软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2018,29(4):1131−1142 [doi: 10.13328/j.cnki.jos.005263]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于权值不确定性的玻尔兹曼机算法*

丁世飞[1,2], 张 健[1,2], 史忠植[2]

[1](中国矿业大学 计算机科学与技术学院,江苏 徐州 221116)

[2](中国科学院 计算技术研究所 智能信息处理重点实验室,北京 100190)

通讯作者: 丁世飞, E-mail: dingsf@cumt.edu.cn

摘 要: 受限制的玻尔兹曼机(RBM)是一种无向图模型.基于RBM的深度学习模型包括深度置信网(DBN)和深度玻尔兹曼机(DBM)等.在神经网络和 RBM 的训练过程中,过拟合问题是一个比较常见的问题.针对神经网络的训练,权值随机变量(weight random variables)、Dropout 方法和早期停止方法已被用于缓解过拟合问题.首先,改变 RBM 模型中的训练参数,使用随机变量代替传统的实值变量,构建了基于随机权值的受限的波尔兹曼机(weight uncertainty RBM,简称 WRBM),接下来,在 WRBM 基础上构建了相应的深度模型:Weight uncertainty Deep Belief Network (WDBN)和 Weight uncertainty Deep Boltzmann Machine(WDBM),并且通过实验验证了 WDBN 和 WDBM 的有效性.最后,为了更好地建模输入图像,引入基于条件高斯分布的 RBM 模型,构建了基于 spike-and-slab RBM(ssRBM)的深度模型,并通过实验验证了模型的有效性.

关键词: 玻尔兹曼机;深度玻尔兹曼机;深度置信网;权值不确定性

中图法分类号: TP183

中文引用格式: 丁世飞,张健,史忠植.基于权值不确定性的玻尔兹曼机算法.软件学报,2018,29(4):1131−1142. http://www.jos.org.cn/1000-9825/5263.htm

英文引用格式: Ding SF, Zhang J, Shi ZZ. Algorithms of Boltzmann machines based on weight uncertainty. Ruan Jian Xue Bao/ Journal of Software, 2018,29(4):1131−1142 (in Chinese). http://www.jos.org.cn/1000-9825/5263.htm

Algorithms of Boltzmann Machines Based on Weight Uncertainty

DING Shi-Fei[1,2], ZHANG Jian[1,2], SHI Zhong-Zhi[2]

[1](School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China)

[2](Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, The Chinese Academy of Sciences, Beijing 100190, China)

Abstract: Based on the restricted Boltzmann machine (RBM), which is a probabilistic graphical model, deep learning models contain deep belief net (DBN) and deep Boltzmann machine (DBM). The overfitting problems commonly exist in neural networks and RBM models. In order to alleviate the overfitting problem, this paper introduces weight random variables to the conventional RBM model and, then builds weight uncertainty deep models based on maximum likelihood estimation. In the experimental section, the paper verifies the effectiveness of the weight uncertainty RBM. In order to improve the image recognition ability, the paper introduces the spike-and-slab RBM (ssRBM) to weight uncertainty RBM and then builds the deep models. The experiments show that the deep models based on weight random variables are effective.

Key words: RBM (restricted Boltzmann machine); DBM (deep Boltzmann machine); DBN (deep belief net); weight uncertainty

* 基金项目: 国家自然科学基金(61672522, 61379101); 国家重点基础研究发展计划(973)(2013CB329502)

Foundation item: National Natural Science Foundation of China (61672522, 61379101); National Basic Research Program of China (973) (2013CB329502)

收稿时间: 2016-09-06; 修改时间: 2016-10-19, 2016-12-06; 采用时间: 2017-01-24; jos 在线出版时间: 2017-03-31

CNKI 网络优先出版: 2017-03-31 21:54:31, http://kns.cnki.net/kcms/detail/11.2560.TP.20170331.2154.002.html

从监督学习的角度看,深度学习模型在解决分类问题时可以看作是一种多层感知器.从误差曲面上看,神经网络收敛的位置取决于权值的初始化值.在这种情况下,如果权值是随机生成的,那么权值可能初始化在误差曲面中比较差的位置上,此时,使用 BP 算法来训练神经网络将会得到比较差的局部最优解.深度学习可以有效地缓解这个问题.通过无监督的预训练过程,神经网络可以调整权值初始化的位置,有助于收敛到更好的局部最优解[1].

自深度学习提出以来,便在学术界和工业界引起了广泛的关注.基于 RBM(restricted Boltzmann machine)的 DBN(deep belief net)模型是深度学习领域中的经典模型之一.RBM 是一种无监督的模型,该模型用于表达输入数据的分布规律[2−4],RBM 的数学基础是概率图模型,许多有效的方法可以用来训练 RBM,例如对比散度算法(CD)、持续的马尔可夫链(persistent Markov chain)、均匀场方法(mean field method)等.在 RBM 的基础上,Hinton 等人在 2006 年提出了 DBN 模型[5].DBN 模型通过无监督的逐层初始化,为多层神经网络提供了一种有效的预训练方法[6].基于 DBN 和 RBM 模型,许多研究成果被提了出来.Lee 等人将 RBM 和卷积神经网络(CNN)相结合,提出了一种卷积深度置信网(convolutional deep belief network)[7,8],为 CNN 的训练提供了一种有效的逐层初始化方法.基于 RBM,另一个经典的模型是深度玻尔兹曼机(DBM)[9].相比于 DBN 模型,DBM 模型在图像的识别和重构方面十分有效[10].RBM 的应用范围还包括语音识别领域,结合递归神经网络(RNN)和 RBM 模型,神经网络在语音识别中同样取得了出色的效果[11,12].与此同时,还有许多其他的模型可以被用于深度学习中,例如自动编码器(AE)、极限学习机(ELM)等[13−15].

然而,在上述的深度模型中依然存在一些不足,其中,过拟合问题是模型训练中常见的问题.为了防止神经网络的过拟合,许多方法被相继提出,其中,Dropout 方法可以被使用在 RBM 模型中[16].然而,通过实验我们发现:Dropout 方法虽然在分类精度上取得了明显的提升,但是 Dropout RBM 的图像重构能力相比于传统 RBM 有所下降.为了解决这个问题,我们尝试在 RBM 模型中引入随机变量来替代实值变量.Weight uncertainty 方法将权值随机变量引入到传统的 BP 算法中,在训练神经网络时取得了比较出色的效果[17].在 Weight uncertainty 方法中,通过把权值看作随机变量,整个神经网络可以看作是一组网络的集成,其中,权值是服从高斯分布的随机变量.从降噪的观点上看,Dropout 方法和权值随机变量均达到了非常类似的效果,权值随机变量相当于对神经网络加入噪声,算法训练的过程同时也是降噪的过程.在实验部分,我们首先将权值随机变量应用到 RBM 中,测试了其分类能力和图像重构能力.然后,我们构造了 WDBN 模型和 WDBM 模型,并与基于 Dropout 方法的 DBN 模型进行了比较.为了进一步提高模型对图像的建模能力,我们希望更好地建模像素点之间的相关性,因此,我们将条件高斯分布的思想引入到模型中,通过使用 spike-and-slab Boltzmann Machine(ssRBM)模型,并对能量函数进行微调,最后将其作为特征提取器,构建了相应的深度模型,并通过实验验证了其有效性.

1 受限的玻尔兹曼机

RBM 是基于能量函数的,该模型由一个可见层 $\vec{v}$ 和一个隐藏层 $\vec{h}$ 组成.如果 RBM 的节点状态都是二值的,那么其能量函数可以表示如下:

$$E(\vec{v},\vec{h})=-\sum_{i=1}^{n_v}a_i v_i-\sum_{j=1}^{n_h}b_j h_j-\sum_{i=1}^{n_v}\sum_{j=1}^{n_h}h_j\times W_{ji}\times v_i \tag{1}$$

其中,$\vec{a}$ 是可见层的偏置;$\vec{b}$ 是隐藏层的偏置;W 是可见层单元和隐藏层单元之间的权值矩阵;$\vec{v}$ 是可见层的状态;$\vec{h}$ 是隐藏层的状态,下标表示其分量;n_v 表示可见层节点数;n_h 表示隐藏层节点数.那么,基于能量函数 $E(\vec{v},\vec{h})$ 的联合概率分布可以表示如下:

$$P(\vec{v},\vec{h})=\frac{1}{Z}\mathrm{e}^{-E(\vec{v},\vec{h})} \tag{2}$$

其中,Z 表示配分函数(partition function):

$$Z=\sum_{v,h}\mathrm{e}^{-E(\vec{v},\vec{h})} \tag{3}$$

我们的目标是使得概率分布函数 $\sum_{i=1}^{n_s} P(\vec{v}^{(i)})$ 最大,其中,n_s 表示样本数,$v^{(i)}$表示第 i 个样本. $P(\vec{v})$ 表示分布 $P(\vec{v},\vec{h})$ 的边缘分布,表达式如下:

$$P(\vec{v})=\sum_h P(\vec{v},\vec{h})=\frac{1}{Z}\sum_h \mathrm{e}^{-E(\vec{v},\vec{h})} \tag{4}$$

极大似然函数定义如下:

$$L_s=\ln\prod_{i=1}^{n_s} P(\vec{v}^{(i)})=\sum_{i=1}^{n_s}\ln P(\vec{v}^{(i)}) \tag{5}$$

其中,n_s 是样本数.这里,我们使用梯度上升方法来最大化似然函数.首先,以一个样本为例对计算过程进行说明.

首先,计算似然函数的偏导数,令 $\theta=(\vec{a},\vec{b},W)$, 我们有:

$$\frac{\partial \ln P(V)}{\partial\theta}=-\sum_h P(\vec{h}\,|\,V)\frac{\partial E(V,\vec{h})}{\partial\theta}+\sum_{v,h}P(\vec{v},\vec{h})\frac{\partial E(\vec{v},\vec{h})}{\partial\theta} \tag{6}$$

其中,V表示一个输入样本,θ是要学习的参数.在RBM模型中,当某一层的节点状态给定时,另一层节点的激活是相互独立的,因此有:

$$p(h_k=1\,|\,\vec{v})=sigmoid\left(b_k+\sum_{i=1}^{n_v}W_{ki}v_i\right) \tag{7}$$

$$p(v_k=1\,|\,\vec{h})=sigmoid\left(a_k+\sum_{j=1}^{n_h}h_jW_{kj}\right) \tag{8}$$

其中,h_k 是 $\vec{h}$ 的分量,v_k 是 $\vec{v}$ 的分量.

Hinton 等人在极大似然估计的基础上,提出了一种对比散度(contrastive divergence,简称 CD)算法来近似求解偏导数[18]:

$$\frac{\partial \ln P(\vec{v})}{\partial W_{ij}}\approx P(h_j=1\,|\,\vec{v}^{(0)})\vec{v}_i^{(0)}-P(h_j=1\,|\,\vec{v}^{(k)})\vec{v}_i^{(k)} \tag{9}$$

$$\frac{\partial \ln P(\vec{v})}{\partial a_i}\approx v_i^{(0)}-v_i^{(k)} \tag{10}$$

$$\frac{\partial \ln P(\vec{v})}{\partial b_i}\approx P(h_i=1\,|\,\vec{v}^{(0)})-P(h_i=1\,|\,\vec{v}^{(k)}) \tag{11}$$

其中,k 是 K 步对比散度算法(K-steps contrastive divergence algorithm,简称 CD-K)中的步数, $\vec{v}^{(0)}$ 表示初始状态下的可见层状态, $\vec{v}^{(k)}$ 表示运行 k 步对比散度算法之后得到的可见层状态.然后,我们用如下公式来更新训练参数:

$$\Delta W_{ij}=\eta(P(h_i=1\,|\,\vec{v}^{(0)})\vec{v}^{(0)}-P(h_i=1\,|\,\vec{v}^{(k)})\vec{v}^{(k)}) \tag{12}$$

$$\Delta a_i=\eta(v_i^{(0)}-v_i^{(k)}) \tag{13}$$

$$\Delta b_i=\frac{\partial \ln P(\vec{v})}{\partial b_i}\approx\eta(P(h_i=1\,|\,\vec{v}^{(0)})-P(h_i=1\,|\,\vec{v}^{(k)})) \tag{14}$$

其中,参数η是学习率.

当输入是实值数据时,我们重新定义如下能量函数:

$$E(\vec{v},\vec{h})=\sum_{i=1}^{n_v}\frac{(v_i-a_i)^2}{2\sigma_i^2}-\sum_{j=1}^{n_h}b_jh_j-\sum_{i=1}^{n_v}\sum_{j=1}^{n_h}\frac{v_i}{\sigma_i}W_{ji}h_j \tag{15}$$

其中,σ表示对角的协方差矩阵.那么,隐藏层的激活概率可以写成如下形式:

$$P(h_k=1\,|\,\vec{v})=sigmoid\left(b_k+\sum_{i=1}^{n_v}W_{ki}v_i\right) \tag{16}$$

可见层的激活概率服从高斯分布,如下表示:

$$P(\vec{v}\mid\vec{h})=\prod_{i=1}\frac{1}{\sigma_i\sqrt{2\pi}}\mathrm{e}^{-\frac{1}{2\sigma_i^2}\left(v_i-a_i-\sigma_i\sum_{j=1}h_jW_{ij}\right)^2} \tag{17}$$

此时的 RBM 又被称为高斯 RBM,也叫均值 RBM[19].

2 RBM 与 BM 的训练算法

关于 RBM 的训练算法有很多,早期 Hinton 等人使用持续的马尔可夫链和模拟退火方法来逼近数据独立期望和数据依赖期望.目前,针对退火、回火算法的研究仍在进行[20,21].虽然使用退火和回火算法可以有效地逼近似然函数,但是退火、回火算法的计算复杂性比较高,因此在处理较大数据时比较困难.另一方面,为了缩短 RBM 的训练时间,均匀场方法被提出以替换 Gibbs 采样[22].在均匀场方法中,隐藏层节点使用实值概率作为其状态值,该方法有助于把 RBM 作为神经网络来训练和解释.但在实践中,以均匀场为代表的变量方法并不适合逼近数据独立性期望.2002 年,Hinton 提出了对比散度算法,这种方法虽然在迭代步长上不够准确,但却保证了梯度方向的正确性,并且具有较快的训练速度.随后,在 CD 算法的基础上,PCD 算法及其改进 FPCD 算法被提了出来[23,24].在 DBM 的训练中,Salakhutdinov 使用均匀场方法来逼近数据依赖性期望,使用持续的马尔可夫链逼近数据独立性期望,取得了良好的效果.

2.1 均匀场方法

在概率图模型中,均匀场推理使用一个近似的后验分布 $Q(h|v;\lambda)$来替换真实的后验分布 $P(h|v;\theta)$.即,选择一个特殊的后验分布 $Q(h|v;\lambda)$来最小化如下 KL 散度:

$$\lambda^*=\arg\min_{\lambda}KL[Q(h\mid v)\,\|\,P(h\mid v)] \tag{18}$$

其中,KL 散度定义为 $KL[Q(h\mid v)\,\|\,P(h\mid v)]=\sum_{\{h\}}Q(h\mid v)\ln\frac{Q(h\mid v)}{P(h\mid v)}$.

对于概率函数 $P(v)$,有:

$$\ln P(v)=\ln\sum_h P(h,v)=\ln\sum_h Q(h\mid v)\frac{P(h,v)}{Q(h\mid v)}\geqslant\sum_h Q(h\mid v)\ln\left(\frac{P(h,v)}{Q(h\mid v)}\right) \tag{19}$$

对于均匀场玻尔兹曼机,有:

$$Q(h\mid v,u)=\prod_{i\in H}u^{S_i}(1-u_i)^{1-S_i} \tag{20}$$

那么,KL 散度可以写成:

$$KL[Q\,\|\,P]=\sum_i(u_i\ln u_i+(1-u_i)\ln(1-u_i))-\sum_{i<j}\theta_{ij}u_iu_j-\sum_i\theta_i^cu_i+\ln Z_c \tag{21}$$

其中,Z_c 是配分函数,$Z_c=\sum_{\{H\}}\left(\exp\left(\sum_{i<j}\theta_{ij}S_iS_j+\sum_i\theta_i^cS_i\right)\right)$,$\theta_i^c=\theta_i+\sum_{j\in V}\theta_{ij}S_j$.$S_i$ 和 S_j 是独立的随机变量,表示模型中的节点,其均值分别为 u_i 和 u_j.接下来对 KL 散度求关于 u_i 的偏导数,并令其为 0,我们得到 $u_i=sigmoid\left(\sum_j\theta_{ij}u_j+\theta_i\right)$.

通常来说,对于均匀场方法,可以使用 EM 算法来求解.有证据显示:对于相同的测试数据,使用均匀场方法要比用 Gibbs 采样快 10~30 倍[25].以 RBM 的形式可以把均匀场方法表示如下:

$$\ln P(v;\theta)\geqslant\sum_{i,j}W_{ij}v_iu_j+\sum_i b_iv_i-\ln Z-\sum_j(u_j\ln u_j+(1-u_j)\ln(1-u_j)) \tag{22}$$

隐藏层变量的激活值可以表示为

$$u_i=sigmoid\left(\sum_j W_{ij}v_j+b_i\right) \tag{23}$$

2.2 持续的马尔可夫链

持续的马尔可夫链又称为随机逼近算法,是一种利用马尔可夫链进行分块 Gibbs 采样的算法.只要马尔可夫链足够长,更新步长不太大,马尔可夫链就可以达到稳态.持续的马尔可夫链可以有效地逼近数据独立性期望.在 RBM 模型中,对于每一个样本 $(\tilde{v}^t,\tilde{h}^t)$, 通过运行一次 Gibbs 采样,得到 $(\tilde{v}^{t+1},\tilde{h}^{t+1})$, 其中,$t$ 表示迭代的时间步.算法的步骤见表 1.

Table 1 Algorithm of persistent Markov chains

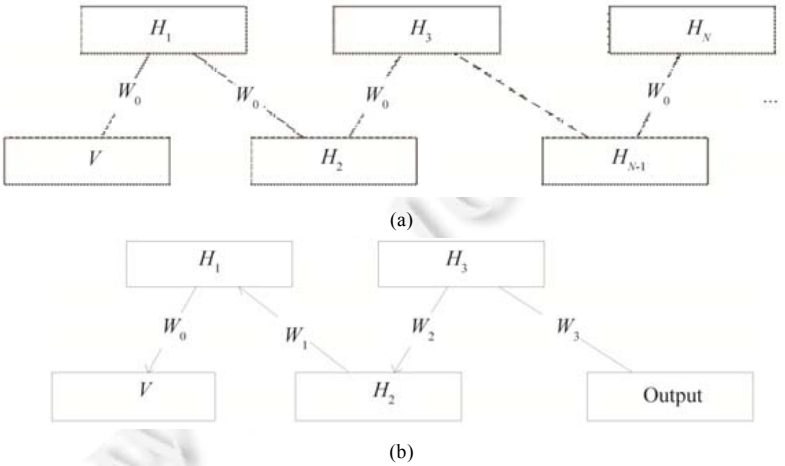
表 1 持续的马尔可夫链算法步骤

持续的马尔可夫链算法.
随机初始化 θ_0 和 M 个样本 $\{\tilde{x}^{0,1},...,\tilde{x}^{0,M}\}$.
for t=0:T(迭代步数)
for i=1:M(parallel Markov chains 数目)
由 $\tilde{x}^{t,i}$ 采样 $\tilde{x}^{t+1,i}$, 利用 $T_{\theta^t}(\tilde{x}^{t+1,i}\leftarrow\tilde{x}^{t,i})$;
End for
Update: $\theta^{t+1}=\theta^t+\alpha_t\left[\Phi(\bar{x})-\frac{1}{M}\sum_{m=1}^{M}\Phi(\tilde{x}^{t+1,m})\right]$
减小 α_t.
End for.

3 深度置信网与深度玻尔兹曼机

3.1 深度置信网

DBN 是一种混合的图模型,其顶端的两层是无向图,用来表示关联记忆.余下的层是一个有向图,构成一个置信网.DBN 的训练是一个贪婪的逐层初始化的过程,首先,假设 DBN 是一个无限多层的模型,而且每一层的节点数相同,那么使用相同的权值 W_0 来初始化整个网络,此时,该模型的训练可以看作是训练一个单层的 RBM,训练结束后,固定第 1 层的权值 W_0 不变,其余层的权值使用 W_1 来替换,然后训练剩余的网络.在这种情况下,先验信息会随着每一层的训练而不断更新.Hinton 等人证明:在这种情况下,每一次 RBM 的贪婪预训练都可以收紧变量边界,也就是说,$\ln p(v|W_1,W_2)\geqslant\ln p(v|W_1)$.预训练过程如图 1 所示,其中,图 1(a)表示把 DBN 看作一个无限层的结构,并且固定权值为 W_0,那么 DBN 的训练等效于 RBM 的训练.然后固定第 1 层权值不变,并且把 V 和 W_0 的乘积作为下一个 DBN 的输入.重复此过程,最后一层替换为输出层,便得到图 1(b)所示的模型结构.将该模型的隐藏层按顺序逐层重新组织,便得到了 DBN 的神经网络结构.



(a)

(b)

Fig.1 Shows the diagram of training process in DBN

图 1 DBN 训练过程的示意图

当完成逐层的预训练之后,从神经网络的角度来看,可以使用 BP 算法对网络进行权值的微调,此时,神经网络的各个节点的激活方式要与 DBN 中各个节点的激活一致.从监督学习的角度来看,该算法有效地缓解了多层感知器极易陷入较差的局部最优解的情况,成为深度学习的经典算法.

3.2 深度玻尔兹曼机

DBM 不同于 DBN,其网络结构仍然是玻尔兹曼机.在 DBM 的训练中,每一层节点的激活取决于与该节点相连的上下两层的节点,可以使用传统的玻尔兹曼机训练算法来训练 DBM,但是效果并不理想.Salakhutdinov 指出,同样可以使用逐层训练的方式初始化 DBM.不同于 DBN,在每一层预训练结束后,不是替换整个先验,而是按比例进行替换.Salakhutdinov 探究了以不同比例替换先验可以取得的不同效果,以 3 层的 DBM 和几何均值为例,首先对输入层和输出层加倍,然后加倍第 1 个隐藏层和第 2 个隐藏层之间的权值 W_2.在 DBM 的预训练过程中,替换的比例为 0.5.DBM 与 DBN 的网络结构如图 2 所示.

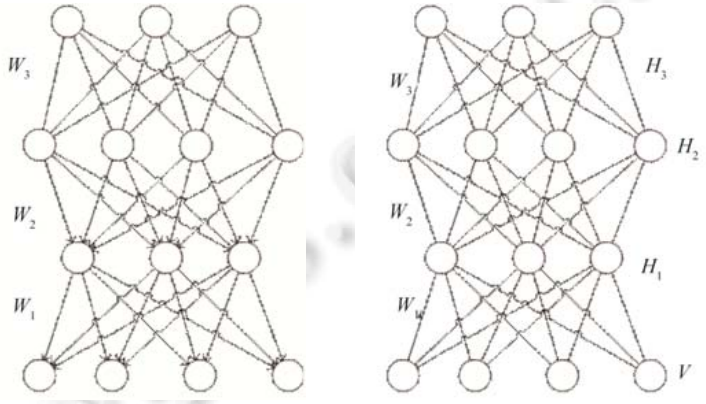


Fig.2 Structure of DBN and DBM

图 2 DBN 和 DBM 的网络结构图

不同于 DBN,在 DBM 中,每一个单元的激活都取决于与其直接相连的所有节点,即,DBM 中隐藏层节点的激活取决于其上下两个相邻层.因此,DBM 的概率函数可以如下表示(其中,上标表示层数,下标表示维度坐标):

$$p(h_j^1=1\mid \vec{v},\vec{h}^2)=sigmoid\left(\sum_i W_{ij}^1 v_i+\sum_m W_{jm}^2 h_m^2+b_j^1\right) \tag{24}$$

$$p(h_m^2=1\mid \vec{h}^1,\vec{h}^3)=sigmoid\left(\sum_j W_{jm}^2 h_j^1+\sum_l W_{ml}^3 h_l^3+b_m^2\right) \tag{25}$$

$$p(h_l^3=1\mid \vec{h}^2)=sigmoid\left(\sum_m W_{ml}^3 h_m^2+b_l^3\right) \tag{26}$$

$$p(v_i=1\mid \vec{h}^1)=sigmoid\left(\sum_j W_{ij}^1 h_j^1+b_j\right) \tag{27}$$

对于 DBM,可以使用随机逼近算法和均匀场算法来逼近偏导数.从神经网络的角度看,DBM 可以看作是一种多层感知器,此时,不同于 DBN,隐藏层单元的激活需要考虑与其相邻的上下两层.

4 基于权值不确定性的深度玻尔兹曼机

4.1 权值不确定性方法

在上述算法中,权值和偏置这两个参数是普通的实值变量,这种情况下,神经网络的训练可能会遇到过拟合问题,在以 RBM 为基础的网络中,Dropout 方法可以有效解决过拟合问题,有效地实现了多个神经网络的集成,也可以认为 Dropout 方法的训练过程是一个消除遮蔽噪声的过程.虽然 Dropout RBM 是一种非常有效的分类算

法,但是对于图像重构问题而言,效果并不理想.我们猜测:由于对神经元的遮蔽是随机的,遮蔽一些节点会使RBM的表达稀疏化,这种稀疏化也许是导致网络重构图像变差的原因.

为了缓解上述问题,我们引入随机变量的思想,将权值看作服从高斯分布的随机变量.只要求得其期望和方差,那么根据该期望和方差生成的采样权值是相对稳定的.同时,每一次权值的采样也可以看作是一个子模型的构建过程.因此,基于权值随机变量的网络模型也可以看作是多个网络的集成.

在Blundell等人的研究中,对于神经网络模型,所有的权值都被表示为随机变量.此时,神经网络的目标函数可以如下表示:

$$\theta' = \arg\min_{\theta} KL[q(W\mid\theta)\,\|\,P(W\mid\theta)] = \arg\min_{\theta} KL[q(W\mid\theta)\,\|\,P(W)] - E_{q(w|\theta)}[\log P(D\mid W)] \tag{28}$$

根据极大后验估计的思想,令:

$$f(W,\theta)=\log q(W|\theta)-\log P(W)P(D|W) \tag{29}$$

其中,先验公式可以表达为高斯分布.在RBM中,为了获得更加出色的图像重构效果和分类效果,我们引入了权值随机变量,这样,RBM的代价函数可以写为极大似然估计$p(v|W)$或者极大后验估计$p(W|v)$的形式.RBM自身的目标函数为$p(v)$,同时也是概率图模型的极大似然函数,因此,为了方便计算,减少超参数的数量,在随机变量的前提下,我们使用极大似然估计的方法来计算RBM的激活概率.假设权值W是符合高斯分布的,其均值为μ,标准差为$\sigma=\log(1+\exp(\rho))$,假设$\varepsilon\sim N(0,I)$.为了方便计算,那么权值可以写成$W=\mu+\log(1+\exp(\rho))\circ\varepsilon$的形式.此时,根据链式求导法则,求导过程变为

$$\frac{\partial\log p(W_{ij})}{\partial W_{ij}}\times\frac{\partial W_{ij}}{\partial\mu}=\left(P(h_j=1\mid\vec{v})v_i-\sum_{v,h}P(v)P(h_j=1\mid\vec{v})v_i\right)\times 1 \tag{30}$$

$$\frac{\partial\log p(W_{ij})}{\partial W_{ij}}\times\frac{\partial W_{ij}}{\partial\rho}=\left(P(h_j=1\mid\vec{v})v_i-\sum_{v,h}P(v)P(h_j=1\mid\vec{v})v_i\right)\times\frac{\varepsilon}{1+\exp(-\rho)} \tag{31}$$

在实验部分,我们基于权值随机变量构建了DBM模型,然后与传统的DBM进行比较.同时,为了与Dropout方法进行对比,我们在实验中测试了Dropout DBN和WDBN的分类能力以及图像的重构误差.通过实验我们发现,权值随机变量在分类精度上达到了与Dropout方法近似的效果.同时,我们的方法具有更低的图像重构误差.

4.2 Boosted CD算法

Boosted CD算法是一种有效的正则化技术,该算法首先被提出用于解决DNA的计算问题[26].在Boosted CD算法中,目标函数可以写成$\sum_n p(\vec{v}_n)+\frac{\lambda c}{2}\sum_j^{n_c}(\vec{v}^{(k)}_{\boldsymbol{n}_c(i-1)+j}-1)^2$,其中,$\vec{v}^{(k)}$表示重构的输入向量,$\vec{v}^{(0)}$表示原始的输入向量.$n_c$表示输入样本中为1的位数,以这种方式进行正则化,形成的梯度称为分类梯度.概率函数可以写成如下形式:

$$\frac{\partial L}{\partial W}\approx \text{Eq.}(2)+\frac{1}{N}\sum_{n=1}^{N} f(v_n^{(k)})h_n^{(k-1)} \tag{32}$$

$$\frac{\partial L}{\partial b}\approx\frac{1}{N}\sum_{n=1}^{N}(\vec{v}_n^{(0)}-\vec{v}_n^{(k)}+f(v_n^{(k)})) \tag{33}$$

其中,$f(v)=v\circ(1-v)\circ g(v), g(v)_i=\sum_{j=1}^{n_c} v_{n_c\left[\frac{i-1}{n_c}\right]+j}-1$,“$\circ$”表示向量对应元素的乘积.

4.3 ssRBM

目前,基于条件高斯分布对图像进行建模的方法主要有高斯-二值受限的玻尔兹曼机(mRBM)、Factored 3 way Boltzmann Machine(cRBM)[27]以及spike and slab Boltzmann Machine(ssRBM)[28,29].其中,mRBM建模的是条件高斯分布的均值,然而mRBM的建模能力比较差,不能很好地还原输入数据的分布.cRBM建模的是条件高斯分布的协方差,此时,建模的精度相比GRBM有所提高.然而,在cRBM中,协方差矩阵是一个非对角矩阵,不能采用分块的Gibbs采样,因此在cRBM中,使用的是混合的蒙特卡洛采样(HMC)方法,然而HMC方法的缺点是参

数过多、运行时间长.在 ssRBM 中,可以同时建模高斯分布的均值和协方差,为了保证可见层单元所服从分布的协方差矩阵是一个对角矩阵,ssRBM 引入了两个随机变量 s 和 h,在给定 s 和 h 的状态时,v 服从条件高斯分布,并且可以使用分块的 Gibbs 采样,修改的能量函数可以如下表示:

$$E(v,s,h)=\frac{1}{2}v^T\Lambda v-\sum_{i=1}^{N}\left(v^TW_is_ih_i-\frac{1}{2}s_i^T\alpha_is_i+b_ih_i\right)=\frac{1}{2}\sum_{d=1}^{D}v_d^2\Lambda_d-\sum_{i=1}^{N}\left(v^TW_is_ih_i-\frac{1}{2}s_i^T\alpha_is_i+\alpha_i\mu_ih_is_i+b_ih_i\right) \quad (34)$$

其中,Λ和α是对角矩阵;W 为可见层单元与隐藏层单元之间的权值矩阵,每一个隐藏层节点 i 关联 2 个变量,即 s_i 和 h_i;μ是一个参数,表示变量 s 的均值.此时,条件概率可以表达如下:

$$P(v\mid s,h)=\frac{P(v,s,h)}{P(s,h)}=N(\Lambda^{-1}\sum_{i=1}^{N}(W_is_ih_i),\Lambda^{-1}) \quad (35)$$

$$P(h_i=1\mid v)=sigmoid\left(\frac{1}{2}v^TW_ih_i\alpha_i^{-1}W_i^Th_iv+v^TW_i\mu_i+b_i\right) \quad (36)$$

$$P(s\mid v,h)=\prod_{i=1}^{N}N(\alpha_i^{-1}v^TW_ih_i+\mu_ih_i,\alpha_i^{-1}) \quad (37)$$

我们的 ssRBM 在原始的模型基础上进行了修正,在使用参数μ的同时,尽量避免过多的参数影响计算.为了建模二值数据,可见层单元的条件概率可以进行如下修改:

$$p(v_j=1\mid s,h)=sigmoid\left(\sum_iW_{ij}s_ih_i+\Lambda_i\right) \quad (38)$$

将 ssRBM 作为我们模型的第 1 层,其余 2 层使用 WRBM 模型.输入为像素点组成的向量,经过第 1 个 ssRBM 之后,使用隐藏层单元 h 的状态作为第 2 个 WRBM 的输入.然后重复该传递过程,直到完成预训练.

5 实验及分析

实验环境如下:CPU:Intel i7 4710hq,内存:16g,GPU:GTX 970m.由于之前并没有权值随机变量在 RBM 中应用的先例,所以我们首先分析了 RBM 和 WRBM 的分类能力和重构能力.使用两个不同大小的 MNIST 数据集来测试 RBM 和 WRBM 的分类和重构能力.为了方便比较权值随机变量的作用,我们使用单层的 RBM 和 WRBM 以及 BP 算法训练.为了形式上的统一,接下来的实验中,BP 过程使用共轭梯度法,迭代次数为 100.在浅层模型中,我们使用的隐藏层节点数为 1 000,模型的学习速率参数选择为 0.005,预训练的迭代次数为 200.在单层 RBM 的实验中,我们使用两个数据集,分别为 MNIST-Basic 和 Rectangles.在多层模型的对比实验中,我们用基于 Dropout 方法建立的深度模型作为对比模型.其中,Dropout DBM 和 Dropout DBN 是在 DBM 模型和 DBN 模型的基础上引入 Dropout 方法构建的深度模型.数据集的属性见表 2.

Table 2 Attributes of data sets

表 2 实验数据的属性

数据集	训练样本数	测试样本数	属性	标签
MNIST-Basic	10 000	50 000	784	10
Rectangles	1 000	50 000	784	2
MNIST	60 000	10 000	784	10

我们首先测试了单层模型来验证权值随机变量的有效性.在这一步的实验中,使用的数据集是 MNIST-Basic 和 Rectangles.

测试精度见表 3,记录的是数据的分类错误率.

Table 3 Number of misclassifications in shallow models

表 3 单层模型的测试精度对比

	MNIST-Basic (%)	Rectangles (%)
RBM-BP	3.628	5.172
Dropout RBM-BP	3.266	4.350
WRBM-BP	3.134	3.958

接下来,我们测试了模型的图像重构能力,记录的是数据的重构误差.

从表 4 中我们可以看出:相对于 Dropout RBM 和 RBM,WRBM 取得了最好的图像重构效果.Dropout RBM 和 WRBM 的重构图像如图 3 所示.

Table 4 Reconstruction errors of RBM and WRBM

表 4 RBM 和 WRBM 的模型重构误差

	MNIST-Basic	MNIST	Rectangles
DBM	153 577	975 660	307
WDBM	144 311	949 322	324
DBN	237 271	1 571 184	1 324
WDBN	201 227	1 513 921	1 109
Dropout DBN	261 735	1 580 317	1 511
Dropout DBM	170 205	979 243	692

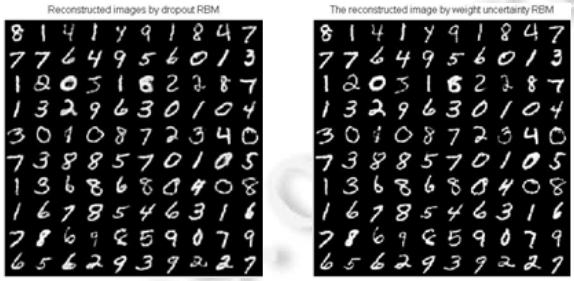


Fig.3 Reconstructed images of Dropout RBM and weight uncertainty RBM

图 3 Dropout RBM 和 weight uncertainty RBM 的重构图像对比

在多层模型中,使用的网络结构为 784-1000-1000-10,为了方便比较,我们使用 BP 算法进行权值的微调,循环次数为 100,MNIST-Basic 的训练样本数为 10 000,测试样本数为 50 000,MNIST 数据集的训练样本数为 60 000,测试样本数为 10 000.

从表 5 中我们可以看出:相对于 DBM,WDBM 在 3 个数据集上的分类能力均有所提升.同时,对于 DBN 模型,WDBN 达到了与 Dropout DBN 相近的分类能力.最后,我们测试 WSDBM 在 MNIST 数据集的分类能力和图像重建能力.重建图像如图 4 所示.

Table 5 Number of misclassifications of DBN and DBM

表 5 DBN 和 DBM 分类精度对比

	MNIST-Basic (%)	MNIST (%)	Rectangles (%)
DBM	2.23	0.156	2.618
WDBM	2.03	0.153	2.278
DBN	2.56	0.175	2.556
WDBN	2.50	0.168	0.734
Dropout DBN	2.51	0.165	1.556
Dropout DBM	2.09	0.151	2.309

Fig.4 Reconstructed image of WSDBM

图 4 WSDBM 的重构图像

在最后的实验中,我们希望在第 1 个 RBM 中既可以提取像素点之间的相关性信息,而且可以进行精确的推理.因此,我们引入 ssRBM 作为第 1 层,构建基于权值随机变量的深度模型 WSDBM.见表 6.

Table 6 Classification accuracies of the algorithms

表 6 分类精度对比

	MNIST-Basic (%)	MNIST (%)	Rectangles (%)
DBM	2.23	0.157	2.618
WDBM	2.032	0.153	2.278
DBN	2.566	0.175	2.556
WDBN	2.502	0.202	0.734
Dropout DBN	2.514	0.165	1.556
Dropout DBM	2.09	0.151	2.309
WSDBM	1.964	0.153	1.514

分类精度的对比图如图 5 所示.

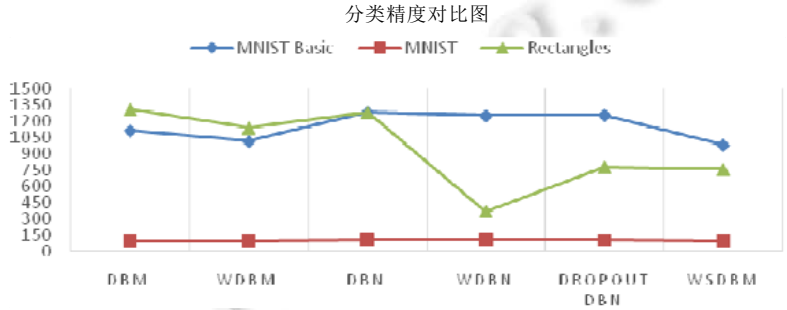


Fig.5 Figure of accuracies

图 5 分类精度对比图

该模型还可以结合卷积操作,被用于处理实数值的图像.我们利用卷积操作测试了模型的特征提取能力,在能量函数中加入卷积操作,将 RBM 中输入数据与权值的乘积转换为卷积 RBM 中输入图像与权值的卷积.卷积层的单元数为 40,卷积核尺寸为 5×5.以 MNIST 数据集为例,融合稀疏编码,我们得到如图 6 所示的特征图(在训练时,我们没有对数据进行二值化处理,而是使用实数值进行表示).

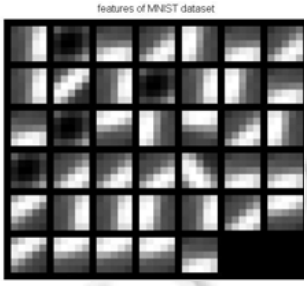


Fig.6 Features of real-valued data

图 6 实值数据的特征图像

6 结 论

本文中,为了缓解神经网络过拟合的问题,提高 RBM 模型的分类能力和图像重构能力,我们引入了 WRBM.在我们的实验中,与 Dropout 方法相比,权值随机变量是有效的.基于实验的表现,我们对模型进一步改进,引入 ssRBM 模型,对能量函数进行略微调整,可以成功地建模实值图像和二值图像.融合卷积和稀疏编码,我们可以

成功地提取图像的边缘特征.在接下来的工作中,我们将重点研究实数值图像的分类和重构问题,目前的研究工作也为我们提供了很好的理论基础.

References:

[1] Erhan D, Bengio Y, Courville A, Manzagol PA, Vincent P, Bengio S. Why does unsupervised pre-training help deep learning. Journal of Machine Learning Research, 2010,11(3):625−660.

[2] Hinton GE. Training products of experts by minimizing contrastive divergence. Neural Computation, 2002,14(8):1771−1800. [doi: 10.1162/089976602760128018]

[3] Roux N, Bengio Y. Representational power of restricted Boltzmann machines and deep belief networks. Neural Computation, 2008, 20(6):1631−1649. [doi: 10.1162/neco.2008.04-07-510]

[4] Liu JW, Liu Y, Luo XL. Research and development on Boltzmann machine. Journal of Computer Research and Development, 2014,51(1):1−16 (in Chinese with English abstract). [doi: 10.7544/issn1000-1239.2014.20121044]

[5] Hinton GE, Osindero S, Th Y. A fast learning algorithm for deep belief nets. Neural Computation, 2006,18(7):1527−1554. [doi: 10.1162/neco.2006.18.7.1527]

[6] Hinton GE, Salakhutdinov R. Reducing the dimensionality of data with neural networks. Science, 2006,313(5786):504−507. [doi: 10.1126/science.1127647]

[7] Lee H, Pham PT, Yan L, Ng A. Unsupervised feature learning for audio classification using convolutional deep belief networks. In: Bengio Y, Schuurmans D, Lafferty JD, Williams CKI, Culotta A, eds. Proc. of the Advances in Neural Information Processing Systems. 2009. 1096−1104. https://papers.nips.cc/book/advances-in-neural-information-processing-systems-22-2009

[8] Norouzi M, Ranjbar M, Mori G. Stacks of convolutional restricted Boltzmann machines for shift-invariant feature learning. In: Proc. of the Computer Vision and Pattern Recognition. 2009. 2735−2742. [doi: 10.1109/CVPR.2009.5206577]

[9] Salakhutdinov R, Larochelle H. Efficient learning of deep Boltzmann machines. Journal of Machine Learning Research, 2010,9(8): 693−700.

[10] Salakhutdinov R, Hinton GE. An efficient learning procedure for deep Boltzmann machines. Neural Computation, 2012,24(8): 1967−2006. [doi: 10.1162/NECO_a_00311]

[11] Boulanger-Lewandowski N, Bengio Y, Vincent P. Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription. Chemistry a European Journal, 2012,18(13):3981−3991.

[12] Hu Z, Fu K, Zhang CS. Audio classical composer identification by deep neural network. Journal of Computer Research and Development, 2014,51(9):1945−1954 (in Chinese with English abstract). [doi: 10.7544/issn1000-1239.2014.20140189]

[13] Zhang J, Ding SF, Zhang N, Shi ZZ. Incremental extreme learning machine based on deep feature embedded. Int'l Journal of Machine Learning and Cybernetics, 2016,7(1):111−120. [doi: 10.1007/s13042-015-0419-5]

[14] Zhang N, Ding SF, Shi ZZ. Denoising Laplacian multi-layer extreme learning machine. Neurocomputing, 2016,171(C):1066−1074. [doi: 10.1016/j.neucom.2015.07.058]

[15] Ding SF, Zhang N, Xu XZ, Guo LL, Zhang J. Deep extreme learning machine and its application in EEG classification. In: Proc. of the Mathematical Problems in Engineering. 2015. 1−11. [doi: 10.1155/2015/129021]

[16] Srivastava N, Hinton GE, Krizhevsky A. Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 2014,15(1):1929−1958.

[17] Blundell C, Cornebise J, Kavukcuoglu K. Weight uncertainty in neural networks. In: Bach F, Blei D, eds. Proc. of the 32nd Int'l Conf. on Machine Learning. Lille, 2015. http://proceedings.mlr.press/v37/

[18] Hinton GE. A practical guide to training restricted Boltzmann machines. Momentum, 2010,9(1):926. [doi: 10.1007/978-3-642-35289-8_32]

[19] Krizhevsky A, Hinton GE. Learning multiple layers of features from tiny images. Technical Report, University of Toronto, 2009.

[20] Salakhutdinov RR. Learning in Markov random fields using tempered transitions. In: Bengio Y, Schuurmans D, Lafferty JD, *et al*., eds. Proc. of the Advances in Neural Information Processing Systems. Curran Associates Inc., 2009. 1598−1606.

[21] Desjardins G, Courville AC, Bengio Y, Vincent P, Delalleau O. Tempered Markov chain Monte Carlo for training of restricted Boltzmann machines. In: Lawrence N, Whye TY, Titterington M, eds. Proc. of the Int'l Conf. on Artificial Intelligence and Statistics, Vol.9. 2010. 145−152.

[22] Peterson C. A mean field theory learning algorithm for neural network. Complex Systems, 1987,1(3):995−1019.

[23] Tieleman T. Training restricted Boltzmann machines using approximations to the likelihood gradient. In: Proc. of the 25th Int'l Conf. on Machine Learning. ACM Press, 2008. 1064−1071. [doi: 10.1145/1390156.1390290]

[24] Tieleman T, Hinton GE. Using fast weights to improve persistent contrastive divergence. In: Proc. of the 26th Int'l Conf. on Machine Learning. ACM Press, 2009. 1033−1040. [doi: 10.1145/1553374.1553506]

[25] Bengio Y. Learning deep architectures for AI. Foundations & Trends in Machine Learning, 2009,2(1):1−127. [doi: 10.1561/2200000006]

[26] Lee T, Yoon S. Boosted categorical restricted Boltzmann machine for computational prediction of splice junctions. In: Bach F, Blei D, eds. Proc. of the Int'l Conf. on Machine Learning. 2015. 2483−2492.

[27] Ranzato M, Krizhevsky A, Hinton GE. Factored 3-way restricted Boltzmann machines for modeling natural images. Journal of Machine Learning Research, 2010,9:621−628.

[28] Courville AC, Bergstra J, Bengio Y. A spike and slab restricted Boltzmann machine. Journal of Machine Learning Research, 2011, 15(15):233−241.

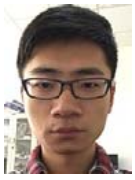
[29] Courville AC, Desjardins G, Bergstra J, Bengio Y. The spike-and-slab RBM and extensions to discrete and sparse data distributions. IEEE Trans. on Software Engineering, 2014,36(9):1874−1887. [doi: 10.1109/TPAMI.2013.238]

附中文参考文献:

[4] 刘建伟,刘媛,罗雄麟.玻尔兹曼机研究进展.计算机研究与发展,2014,51(1):1−16. [doi: 10.7544/issn1000-1239.2014.20121044]

[12] 胡振,傅昆,张长水.基于深度学习的作曲家分类问题.计算机研究与发展,2014,51(9):1945−1954. [doi: 10.7544/issn1000-1239.2014.20140189]

丁世飞(1963－),男,山东青岛人,博士,教授,博士生导师,CCF 杰出会员,主要研究领域为智能信息处理,人工智能,模式识别,机器学习,数据挖掘,粗糙集,软计算,大数据分析,云计算.



张健(1990－),男,学士,主要研究领域为机器学习,模式识别.

史忠植(1941－),男,博士,教授,博士生导师,CCF 会士,主要研究领域为智能科学,人工智能,机器学习.