補助関数法による制約付きボルツマンマシンの学習アルゴリズムの検討*

☆高宗典玄，石原達馬（東大院・情報理工研），亀岡弘和（東大院・情報理工，NTT CS 研）

1 はじめに

近年，Deep learning の有効性は音声認識をはじめ様々な分野で示されている．Deep learning において，大量データの下で学習をいかに効率的に行えるかは重要課題の一つである．Deep Neural Network (DNN) の一種である Deep Belief Network (DBN)[1, 2] は制約付きボルツマンマシン (Restricted Boltzmann Machine; RBM)[3] を多層に積み上げたものと見なせ，各層の RBM の教師なし学習を順次行っていくことにより初期学習を行う方式が DBN の学習において効果的であることが知られている．RBM の学習アルゴリズムとして，Contrastive Divergence (CD) 法 [1, 4] が非常に有名であるが，RBM の学習をいかに効率的に行えるかが DBN の全体の学習にかかる計算時間に直結する．

我々の研究室では，これまで様々な音響信号処理問題における最適化問題に対し，補助関数法と呼ぶ原理に基づく最適化アルゴリズムを導出し，その効果を示してきた（例えば [5]）．RBM の学習においても補助関数法に基づく学習則を導出することができれば，DBN の初期学習方式として高い効果を発揮する可能性がある．以上の動機のもと，本発表では RBM の学習問題に焦点を当て，補助関数法に基づく新しい学習則を提案する．

2 制約付きボルツマンマシン

2.1 RBM 学習における目的関数

RBM は，Fig. 1 で示されるように完全 2 部グラフの無向グラフの構造を持ち，観測される状態を可視層，背後にある状態を隠れ層と呼ぶ．このとき，可視層同士や隠れ層同士には結合が無いため “制約付き” と呼ばれる．このとき，可視層の状態数を I，可視層の状態を $\boldsymbol{v} = \{v_i\} \in \{0,1\}^I$，隠れ層の状態数を J，隠れ層の状態を $\boldsymbol{h} = \{h_j\} \in \{0,1\}^J$ とすると可視層の状態と隠れ層の状態を確率変数とした同時確率は

$$p(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta}) = \frac{\exp(-E(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta}))}{Z(\boldsymbol{\Theta})} \tag{1}$$

で定義される．ここで，

$$E(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta}) = -\sum_i b_i^{\mathrm{V}} v_i - \sum_j b_j^{\mathrm{H}} h_j - \sum_{i,j} W_{ij} v_i h_j, \tag{2}$$

$$Z(\boldsymbol{\Theta}) = \sum_{\boldsymbol{v}} \sum_{\boldsymbol{h}} \exp(-E(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta})) \tag{3}$$

であり，$\boldsymbol{\Theta} = \{b_i^{\mathrm{V}}, b_j^{\mathrm{H}}, W_{ij}\}$ は分布パラメータである．

RBM の学習問題とは観測される N 個の可視層のデータ $\boldsymbol{v}^{(1)}, \cdots, \boldsymbol{v}^{(N)}$ からこのパラメータ $\boldsymbol{\Theta}$ を推定することである．

このとき RBM の学習問題に対するよく用いられる目的関数として，次の周辺分布の対数尤度関数が挙げられる．

$$\begin{aligned} J(\boldsymbol{\Theta}) &= \frac{1}{N}\sum_n \log p\left(\boldsymbol{v}^{(n)}|\boldsymbol{\Theta}\right) \\ &= \frac{1}{N}\sum_n \log \sum_{\boldsymbol{h}} p\left(\boldsymbol{v}^{(n)},\boldsymbol{h}|\boldsymbol{\Theta}\right). \end{aligned} \tag{4}$$

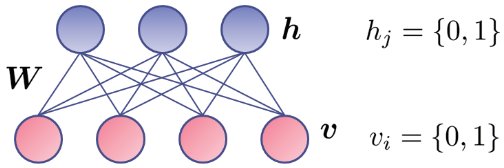

Fig. 1 RBM のグラフ表現

2.2 Contrastive Divergence 法 [1, 4]

最急降下法により，式 (4) で表される周辺分布の対数尤度関数 $J(\boldsymbol{\Theta})$ の最大化を考える．$J(\boldsymbol{\Theta})$ を Θ に関して微分すると，

$$\begin{aligned} &\frac{\partial J}{\partial \boldsymbol{\Theta}}(\boldsymbol{\Theta}) = \\ &\quad -\frac{1}{N}\sum_n \sum_{\boldsymbol{h}} p\left(\boldsymbol{h}|\boldsymbol{v}^{(n)},\boldsymbol{\Theta}\right) \frac{\partial E}{\partial \boldsymbol{\Theta}}\left(\boldsymbol{v}^{(n)},\boldsymbol{h}|\boldsymbol{\Theta}\right) \\ &\quad + \sum_{\boldsymbol{v}} \sum_{\boldsymbol{h}} p(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta}) \frac{\partial E}{\partial \boldsymbol{\Theta}}(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta}) \end{aligned} \tag{5}$$

となる．このため，最急降下法によるパラメータの更新は

$$\boldsymbol{\Theta} \leftarrow \boldsymbol{\Theta}^{\mathrm{old}} + \epsilon \Delta_{\mathrm{cd}} \boldsymbol{\Theta} \tag{6}$$

となる．ただし，ϵ は学習率，$\Delta_{\mathrm{cd}}\boldsymbol{\Theta} = \partial J/\partial \boldsymbol{\Theta}$ である．

ここで，$\boldsymbol{v}$ や $\boldsymbol{h}$ についての和に関しては厳密に計算しようとすると $O(2^I)$ や $O(2^J)$ の計算量となるため，現実的ではない．しかし，RBM の特徴として可視層同士，隠れ層同士の依存関係が無いため，

$$p(\boldsymbol{v}|\boldsymbol{h},\boldsymbol{\Theta}) = \prod_i p(v_i|\boldsymbol{h},\boldsymbol{\Theta}), \tag{7}$$

$$p(\boldsymbol{h}|\boldsymbol{v},\boldsymbol{\Theta}) = \prod_j p(h_j|\boldsymbol{v},\boldsymbol{\Theta}) \tag{8}$$

という関係がある．ただし，

$$p(v_i = 1|\boldsymbol{h},\boldsymbol{\Theta}) = \frac{1}{1+\exp\left(-b_i^{\mathrm{V}} - \sum_j W_{ij} h_j\right)}, \tag{9}$$

$$p(h_j = 1|\boldsymbol{v},\boldsymbol{\Theta}) = \frac{1}{1+\exp\left(-b_j^{\mathrm{H}} - \sum_i W_{ij} v_i\right)} \tag{10}$$

である．そこで，式 (5) の第 1 項に関しては周辺化を行うことにより，$\sum_{\boldsymbol{h}}$ を $\sum_j$ にすることが出来る．

しかし，式 (5) の第 2 項に関してはどうしても計算が困難である．そこで，式 (5) の第 2 項は同時確率による期待値計算であることに注目すると，次式のようなギブスサンプリングによる同時確率の近似を用いることで計算量を削減することが考えられる．

$$p(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta}) \approx \frac{1}{M}\sum_m \delta\left(\boldsymbol{v} - \boldsymbol{v}^{(m)}\right) p\left(\boldsymbol{h}|\boldsymbol{v}^{(m)},\boldsymbol{\Theta}\right) \tag{11}$$

ここで，式 (7)，(8) からギブスサンプリングは

$$h_j^{d-1} \sim p\left(h_j|\boldsymbol{v}^{d-1},\boldsymbol{\Theta}\right), \tag{12}$$

$$v_i^d \sim p\left(v_i|\boldsymbol{h}^{d-1},\boldsymbol{\Theta}\right) \tag{13}$$

* “Training Restricted Boltzmann Machine with Auxiliary Function Approach” by Takamune Norihiro, Ishihara Tatsuma (Univ. of Tokyo), Kameoka Hirokazu (Univ. of Tokyo, NTT CS Lab.).

と容易に行うことが可能である．
式 (6) による更新を式 (11) のギブスサンプリングによる近似で行う手法を CD 法という．

3 補助関数法による RBM の学習アルゴリズム

3.1 補助関数法 1

補助関数法による目的関数 $F(\boldsymbol{x})$ の最大化問題の最適化アルゴリズムは，補助変数 $\boldsymbol{y}$ を導入して，任意の $\boldsymbol{x}$，$\boldsymbol{y}$ で $F(\boldsymbol{x}) \geq F^{+}(\boldsymbol{x},\boldsymbol{y})$ となり，$F(\boldsymbol{x}) = \min_{\boldsymbol{y}} F^{+}(\boldsymbol{x},\boldsymbol{y})$ となるような下限関数 $F^{+}(\boldsymbol{x},\boldsymbol{y})$ を設計して，$F^{+}(\boldsymbol{x},\boldsymbol{y})$ を $\boldsymbol{x}$ についての最小化と $\boldsymbol{y}$ についての最小化を交互に行うことである．ここで重要なのは，$F^{+}(\boldsymbol{x},\boldsymbol{y})$ を $\boldsymbol{x}$ についての最小化が容易に行えるように $F^{+}(\boldsymbol{x},\boldsymbol{y})$ を設計できるかであるので，本研究では $\boldsymbol{x}$ の各変数の相互依存をなくすような $F^{+}(\boldsymbol{x},\boldsymbol{y})$ を設計することを目指す．そこで，式 (4) による目的関数を考えたとき，Jensen の不等式から

$$\begin{aligned} J(\boldsymbol{\Theta}) &= \frac{1}{N}\sum_{n}\log\sum_{\boldsymbol{h}} p\left(\boldsymbol{v}^{(n)},\boldsymbol{h}|\boldsymbol{\Theta}\right) \\ &\geq \frac{1}{N}\sum_{n}\sum_{\boldsymbol{h}}\lambda_{n,\boldsymbol{h}}\log p\left(\boldsymbol{v}^{(n)},\boldsymbol{h}|\boldsymbol{\Theta}\right) \\ &\quad -\sum_{\boldsymbol{h}}\lambda_{n,\boldsymbol{h}}\log\lambda_{n,\boldsymbol{h}}. \end{aligned} \tag{14}$$

となる．ここで，等号の成立は

$$\lambda_{n,\boldsymbol{h}} \leftarrow p\left(\boldsymbol{h}|\boldsymbol{v}^{(n)},\boldsymbol{\Theta}^{\mathrm{old}}\right) \tag{15}$$

である．式 (14) を整理すると，

$$\begin{aligned} J(\boldsymbol{\Theta}) \geq &-\frac{1}{N}\sum_{n}\sum_{\boldsymbol{h}}\lambda_{n,\boldsymbol{h}}E\left(\boldsymbol{v}^{(n)},\boldsymbol{h}|\boldsymbol{\Theta}\right) \\ &-\log Z(\boldsymbol{\Theta}) - \sum_{\boldsymbol{h}}\lambda_{n,\boldsymbol{h}}\log\lambda_{n,\boldsymbol{h}} \end{aligned} \tag{16}$$

となる．ここで，この式の第 2 項について考えると，負の対数関数は凸関数であるので，接線の方程式を用いて下から抑えることが出来る．

$$-\log Z(\boldsymbol{\Theta}) \geq -\frac{Z(\boldsymbol{\Theta})}{\zeta} - \log\zeta + 1. \tag{17}$$

ここで，等号の成立は接点となるので，

$$\zeta \leftarrow Z\left(\boldsymbol{\Theta}^{\mathrm{old}}\right) \tag{18}$$

となる．ここで，式 (2) から $E(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta})$ はパラメータ $\boldsymbol{\Theta}$ に対し線形であるので

$$E(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta}) = -\sum_{k} a_k(\boldsymbol{v},\boldsymbol{h})\,\theta_k \tag{19}$$

とおく．ここで注意すべきことは $a_k \in \{0,1\}$ である．このとき，$-\exp\left(\sum_k a_k(\boldsymbol{v},\boldsymbol{h})\,\theta_k\right)$ に対して，複素 NMF[5] で用いられている Jensen の不等式を用いた補助関数の設計法を用いると，

$$\begin{aligned} &-\exp\left(\sum_{k} a_k(\boldsymbol{v},\boldsymbol{h})\,\theta_k\right) \\ &\geq -\sum_{k}\beta_k(\boldsymbol{v},\boldsymbol{h}) \\ &\times\exp\left(\frac{a_k(\boldsymbol{v},\boldsymbol{h})\,\theta_k - \alpha_k(\boldsymbol{v},\boldsymbol{h})}{\beta_k(\boldsymbol{v},\boldsymbol{h})}\right) \end{aligned} \tag{20}$$

となり，等号の成立は

$$\begin{aligned} &\forall\beta_k(\boldsymbol{v},\boldsymbol{h}) \in [0,1], \\ &\sum_{k}\beta_k(\boldsymbol{v},\boldsymbol{h}) = 1, \end{aligned} \tag{21}$$

$$\begin{aligned} \alpha_k(\boldsymbol{v},\boldsymbol{h}) \leftarrow &a_k(\boldsymbol{v},\boldsymbol{h})\,\theta_k^{\mathrm{old}} \\ &-\beta_k(\boldsymbol{v},\boldsymbol{h})\sum_{l} a_l(\boldsymbol{v},\boldsymbol{h})\,\theta_l^{\mathrm{old}} \end{aligned} \tag{22}$$

となる．ここで，$\beta_k(\boldsymbol{v},\boldsymbol{h})$ は任意に設計できるので，$\boldsymbol{v}$，$\boldsymbol{h}$ に依存しない定数 β_k とし，式 (20) に補助変数の更新式 (22) を代入し，式 (17) を式 (3), (19) 用いて整理すると，

$$\begin{aligned} &-\log Z(\boldsymbol{\Theta}) \\ &\geq -\sum_{\boldsymbol{v}}\sum_{\boldsymbol{h}}\frac{\exp\left(\sum_l a_l(\boldsymbol{v},\boldsymbol{h})\,\theta_l^{\mathrm{old}}\right)}{\zeta} \\ &\quad\times\sum_{k}\beta_k\exp\left(\frac{a_k(\boldsymbol{v},\boldsymbol{h})\left(\theta_k-\theta_k^{\mathrm{old}}\right)}{\beta_k}\right) \\ &\quad-\log\zeta + 1 \end{aligned} \tag{23}$$

となる．ここで，式 (1)，(18) から式 (23) は

$$\begin{aligned} &-\log Z(\boldsymbol{\Theta}) \\ &\geq -\sum_{k}\beta_k\sum_{\boldsymbol{v}}\sum_{\boldsymbol{h}} p\left(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta}^{\mathrm{old}}\right) \\ &\quad\times\exp\left(\frac{a_k(\boldsymbol{v},\boldsymbol{h})\left(\theta_k-\theta_k^{\mathrm{old}}\right)}{\beta_k}\right) \\ &\quad-\log\zeta + 1 \end{aligned} \tag{24}$$

となる．式 (15)，(16)，(18)，(24) より，$\bar{\boldsymbol{\Theta}} = \{\boldsymbol{\Theta}^{\mathrm{old}},\beta_k\}$ とおいたとき，$J(\boldsymbol{\Theta})$ の下限関数 $J^{+}\left(\boldsymbol{\Theta},\bar{\boldsymbol{\Theta}}\right)$ は

$$\begin{aligned} &J^{+}\left(\boldsymbol{\Theta},\bar{\boldsymbol{\Theta}}\right) \\ &= \frac{1}{N}\sum_{n}\sum_{\boldsymbol{h}} p\left(\boldsymbol{h}|\boldsymbol{v}^{(n)},\boldsymbol{\Theta}^{\mathrm{old}}\right) \\ &\quad\times\sum_{k} a_k\left(\boldsymbol{v}^{(n)},\boldsymbol{h}\right)\theta_k \\ &\quad-\sum_{k}\beta_k\sum_{\boldsymbol{v}}\sum_{\boldsymbol{h}} p\left(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta}^{\mathrm{old}}\right) \\ &\quad\times\exp\left(\frac{a_k(\boldsymbol{v},\boldsymbol{h})\left(\theta_k-\theta_k^{\mathrm{old}}\right)}{\beta_k}\right) \\ &\quad+C\left(\bar{\boldsymbol{\Theta}}\right) \end{aligned} \tag{25}$$

と定義できる．ただし $C\left(\bar{\boldsymbol{\Theta}}\right)$ は $\boldsymbol{\Theta}$ に対して定数の項である．

ここで，式 (25) を θ_k について微分すると，

$$\begin{aligned} &\frac{\partial J^{+}}{\partial\theta_k}\left(\boldsymbol{\Theta},\bar{\boldsymbol{\Theta}}\right) \\ &= \frac{1}{N}\sum_{n}\sum_{\boldsymbol{h}} p\left(\boldsymbol{h}|\boldsymbol{v}^{(n)},\boldsymbol{\Theta}^{\mathrm{old}}\right) a_k\left(\boldsymbol{v}^{(n)},\boldsymbol{h}\right) \\ &\quad-\sum_{\boldsymbol{v}}\sum_{\boldsymbol{h}} p\left(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta}^{\mathrm{old}}\right) a_k(\boldsymbol{v},\boldsymbol{h}) \\ &\quad\times\exp\left(\frac{a_k(\boldsymbol{v},\boldsymbol{h})\left(\theta_k-\theta_k^{\mathrm{old}}\right)}{\beta_k}\right) \end{aligned} \tag{26}$$

となる．ここで，$a_k(\boldsymbol{v},\boldsymbol{h}) \in \{0,1\}$ である事実を用

いると，式 (26) の第二項は

$$\begin{aligned}&\sum_{\boldsymbol{v}}\sum_{\boldsymbol{h}} p\left(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta}^{\mathrm{old}}\right) a_k\left(\boldsymbol{v},\boldsymbol{h}\right)\\ &\quad\times\exp\left(\frac{a_k\left(\boldsymbol{v},\boldsymbol{h}\right)\left(\theta_k-\theta_k^{\mathrm{old}}\right)}{\beta_k}\right)\\ &=\exp\left(\frac{\theta_k-\theta_k^{\mathrm{old}}}{\beta_k}\right)\\ &\quad\times\sum_{\boldsymbol{v}}\sum_{\boldsymbol{h}} p\left(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta}^{\mathrm{old}}\right) a_k\left(\boldsymbol{v},\boldsymbol{h}\right)\end{aligned} \tag{27}$$

となる．よって，$\partial J^{+}/\partial\theta_k=0$ を解析的に求めることができ，$\Delta_{\mathrm{af}}\theta_k$ を

$$\begin{aligned}&\Delta_{\mathrm{af}}\theta_k\\ &=\log\frac{1}{N}\sum_n\sum_{\boldsymbol{h}} p\left(\boldsymbol{h}|\boldsymbol{v}^{(n)},\boldsymbol{\Theta}^{\mathrm{old}}\right) a_k\left(\boldsymbol{v}^{(n)},\boldsymbol{h}\right)\\ &-\log\sum_{\boldsymbol{v}}\sum_{\boldsymbol{h}} p\left(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta}^{\mathrm{old}}\right) a_k\left(\boldsymbol{v},\boldsymbol{h}\right)\end{aligned} \tag{28}$$

と定義すると，θ_k の更新式は，

$$\theta_k\leftarrow\theta_k^{\mathrm{old}}+\beta_k\Delta_{\mathrm{af}}\theta_k \tag{29}$$

となる．ここで，式 (28) の右辺の第 2 項は 2.2 節で言及したように厳密に計算することは困難である．そこで，CD 法と同様にギブスサンプリングによる $\boldsymbol{v}$ の周辺確率の近似を行うことで，式 (29) の計算が可能となる．

さて，式 (6)，(29) を比較すると，式 (29) の更新式はあたかも学習率が β_k であるかのような形となっている．ところが，式 (21) よりパラメータ数が多くなると β_k の平均的な値は小さくなるため，収束が遅くなることが予想される．そこで，収束を速くするために以下の近似を考える．

$$\beta_k'\leftarrow\beta_k^{\gamma}. \tag{30}$$

このとき，$\gamma\in[0,1]$ ならば，$\beta_k\in[0,1]$ を満たすので β_k' は β_k よりも大きくなる．そのため，式 (29) による更新が速くなることが期待できる．ただし，β_k' は式 (21) を満たさないため，補助関数法における収束性は保証されないことには注意されたい．

以上より補助関数法による 1 つ目の学習アルゴリズムは，次の 1)〜3) を反復することである．1) 補助変数 $\bar{\boldsymbol{\Theta}}$ を求める．2) ギブスサンプリングで $p\left(\boldsymbol{v},\boldsymbol{h}|\boldsymbol{\Theta}^{\mathrm{old}}\right)$ の近似値を求める．3) 式 (29) の更新式でパラメータを更新．

3.2 補助関数法 2

次に，補助関数法を用いた学習アルゴリズムとして，今度は目的関数が他と異なるものを導出する．ここで用いる目的関数は可視層に観測データが来たときに，ギブスサンプリングを 1 回行ったときに元の観測データが再現される確率の対数を尤度としたもので，

$$J_r\left(\boldsymbol{\Theta}\right)=\frac{1}{N}\sum_n\log\sum_{\boldsymbol{h}} p\left(\boldsymbol{v}^{(n)}|\boldsymbol{h},\boldsymbol{\Theta}\right)p\left(\boldsymbol{h}|\boldsymbol{v}^{(n)},\boldsymbol{\Theta}\right) \tag{31}$$

と表される．この式を直接最大化するのは困難であるので，この式に対して，

$$\boldsymbol{h}^{(n)}\sim p\left(\boldsymbol{h}|\boldsymbol{v}^{(n)},\boldsymbol{\Theta}\right) \tag{32}$$

でサンプリングした値を元に

$$\begin{aligned}&J_r\left(\boldsymbol{\Theta}\right)\\ &\approx\frac{1}{N}\sum_n\log p\left(\boldsymbol{v}^{(n)}|\boldsymbol{h}^{(n)},\boldsymbol{\Theta}\right)p\left(\boldsymbol{h}^{(n)}|\boldsymbol{v}^{(n)},\boldsymbol{\Theta}\right)\end{aligned} \tag{33}$$

と近似することを考える．この右辺を $\tilde{J}_r$ とおくと，式 (7)，(8) より，

$$\begin{aligned}\tilde{J}_r=\frac{1}{N}\sum_n\Bigg\{&\sum_i v_i^{(n)}\log q_{ni}^{\mathrm{V}}\\ &+\sum_i\left(1-v_i^{(n)}\right)\log\left(1-q_{ni}^{\mathrm{V}}\right)\\ &+\sum_j h_j^{(n)}\log q_{nj}^{\mathrm{H}}\\ &+\sum_j\left(1-h_j^{(n)}\right)\log\left(1-q_{nj}^{\mathrm{H}}\right)\Bigg\}\end{aligned} \tag{34}$$

となる．ただし，$q_{ni}^{\mathrm{V}}=p\left(v_i^{(n)}=1|\boldsymbol{h}^{(n)},\boldsymbol{\Theta}\right)$，$q_{nj}^{\mathrm{H}}=p\left(h_j^{(n)}=1|\boldsymbol{v}^{(n)},\boldsymbol{\Theta}\right)$ である．よって，式 (9)，(10) より式 (34) を整理すると，

$$\begin{aligned}\tilde{J}_r=-\frac{1}{N}\sum_n\Bigg\{&\sum_i v_i^{(n)}\log\left(1+\exp\left(-f_{ni}^{\mathrm{V}}\right)\right)\\ &+\sum_i\left(1-v_i^{(n)}\right)\log\left(1+\exp f_{ni}^{\mathrm{V}}\right)\\ &+\sum_j h_j^{(n)}\log\left(1+\exp\left(-f_{nj}^{\mathrm{H}}\right)\right)\\ &+\sum_j\left(1-h_j^{(n)}\right)\log\left(1+\exp f_{nj}^{\mathrm{H}}\right)\Bigg\}\end{aligned} \tag{35}$$

となる．ただし，$f_{ni}^{\mathrm{V}}=b_i^{\mathrm{V}}+\sum_j W_{ij}h_j^{(n)}$，$f_{nj}^{\mathrm{H}}=b_j^{\mathrm{H}}+\sum_i W_{ij}v_i^{(n)}$ である．ここで，式 (35) のそれぞれの項は負の対数関数であり，その引数の項をみると，式 (17)〜(24) と同様の論理で下限関数が設計できることが分かる．

よって，下限関数 $\tilde{J}_r^{+}\left(\boldsymbol{\Theta},\bar{\boldsymbol{\Theta}}\right)$ は

$$\begin{aligned}&\tilde{J}_r^{+}\left(\boldsymbol{\Theta},\bar{\boldsymbol{\Theta}}\right)\\ &=-\frac{1}{N}\Bigg\{\sum_i v_i^{(n)}\left(1-\hat{q}_{ni}^{\mathrm{V}}\right)\xi_{ni}^{\mathrm{V}}\\ &\qquad+\sum_i\left(1-v_i^{(n)}\right)\hat{q}_{ni}^{\mathrm{V}}\eta_{ni}^{\mathrm{V}}\\ &\qquad+\sum_j h_j^{(n)}\left(1-\hat{q}_{nj}^{\mathrm{H}}\right)\xi_{nj}^{\mathrm{H}}\\ &\qquad+\sum_j\left(1-h_j^{(n)}\right)\hat{q}_{nj}^{\mathrm{H}}\eta_{nj}^{\mathrm{H}}\Bigg\}+C\left(\bar{\boldsymbol{\Theta}}\right)\end{aligned} \tag{36}$$

となる．ただし，$\hat{q}_{ni}^{\mathrm{V}}=p\left(v_i^{(n)}=1|\boldsymbol{h}^{(n)},\boldsymbol{\Theta}^{\mathrm{old}}\right)$，$\hat{q}_{nj}^{\mathrm{H}}=p\left(h_j^{(n)}=1|\boldsymbol{v}^{(n)},\boldsymbol{\Theta}^{\mathrm{old}}\right)$，$\xi_{ni}^{\mathrm{V}}=\beta_{i0}^{\mathrm{V}}e^{-\hat{b}_i^{\mathrm{V}}}+\sum_j\beta_{ij}^{\mathrm{V}}e^{-\hat{W}_{nij}^{\mathrm{V}}}$，$\eta_{ni}^{\mathrm{V}}=\beta_{i0}^{\mathrm{V}}e^{\hat{b}_i^{\mathrm{V}}}+\sum_j\beta_{ij}^{\mathrm{V}}e^{\hat{W}_{nij}^{\mathrm{V}}}$，$\xi_{nj}^{\mathrm{H}}=\beta_{0j}^{\mathrm{H}}e^{-\hat{b}_j^{\mathrm{H}}}+\sum_i\beta_{ij}^{\mathrm{H}}e^{-\hat{W}_{nij}^{\mathrm{H}}}$，$\eta_{nj}^{\mathrm{H}}=\beta_{0j}^{\mathrm{H}}e^{\hat{b}_j^{\mathrm{H}}}+\sum_i\beta_{ij}^{\mathrm{H}}e^{\hat{W}_{nij}^{\mathrm{H}}}$，$\hat{b}_i^{\mathrm{V}}=\frac{b_i^{\mathrm{V}}-b_i^{\mathrm{V,old}}}{\beta_{i0}^{\mathrm{V}}}$，$\hat{b}_j^{\mathrm{H}}=\frac{b_j^{\mathrm{H}}-b_j^{\mathrm{H,old}}}{\beta_{0j}^{\mathrm{H}}}$，$\hat{W}_{nij}^{\mathrm{V}}=\frac{W_{ij}-W_{ij}^{\mathrm{old}}}{\beta_{ij}^{\mathrm{V}}}h_j^{(n)}$，$\hat{W}_{nij}^{\mathrm{H}}=\frac{W_{ij}-W_{ij}^{\mathrm{old}}}{\beta_{ij}^{\mathrm{H}}}v_i^{(n)}$，であり，$\beta_{i0}^{\mathrm{V}}$，$\beta_{ij}^{\mathrm{V}}$，$\beta_{0j}^{\mathrm{H}}$，$\beta_{ij}^{\mathrm{H}}$ は

$$\begin{aligned}&\beta_{i0}^{\mathrm{V}},\beta_{ij}^{\mathrm{V}},\beta_{0j}^{\mathrm{H}},\beta_{ij}^{\mathrm{H}}\in[0,1],\\ &\beta_{i0}^{\mathrm{V}}+\sum_j\beta_{ij}^{\mathrm{V}}=1,\quad\forall i,\\ &\beta_{0j}^{\mathrm{H}}+\sum_i\beta_{ij}^{\mathrm{H}}=1,\quad\forall j,\end{aligned} \tag{37}$$

を満たす任意定数である．また，$\bar{\boldsymbol{\Theta}} = \{\boldsymbol{\Theta}^{\mathrm{old}}, \beta_{i0}^{\mathrm{V}}, \beta_{ij}^{\mathrm{V}}, \beta_{0j}^{\mathrm{H}}, \beta_{ij}^{\mathrm{H}}\}$ であり，$C\left(\bar{\boldsymbol{\Theta}}\right)$ は $\boldsymbol{\Theta}$ に対して定数の項である．

ここで，$\partial \tilde{J}_r^{+}/\partial \boldsymbol{\Theta} = 0$ を解くことについて考えると，b_i^{V}，b_j^{H} については解析的に解くことが出来るが，W_{ij} については解析的に解くことが出来ない．これは，$\partial^2 \tilde{J}_r^{+}/\partial W_{ij}^2 < 0$ であるので，Newton 法を用いて効率的に求めることが出来る．

また，β_{i0}^{V}，β_{ij}^{V}，β_{0j}^{H}，β_{ij}^{H} が 1 つ目の補助関数法を用いた学習アルゴリズムと同様に学習率を担っていると考えられるため，これらについても γ 乗にする近似を考える．

以上より補助関数法による 3 つ目の学習アルゴリズムは，次の 1)〜3) を反復することとなる．1) サンプリングにより $h_j^{(n)}$ を求める．2) 補助変数 $\bar{\boldsymbol{\Theta}}$ を求める．3)$\partial \tilde{J}_r^{+}/\partial \boldsymbol{\Theta} = 0$ から求まる更新式でパラメータを更新．

4 動作確認実験

2，3 章で説明した各学習アルゴリズムがどのような挙動を示すかについて，可視層の状態数 $I = 10$，隠れ層の状態数 $J = 8$ という非常に小さな系で実験を行った．このとき，可視層に入力するデータは乱数で 20 個生成し，生成したそれぞれに対し，80 %のノイズをかけたものを 100 個づつ用意した．つまり，入力するデータ数は $N = 2000$ となる．また，学習の反復回数 T は 500 回とし，各パラメータの初期値は $[-1, 1]$ の一様乱数から生成し，すべてのアルゴリズムで共通の初期値とした．

次に，β_k，β_{i0}^{V}，β_{ij}^{V}，β_{0j}^{H}，β_{ij}^{H} は一様，つまり，

$$\beta_k = \frac{1}{I + J + IJ}, \tag{38}$$

$$\beta_{i0}^{\mathrm{V}} = \beta_{ij}^{\mathrm{V}} = \frac{1}{1 + J}, \tag{39}$$

$$\beta_{0j}^{\mathrm{H}} = \beta_{ij}^{\mathrm{H}} = \frac{1}{1 + I} \tag{40}$$

とし，CD 法の学習率 ϵ や式 (30) に示す γ のスケジューリングを t 回目の反復のとき

$$\epsilon\left(t\right) = \epsilon_{\mathrm{init}} \left(\frac{\epsilon_{\mathrm{end}}}{\epsilon_{\mathrm{init}}}\right)^{\frac{t-1}{T-1}}, \tag{41}$$

$$\gamma\left(t\right) = \gamma_{\mathrm{init}} \left(\frac{\gamma_{\mathrm{end}}}{\gamma_{\mathrm{init}}}\right)^{\frac{t-1}{T-1}} \tag{42}$$

とした．このとき，$\epsilon_{\mathrm{init}} = 1$，$\epsilon_{\mathrm{end}} = 0.1$，$\gamma_{\mathrm{init}} = 0.1$，$\gamma_{\mathrm{init}} = 1$ とした．また，ギブスサンプリングの回数を 1 回，Newton 法を用いる場合はその反復数を 1 回とした．

MATLAB による実装により計算時間を計ったところ，CD 法と比べ，補助関数法 1 による学習時間はおおよそ同じくらいであり，補助関数法 2 による学習時間はおおよそ 3/4 倍となった．また，各学習アルゴリズムにより式 (4) に示す対数尤度がどのように遷移したかを Fig. 2 に示す．予想したように γ による学習率の近似をしなかったアルゴリズムは収束が遅く，それに対し，γ による学習率の近似をしたアルゴリズムは非常に速く収束するという挙動が観測された．また，γ の加速を行うことにより，CD 法よりも速くなる可能性を示す結果が得られた．

本来，補助関数法は設計パラメータが少ないという利点があるが，このときは γ という設計パラメータが生じてしまう．しかし，補助関数法の原理より，ギブスサンプリングの近似が十分ならば $\gamma = 1$ のときは安定して収束するので，$\gamma = 1$ に近づくようなスケジューリングをすれば良いことから，CD 法の学習率のスケジューリングより設計の指針がはっきりしていると考えられる．

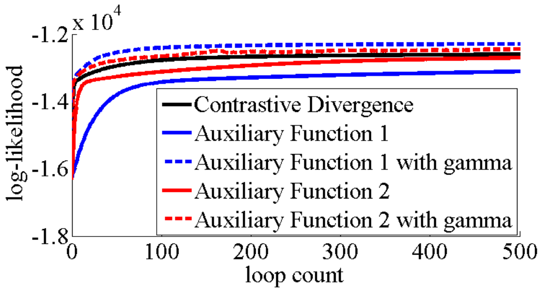

Fig. 2　各学習アルゴリズムにおける反復毎の対数尤度．黒い実線は CD 法を表し，青，赤の実線と破線はそれぞれ提案手法の γ による学習率の近似をしなかったものとしたものを表す．

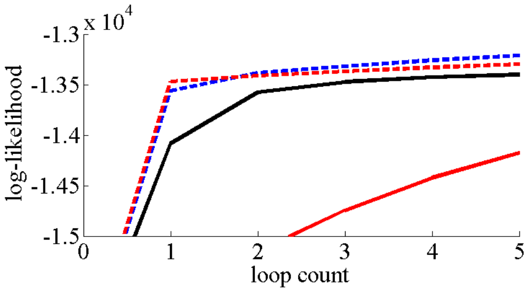

Fig. 3　Fig. 2 の一部拡大

さらに，興味深いことは，本来，目的関数が式 (4) とは異なるものから出発した学習アルゴリズムである 2 つ目のアルゴリズムが対数尤度に対して増加傾向にあることである．また，このアルゴリズムは γ による学習の加速を行うと，始めの数反復において他のアルゴリズムより対数尤度の上昇が見られた．

5 まとめ

本稿では，RBM の学習アルゴリズムとして，補助関数法を用いて新たに 2 つの学習アルゴリズムを導出した．そして，小規模の動作確認実験を通して，既存手法と同等以上の性能を見込めることが確認できた．今後の課題として，可視層と隠れ層の状態数が多くなったときや多層に重ねて Deep learning を行ったときにどのような挙動を示すかの観察が挙げられる．

参考文献

[1] Hinton, G. E., et al. *"A fast learning algorithm for deep belief nets,"* Neural Computation, 2006, 18.7, 1527–1554.

[2] Bengio, Y., et al. *"Greedy layer-wise training of deep networks,"* NIPS, 2007, 19: 153.

[3] Smolensky, P., *"Information processing in dynamical systems: Foundations of harmony theory, "* MIT Press, 1986, pp. 194–281.

[4] Hinton, G. E., *"Training Products of Experts by Minimizing Contrastive Divergence,"* Neural Computation, 2002, 14.8: 1771-1800.

[5] H. Kameoka, et al. *"Complex NMF: A new sparse representation for acoustic signals,"* In Proc of ICASSP, 2009, p. 3437-3440.