

ニューラルネット PCA によるデータ特徴ベクトルの推定

井上由美子 (指導教員: 吉田裕亮)

1 ニューラルネット

ニューラル・ネットワーク・モデルとは生物の神経系の特徴的な機能に着目してそのモデル化を行い、情報処理に応用しようとするものである。近年の基礎的研究の蓄積ならびに様々な分野での数多くの応用例とともに、その有用性が認識されつつある。

生物の神経系とニューロンのモデル

生物の神経系は、多数のニューロンが複雑に結合され、それぞれが並列に処理を行っている。各ニューロンの大まかな構造は、入力端子である樹状突起、出力端子である軸索ならびに本体の細胞体から成り立っている。

各ニューロンの樹状突起はシナプスを通して他のいくつかのニューロンからの入力信号を受け取る。ニューロン間の信号伝達は、電気パルスによって行われ、ニューロンの状態が変化する。このニューロンの状態の変化は、シナプス結合の種類に依存する。

1943年に、McCulloch - Pitts はこうした生物系内のニューロンの動作原理に基づいて数式で示されるニューロンのモデルを提案した。ニューラルネットが学習によって知能を獲得することが知能を有するシステムの構築に役立つといわれている。

ニューラルネットワークにおける学習メカニズムは、ニューロン間の結合強度を何らかの学習則に基づいて更新することにより、ネットワークが望ましい出力を出すように学習をさせることである。

ニューラルネットの学習方式としては入力データに対して理想的と考えられる出力値、すなわち教師信号が与えられている場合とそうでない場合がある。前者の学習形態を教師あり学習と呼び、後者の学習形態を教師なし学習と呼ぶ。教師あり学習の場合は、ニューラルネットからの出力と理想的な出力(教師信号)を比較することによってその差をできるだけ小さくするよう結合強度の値を変更する。一方、教師なし学習の場合は、理想的な出力は外部から与えられないのでネットワーク内での評価基準を内蔵しておくことが必要となる。

2 ニューラルネット PCA

主成分分析について

多次元確率ベクトルの変動をできるだけ少数個の変量で説明する手法のひとつとして主成分分析(PCA)が用いられる。

主成分分析は K. Pearson (1901) によって創始され、H. Hotelling (1933) によって、より一般的な形で再発見された。

主成分は元の変数の線形結合を考え、その分散が最大

となる線形結合を求める。このとき、係数を大きくすると分散はいくらでも大きくなるので、係数の 2 乗和が 1 であるという条件の下で分散が最大となるようにする。こうして第 1 主成分が定まる。次に、この主成分とは無相関なものの中から分散が最大になるものを定める。以下、この手続きを繰り返して順次、主成分が定義される。

ニューラルネット PCA

この主成分を決定するためにニューラルネット・モデルを用いる手法はニューラルネット PCA と呼ばれる。すなわち、 n 次元の入力データ x を (x_1, x_2, \dots, x_n) とし、データ集合をベクトル ω に平行な軸へ投影した分布が最大の分散を持つように、 ω (結合強度)を学習し計算していく。 $\sum \omega_i^2 = 1$ の場合、 ω への x 射影は、まさに $y = \sum \omega_i x_i$ であり、射影した分布の分散は、 y の分散に等しくなる。ニューラルネット PCA は確率ベクトルを順次入力しその特徴を学習して行く、いわゆるオンライン学習であり、したがって時間と共に変動する特徴を捉えることへの応用なども考えられる。

本研究ではニューラルネット PCA の学習則のひとつとして知られている Recursive Least Squares 学習アルゴリズムを用いる。与えられたデータに対して、学習回数の増大と共に、ニューラルネット PCA で推定される特徴ベクトルとデータから得られる第 1 主成分ベクトルとの差の 2 乗ノルムの収束のオーダーは既に調べられている。

本研究では、さらに第 2 以降の固有ベクトルについて同様にニューラルネット PCA によって推定される特徴ベクトルと第 2 以降の主成分ベクトルとの差の 2 乗ノルムの収束のオーダーを調べる数値実験を試みた。

3 RLS (Recursive Least Squares) 学習アルゴリズム

Sanger が提案した RLS (Recursive Least Squares) 学習アルゴリズムとは、出力ユニットを複数個用意することにより、欲しい個数まで主成分を検出するニューラルネット PCA である。これは、1 番目の出力ユニットで上に述べた方法で主成分を検出し、入力からこの主成分を取り除いた信号を生成する (Schmidt の直交化を行っていることになる)。次に、これを 2 番目の出力ユニットの入力とすることで第 2 主成分を検出する。以下、逐次必要な個数まで主成分を求める手法である。

第 m 固有ベクトルを求める RLS 学習アルゴリズム RLS 学習アルゴリズムは以下のようなになる。

Step1: ネットワークの初期化

入力層と出力層の間の重みの初期値 $\omega_m(0)$ を乱数を用いて設定する。

Step2: 入力ベクトルの入力

数値実験では、適当な共分散行列を持つ N 次元正規乱

数 $x(t) = (x_1, x_2, \dots, x_N)$ を発生させて入力する。

Step3: ネットワークの出力 (中間層) を次式に従って計算する

$$h_m(t) = \omega_m^t(t-1)x(t),$$

$$K_m(t) = \frac{P_m(t-1)h_m(t)}{1 + h_m^2(t)P_m(t-1)}.$$

Step4: 結合強度の更新

- 結合強度を下の式に従って更新する。

$$\omega_m(t) = \omega(t-1) + K_m(t)[d_m(t) - h_m(t)\omega_m(t-1)].$$

- ここで教師信号 $d_m(t)$ は

$$d_m(t) = x(t) - \sum_{i < m} \hat{h}_i(t)\hat{w}_i.$$

ただし $\sum_{i < m} \hat{h}_i(t)\hat{w}_i$ は $m-1$ 番目までの主成分の総和を表す。

Step5: 下式に従って、 $P_m(t)$ の値を定める。

$$P_m(t) = [1 - K_m(t)h_m(t)]P_m(t-1)$$

Step6: 収束の判定

$l_m = (C\omega_m(t)/\omega_m(t))$ の要素の平均) としたとき

$$C\omega_m(t) = l_m\omega_m(t)$$

というベクトルの各要素が設定値以下になれば学習が収束したと判断して終了する。

収束しなければ Step2 から Step6 までの操作を繰り返す。

4 数値実験

Recursive Least Squares 学習アルゴリズムによって推定された固有ベクトルと真の固有ベクトルとの差の 2 乗ノルムの収束に関する評価を行う。

なお、最大固有ベクトルの 2 乗ノルムの収束オーダーは、データの次元数 20 までではほぼ一定であることがわかっている。

他の固有ベクトルについても、それぞれ同様な計算を行い、どのように収束するのかを調べる実験を試みる。

入力データとして 5 次元データ、10 次元データ、15 次元データ、20 次元データを用意する。

入力データについて

平均が μ_1, \dots, μ_N で、分散および共分散が σ_{ij} ($1 \leq i, j \leq N$) の N 次元正規分布に従う乱数 x_1, \dots, x_N は、互いに独立な標準正規乱数 y_1, \dots, y_N から次のようにして作ることができる。

$$x_1 = \mu_1 + a_{11}y_1$$

$$x_2 = \mu_2 + a_{21}y_1 + a_{22}y_2$$

...

$$x_N = \mu_N + a_{N1}y_1 + a_{N2}y_2 + \dots + a_{NN}y_N$$

まず、 x_i が平均値 μ_i の正規分布をすることは明らかである。

x_i の分散 σ_{ii} は、

$$\sigma_{ii} = a_{i1}^2 + a_{i2}^2 + \dots + a_{ii}^2$$

に等しい。また、 x_i と x_j の共分散 σ_{ij} ($i > j$) は、

$$\sigma_{ij} = a_{i1}a_{j1} + a_{i2}a_{j2} + \dots + a_{ij}a_{jj}$$

に等しい。従って、発生すべき乱数 x_1, x_2, \dots, x_N の分散および共分散 σ_{ij} ($1 \leq j \leq i \leq N$) が与えられると、上式を満たす a_{ij} ($1 \leq j \leq i \leq N$) が求められる。

以下のように学習回数と 2 乗ノルムの収束のオーダーを調べる。

- データの次元数: 5 次元, 10 次元, 15 次元, 20 次元

縮約した次元数: 3 次元

計算した固有ベクトル: 第 1, 第 2, 第 3

- データの次元数: 15 次元, 20 次元

縮約した次元数: 11 次元

計算した固有ベクトル: 第 1, 第 2, 第 3, 第 5, 第 8, 第 11

- データの次元数: 10 次元

縮約した次元数: 8 次元

計算した固有ベクトル: 第 1, 第 2, 第 3, 第 5, 第 8

まず、これら 4 種類の次元について、それぞれのデータ ((学習回数より大きい数, 次元数) 行列) を 1 個ずつ用意して、計算した。その結果、最大固有ベクトルの収束のオーダーは、既に知られている結果と似ている実験結果が得られた。しかし、他の固有ベクトルについては最大固有ベクトルとは違う実験結果も得られた。

よって、データによっては RLS 学習アルゴリズムから学習回数とともに、精度の高い全てのデータ特徴ベクトルを推定できないのかもしれないということがわかった。

5 まとめと今後の課題

上で示した実験で他の固有ベクトルについてもそれぞれ最大固有ベクトルと同様な結果が得られれば、RLS アルゴリズムによって第 2 固有ベクトル, 第 3 固有ベクトル, ... と次々に固有ベクトルを推定できる。(参考までに最大固有ベクトルについては下のような結果が知られている。) 今後の課題には、時系列データ (AR モデル) の解析に RLS アルゴリズムを応用があげられる。

図 2: 学習回数 (Recursive Least Squares 学習アルゴリズム)

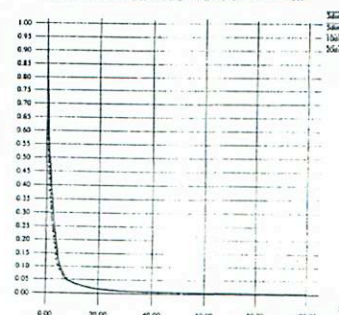


図 3: log-log プロットしたもの

