taylor *et al.*, 2010) 以及内置的概率最大池化操作卷积深度网络(Lee *et al.*, 2009a,b; Krizhevsky, 2010)。其他的无监督特征学习方法也适用于卷积设置包括PSD (Kavukcuoglu *et al.*, 2009, 2010; Jarrett *et al.*, 2009; Henaff *et al.*, 2011)，稀疏编码的卷积版本称为反卷积网络(Zeiler *et al.*, 2010)，地形ICA (Le *et al.*, 2010)，以及Kivinen and Williams (2012)应用的mPoT 建模自然纹理。Gregor and LeCun (2010a); Le *et al.* (2010) 演示了分块卷积技术，其中参数为仅在接受域为$k$步骤的特征提取器之间共享远离(所以那些关注邻近位置的人不会共享)。这使得池化单元不仅可以保持不变翻译，是卷积网络和早期的具有局部连接但没有权重的神经网络分享(LeCun, 1986, 1989)。

### 共享的替代方案

或者，人们也可以使用显性知识期望不变量的数学表达来定义对已知的变换族鲁棒的变换输入变形，使用所谓的散射算子(Mallat, 2012; Bruna and Mallat, 2011)，哪些可以以类似于深度卷积网络的有趣方式进行计算还有小波。与卷积网络一样，散射算子交替执行两种类型的操作: 卷积和池化(作为范数)。与卷积不同网络，所提出的方法在每一层保存所有关于输入的信息(以一种可以反向的方式)，并自动生成非常稀疏(但维度非常高)的表示。另一个区别是过滤器不是学习的，而是设置为保证鲁棒地实现了先验指定的不变性。仅仅几个级别就足以在几个基准测试中取得令人印象深刻的结果数据集。

## 11.3 时间一致性和慢特征

识别缓慢移动/变化因素的原则时态/空间数据已被调查很多(Becker and Hinton, 1992; Wiskott and Sejnowski, 2002; Hurri and Hyvärinen, 2003; Körding *et al.*, 2004; Cadieu and Olshausen, 2009) 作为寻找有用表示的原则。特别是这个想法应用于图像序列和作为一个解释V1简单细胞和复杂细胞的行为方式。a 好的概述可以在Hurri and Hyvärinen (2003); Berkes and Wiskott (2005)找到。

最近，时间相干性被成功地深入研究建模视频的架构(Mobahi *et al.*, 2009)。还发现，时间连贯性发现了与获得的视觉特征相似的视觉特征通过普通的无监督特征学习(Bergstra and Bengio, 2009)，时间一致性惩罚与训练相结合 无监督特征学习准则(Zou *et al.*, 2011)，使用L1正则化的稀疏自编码器，在这种情况下，得到改进分类性能。

时间一致性先验可以用几种方式表示，最简单的一种有时是特征值之间的平方差$t$ 和$t+1$。其他合理的时间一致性先验包括以下内容。首先，不是惩罚平方变化，而是惩罚绝对值(或者类似的稀疏性惩罚) 大多数情况下变化值应该正好为0，这是什么意思从直觉上讲，这对我们周围的现实生活因素是有意义的。第二，人们会认为，不同的因素可以改变，而不是慢慢改变与自己不同的时间尺度相关联。他们时间尺度的特殊性因此可以成为解开解释因素的线索。第三，这是意料之中的有些因数应该用一组数字来表示(如$x$、$y$、$z$物体在空间中的位置姿势参数Hinton *et al.* (2011))而不是by 一个单一的标量，这些群体倾向于一起移动。结构化稀疏性惩罚(Kavukcuoglu *et al.*, 2009; Jenatton *et al.*, 2009; Bach *et al.*, 2011; Gregor *et al.*, 2011) 可以用于此目的。

### 11.4 解缠变异因子的算法

目的是消除表示对方向的敏感性对手头的任务没有信息的数据的方差。无论如何通常情况下特征提取的目标是解开或分离数据中有许多不同但信息量大的因素，例如，在视频中人物:主体身份、执行的动作、主体相对于相机的姿态等。在这种情况下，生成不变特征的方法，如特征池化，可能是不够的。

构建不变特征的过程可以看作由两个步骤组成。首先，恢复帐户的低级特征以获取数据。其次，将这些低级特征的子集进行池化一起形成更高层次的不变特征，例如池化以及卷积神经网络的下采样层。池化特征形成的不变表示提供了一个以不完全窗口为数据的详细表示较低级别的特征在池化过程中被抽象掉。同时我们希望更高层次的特征更抽象，表现得更丰富不变性，我们几乎无法控制哪些信息丢失了池化。真正希望的是特定的功能集是对不相关的特征不变，并解缠相关的特征特征。不幸的是，通常很难确定先验 哪组特征最终将与at的任务相关手。

一个有趣的方法来利用一些已知数据中存在的变异因子就是变换自动编码器(Hinton *et al.*, 2011):而不是标量模式检测器(例如;对应于输入中出现特定形式的概率) 我们可以将这些特征组织为包含两者的组模式检测器和指定属性的姿态参数 检测到的模式。在(Hinton *et al.*, 2011)中，假设是什么先验是成对的(或连续的)样本被观察到姿态参数中对应变化的关联值。例如，控制眼睛的动物知道眼睛的变化当从视网膜上的一个图像到下一个图像时，它的眼睛运动系统被应用。在该工作中，还假设姿态变化是相同的所有的模式检测器，这对于全局变化是有意义的随着图像平移和相机几何形状的变化。相反，我们希望来发现应该关联的姿态参数和属性使用每个特征检测器，而不必提前指定什么它们应该是一样的，强制它们对于所有的功能都是一样的观察所有姿态参数或属性的变化。

讨论了流形切线分类器的最新研究进展在8.3 部分，在这方面很有趣。在没有任何监督或先验知识的情况下，它会发现显著的局部变异因子(切线向量) 流形，从CAE中提取，解释为局部有效输入“变形”)。更高层次的功能被鼓励使用对这些变异因素是不变的，所以它们必须依赖于其他特征。从某种意义上说，这种方法是解缠有效的局部沿着数据流形的变形，来自其他更剧烈的变化，与其他因素相关联，比如那些影响阶级的因素身份。[26]

一个解决信息丢失问题的方法特征池化范式，就是考虑许多重叠的池基于相同的低级特征集的特征。这样的结构会是否有可能学习到一组冗余的不变特征不会造成重大信息损失。然而，这并不是显而易见的可以应用学习原则来确保特征是正确的不变的同时保持尽可能多的信息。在深深的时候信念网络(Belief Network)或深度玻尔兹曼机(Deep Boltzmann Machine 4和10.2)，两个隐藏层将，原则上，能够将信息保存到“池‘中隐藏层中，不能保证第二层特征多比“低层”的第一层特征不变性。然而，有一些经验证据表明，DBN的第二层往往显示更多不变性比第一层(Erhan *et al.*, 2010a)。

一种更原则性的方法，从确保更稳健的角度来看紧凑的特征表示，可以通过重新考虑特征的解缠来构想通过它的生成等价物——特征组成的镜头。自从许多无监督学习算法都有一个生成式解释(或一种从它们的输入中重建输入的方法高层次表示)，生成视角可以提供如何进行的洞察考虑一下分解因素。目前大多数模型使用构造不变特征具有其底层特征的解释性线性组合来构造数据。[27]这是这是一个相当基本的特征组合形式局限性。例如，不可能将特征与泛型线性组合转换(如翻译)以生成特征的转换版本。我们甚至不能考虑通用颜色特征与灰度刺激线性结合Pattern生成彩色图案看来如果我们要采取解缠的概念我们需要更丰富的互动比简单的线性组合所提供的特性要好。

26. 在输入空间中，影响类身份的变化实际上可能是相似的幅度到局部变形，但不跟随流形，例如交叉低密度区域。

27. 顺便说一句，如果我们只得到的值对于更高层次的池化特征，我们无法准确地恢复数据因为我们不知道如何分配池化特征的积分值转换为较低级别的特征。这只是生成版本信息汇集导致信息丢失的后果。

## 12 结论

本文对表示学习和深度学习进行了综述三种明显不相关的主要方法:概率模型(有向类型，如稀疏编码和无向类型，如稀疏编码如Boltzmann machines)，基于重构的算法相关自动编码器和几何驱动的流形学习方法。目前，将这些方法联系起来是一个非常活跃的研究领域，并有可能继续产生利用每种范式的相对优势的模型和方法。

实际问题和指导方针。其中一个批评是针对人工神经网络和深度学习算法的关键在于它们有很多超参数和变体，探索它们的配置和体系结构是一个艺术。这激发了早期一本关于“”技巧的书贸易”(Orr and Muller, 1998)其中LeCun *et al.* (1998a)还在与训练深度架构相关，特别是所关注的内容初始化，条件不良和随机梯度下降。一个好的和更现代的好的训练方法的概要，特别适应培训RBMs，在Hinton (2010)中提供，而类似的可以找到更多面向深度神经网络的指南在Bengio (2013)，它们都是小说的一部分以上书的版本。超参数自动化的最新工作搜索(Bergstra and Bengio, 2012; Bergstra *et al.*, 2011; Snoek *et al.*, 2012)也使它更方便、高效和可复制。

结合通用**ai**级别先验。我们已经介绍了许多我们认为可以的高级通用先验通过改进使机器学习更接近人工智能表示学习。这些先验大多与假设有关存在多种变异的潜在因素，其变异是在某种意义上是相互正交的。他们是被期待的要在多个抽象层次上组织，因此需要深入架构，这也有统计上的优势，因为它们允许以组合高效的方式重用参数。只有其中一些因素通常与任何特定的例子有关，表示稀疏性的证明。这些因素预计会与简单的*(*例如，线性*)*依赖关系相关，以及这些依赖关系的子集解释感兴趣的不同随机变量*(*输入、任务*)*在时间和空间上以结构化的方式变化*(*时间和空间连贯性*)*。我们期待表示学习在未来的成功应用为了完善和增加前科记录，把它们中的大部分结合起来，而不是只专注于一个。研究更好地考虑这些先验的培训标准有可能让我们更接近发现的长期目标吗能够解开潜在解释因素的学习算法。

推理。我们预测的方法直接基于将表示函数参数化将包含越来越多的在推理过程中发现的迭代计算类型概率潜变量模型。已经有行动了另一方面，利用概率潜变量模型自身学习的近似推理机制(即: 生成表示函数的参数化描述)。主要的概率模型的吸引力在于潜在的语义变量是清晰的，这允许清晰地分离的问题建模(选择能量函数)、推断(估计$P(h|x)$)和学习(优化参数)，在每种情况下使用通用工具。在另一方面，做近似推断而不是近似在学习的近似优化中明确说明可以有有害影响，因此需要学习近似推理。更根本的是，存在多模态的问题后$P(h|x)$。如果有指数级的可能构型的价值观的因素$h_i$，可以解释$x$，然后我们似乎陷入非常糟糕的推理，要么专注于单一模式(MAP 假设存在某种强因子分解(如变分) 推理)或使用无法访问$P(h|x)$的足够模式的MCMC。我们作为思想食物的提议是放弃对的显式表示要求的想法后验，并满足于利用的隐式表征$P(h|x)$中的潜在结构，以便简洁地表示它:even 虽然$P(h|x)$可能有指数级的模式，但这是可能的用一组数字来表示它。例如，考虑计算a 确定性特征表示$f(x)$，隐式捕获关于信息的高度多模态$P(h|x)$，在这个意义上，所有的问题(例如对目标概念进行一些预测)可以在$P(h|x)$上提问，也可以在$f(x)$上回答。

优化。要更好地理解它，还有很多工作要做训练深度架构的成功和失败，均在监督中案例(最近有很多成功案例)和无

监督案例(其中有很多需要做更多的工作)。尽管正则化效果很重要在小数据集上，这种影响在非常大的数据集上仍然存在其中涉及一些优化问题。它们更多是因为局部最小值吗(我们现在知道它们数量巨大)和动态培训程序?或者它们主要是由于条件不佳，可能会被处理通过近似二阶方法?这些基本问题仍然没有答案值得更多的研究。

致谢

作者要感谢David ward - farley、Razvan Pascanu和Ian Goodfellow的有用反馈，以及NSERC、CIFAR和加拿大研究资助主席。

## REFERENCES

Alain, G. and Bengio, Y. (2012). What regularized auto-encoders learn from the data generating distribution. Technical Report Arxiv report 1211.4246, Université de Montréal.

Amari, S. (1998). Natural gradient works efficiently in learning. *Neural Computation*, **10**(2), 251–276.

Bach, F., Jenatton, R., Mairal, J., and Obozinski, G. (2011). Structured sparsity through convex optimization. *CoRR*, **abs/1109.2397**.

Bagnell, J. A. and Bradley, D. M. (2009). Differentiable sparse coding. In *NIPS'2009*, pages 113–120.

Baird, H. (1990). Document image defect models. In *IAPR Workshop, Syntactic & Structural Patt. Rec.*, pages 38–46.

Becker, S. and Hinton, G. (1992). A self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, **355**, 161–163.

Belkin, M. and Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, **15**(6), 1373–1396.

Bell, A. and Sejnowski, T. J. (1997). The independent components of natural scenes are edge filters. *Vision Research*, **37**, 3327–3338.

Bengio, Y. (1993). A connectionist approach to speech recognition. *International Journal on Pattern Recognition and Artificial Intelligence*, **7**(4), 647–668.

Bengio, Y. (2008). Neural net language models. *Scholarpedia*, **3**(1).

Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, **2**(1), 1–127. Also published as a book. Now Publishers, 2009.

Bengio, Y. (2011). Deep learning of representations for unsupervised and transfer learning. In *JMLR W&CP: Proc. Unsupervised and Transfer Learning*.

Bengio, Y. (2013). Practical recommendations for gradient-based training of deep architectures. In K.-R. Müller, G. Montavon, and G. B. Orr, editors, *Neural Networks: Tricks of the Trade*. Springer.

Bengio, Y. and Delalleau, O. (2009). Justifying and generalizing contrastive divergence. *Neural Computation*, **21**(6), 1601–1621.

Bengio, Y. and Delalleau, O. (2011). On the expressive power of deep architectures. In *ALT'2011*.

Bengio, Y. and LeCun, Y. (2007). Scaling learning algorithms towards AI. In L. Bottou, O. Chapelle, D. DeCoste, and J. Weston, editors, *Large Scale Kernel Machines*. MIT Press.

Bengio, Y. and Monperrus, M. (2005). Non-local manifold tangent learning. In *NIPS'2004*, pages 129–136. MIT Press.

Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, **5**(2), 157–166.

Bengio, Y., Ducharme, R., Vincent, P., and Jauvin, C. (2003). A neural probabilistic language model. *JMLR*, **3**, 1137–1155.

Bengio, Y., Paiement, J.-F., Vincent, P., Delalleau, O., Le Roux, N., and Ouimet, M. (2004). Out-of-sample extensions for LLE, Isomap, MDS, Eigenmaps, and Spectral Clustering. In *NIPS'2003*.

Bengio, Y., Delalleau, O., and Le Roux, N. (2006a). The curse of highly variable functions for local kernel machines. In *NIPS'2005*.

Bengio, Y., Larochelle, H., and Vincent, P. (2006b). Non-local manifold Parzen windows. In *NIPS'2005*. MIT Press.

Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H. (2007). Greedy layer-wise training of deep networks. In *NIPS'2006*.

Bengio, Y., Louradour, J., Collobert, R., and Weston, J. (2009). Curriculum learning. In *ICML'09*.

Bengio, Y., Delalleau, O., and Simard, C. (2010). Decision trees do not generalize to new variations. *Computational Intelligence*, **26**(4), 449–467.

Bengio, Y., Alain, G., and Rifai, S. (2012). Implicit density estimation by local moment matching to sample from auto-encoders. Technical report, arXiv:1207.0057.

Bengio, Y., Mesnil, G., Dauphin, Y., and Rifai, S. (2013). Better mixing via deep representations. In *ICML'2013*.

Bergstra, J. and Bengio, Y. (2009). Slow, decorrelated features for pretraining complex cell-like networks. In *NIPS'2009*.

Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *J. Machine Learning Res.*, **13**, 281–305.

Bergstra, J., Bardenet, R., Bengio, Y., and Kégl, B. (2011). Algorithms for hyper-parameter optimization. In *NIPS'2011*.

Berkes, P. and Wiskott, L. (2005). Slow feature analysis yields a rich repertoire of complex cell properties. *Journal of Vision*, **5**(6), 579–602.

Besag, J. (1975). Statistical analysis of non-lattice data. *The Statistician*, **24**(3), 179–195.

Bordes, A., Glorot, X., Weston, J., and Bengio, Y. (2012). Joint learning of words and meaning representations for open-text semantic parsing. *AISTATS'2012*.

Boulanger-Lewandowski, N., Bengio, Y., and Vincent, P. (2012). Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription. In *ICML'2012*.

Boureau, Y., Ponce, J., and LeCun, Y. (2010). A theoretical analysis of feature pooling in vision algorithms. In *ICML'10*.

Boureau, Y., Le Roux, N., Bach, F., Ponce, J., and LeCun, Y. (2011). Ask the locals: multi-way local pooling for image recognition. In *ICCV'11*.

Bourlard, H. and Kamp, Y. (1988). Auto-association by multilayer perceptrons and singular value decomposition. *Biological Cybernetics*, **59**, 291–294.

Brand, M. (2003). Charting a manifold. In *NIPS'2002*, pages 961–968. MIT Press.

Breuleux, O., Bengio, Y., and Vincent, P. (2011). Quickly generating representative samples from an RBM-derived process. *Neural Computation*, **23**(8), 2053–2073.

Bruna, J. and Mallat, S. (2011). Classification with scattering operators. In *ICPR'2011*.

Cadieu, C. and Olshausen, B. (2009). Learning transformational invariants from natural movies. In *NIPS'2009*, pages 209–216. MIT Press.

Carreira-Perpiñan, M. A. and Hinton, G. E. (2005). On contrastive divergence learning. In *AISTATS'2005*, pages 33–40.

Chen, M., Xu, Z., Winberger, K. Q., and Sha, F. (2012). Marginalized denoising autoencoders for domain adaptation. In *ICML'2012*.

Cho, K., Raiko, T., and Ilin, A. (2010). Parallel tempering is efficient for learning restricted Boltzmann machines. In *IJCNN'2010*.

Cho, K., Raiko, T., and Ilin, A. (2011). Enhanced gradient and adaptive learning rate for training restricted Boltzmann machines. In *ICML'2011*, pages 105–112.

Ciresan, D., Meier, U., and Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. Technical report, arXiv:1202.2745.

Ciresan, D. C., Meier, U., Gambardella, L. M., and Schmidhuber, J. (2010). Deep big simple neural nets for handwritten digit recognition. *Neural Computation*, **22**, 1–14.

Coates, A. and Ng, A. Y. (2011a). The importance of encoding versus training with sparse coding and vector quantization. In *ICML'2011*.

Coates, A. and Ng, A. Y. (2011b). Selecting receptive fields in deep networks. In *NIPS'2011*.

Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *ICML'2008*.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from

scratch. *Journal of Machine Learning Research*, **12**, 2493–2537.

Courville, A., Bergstra, J., and Bengio, Y. (2011a). A spike and slab restricted Boltzmann machine. In *AISTATS'2011*.

Courville, A., Bergstra, J., and Bengio, Y. (2011b). Unsupervised models of images by spike-and-slab RBMs. In *ICML'2011*.

Dahl, G. E., Ranzato, M., Mohamed, A., and Hinton, G. E. (2010). Phone recognition with the mean-covariance restricted Boltzmann machine. In *NIPS'2010*.

Dahl, G. E., Yu, D., Deng, L., and Acero, A. (2012). Context-dependent pre-trained deep neural networks for large vocabulary speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, **20**(1), 33–42.

Deng, L., Seltzer, M., Yu, D., Acero, A., Mohamed, A., and Hinton, G. (2010). Binary coding of speech spectrograms using a deep auto-encoder. In *Interspeech 2010*, Makuhari, Chiba, Japan.

Desjardins, G. and Bengio, Y. (2008). Empirical evaluation of convolutional RBMs for vision. Technical Report 1327, Dept. IRO, U. Montréal.

Desjardins, G., Courville, A., Bengio, Y., Vincent, P., and Delalleau, O. (2010). Tempered Markov chain Monte Carlo for training of restricted Boltzmann machine. In *AISTATS'2010*, volume 9, pages 145–152.

Desjardins, G., Courville, A., and Bengio, Y. (2011). On tracking the partition function. In *NIPS'2011*.

Desjardins, G., Courville, A., and Bengio, Y. (2012). On training deep Boltzmann machines. Technical Report arXiv:1203.4416v1, Université de Montréal.

DiCarlo, J., Zoccolan, D., and Rust, N. (2012). How does the brain solve visual object recognition? *Neuron*.

Donoho, D. L. and Grimes, C. (2003). Hessian eigenmaps: new locally linear embedding techniques for high-dimensional data. Technical Report 2003-08, Dept. Statistics, Stanford University.

Eisner, J. (2012). Learning approximate inference policies for fast prediction. Keynote talk at ICML Workshop on Inferning: Interactions Between Search and Learning.

Erhan, D., Courville, A., and Bengio, Y. (2010a). Understanding representations learned in deep architectures. Technical Report 1355, Université de Montréal/DIRO.

Erhan, D., Bengio, Y., Courville, A., Manzagol, P.-A., Vincent, P., and Bengio, S. (2010b). Why does unsupervised pre-training help deep learning? *Journal of Machine Learning Research*, **11**, 625–660.

Freund, Y. and Haussler, D. (1994). Unsupervised learning of distributions on binary vectors using two layer networks. Technical Report UCSC-CRL-94-25, University of California, Santa Cruz.

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, **36**, 193–202.

Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *AISTATS'2010*.

Glorot, X., Bordes, A., and Bengio, Y. (2011a). Deep sparse rectifier neural networks. In *AISTATS'2011*.

Glorot, X., Bordes, A., and Bengio, Y. (2011b). Domain adaptation for large-scale sentiment classification: A deep learning approach. In *ICML'2011*.

Goodfellow, I., Le, Q., Saxe, A., and Ng, A. (2009). Measuring invariances in deep networks. In *NIPS'2009*, pages 646–654.

Goodfellow, I., Courville, A., and Bengio, Y. (2011). Spike-and-slab sparse coding for unsupervised feature discovery. In *NIPS Workshop on Challenges in Learning Hierarchical Models*.

Goodfellow, I. J., Courville, A., and Bengio, Y. (2012). Spike-and-slab sparse coding for unsupervised feature discovery. arXiv:1201.3382.

Gregor, K. and LeCun, Y. (2010a). Emergence of complex-like cells in a temporal product network with local receptive fields. Technical report, arXiv:1006.0448.

Gregor, K. and LeCun, Y. (2010b). Learning fast approximations of sparse coding. In *ICML'2010*.

Gregor, K., Szlam, A., and LeCun, Y. (2011). Structured sparse coding via lateral inhibition. In *NIPS'2011*.

Gribonval, R. (2011). Should penalized least squares regression be interpreted as Maximum A Posteriori estimation? *IEEE Transactions on Signal Processing*, **59**(5), 2405–2410.

Grosse, R., Raina, R., Kwong, H., and Ng, A. Y. (2007). Shift-invariant sparse coding for audio classification. In *UAI'2007*.

Grubb, A. and Bagnell, J. A. D. (2010). Boosted backpropagation learning for training deep modular networks. In *ICML'2010*.

Gutmann, M. and Hyvarinen, A. (2010). Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *AISTATS'2010*.

Hamel, P., Lemieux, S., Bengio, Y., and Eck, D. (2011). Temporal pooling and multiscale learning for automatic annotation and ranking of music audio. In *ISMIR*.

Håstad, J. (1986). Almost optimal lower bounds for small depth circuits. In *STOC'86*, pages 6–20.

Håstad, J. and Goldmann, M. (1991). On the power of small-depth threshold circuits. *Computational Complexity*, **1**, 113–129.

Henaff, M., Jarrett, K., Kavukcuoglu, K., and LeCun, Y. (2011). Unsupervised learning of sparse features for scalable audio classification. In *ISMIR'11*.

Hinton, G., Krizhevsky, A., and Wang, S. (2011). Transforming auto-encoders. In *ICANN'2011*.

Hinton, G., Deng, L., Dahl, G. E., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T., and Kingsbury, B. (2012). Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Processing Magazine*, **29**(6), 82–97.

Hinton, G. E. (1986). Learning distributed representations of concepts. In *Proc. 8th Conf. Cog. Sc. Society*, pages 1–12.

Hinton, G. E. (1999). Products of experts. In *ICANN'1999*.

Hinton, G. E. (2000). Training products of experts by minimizing contrastive divergence. Technical Report GCNU TR 2000-004, Gatsby Unit, University College London.

Hinton, G. E. (2010). A practical guide to training restricted Boltzmann machines. Technical Report UTML TR 2010-003, Department of Computer Science, University of Toronto.

Hinton, G. E. and Roweis, S. (2003). Stochastic neighbor embedding. In *NIPS'2002*.

Hinton, G. E. and Salakhutdinov, R. (2006). Reducing the dimensionality of data with neural networks. *Science*, **313**(5786), 504–507.

Hinton, G. E. and Zemel, R. S. (1994). Autoencoders, minimum description length, and helmholtz free energy. In *NIPS'1993*.

Hinton, G. E., Osindero, S., and Teh, Y. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, **18**, 1527–1554.

Hubel, D. H. and Wiesel, T. N. (1959). Receptive fields of single neurons in the cat's striate cortex. *Journal of Physiology*, **148**, 574–591.

Hurri, J. and Hyvärinen, A. (2003). Temporal coherence, natural image sequences, and the visual cortex. In *NIPS'2002*.

Hyvärinen, A. (2005). Estimation of non-normalized statistical models using score matching. *J. Machine Learning Res.*, **6**.

Hyvärinen, A. (2007). Some extensions of score matching. *Computational Statistics and Data Analysis*, **51**, 2499–2512.

Hyvärinen, A. (2008). Optimal approximation of signal priors. *Neural Computation*, **20**(12), 3087–3110.

Hyvärinen, A. and Hoyer, P. (2000). Emergence of phase and shift invariant features by decomposition of natural images into independent feature subspaces. *Neural Computation*, **12**(7).

Hyvärinen, A., Karhunen, J., and Oja, E. (2001a). *Independent Component Analysis*. Wiley-Interscience.

Hyvärinen, A., Hoyer, P. O., and Inki, M. (2001b). Topographic independent component analysis. *Neural Computation*, **13**(7), 1527–1558.

Hyvärinen, A., Hurri, J., and Hoyer, P. O. (2009). *Natural Image Statistics: A probabilistic approach to early computational vision*. Springer-Verlag.

Jaeger, H. (2007). Echo state network. *Scholarpedia*, **2**(9), 2330.

Jain, V. and Seung, S. H. (2008). Natural image denoising with convolutional networks. In *NIPS'2008*.

Jarrett, K., Kavukcuoglu, K., Ranzato, M., and LeCun, Y. (2009). What is the best multi-stage architecture for object recognition? In *ICCV'09*.

Jenatton, R., Audibert, J.-Y., and Bach, F. (2009). Structured variable selection with sparsity-inducing norms. Technical report, arXiv:0904.3523.

Jutten, C. and Herault, J. (1991). Blind separation of sources, part I: an adaptive algorithm based on neuromimetic architecture. *Signal Processing*, **24**, 1–10.

Kavukcuoglu, K., Ranzato, M., and LeCun, Y. (2008). Fast inference in sparse coding algorithms with applications to object recognition. CBLL-TR-2008-12-01, NYU.

Kavukcuoglu, K., Ranzato, M.-A., Fergus, R., and LeCun, Y. (2009). Learning invariant features through topographic filter maps. In *CVPR'2009*.

Kavukcuoglu, K., Sermanet, P., Boureau, Y.-L., Gregor, K., Mathieu, M., and LeCun, Y. (2010). Learning convolutional feature hierarchies for visual recognition. In *NIPS'2010*.

Kingma, D. and LeCun, Y. (2010). Regularized estimation of image statistics by score matching. In *NIPS'2010*.

Kivinen, J. J. and Williams, C. K. I. (2012). Multiple texture Boltzmann machines. In *AISTATS'2012*.

Körding, K. P., Kayser, C., Einhäuser, W., and König, P. (2004). How are complex cell properties adapted to the statistics of natural stimuli? *J. Neurophysiology*, **91**.

Krizhevsky, A. (2010). Convolutional deep belief networks on CIFAR-10. Technical report, U. Toronto.

Krizhevsky, A. and Hinton, G. (2009). Learning multiple layers of features from tiny images. Technical report, U. Toronto.

Krizhevsky, A., Sutskever, I., and Hinton, G. (2012). ImageNet classification with deep convolutional neural networks. In *NIPS'2012*.

Larochelle, H. and Bengio, Y. (2008). Classification using discriminative restricted Boltzmann machines. In *ICML'2008*.

Larochelle, H., Bengio, Y., Louradour, J., and Lamblin, P. (2009). Exploring strategies for training deep neural networks. *Journal of Machine Learning Research*, **10**, 1–40.

Lazebnik, S., Schmid, C., and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR'2006*.

Le, H.-S., Oparin, I., Allauzen, A., Gauvin, J.-L., and Yvon, F. (2013). Structured output layer neural network language models for speech recognition. *IEEE Trans. Audio, Speech & Language Processing*.

Le, Q., Ngiam, J., Chen, Z., hao Chia, D. J., Koh, P. W., and Ng, A. (2010). Tiled convolutional neural networks. In *NIPS'2010*.

Le, Q., Ngiam, J., Coates, A., Lahiri, A., Prochnow, B., and Ng, A. (2011a). On optimization methods for deep learning. In *ICML'2011*.

Le, Q. V., Karpenko, A., Ngiam, J., and Ng, A. Y. (2011b). ICA with reconstruction cost for efficient overcomplete feature learning. In *NIPS'2011*.

Le, Q. V., Zou, W. Y., Yeung, S. Y., and Ng, A. Y. (2011c). Learning hierarchical spatio-temporal features for action recognition with independent subspace analysis. In *CVPR'2011*.

Le Roux, N., Bengio, Y., Lamblin, P., Joliveau, M., and Kegl, B. (2008a). Learning the 2-D topology of images. In *NIPS'07*.

Le Roux, N., Manzagol, P.-A., and Bengio, Y. (2008b). Topmoumoute online natural gradient algorithm. In *NIPS'07*.

LeCun, Y. (1986). Learning processes in an asymmetric threshold network. In *Disordered Systems and Biological Organization*, pages 233–240. Springer-Verlag.

LeCun, Y. (1987). *Modèles connexionistes de l'apprentissage*. Ph.D. thesis, Université de Paris VI.

LeCun, Y. (1989). Generalization and network design strategies. In *Connectionism in Perspective*. Elsevier Publishers.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*.

LeCun, Y., Bottou, L., Orr, G. B., and Müller, K. (1998a). Efficient backprop. In *Neural Networks, Tricks of the Trade*.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998b). Gradient based learning applied to document recognition. *Proc. IEEE*.

Lee, H., Ekanadham, C., and Ng, A. (2008). Sparse deep belief net model for visual area V2. In *NIPS'07*.

Lee, H., Grosse, R., Ranganath, R., and Ng, A. Y. (2009a). Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *ICML'2009*.

Lee, H., Pham, P., Largman, Y., and Ng, A. (2009b). Unsupervised feature learning for audio classification using convolutional deep belief networks. In *NIPS'2009*.

Lin, Y., Tong, Z., Zhu, S., and Yu, K. (2010). Deep coding network. In *NIPS'2010*.

Lowe, D. (1999). Object recognition from local scale invariant features. In *ICCV'99*.

Mallat, S. (2012). Group invariant scattering. *Communications on Pure and Applied Mathematics*.

Marlin, B. and de Freitas, N. (2011). Asymptotic efficiency of deterministic estimators for discrete energy-based models: Ratio matching and pseudolikelihood. In *UAI'2011*.

Marlin, B., Swersky, K., Chen, B., and de Freitas, N. (2010). Inductive principles for restricted Boltzmann machine learning. In *AISTATS'2010*, pages 509–516.

Martens, J. (2010). Deep learning via Hessian-free optimization. In *ICML'2010*, pages 735–742.

Martens, J. and Sutskever, I. (2011). Learning recurrent neural networks with Hessian-free optimization. In *ICML'2011*.

Memisevic, R. and Hinton, G. E. (2010). Learning to represent spatial transformations with factored higher-order Boltzmann machines. *Neural Comp.*, **22**(6).

Mesnil, G., Dauphin, Y., Glorot, X., Rifai, S., Bengio, Y., Goodfellow, I., Lavoie, E., Muller, X., Desjardins, G., Warde-Farley, D., Vincent, P., Courville, A., and Bergstra, J. (2011). Unsupervised and transfer learning challenge: a deep learning approach. In *JMLR W&CP: Proc. Unsupervised and Transfer Learning*, volume 7.

Mikolov, T., Deoras, A., Kombrink, S., Burget, L., and Cernocky, J. (2011). Empirical evaluation and combination of advanced language modeling techniques. In *INTERSPEECH'2011*.

Mobahi, H., Collobert, R., and Weston, J. (2009). Deep learning from temporal coherence in video. In *ICML'2009*.

Mohamed, A., Dahl, G., and Hinton, G. (2012). Acoustic modeling using deep belief networks. *IEEE Trans. on Audio, Speech and Language Processing*, **20**(1), 14–22.

Montufar, G. F. and Morton, J. (2012). When does a mixture of products contain a product of mixtures? Technical report, arXiv:1206.0387.

Murray, I. and Salakhutdinov, R. (2009). Evaluating probabilities under high-dimensional latent variable models. In *NIPS'2008*, pages 1137–1144.

Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted Boltzmann machines. In *ICML'10*.

Neal, R. M. (1992). Connectionist learning of belief networks. *Artificial Intelligence*, **56**, 71–113.

Neal, R. M. (1993). Probabilistic inference using Markov chain Monte-Carlo methods. Technical Report CRG-TR-93-1, Dept. of Computer Science, University of Toronto.

Ngiam, J., Chen, Z., Koh, P., and Ng, A. (2011). Learning deep energy models. In *Proc. ICML'2011*. ACM.

Olshausen, B. A. and Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, **381**, 607–609.

Orr, G. and Muller, K.-R., editors (1998). *Neural networks: tricks of the trade*. Lect. Notes Comp. Sc. Springer-Verlag.

Pascanu, R. and Bengio, Y. (2013). Natural gradient revisited. Technical report, arXiv:1301.3584.

Raiko, T., Valpola, H., and LeCun, Y. (2012). Deep learning made easier by linear transformations in perceptrons. In *AISTATS'2012*.

Raina, R., Battle, A., Lee, H., Packer, B., and Ng, A. Y. (2007). Self-taught learning: transfer learning from unlabeled data. In *ICML'2007*.

Ranzato, M. and Hinton, G. H. (2010). Modeling pixel means and covariances using factorized third-order Boltzmann machines. In *CVPR'2010*, pages 2551–2558.

Ranzato, M., Poultney, C., Chopra, S., and LeCun, Y. (2007). Efficient learning of sparse representations with an energy-based model. In

*NIPS'2006*.

Ranzato, M., Boureau, Y., and LeCun, Y. (2008). Sparse feature learning for deep belief networks. In *NIPS'2007*.

Ranzato, M., Krizhevsky, A., and Hinton, G. (2010a). Factored 3-way restricted Boltzmann machines for modeling natural images. In *AISTATS'2010*, pages 621–628.

Ranzato, M., Mnih, V., and Hinton, G. (2010b). Generating more realistic images using gated MRF's. In *NIPS'2010*.

Ranzato, M., Susskind, J., Mnih, V., and Hinton, G. (2011). On deep generative models with applications to recognition. In *CVPR'2011*.

Riesenhuber, M. and Poggio, T. (1999). Hierarchical models of object recognition in cortex. *Nature Neuroscience*.

Rifai, S., Vincent, P., Muller, X., Glorot, X., and Bengio, Y. (2011a). Contractive auto-encoders: Explicit invariance during feature extraction. In *ICML'2011*.

Rifai, S., Mesnil, G., Vincent, P., Muller, X., Bengio, Y., Dauphin, Y., and Glorot, X. (2011b). Higher order contractive auto-encoder. In *ECML PKDD*.

Rifai, S., Dauphin, Y., Vincent, P., Bengio, Y., and Muller, X. (2011c). The manifold tangent classifier. In *NIPS'2011*.

Rifai, S., Bengio, Y., Dauphin, Y., and Vincent, P. (2012). A generative process for sampling contractive auto-encoders. In *ICML'2012*.

Roweis, S. (1997). EM algorithms for PCA and sensible PCA. CNS Technical Report CNS-TR-97-02, Caltech.

Roweis, S. and Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, **290**(5500).

Salakhutdinov, R. (2010a). Learning deep Boltzmann machines using adaptive MCMC. In *ICML'2010*.

Salakhutdinov, R. (2010b). Learning in Markov random fields using tempered transitions. In *NIPS'2010*.

Salakhutdinov, R. and Hinton, G. E. (2007). Semantic hashing. In *SIGIR'2007*.

Salakhutdinov, R. and Hinton, G. E. (2009). Deep Boltzmann machines. In *AISTATS'2009*, pages 448–455.

Salakhutdinov, R. and Larochelle, H. (2010). Efficient learning of deep Boltzmann machines. In *AISTATS'2010*.

Salakhutdinov, R., Mnih, A., and Hinton, G. E. (2007). Restricted Boltzmann machines for collaborative filtering. In *ICML 2007*.

Savard, F. (2011). *Réseaux de neurones à relaxation entraînés par critère d'autoencodeur débruitant*. Master's thesis, U. Montréal.

Schmah, T., Hinton, G. E., Zemel, R., Small, S. L., and Strother, S. (2009). Generative versus discriminative training of RBMs for classification of fMRI images. In *NIPS'2008*, pages 1409–1416.

Schölkopf, B., Smola, A., and Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, **10**, 1299–1319.

Schwenk, H., Rousseau, A., and Attik, M. (2012). Large, pruned or continuous space language models on a gpu for statistical machine translation. In *Workshop on the future of language modeling for HLT*.

Seide, F., Li, G., and Yu, D. (2011a). Conversational speech transcription using context-dependent deep neural networks. In *Interspeech 2011*, pages 437–440.

Seide, F., Li, G., and Yu, D. (2011b). Feature engineering in context-dependent deep neural networks for conversational speech transcription. In *ASRU'2011*.

Serre, T., Wolf, L., Bileschi, S., and Riesenhuber, M. (2007). Robust object recognition with cortex-like mechanisms. *IEEE Trans. Pattern Anal. Mach. Intell.*, **29**(3), 411–426.

Seung, S. H. (1998). Learning continuous attractors in recurrent networks. In *NIPS'1997*.

Simard, D., Steinkraus, P. Y., and Platt, J. C. (2003). Best practices for convolutional neural networks. In *ICDAR'2003*.

Simard, P., Victorri, B., LeCun, Y., and Denker, J. (1992). Tangent prop - A formalism for specifying selected invariances in an adaptive network. In *NIPS'1991*.

Simard, P. Y., LeCun, Y., and Denker, J. (1993). Efficient pattern recognition using a new transformation distance. In *NIPS'92*.

Smolensky, P. (1986). Information processing in dynamical systems: Foundations of harmony theory. In D. E. Rumelhart and J. L. McClelland, editors, *Parallel Distributed Processing*, volume 1, chapter 6, pages 194–281. MIT Press, Cambridge.

Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. In *NIPS'2012*.

Socher, R., Huang, E. H., Pennington, J., Ng, A. Y., and Manning, C. D. (2011a). Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In *NIPS'2011*.

Socher, R., Pennington, J., Huang, E. H., Ng, A. Y., and Manning, C. D. (2011b). Semi-supervised recursive autoencoders for predicting sentiment distributions. In *EMNLP'2011*.

Srivastava, N. and Salakhutdinov, R. (2012). Multimodal learning with deep boltzmann machines. In *NIPS'2012*.

Stoyanov, V., Ropson, A., and Eisner, J. (2011). Empirical risk minimization of graphical model parameters given approximate inference, decoding, and model structure. In *AISTATS'2011*.

Sutskever, I. (2012). *Training Recurrent Neural Networks*. Ph.D. thesis, Departement of computer science, University of Toronto.

Sutskever, I. and Tieleman, T. (2010). On the Convergence Properties of Contrastive Divergence. In *AISTATS'2010*.

Sutskever, I., Hinton, G., and Taylor, G. (2009). The recurrent temporal restricted Boltzmann machine. In *NIPS'2008*.

Swersky, K. (2010). *Inductive Principles for Learning Restricted Boltzmann Machines*. Master's thesis, University of British Columbia.

Swersky, K., Ranzato, M., Buchman, D., Marlin, B., and de Freitas, N. (2011). On score matching for energy based models: Generalizing autoencoders and simplifying deep learning. In *Proc. ICML'2011*. ACM.

Taylor, G. and Hinton, G. (2009). Factored conditional restricted Boltzmann machines for modeling motion style. In *ICML'2009*.

Taylor, G., Fergus, R., LeCun, Y., and Bregler, C. (2010). Convolutional learning of spatio-temporal features. In *ECCV'10*.

Tenenbaum, J., de Silva, V., and Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, **290**(5500), 2319–2323.

Tieleman, T. (2008). Training restricted Boltzmann machines using approximations to the likelihood gradient. In *ICML'2008*, pages 1064–1071.

Tieleman, T. and Hinton, G. (2009). Using fast weights to improve persistent contrastive divergence. In *ICML'2009*.

Tipping, M. E. and Bishop, C. M. (1999). Probabilistic principal components analysis. *J. Roy. Stat. Soc. B*, (3).

Turaga, S. C., Murray, J. F., Jain, V., Roth, F., Helmstaedter, M., Briggman, K., Denk, W., and Seung, H. S. (2010). Convolutional networks can learn to generate affinity graphs for image segmentation. *Neural Computation*, **22**, 511–538.

van der Maaten, L. (2009). Learning a parametric embedding by preserving local structure. In *AISTATS'2009*.

van der Maaten, L. and Hinton, G. E. (2008). Visualizing data using t-SNE. *J. Machine Learning Res.*, **9**.

Vincent, P. (2011). A connection between score matching and denoising autoencoders. *Neural Computation*, **23**(7).

Vincent, P. and Bengio, Y. (2003). Manifold Parzen windows. In *NIPS'2002*. MIT Press.

Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders. In *ICML 2008*.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., and Manzagol, P.-A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J. Machine Learning Res.*, **11**.

Weinberger, K. Q. and Saul, L. K. (2004). Unsupervised learning of image manifolds by semidefinite programming. In *CVPR'2004*, pages 988–995.

Welling, M. (2009). Herding dynamic weights for partially observed random field models. In *UAI'2009*.

Welling, M., Hinton, G. E., and Osindero, S. (2003). Learning sparse topographic representations with products of Student-t distributions. In *NIPS'2002*.

Weston, J., Ratle, F., and Collobert, R. (2008). Deep learning via

semi-supervised embedding. In *ICML 2008*.

Weston, J., Bengio, S., and Usunier, N. (2010). Large scale image annotation: learning to rank with joint word-image embeddings. *Machine Learning*, **81**(1), 21–35.

Wiskott, L. and Sejnowski, T. (2002). Slow feature analysis: Unsupervised learning of invariances. *Neural Computation*, **14**(4), 715–770.

Younes, L. (1999). On the convergence of Markovian stochastic algorithms with rapidly decreasing ergodicity rates. *Stochastics and Stochastic Reports*, **65**(3), 177–228.

Yu, D., Wang, S., and Deng, L. (2010). Sequential labeling using deep-structured conditional random fields. *IEEE Journal of Selected Topics in Signal Processing*.

Yu, K. and Zhang, T. (2010). Improved local coordinate coding using local tangents. In *ICML'2010*.

Yu, K., Zhang, T., and Gong, Y. (2009). Nonlinear learning using local coordinate coding. In *NIPS'2009*.

Yu, K., Lin, Y., and Lafferty, J. (2011). Learning image representations from the pixel level via hierarchical sparse coding. In *CVPR*.

Yuille, A. L. (2005). The convergence of contrastive divergences. In *NIPS'2004*, pages 1593–1600.

Zeiler, M., Krishnan, D., Taylor, G., and Fergus, R. (2010). Deconvolutional networks. In *CVPR'2010*.

Zou, W. Y., Ng, A. Y., and Yu, K. (2011). Unsupervised learning of visual invariance with temporal coherence. In *NIPS 2011 Workshop on Deep Learning and Unsupervised Feature Learning*.