パターン認識と機械学習の学習

Learning of Pattern Recognition and Machine Learning

ベイズ理論に挫折しないための数学

光成 滋生 著

暗黒$\sum_{\infty}^{n}$信団
Since 2656

補　講

第2版で加筆された第5章の全文と第11章の本文訂正を配布いたします

第 5 章

「ニューラルネットワーク」の補足

PRML5 章では, 本来別の記号を割り当てるべきところに同じ記号を用いることがあり, 初読時には混乱しやすい. 慣れてしまえば読むのは難しくないが, ここでは出来るだけ区別してみる.

5.1 フィードフォワードネットワーク関数

3 章, 4 章でやったモデルは基底関数 ϕ_j とパラメータ w_j の線形和を非線形活性化関数に入れたものだった. ここではそれを拡張する. $x_1, \ldots, x_D$ を入力変数とし $x_0 = 1$ をバイアス項（定数項）に対応する変数, $w_{ji}^{(1)}$ をパラメータとして

$$\hat{a}_j = \sum_{i=0}^{D} w_{ji}^{(1)} x_i$$

とする. PRML では上記の $\hat{a}_j$ を a_j と書いているがすぐあとに出てくる a_k とは無関係である. ここでは異なることを強調するために $\hat{a}_j$ とする.

$\hat{a}_j$ を活性化関数 h で変換する.

$$z_j = h(\hat{a}_j).$$

h としてはロジスティックシグモイドなどのシグモイド関数が用いられる. これらの線形和をとって出力ユニット活性を求める. $z_0 = 1$ をバイアス項に対応する変数として

$$a_k = \sum_{j=0}^{M} w_{kj}^{(2)} z_j.$$

この出力ユニット活性を活性化関数を通してネットワークの出力 y_k とする. 2 クラス分類問題ならロジスティックシグモイド関数を使う.

$$y_k = y_k(x, w) = \sigma(a_k).$$

ここで w は $\{w_{ji}^{(1)}, w_{kj}^{(2)}\}$ をまとめたベクトルである. これらの式を組み合わせると

$$y_k = \sigma\left(\sum_{j=0}^{M} w_{kj}^{(2)} h\left(\sum_{i=0}^{D} w_{ji}^{(1)} x_i\right)\right).$$

5.2 ネットワーク訓練

回帰問題を考える. 入力ベクトル x と K 次元の目標変数 t があり, x, w における t の条件付き確率が精度 βI のガウス分布とする. N 個の同時独立分布 $x = \{x_1, \ldots, x_N\}$ と対応する目標値 $t = \{t_1, \ldots, t_N\}$ を用意し, 出力ユニットの活性化関数を恒等写像として $y_n = y(x_n, w)$ とする.

$$p(t|x, w) = \prod_{n=1}^{N} \mathcal{N}(t_n|y(x_n, w), \beta^{-1} I).$$

対数をとると

$$\begin{aligned}\log p(t|x, w) &= -\sum_n \left(\frac{1}{2}(t_n - y_n)^{\boldsymbol{T}} (\beta I)(t_n - y_n) \right) - \sum_n \frac{K}{2} \log(2\pi) - \sum_n \frac{1}{2} \log |\beta^{-1} I| \\ &= -\frac{\beta}{2} \sum_n ||t_n - y(x_n, w)||^2 - \frac{DN}{2} \log(2\pi) + \frac{NK}{2} \log \beta. \end{aligned} \tag{5.1}$$

そうするとこの関数の w についての最大化は最初の項の最小化, つまり

$$E(w) = \frac{1}{2} \sum_n ||y(x_n, w) - t_n||^2$$

の最小化と同等である. 最小値を与える $w(= w_{\mathrm{ML}})$ をなんらかの方法で求める. その値を式 (5.1) に代入して β で微分して 0 とおくと

$$-\frac{1}{2} \sum_n ||t_n - y(x_n, w_{\mathrm{ML}})||^2 + \frac{NK}{2} \frac{1}{\beta} = 0.$$

よって

$$\frac{1}{\beta_{\mathrm{ML}}} = \frac{1}{NK} \sum_n ||t_n - y(x_n, w_{\mathrm{ML}})||^2.$$

5.2.1 問題に応じた関数の選択

回帰問題を考える. a_k の活性化関数を恒等写像にとる. すると二乗和誤差関数の微分は

$$\frac{\partial E}{\partial a_k} = y_k - t_k.$$

クラス分類問題でも同様の関係式が成り立つことを確認しよう.

目標変数 t が $t = 1$ でクラス C_1, $t = 0$ でクラス C_2 を表す 2 クラス分類問題を考える. 活性化関数をロジスティックシグモイド関数に選ぶ.

$$y = \sigma(a) = \frac{1}{1 + \exp(-a)}.$$

この微分は $dy/da = \sigma(a)(1 - \sigma(a)) = y(1 - y)$ であった. $p(t = 1|x) = y(x, w)$, $p(t = 0|x) = 1 - y(x, w)$ なので

$$p(t|x, w) = y(x, w)^t \left(1 - y(x, w)\right)^{1-t}.$$

よって 4 章と同様にして交差エントロピー誤差関数は

$$E(w) = -\sum_{n=1}^{N} \left(t_n \log y_n + (1-t_n)\log(1-y_n)\right).$$

これを a_k で微分すると

$$\begin{aligned}\frac{\partial E}{\partial a_k} &= -\left(t_k \frac{y_k(1-y_k)}{y_k} + (1-t_k)\frac{-y_k(1-y_k)}{1-y_k}\right) \\ &= -(t_k - t_k y_k - y_k + y_k t_k) \\ &= y_k - t_k.\end{aligned}$$

K 個の 2 クラス分類問題を考える．それぞれの活性化関数がロジスティックシグモイド関数とする．

$$p(t|x,w) = \prod_{k=1}^{K} y_k(x,w)^{t_k}(1-y_k(x,w))^{1-t_k}.$$

$y_{nk} = y_k(x_n, w)$ とし n 番目の入力 x_n に対する目標変数を t_{nk} で表す．$t_{nk} \in \{0,1\}$ であり，$\sum_k t_{nk} = 1$ である．

$$E(w) = -\prod_{n,k}(t_{nk}\log y_{nk} + (1-t_{nk})\log(1-y_{nk})).$$

y_{nj} に対応する a を a_{nj} とすると

$$\frac{\partial E(w)}{\partial a_{nj}} = -(t_{nj}(1-y_{nj}) + (1-t_{nj})(-y_{nj})) = y_{nj} - t_{nj}.$$

最後に K クラス分類問題を考える．同様に n 番目の入力 x_n に対する目標変数 t_{nk} で表す．$y_k(x_n, w)$ を t_{nk} が 1 となる確率 $p(t_{nk} = 1|x_n)$ とみなす．

$$E(w) = -\log p(t|x,w) = -\sum_{n,k} t_{nk}\log y_k(x_n,w).$$

活性化関数はソフトマックス関数で

$$y_k(x,w) = \frac{\exp a_k(x,w)}{\sum_j \exp(a_j(x,w))}$$

のとき

$$\frac{\partial y_k}{\partial a_j} = y_k(\delta_{kj} - y_j).$$

よって $a_{nj} = a_j(x_n, w)$ とすると

$$\frac{\partial}{\partial a_{nj}}\log y_k(x_n,w) = \frac{1}{y_{nk}}\frac{\partial y_{nk}}{\partial a_{nj}} = \delta_{kj} - y_{nj}.$$

よって

$$\frac{\partial E}{\partial a_{nj}} = -\sum_k t_{nk}(\delta_{kj} - y_{nj}) = -t_{nj} + \left(\sum_k t_{nk}\right) y_{nj} = y_{nj} - t_{nj}.$$

5.3 局所二次近似

スカラー x についての関数 $E(x)$ の $x = a$ におけるテイラー展開を 2 次の項で打ち切った近似式は

$$E(x) \approx E(a) + E'(a)(x-a) + \frac{1}{2}E''(a)(x-a)^2$$

であった. これを n 変数関数 $E(x_1, \ldots, x_n)$ に拡張する. $x = (x_1, \ldots, x_n)^{\boldsymbol{T}}$, $a = (a_1, \ldots, a_n)^{\boldsymbol{T}}$ とおいて

$$\begin{aligned} E(x) &\approx E(a) + (x-a)^{\boldsymbol{T}} \left(\frac{\partial}{\partial x_i} E\right) + \frac{1}{2}(x-a)^{\boldsymbol{T}} \left(\frac{\partial^2 E}{\partial x_i \partial x_j}\right)(x-a) \\ &= E(a) + (x-a)^{\boldsymbol{T}}(\nabla E) + \frac{1}{2}(x-a)^{\boldsymbol{T}} H(E)(x-a) \end{aligned}$$

となる. $E(x)$ が $x = a$ の付近で極小ならば, そこでの勾配 ∇E は 0 なので

$$E(x) \approx E(a) + \frac{1}{2}(x-a)^{\boldsymbol{T}} H(E)(x-a).$$

$H(E)$ は対称行列なので 3.6 節の議論より対角化することで

$$E(x) \approx E(a) + \frac{1}{2}\sum_i \lambda_i y_i^2$$

の形にできる. そして $E(x)$ が $x = a$ の付近で極小となるのは $H(E) > 0$（正定値）であるときとわかる.

なお, $H(f) = \nabla^2 f = \nabla(\nabla f)$ という表記をすることがある. 微分作用素 ∇ を 2 回するので 2 乗の形をしている. ただ ∇f が縦ベクトルならもう一度 ∇ をするときは結果が行列になるように, 入力ベクトルの転置を取って作用するとみなす. n^2 次元の長いベクトルになるわけではない.

5.4 誤差関数微分の評価

与えられたネットワークに対して誤差関数の変化の割合を調べる. この節ではどの変数がどの変数に依存しているか気をつけて微分する必要がある. 誤差関数が訓練集合の各データに対する誤差の和で表せると仮定する :

$$E(w) = \sum_{n=1}^{N} E_n(w).$$

一般のフィードフォワードネットワークで

$$a_j = \sum_i w_{ji} z_i, \quad z_j = h(a_j) \tag{5.2}$$

とする. 入力 z_i が出力ユニット a_j に影響を与え, その a_j が非線形活性化関数 $h()$ を通して z_j に影響を与える. ある特定のパターン E_n の重み w_{ji} に関する微分を考える. 以下, 特定のパターン

を固定することで E_n 以外の添え字の n を省略する．式 (5.2) のように E_n は非線形活性化関数 h の変数 a_j を通して w_{ji} に依存している．

$$\frac{\partial E_n}{\partial w_{ji}} = \frac{\partial E_n}{\partial a_j}\frac{\partial a_j}{\partial w_{ji}}.$$

a_j は w_{ji} に関しては線形なので

$$\frac{\partial a_j}{\partial w_{ji}} = z_i.$$

誤差と呼ばれる記号 $\delta_j = \partial E_n/\partial a_j$ を導入すると

$$\frac{\partial E_n}{\partial w_{ji}} = \delta_j z_i$$

とかける．δ_j は $h()$ が正準連結関数の場合は 5.2 節での考察により

$$\delta_j = \frac{\partial E_n}{\partial a_j} = y_j - t_j$$

で計算できる．ユニット j につながっているユニット k を通して E_n への a_j の影響があると考えると

$$\frac{\partial E_n}{\partial a_j} = \sum_k \frac{\partial E_n}{\partial a_k}\frac{\partial a_k}{\partial a_j}. \tag{5.3}$$

ここで a_j と a_k は z を経由して関係していると考えているので $\partial a_k/\partial a_j = \delta_{jk}$（クロネッカーのデルタ）にはならないことに注意する．実際,

$$a_k = \sum_i w_{ki} h(a_i)$$

より

$$\frac{\partial a_k}{\partial a_j} = w_{kj} h'(a_j).$$

これを式 (5.3) に代入して

$$\delta_j = \frac{\partial E_n}{\partial a_j} = \sum_k \delta_k w_{kj} h'(a_j) = h'(a_j)\sum_k w_{kj}\delta_k.$$

5.5 外積による近似

$$E(w) = \frac{1}{2}\sum_{n=1}^{N}(y_n - t_n)^2$$

のときのヘッセ行列は

$$H = H(E) = \sum_n (\nabla y_n)(\nabla y_n)^{\boldsymbol{T}} + \sum_n (y_n - t_n)H(y_n).$$

一般には成立しないがもしよく訓練された状態で y_n が目標値 t_n に十分近ければ第 2 項を無視できる. その場合 $b_n = \nabla y_n = \nabla a_n$（活性化関数が恒等写像なので）とおくと

$$H \approx \sum_n b_n {b_n}^{\boldsymbol{T}}.$$

これを Levenberg-Marquardt 近似という.

$$E = \frac{1}{2} \iint (y(x,w) - t)^2 p(x,t)\, dxdt$$

のときのヘッセ行列を考えてみると

$$\frac{\partial E}{\partial w_i} = \iint (y-t) \frac{\partial y}{\partial w_i} p(x,t)\, dxdt.$$

$$\begin{aligned}
\frac{\partial^2 E}{\partial w_i \partial w_j} &= \iint \left(\frac{\partial y}{\partial w_j}\frac{\partial y}{\partial w_i} + (y-t)\frac{\partial^2 y}{\partial w_i \partial w_j} \right) p(x,t)\, dxdt. \\
&\quad (p(x,t) = p(t|x)p(x) \text{ より}) \\
&= \int \frac{\partial y}{\partial w_j}\frac{\partial y}{\partial w_i} \left(\int p(t|x)\, dt \right) p(x)\, dx \\
&+ \int \frac{\partial^2 y}{\partial w_i \partial w_j} \left(\int (y-t)p(t|x)\, dt \right) p(x)\, dx \\
&\quad (\text{第 1 項のカッコ内は 1.} \quad \text{第 2 項は } y(x) = \int tp(t|x)\, dt \text{ を使うと 0}) \\
&= \int \frac{\partial y}{\partial w_j}\frac{\partial y}{\partial w_i} p(x)\, dx.
\end{aligned}$$

ロジスティックシグモイドのときは

$$\begin{aligned}
\nabla E(w) &= \sum_n \frac{\partial E}{\partial a_n} \nabla a_n = -\sum_n \left(\frac{t_n y_n (1-y_n)}{y_n} - \frac{(1-t_n) y_n (1-y_n)}{1-y_n} \right) \nabla a_n \\
&= \sum_n (y_n - t_n) \nabla a_n.
\end{aligned}$$

よって $y_n \approx t_n$ なら

$$\begin{aligned}
\nabla^2 E(w) &= \sum_n \frac{\partial y_n}{\partial a_n} \nabla a_n {\nabla a_n}^{\boldsymbol{T}} + \sum_n (y_n - t_n) \nabla^2 a_n \\
&\approx \sum_n y_n (1-y_n) \nabla a_n {\nabla a_n}^{\boldsymbol{T}}.
\end{aligned}$$

5.6 ヘッセ行列の厳密な評価

この節は計算は難しくはないが, 記号がややこしいので書いてみる. 変数の関係式は $\hat{a}_j = \sum_i w^{(1)}_{ji} x_i$, $z_j = h(\hat{a}_j)$, $a_k = \sum_j w^{(2)}_{kj} z_j$, $y_k = a_k$ である. PRML の a_j と a_k は違う対象であることに注意する. ここでは a_j の代わりに $\hat{a}_j$ を使う.

添え字の i, i' は入力, j, j' は隠れユニット, k, k' は出力である. また

$$\delta_k = \frac{\partial E_n}{\partial a_k}, \quad M_{kk'} = \frac{\partial^2 E_n}{\partial a_k \partial a_{k'}}$$

という記号を導入する. E_n 以外の添え字 n を省略する.

5.6.1 両方の重みが第 2 層にある

$$\frac{\partial a_k}{\partial w_{kj}^{(2)}} = z_j, \quad \frac{\partial E_n}{\partial w_{kj}^{(2)}} = \frac{\partial a_k}{\partial w_{kj}^{(2)}} \frac{\partial E_n}{\partial a_k} = z_j \delta_k.$$

よって

$$\frac{\partial^2 E_n}{\partial w_{kj}^{(2)} \partial w_{k'j'}^{(2)}} = \frac{\partial a_k'}{\partial w_{k'j'}^{(2)}} \frac{\partial}{\partial a_{k'}} \left(\frac{\partial E_n}{\partial w_{kj}^{(2)}} \right) = z_{j'} z_j \frac{\partial \delta_k}{\partial a_{k'}} = z_j z_{j'} M_{kk'}.$$

5.6.2 両方の重みが第 1 層にある

$$\frac{\partial a_k}{\partial \hat{a}_j} = w_{kj}^{(2)} h'(\hat{a}_j), \quad \frac{\partial \hat{a}_j}{\partial w_{ji}^{(1)}} = x_i$$

より

$$\frac{\partial E_n}{\partial w_{ji}^{(1)}} = \frac{\partial \hat{a}_j}{\partial w_{ji}^{(1)}} \frac{\partial E_n}{\partial \hat{a}_j} = x_i \sum_k \frac{\partial a_k}{\partial \hat{a}_j} \frac{\partial E_n}{\partial a_k} = x_i \sum_k w_{kj}^{(2)} h'(\hat{a}_j) \delta_k.$$

$$\frac{\partial^2 E_n}{\partial w_{ji}^{(1)} \partial w_{j'i'}^{(1)}} = \frac{\partial \hat{a}_{j'}}{\partial w_{j'i'}^{(1)}} \frac{\partial}{\partial \hat{a}_{j'}} \left(\frac{\partial E_n}{\partial w_{ji}^{(1)}} \right) = x_i x_{i'} \underbrace{\frac{\partial}{\partial \hat{a}_{j'}} \left(h'(\hat{a}_j) \sum_k w_{kj}^{(2)} \delta_k \right)}_{=:A}.$$

$j = j'$ のとき

$$A = h''(\hat{a}_{j'}) \sum_k w_{kj}^{(2)} \delta_k + B, \quad B := h'(\hat{a}_j) \frac{\partial}{\partial \hat{a}_{j'}} \left(\sum_k w_{kj}^{(2)} \delta_k \right).$$

$j \neq j'$ のとき

$$B = h'(\hat{a}_j) \sum_{k'} \frac{\partial a_{k'}}{\partial \hat{a}_j} \frac{\partial}{\partial a_{k'}} \left(\sum_k w_{kj}^{(2)} \delta_k \right) = \sum_{k,k'} h'(\hat{a}_j) h'(\hat{a}_{j'}) w_{k'j'}^{(2)} w_{kj}^{(2)} M_{kk'}.$$

二つをまとめて

$$\frac{\partial^2 E_n}{\partial w_{ji}^{(1)} \partial w_{j'i'}^{(1)}} = x_i x_{i'} \left\{ h''(\hat{a}_{j'}) \delta_{jj'} \sum_k w_{kj}^{(2)} \delta_k + h'(\hat{a}_j) h'(\hat{a}_{j'}) \sum_{k,k'} w_{kj}^{(2)} w_{k'j'}^{(2)} M_{kk'} \right\}.$$

$\delta_{jj'}$ はクロネッカーのデルタ.

5.6.3 重みが別々の層に一つずつある

$$\frac{\partial E_n}{\partial w_{kj'}^{(2)}} = z_{j'} \delta_k, \quad \frac{\partial^2 E_n}{\partial w_{ji}^{(1)} \partial w_{kj'}^{(2)}} = \frac{\partial \hat{a}_j}{\partial w_{ji}^{(1)}} \underbrace{\frac{\partial}{\partial \hat{a}_j} (z_{j'} \delta_k)}_{=:A}, \quad \frac{\partial \hat{a}_j}{\partial w_{ji}^{(1)}} = x_i.$$

$j = j'$ のとき

$$A = h'(\hat{a}_{j'})\delta_k + B, \quad B := z_{j'}\frac{\partial \delta_k}{\partial \hat{a}_j}.$$

$j \neq j'$ のとき

$$B = z_{j'}\sum_{k'}\frac{\partial a_{k'}}{\partial \hat{a}_j}\frac{\partial \delta_k}{\partial a_{k'}} = z_{j'}\sum_{k'} w^{(2)}_{k'j}h'(\hat{a}_j)M_{kk'}.$$

よって

$$\frac{\partial^2 E_n}{\partial w^{(1)}_{ji}\partial w^{(2)}_{kj'}} = x_i h'(\hat{a}_j)\left\{\delta_{jj'}\delta_k + z_{j'}\sum_{k'} w^{(2)}_{k'j}M_{kk'}\right\}.$$

5.7 ヘッセ行列の積の高速な計算

応用面を考えると最終的に必要なものはヘッセ行列 H そのものではなくあるベクトル v と H の積であることが多い. H を計算せず直接 $v^{\boldsymbol{T}}H = v^{\boldsymbol{T}}\nabla\nabla$ を計算するために, 左半分だけを取り出して $\mathcal{R}\{\cdot\} = v^{\boldsymbol{T}}\nabla$ という記法を導入する. 5.3 節の終わりに書いたようにこの ∇ は入力が縦ベクトルなら転置を取ってから作用するとみなす. なお, v に依存するものをあたかも依存しないかのように $\mathcal{R}\{\cdot\}$ と書いてしまうのは筋がよいとは思わない.

簡単な例を見てみよう. 2 変数関数 $y = f(x_1, x_2)$ について

$$\mathcal{R}\{\cdot\} = (v_1, v_2)\nabla = (v_1, v_2)\begin{pmatrix}\frac{\partial}{\partial x_1}\\ \frac{\partial}{\partial x_2}\end{pmatrix}.$$

よって

$$\mathcal{R}\{x_1\} = (v_1, v_2)\begin{pmatrix}1\\0\end{pmatrix} = v_1,$$
$$\mathcal{R}\{x_2\} = (v_1, v_2)\begin{pmatrix}0\\1\end{pmatrix} = v_2,$$

これを, $\mathcal{R}\{\}$ は入力値の x_i をその添え字に対応する v_i に置き換える作用と考えることにする. $\mathcal{R}\{\}$ は x_i について明らかに線形, つまり

$$\mathcal{R}\{ax_1 + bx_2\} = av_1 + bv_2 = a\mathcal{R}\{x_1\} + b\mathcal{R}\{x_2\}.$$

前節と同じ 2 層ネットワークで考えてみる. $a_j = \sum_i w^{(1)}_{ji}x_i$, $z_j = h(a_j)$, $y_k = \sum_j w^{(2)}_{kj}z_j$ である. a_k の代わりに y_k を使うので $\hat{a}_j$ ではなく PRML と同じ a_j にする. PRML では w_{ji} の肩の添え字を省略しているが念のためここではつけておく.

$w^{(1)}_{ji}$ に対応する値を v_{ji} とすると $w^{(1)}_{ji}$ の線形和である a_j について

$$\mathcal{R}\{a_j\} = \sum_i x_i\mathcal{R}\left\{w^{(1)}_{ji}\right\} = \sum_i v_{ji}x_i.$$

$$\mathcal{R}\{z_j\} = v^{\boldsymbol{T}}\nabla\left(\sum_j w^{(1)}_{ji}h(a_j)\right) = v^{\boldsymbol{T}}\left(\frac{\partial h(a_j)}{\partial a_j}\nabla a_j\right) = h'(a_j)\mathcal{R}\{a_j\}.$$

$$\mathcal{R}\{y_k\} = v^{(2)\boldsymbol{T}}\left(\nabla\left(\sum_j w_{kj}^{(2)} z_j\right)\right) = v^{(2)\boldsymbol{T}}\left(\sum_j \left(\nabla w_{kj}^{(2)}\right) z_j + \sum_j w_{kj}^{(2)} \nabla z_j\right)$$
$$= \sum_j v_{kj}^{(2)} z_j + \sum_j w_{kj}^{(2)} \mathcal{R}\{z_j\}.$$

なんとなくルールが見えてきたであろう. $\mathcal{R}\{\cdot\}$ は $\mathcal{R}\{w\} = v$ という記号の置き換え以外は積や合成関数の微分のルールの形に従っている（もともと微分作用素を用いて定義しているので当然ではあるが）.

逆伝播の式 :

$$\delta_k^{(2)} = y_k - t_k,$$
$$\delta_j^{(1)} = h'(a_j) \sum_k w_{kj}^{(2)} \delta_k^{(2)}$$

で考えてみると

$$\mathcal{R}\left\{\delta_k^{(2)}\right\} = \mathcal{R}\{y_k\}.$$

$$\mathcal{R}\left\{\delta_j^{(1)}\right\} = h''(a_j)\mathcal{R}\{a_j\}\left(\sum_k w_{kj}^{(2)} \delta_k^{(2)}\right) + h'(a_j)\left(\sum_k v_{kj}^{(2)} \delta_k^{(2)} + \sum_k w_{kj}^{(2)} \mathcal{R}\left\{\delta_k^{(2)}\right\}\right).$$

誤差の微分の式:

$$\frac{\partial E}{\partial w_{kj}^{(2)}} = \delta_k^{(2)} z_j.$$
$$\frac{\partial E}{\partial w_{jk}^{(1)}} = \delta_j^{(1)} x_i.$$

より

$$\mathcal{R}\left\{\frac{\partial E}{\partial w_{kj}^{(2)}}\right\} = \mathcal{R}\left\{\delta_k^{(2)}\right\} z_j + \delta_k^{(2)} \mathcal{R}\{z_j\}.$$
$$\mathcal{R}\left\{\frac{\partial E}{\partial w_{jk}^{(1)}}\right\} = x_i \mathcal{R}\left\{\delta_j^{(1)}\right\}.$$

5.8 ソフト重み共有

ネットワークの, あるグループに属する重みを等しくすることで複雑さを減らす手法がある. しかし重みが等しいという制約は厳しい. ソフト重み共有はその制約を外し, 代わりに正則化項を追加することで, あるグループに属する重みが似た値をとれるようにする手法である. π_k を混合係数として確率密度関数は

$$p(w) = \prod_i p(w_i), \quad p(w_i) = \sum_{k=1}^{M} \pi_k \mathcal{N}(w_i|\mu_k, \sigma_k^2).$$

$p(w_i)$ が確率分布なので混合係数は $\sum_k \pi_k = 1,\ 0 \leq \pi_k \leq 1$ を満たす. 2 乗ノルムの正規化項は平均 0 のガウス事前分布の負の対数尤度関数とみなせた. ここでは複数個の重みに対応させるため混合ガウス分布を用いてみる.

$$\Omega(w) = -\log p(w) = -\sum_i \log \left(\sum_{k=1}^{M} \pi_k \mathcal{N}(w_i|\mu_k, \sigma_k^2) \right).$$

最小化したい目的関数は誤差関数と正則化項の和で

$$\tilde{E}(w) = E(w) + \Omega(w).$$

$p(j) = \pi_j$ とおいて負担率を導入する.

$$\gamma_j(w_i) = p(j|w_i) = \frac{p(j)p(w_i|j)}{p(w_i)} = \frac{\pi_j \mathcal{N}(w_i|\mu_j, \sigma_j^2)}{p(w_i)}.$$

正規分布の微分

$$\begin{aligned}
\frac{\partial}{\partial x}\mathcal{N}(x|\mu,\sigma) &= \mathcal{N}(x|\mu,\sigma)\left(-\frac{x-\mu}{\sigma^2}\right), \\
\frac{\partial}{\partial \mu}\mathcal{N}(x|\mu,\sigma) &= \mathcal{N}(x|\mu,\sigma)\left(\frac{x-\mu}{\sigma^2}\right) \\
\frac{\partial}{\partial \sigma}\mathcal{N}(x|\mu,\sigma) &= \mathcal{N}(x|\mu,\sigma)\left(-\frac{1}{\sigma} + \frac{(x-\mu)^2}{\sigma^3}\right)
\end{aligned}$$

を思い出しておく. $\log p(w_i)$ を w_i で微分すると

$$\begin{aligned}
\frac{\partial}{\partial w_i}\log p(w_i) &= \frac{1}{p(w_i)}\left(\sum_k \pi_k \frac{\partial}{\partial w_i}\mathcal{N}(w_i|\mu_k,\sigma_k^2)\right) \\
&= \frac{1}{p(w_i)}\left(\sum_k \pi_k \mathcal{N}(w_i|\mu_k,\sigma_k^2)\left(-\frac{w_i-\mu_k}{\sigma_k^2}\right)\right) \\
&= -\sum_k \frac{\pi_k \mathcal{N}(w_i|\mu_k,\sigma_k)}{p(w_i)}\frac{w_i-\mu_k}{\sigma_k^2} \\
&= -\sum_k \gamma_k(w_i)\frac{w_i-\mu_k}{\sigma_k^2}.
\end{aligned}$$

よって

$$\frac{\partial \tilde{E}}{\partial w_i} = \frac{\partial E}{\partial w_i} + \sum_k \gamma_k(w_i)\frac{w_i-\mu_k}{\sigma_k^2}.$$

同様に

$$\begin{aligned}
\frac{\partial}{\partial \mu_k}\log p(w) &= \sum_j \frac{\partial}{\partial \mu_k}\log p(w_j) = \sum_j \frac{\pi_k \mathcal{N}(w_j|\mu_k,\sigma_k^2)}{p(w_j)}\frac{w_j-\mu_k}{\sigma_k^2} \\
&= \sum_j \gamma_k(w_j)\frac{w_j-\mu_k}{\sigma_k^2}.
\end{aligned}$$

よって

$$\frac{\partial \tilde{E}}{\partial \mu_k} = \sum_j \gamma_k(w_j)\frac{\mu_k-w_j}{\sigma_k^2}.$$

$$\frac{\partial \tilde{E}}{\partial \sigma_k} = -\sum_j \frac{\partial}{\partial \sigma_k} \log p(w_j) = -\sum_j \frac{\pi_k \mathcal{N}(w_j|\mu_k, \sigma_k^2)}{p(w_j)} \left(-\frac{1}{\sigma_k} + \frac{(w_j - \mu_k)^2}{\sigma_k^3} \right)$$
$$= \sum_j \gamma_k(w_j) \left(\frac{1}{\sigma_k} - \frac{(w_j - \mu_k)^2}{\sigma_k^3} \right).$$

π_j に関する制約より補助変数 η_j を用いて

$$\pi_j = \frac{\exp(\eta_j)}{\sum_k \exp(\eta_k)}$$

と表すと 4.13 節式 (4.1) より

$$\frac{\partial \pi_k}{\partial \eta_j} = \pi_k(\delta_{kj} - \pi_j).$$

よって

$$\frac{\partial \tilde{E}}{\partial \eta_j} = -\sum_i \frac{\partial}{\partial \eta_j} \log p(w_i) = -\sum_i \frac{\partial}{\partial \eta_j} \log \left(\sum_k \pi_k \mathcal{N}(w_i|\mu_k, \sigma_k^2) \right)$$
$$= -\sum_{i,k} \frac{\mathcal{N}(w_i|\mu_k, \sigma_k^2)}{p(w_i)} \frac{\partial \pi_k}{\partial \eta_j}$$
$$= -\sum_{i,k} \frac{\mathcal{N}(w_i|\mu_k, \sigma_k^2)}{p(w_i)} \pi_k(\delta_{kj} - \pi_j)$$
$$= -\sum_i \left(\frac{\pi_j \mathcal{N}(w_i|\mu_j, \sigma_j^2)}{p(w_i)} - \frac{\pi_j \sum_k \pi_k \mathcal{N}(w_i|\mu_k, \sigma_k^2)}{p(w_i)} \right)$$
$$= -\sum_i (\gamma_j(w_i) - \pi_j) = \sum_i (\pi_j - \gamma_j(w_i)).$$

5.9 混合密度ネットワーク

$$p(t|x) = \sum_{k=1}^{K} \pi_k(x) \mathcal{N}(t|\mu_k(x), \sigma_k^2(x) I)$$

という分布のモデルを考える．このモデルのパラメータを, x を入力としてえられるニューラルネットワークの出力となるようにとることで推論する．前節と同様 $\sum_k \pi_k(x) = 1$, $0 \leq \pi_k(x) \leq 1$ という制約があるので変数 a_l^π を導入し

$$\pi_k(x) = \frac{\exp(a_k^\pi)}{\sum_l \exp(a_l^\pi)}$$

とする．分散は 0 以上という制約があるので変数 a_k^σ を導入し

$$\sigma_k(x) = \exp(a_k^\sigma)$$

とする．平均は特に制約がないので

$$\mu_{kj}(x) = a_{kj}^\mu$$

とする.

$$\mathcal{N}_{nk} = \mathcal{N}(t_n|\mu_k(x_n), \sigma_k^2(x_n)I)$$

とおくとデータが独立の場合, 誤差関数は

$$E(w) = -\sum_n \log\left(\sum_k \pi_k(x_n)\mathcal{N}_{nk}\right).$$

前節と同様 $p(k|x) = \pi_k(x)$ とおいて負担率を

$$\gamma_{nk}(t_n|x_n) = p(k|t_n, x_n) = \frac{p(k|x_n)p(t_n|k)}{p(t_n|x_n)} = \frac{\pi_k\mathcal{N}_{nk}}{\sum_l \pi_l\mathcal{N}_{nl}}$$

とする.

$$\frac{\partial \pi_j}{\partial a_k^\pi} = \pi_j(\delta_{kj} - \pi_k)$$

より

$$\frac{\partial E_n}{\partial a_k^\pi} = -\frac{\sum_j \pi_j(\delta_{kj} - \pi_k)\mathcal{N}_{nj}}{\sum_j \pi_j\mathcal{N}_{nj}} = -(\gamma_{nk} - \pi_k) = \pi_k - \gamma_{nk}.$$

$$\mathcal{N}(t|\mu, \sigma^2 I) = \frac{1}{(2\pi)^{L/2}}\frac{1}{\sigma^L}\exp\left(-\frac{1}{2\sigma^2}\sum_{l=1}^{L}(t_l - \mu_l)^2\right)$$

なので

$$\frac{\partial}{\partial \mu_l}\mathcal{N}(t|\mu, \sigma^2 I) = \mathcal{N}(t|\mu, \sigma^2 I)\left(-\frac{t_l - \mu_l}{\sigma^2}\right).$$

よって

$$\frac{\partial E_n}{\partial a_{kl}^\mu} = -\frac{\mathcal{N}_{nk}}{\sum_j \pi_j\mathcal{N}_{nj}}\frac{t_{nl} - \mu_{kl}}{\sigma_k^2} = \gamma_{nk}\left(\frac{\mu_{kl} - t_{nl}}{\sigma_k^2}\right).$$

同様に

$$\frac{\partial}{\partial \sigma}\mathcal{N}(t|\mu, \sigma^2 I) = \mathcal{N}(t|\mu, \sigma^2 I)\left(-\frac{L}{\sigma} + \frac{||t - \mu||^2}{\sigma^3}\right)$$

より

$$\frac{\partial}{\partial a_{kl}^\mu}\mathcal{N}_{nj} = \delta_{jk}\mathcal{N}_{nj}\left(\frac{t_{nl} - \mu_{kl}}{\sigma_k^2}\right).$$

よって

$$\frac{\partial \mathcal{N}_{nk}}{\partial a_k^\sigma} = \frac{\partial \sigma_k}{\partial a_k^\sigma}\frac{\partial \mathcal{N}_{nk}}{\partial \sigma_k} = \sigma_k\mathcal{N}_{nk}\left(-\frac{L}{\sigma_k} + \frac{||t_n - \mu_k||^2}{\sigma_k^3}\right) = \mathcal{N}_{nk}\left(-L + \frac{||t_n - \mu_k||^2}{\sigma_k^2}\right).$$

よって

$$\frac{\partial E_n}{\partial a_k^\sigma} = \gamma_{nk}\left(L - \frac{||t_n - \mu_k||^2}{\sigma_k^2}\right).$$

条件付き平均についての密度関数の分散は

$$\begin{aligned}
s^2(x) &= E\left[||t - E[t|x]||^2 \middle| x\right] \\
&= \sum_k \pi_k \int \left(||t||^2 - 2t^{\boldsymbol{T}} E[t|x] + ||E[t|x]||^2\right) \mathcal{N}(t|\mu_k, \sigma_k^2 I)\, dt \\
&= \sum_k \pi_k \left(\sigma_k^2 + ||\mu_k||^2 - 2{\mu_k}^{\boldsymbol{T}} E[t|x] + ||E[t|x]||^2\right) \\
&= \sum_k \pi_k(x) \left(\sigma_k(x)^2 + ||\mu_k - \sum_j \pi_j(x)\mu_j(x)||^2\right).
\end{aligned}$$

5.10　クラス分類のためのベイズニューラルネットワーク

ロジスティックシグモイド出力を一つ持つネットワークによる 2 クラス分類問題を考える．そのモデルの対数尤度関数は $t_n \in \{0, 1\}$, $y_n = y(x_n, w)$ として

$$\log p(\mathcal{D}|w) = \sum_n (t_n \log y_n + (1 - t_n) \log(1 - y_n)).$$

事前分布を

$$p(w|\alpha) = \mathcal{N}(w|0, \alpha^{-1} I) = \frac{1}{\sqrt{2\pi}^W |\alpha^{-1}|^{W/2}} \exp\left(-\frac{1}{2}\alpha {w^{\boldsymbol{T}}} w\right)$$

とする（W は w に含まれるパラメータの総数）．ノイズがないので β を含まない．

$$E(w) = \log p(\mathcal{D}|w) + \frac{\alpha}{2} {w^{\boldsymbol{T}}} w$$

の最小化で w_{MAP} を求め，$A = -\nabla^2 \log p(w|\mathcal{D})\big|_{w=w_{\mathrm{MAP}}}$ を何らかの方法で求める．ラプラス近似を使って事後分布をガウス近似すると

$$q(w|\mathcal{D}) = \mathcal{N}(w|w_{\mathrm{MAP}}, A^{-1}).$$

正規化項を求める 4.17 節式 (4.2) を使って

$$\begin{aligned}
\log p(\mathcal{D}|\alpha) &\approx \log\left(p(\mathcal{D}|w_{\mathrm{MAP}}) p(w_{\mathrm{MAP}}|\alpha) \sqrt{\frac{(2\pi)^W}{|A|}}\right) \\
&= \log p(\mathcal{D}|w_{\mathrm{MAP}}) - \frac{W}{2}\log(2\pi) + \frac{W}{2}\log\alpha - \frac{1}{2}\alpha {w_{\mathrm{MAP}}}^{\boldsymbol{T}} w_{\mathrm{MAP}} \\
&\quad + \frac{W}{2}\log(2\pi) - \frac{1}{2}\log|A| \\
&= -E(w_{\mathrm{MAP}}) - \frac{1}{2}\log|A| + \frac{W}{2}\log\alpha.
\end{aligned}$$

$$E(w_{\mathrm{MAP}}) = -\sum_n (t_n \log y_n + (1 - t_n)\log(1 - y_n)) + \frac{1}{2}\alpha {w_{\mathrm{MAP}}}^{\boldsymbol{T}} w_{\mathrm{MAP}}.$$

予測分布を考える．出力ユニットの活性化関数を線形近似する．

$$\begin{aligned}
a(x, w) &\approx a(x, w_{\mathrm{MAP}}) + \nabla a(x, w_{\mathrm{MAP}})^{\boldsymbol{T}} (w - w_{\mathrm{MAP}}) \\
&\qquad (a_{\mathrm{MAP}}(x) = a(x, w_{\mathrm{MAP}}), \quad b = \nabla a(x, w_{\mathrm{MAP}}) \text{ として}) \\
&= a_{\mathrm{MAP}}(x) + b^{\boldsymbol{T}} (w - w_{\mathrm{MAP}}).
\end{aligned}$$

$$p(a|x,\mathcal{D}) = \int \delta(a - a(x,w))q(w|\mathcal{D})\,dw$$
$$= \int \delta(a - a_{\mathrm{MAP}}(x) - {w_{\mathrm{MAP}}}^{\boldsymbol{T}} b + w^{\boldsymbol{T}} b)q(w|\mathcal{D})\,dw.$$

平均は

$$E[a] = \int a p(a|x,\mathcal{D})\,da = \int \delta(a - a(x,w))w(w)a\,dadw$$
$$= \int a(x,w)q(w)\,dw = (a_{\mathrm{MAP}}(x) - {w_{\mathrm{MAP}}}^{\boldsymbol{T}} b) + \int b^{\boldsymbol{T}} w q(w)\,dw$$
$$= a_{\mathrm{MAP}}(x) - {w_{\mathrm{MAP}}}^{\boldsymbol{T}} b + b^{\boldsymbol{T}} w_{\mathrm{MAP}} = a_{\mathrm{MAP}}(x).$$

分散は $w^{\boldsymbol{T}} b$ が効くので

$$\sigma_a^2(x) = b^{\boldsymbol{T}} A^{-1} b(x)$$

予測分布は 4.20 節式 (4.4) の近似式を使って

$$p(t=1|x,\mathcal{D}) = \int \sigma(a) p(a|x,\mathcal{D})\,da \approx \sigma(\kappa(\sigma_a^2) a_{\mathrm{MAP}}(x)).$$

このボルツマン因子をより基本的な統計力学の定理から導くこともできるが，それは本筋からそれるので省略する．

11.1.3 ポテンシャルエネルギー

上記の話では，E は気体分子の運動エネルギーについてのみ議論したが，分子は運動エネルギーに加えて位置エネルギー（ポテンシャルエネルギー）も持っている（無重力とかなら話は別だが）．そしてポテンシャルエネルギーについてもボルツマン因子は有効である．たとえば地面を無限に広い平面で近似し，各分子がポテンシャルエネルギー $E = m \cdot g \cdot h$ を持つとする（h は地面からの高さ）．そうすると高いところの分子は低いところに比べて高いポテンシャルエネルギーを持っているといえる．当然のことながら，ボルツマン因子によれば高いところまでのぼれる分子はそう多くない．高さに対して指数関数的に少なくなる．つまり上空ほど空気は薄くなる．これは私たちの常識と一致している．

ちなみに温度 300K のときに窒素分子 (分子量 28) の存在確率 p をボルツマン因子で計算すると，標高 0m と標高 2000m での p の比が 0.80 倍と出る．温度 300K というのは 27 ℃に相当し，窒素は空気の大半を占めることを考えれば，これはそう悪くない大気の近似である．そして標高 2000m での一般的な気圧を Google で調べたら 0.80 気圧のようだ．まあ上空に行くほど気温が下がるし，地球は平板ではないので，この近似はいろいろと問題があるが，しかし参考にはなるだろう．

11.1.4 サンプリングへの応用

さてやっと本題に入ろう．私たちは $p(z)$ を考えている．ここでの p はボルツマン因子の p ではなくて，サンプリングしたい対象の確率の p である．この $p(z)$ の分布にそった出現頻度の z の集合がほしい．それならば，この p から対応するポテンシャル E を構成し，その中で物理学的なシミュレーションをすれば，その（仮想的な）粒子の位置 z のログは，サンプリングにふさわしいものになりそうではないか[*4]．これが基本的なアイデアである．E をうまく作れれば，（そして精密に物理学を再現しさえすれば）期待通りの確率で z がサンプルできる．

ということで，ついに (11.54) に似た式が登場する（この式はもちろん統計力学のボルツマン因子に由来している）．

$$p(z) = \frac{1}{Z} \exp\left(-\frac{E(z)}{kT}\right)$$

教科書とは違い，私はまだこの kT を 1 に置きかえない．この式だと，E から p を導く式のように見えておかしいので，これを E について解いておこう．

$$E(z) = -kT \cdot \ln\left(p(z) \cdot Z\right)$$

[*4] z が位置を表す・・・という表現がよくわからなければ，z は仮想粒子の地面からの高さを表す，とでも思ってほしい．z がスカラーであるときは，このたとえは悪くないと思う．

与えられた確率分布 p に対して，こういうポテンシャルエネルギー E を持つ系の中での仮想的な分子の動きをシミュレーションすればいい．つまりはそういうことだ．

PRML ではここから解析力学に突入するのだが，これは難易度が上がってしまうので私は違う方法を選んだ．大丈夫，そんな小難しい理論を使わなくても，十分に説明できる．物理は私のようなものでも理解できるほどに簡単なのだ．ということで，私は高校物理でおなじみのニュートン力学で進める．

さて，この仮想分子にはきっと質量があるだろう．これをとりあえず m としよう．この仮想分子は常にポテンシャルエネルギーから力を受けているのだが，その力は，ポテンシャルエネルギーを z で偏微分し，-1 を乗じれば求められる．

$$F = -\frac{dE}{dz} = kT \cdot \frac{1}{p} \cdot \frac{dp}{dz}$$

この仮想分子の加速度を a とおけば，分子の質量 m は時間変動しないので， $F = m \cdot a$ となり，

$$a = \frac{kT}{m} \cdot \frac{1}{p} \cdot \frac{dp}{dz}$$

となる．最初の因子 kT/m は適当に 1 にしてしまってもいいかもしれない．また p も a に対してこの形でしか現れないので，つまり微分した関数との比だけが重要なので，p を正規化し忘れていても a の値は変わらない．

これでこの仮想粒子をこの加速度に沿って動かしていくシミュレーションをして z のログをとればいいだけなのだが，少し問題というか注意点がある．それについて，次の節に書こう．

11.1.5 注意点

私たちは，統計力学という「多数の分子が衝突しながら乱雑に動く」物理学を使っている．ということは，このシミュレーションにおいて分子はどのくらいの個数を扱わなければいけないのだろう．100 個くらいでいいのか？ 1 万個か？ 100 万個くらいだろうか．・・・いやいや欲を言って厳密を期するのであれば 1 モルくらいはほしいかもしれない．つまりは 10^{23} 個程度である．

もちろんそんなことはやっていられない（メモリの消費量が尋常ではない）．ということで 1 個の分子の動きをシミュレーションしていくだけで同等の結果を得る方法を考えよう．これはいわば本質ではなくただの技巧（テクニック）である．

やることは簡単で，まずシミュレーション時間を十分に長くとることだ．そうすれば分子は（たとえ一つであったとしても）さまざまなポテンシャルの場所を探検してくれる．そうすれば一分子ながら $E(z)$ 全体を十分に反映した z のログができて，サンプリングができる．・・・おっと言い忘れたが，もちろんシミュレーション内に複数の分子を配し（つまり z や速度 v の初期値が違う），それぞれシミュレーションしてもいい．それは並列化が有効な手法である．

またシミュレーションの最初のころの z は信用できないとしてログからは捨てることを推奨する．しばらくシミュレーションしていると仮想分子は $E(z)$ を反映した場所をうろつくようになるが，それまでは初期値の選び方の影響を強く受けてしまい，サンプリングに際して有効とは思えない．なお，初期値に恵まれないと（ポテンシャルエネルギー的な意味合いで）どこかのくぼみにハ

マってなかなか出られない，なんてこともありうるだろう．こうなると $E(z)$ 全体を十分に反映しているとは言い難い．そこで，たまに zを 乱数で初期化したらより良いと私は思う．つまり分子はワープするのだ．ただワープ後しばらくはやはり z は $E(z)$ を反映していないので，サンプリング用のログとしては使わずに捨てるのを推奨する．

次は温度の問題を論じよう．私たちは先ほど気楽に kT/m という係数を 1 と置いたが，これは T を定数として扱っていることになる．なぜなら，k も m も定数だからだ．つまりこのシミュレーションは温度が一定の仮想世界を考えているということである．

しかし普通のニュートン力学でシミュレーションをしているだけだと，これは達成できない．なぜなら，普通のニュートン力学だけのシミュレーションでは，系のエネルギーが（計算誤差を除いて）一定値をとるからだ．つまり，ポテンシャルエネルギーが低い地点では運動エネルギーが多くあって，高い地点では運動エネルギーが少ないことになる．しかし自然界の温度一定系では，そうなってはいない．分子はほかの分子との衝突やもしくは輻射のやりとりで運動エネルギーを得たり失ったりしており，結果的にポテンシャルエネルギーの値とは独立に運動エネルギーを得ている．運動エネルギーの平均値は（周囲の）温度にしか依存しない．つまり，温度一定とエネルギー一定は一般には両立しないのだ．

ということで，運動エネルギーをたまに調整してやらねばならない．これは分子同士の衝突現象の代用である．つまりは速度を適当に決めなおすということだ．これをやらないと $p(z)$ の分布は実現しないので重要である（実は当初私はこれをミスっておかしなサンプリング結果になり泣かされた）．速度ベクトルの各成分は乱数で決めればいいだろう．運動エネルギーもボルツマン因子に従うはずなので，正規分布な乱数を使えばいいだろう．このとき，温度や質量の関係が加速度 a の算出に使ったものと矛盾しないようにすべきだろう．私がここで言わんとしていることは，速度を決めなおすときに乱数を適当に使うわけだが，その速度による運動エネルギーの期待値が，$\frac{kT}{2} \times (z$ の次元数) くらいになるように，乱数の分布に気を配ってほしいということである．

もし，z が 4 次元以上のベクトルであれば，それはもはや物理学からは離れてしまうが，ニュートン力学は第 4 の座標変数があったとしても自然に拡張可能であり（誰でも容易に類推できる），それでおそらく問題はないであろう．

はみだしコラム「分配関数 Z は z の関数なのか？」

分配関数 Z は，ボルツマン因子を確率として正規化するための定数である．

$$p(z) = \frac{1}{Z} \exp\left(-\frac{E(z)}{kT}\right) \qquad \text{107 ページ参照}$$

この Z は，系が取りうるエネルギーのついてのボルツマン因子をすべて計算し，それを足し合わせることで求められる．これは確率の正規化の基本を思い起こしてもらえれば自明である．つまり，すべての確率を足したら 1 にならなければいけないので，正規化前のものをとにかく全部足して，それで割ってやればよいということだ．・・・正規化定数はその名の通り定数なのであるが，しかし別の視点から見ると関数でもある．それゆえに Z は分配「関数」などという別名があるのだ．その話を私はしたい．

いろいろと理屈をこねてもいいのだが，とにかく一回 Z を計算してみようではないか．それが

一番わかりやすいだろう．今ここに，エネルギーがとびとびの値しか取れない実験装置がある．階段のような地形でしかも物体がなんらかの理由で地面から離れられないと想定すればいいだろう．もしくは，量子力学的に量子化された状態だと思ってもらってもいい．とにかく，ここではエネルギー E が $0, 1, 2, 3, 4, ...$ と整数値しかとりえない．そういう状況を考えてほしい[*5]．

このケースで Z を計算してみよう．実に簡単である．

$$Z = 1 + \exp\left(-\frac{1}{kT}\right) + \exp\left(-\frac{2}{kT}\right) + \exp\left(-\frac{3}{kT}\right) + ...$$

これは無限等比数列の和なので，簡単に整理できる．

$$Z = \frac{1}{1 - \exp\left(-\frac{1}{kT}\right)}$$

さてこの Z を見てほしい．これは何の関数だろうか．なんの定数だろうか．・・・まず，Z は E を一切含んでいない．それは当然だ，なぜなら E に値を代入して数列を作り，それを全部足したのだから．代入したのだから，式中に E はもう残っていない．だから Z は E に対しては定数である．また Z の式の中には T という値が残っている．つまり Z は T の関数なのだ．

私たちの考えている E は，$p(z)$ から構成したものなので，当然のことながら z の関数であった．しかしこの E はもう Z には残っていないので，Z は z の関数ではない．・・・おっとこれは言い過ぎかもしれない．もし温度が場所によって違うような系を考えているのなら，T が z の関数になるので Z は z の関数であると言えるだろう．しかし私はそういう複雑な状況を今回は想定していない．温度は系全体で共通な定数だと想定している．

さらに註・・・というかもはや追記

私はこの説明において，あまりよく考えずに PRML の流れに合わせて kT/m を 1 としてしまったが，温度は本当はそんなに軽く扱っていいものではない．いやそれを言ったら質量だって適当にしてはいけないかもしれない．物理屋の意地があるので少々語ることにする．

まず温度だが，もし温度があまりに低いと分子はほとんどエネルギーを持てないので，最寄りのポテンシャルエネルギーの低い場所にすぐに収まってしまって，そこから二度と出てこない．もちろん z の初期値に恵まれて，そこから下る過程で高い運動エネルギーを一時的には持てるかもしれないが，それも温度を考慮した速度 v の取り直しによって，結局失ってしまう．この場合，結局仮想粒子はごく狭い範囲をちまちま動くことしかできず，それはつまり z の初期値に強く影響されたサンプリング，言い換えれば $p(z)$ 全体をほとんど反映していないサンプリングとなる．これではもちろんいけないだろう．

では温度を高くしてやればいいのか．確かにそうすればポテンシャルの丘が多少あっても難なく飛び越えていくだろう．つまり仮想粒子は $p(z)$ 全体を十分に探検できるようになる．・・・しかし話はそう単純ではないのだ．もし温度が不適切なほど高ければ，もはや仮想分子は $p(z)$ に影響されなくなる．なぜなら，自分が温度由来で与えられている速度 v に対して，ポテンシャルから与

[*5] 何か適当な定数 c を考えて，$E = 0, c, 2c, 3c, 4c, ...$ とすればより一般的になるが，私は定数と言えども文字を増やして話をややこしくしたくなかったのであえて単なる整数とした．エネルギーの単位を適当に調整したと思ってもらってもいい．いずれにせよ，結論は全く変わらない．

えられた加速度 a が小さく，もはやノイズ程度にしかならないからだ．こうなってしまうと，どの z に対しても $p(z)$ は同じ，みたいな系のサンプリングをしたような結果にしかならず，これも不本意だろう．

今度は質量 m について考えてみよう．質量はポテンシャルエネルギーの傾きがどのくらい加速度に変換されるかの比例定数である．分かりやすくするために極端な例を考えよう．もし質量が無限大だったらどうだろう．そうとも，加速度 a は常にゼロとなり，仮想粒子は $p(z)$ に影響されることなく，温度をベースに与えられた v のまま等速直進運動をすることになる．これは温度が高すぎた場合によく似ている．これはダメだ！・・・では質量が 0 にかなり近い小さい値だったらどうか．これは温度が 0 だった場合のようにふるまうことになり，これもダメだ．

・・・と言いたいところなのだが，速度の初期値や取り直しの際に，私のアドバイスに従い運動エネルギー $\frac{1}{2}mv^2$ の期待値が $\frac{1}{2}kT \times (z$ の次元数$)$ となるようにとっているのであれば，m が大きいときには v が小さくなり，m が小さいときには v が大きくなるので，加速度 a のスケールと自動的に同じになる．ということで，質量 m をどのくらいの値にすべきかについては，あまり深刻に悩まずともよさそうである．

さらに温度についてもいいニュースがある．私たちはポテンシャルエネルギーを定義するときに $E(z) = -kT \cdot \ln(p(z) \cdot Z)$ としている．つまり温度が高い場合は，相応にポテンシャルの起伏も激しくなるのだ．温度が低い場合は，それに合わせてポテンシャルの起伏もなだらかになるのだ．だからたぶん温度についてもそこまで神経をとがらす必要はない．

あとは v の取り直しの頻度についても考えてみよう．v の取り直しが頻繁に起こる場合，これは仮想世界中の粒子数が非常に多くて過密であることを意味する．だからしょっちゅうぶつかっているわけだ．また，取り直しの頻度が少ない場合，これはかなり希薄な気体をシミュレーションしているということになる．

まず頻度が低すぎるというのはよろしくない．なぜなら温度が安定しなくなるからだ．そもそもそんなに希薄な気体では総分子数は相当に少ないということになるだろうが，そんな系では統計力学的な温度の定義が通用しなくなる．統計力学は十分に多数の分子がエネルギーをやり取りしているような状況を前提に組まれているのだ．だからこそやっかい極まりない揺らぎの問題を解消できている．それが通用しなくなるほど希薄だとするとさまざまな前提が崩れて，今までの説明通りにはいかなくなってしまう恐れが出てくる．・・・別の言い方をするなら，この場合の仮想粒子は，運動エネルギーとポテンシャルエネルギーの和が一定になるような運動を過剰に長く続けてしまう．これは先に書いたように，温度一定とは異なる挙動である．

・・・では頻度をうんと上げるのはどうだろう．今度は $p(z)$ がほとんど反映されなくなってしまう．ポテンシャルエネルギーから加速度を決めてようやくその向きに進み始めたところで，v がリセットされてしまえば，結局 $p(z)$ は仮想粒子の運動にほとんど影響できなかったことになる．ということは上記で温度が大きすぎた場合の考察のような，残念な結果になるだろう．

11.1.6 余談

PRML 下巻の p.264 では**ハミルトン**を紹介しているが，私ならその場所に，**ルートヴィッヒ・ボルツマン**を置くだろう．興味があれば wikipedia で彼について調べてみてほしい．