バギングを用いた2次元非線形判別曲線の推定

白川　聖子　(指導教官 : 吉田 裕亮)

1　はじめに

ニューラルネットワーク法などによる非線形判別は線形判別に比べて, 一般に多くのパラメータの推定が必要であり, 計算が繁雑である. そこで現実にはより簡単な方法で非線形な判別曲線を推定することが求められる. 本研究ではこのような方法のひとつとして, バギング法を用いた非線形判別曲線の推定手法を提案する. バギングにおける 1 つ 1 つの仮説は, 弱仮説と呼ばれるものであり比較的簡単な推定であるが, それを重ね合わせることにより最終的に強い仮説を推定する.

2　バギング法

バギング (bagging) とは bootstrap aggregating に由来し, その名のとおりブートストラップ法により例題をリサンプリングして異なる弱い仮説を多数作り, それらから集合体を構成することによって最終的な仮説を作る方法一般を指す. なお仮説の生成は並列的であり, リサンプリングは独立に行うので, 弱仮説どうしは互いに影響しない.

3　非線形判別曲線の推定法

2 次元データの非線形 2 群判別にバギング法を用いた推定を行うため, 以下のような操作を施す.

3.1　推定のアルゴリズム

1. データ領域内にランダム領域をとる.
2. ランダム領域内からデータをいくつか抽出し, それぞれ 0 値, 1 値を取るものに判別する.
3. 0 値を取るもの, 1 値を取るもの各群の重心座標を求め, ランダム領域内で 2 点に関するマハラノビス距離による中点の軌跡を求める.
4. 工程 1 ～ 3 を N 回繰り返し, データ領域内において n 回以上通った点を再度プロットする.
   N: 指定する十分大きな自然数
   n: N に応じて適宜指定される自然数

3.2　ランダム領域

本研究ではデータ領域よりも大きな, 架空領域を考える. 架空領域内で乱数 s_1, t_1, u を与え (s_1, t_1) 座標を 1 点, u を一辺の長さとする正方形をランダム領域とする.

この手法は簡単な領域の取り方である. しかしデータ領域内だけで行うと偏った部分に多く領域が現れる. そこで架空領域を考える. するとランダム領域は, 架空領域内では偏りがあるものの, データ領域内での偏りはかなり減少させることができる.

3.3　シミュレーションデータでの実験

2 次元の有界領域 $D = [-2, 2] \times [-2, 2]$ 内で与えた曲線により, それぞれ 0 値, 1 値が割り当てられたデータを計 1000 個用意する. 以下のシミュレーションは $N = 8000, n = 5$ と設定されている.

群の設定：　$y \geq x^3 - x \rightarrow \{0\}$
　　$y < x^3 - x \rightarrow \{1\}$

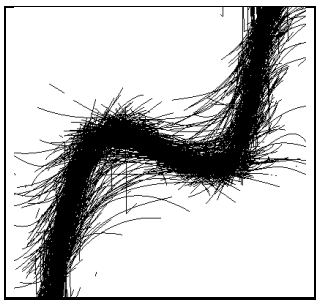


図 1: N=8000

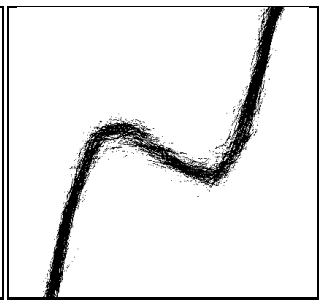


図 2: n=5

図 1 はアルゴリズムによって描かれた, 8000 本の曲線である. 図 2 は図 1 において, 5 回以上通った点のみを取り出したものである.

図 2 を可能な限り 1 本の曲線に近づけるため, 本研究では以下の 2 段階の画像補正を施した.

4　画像補正

4.1　補正アルゴリズム 1

1. 画像を pbm データとして保存する.
2. ある点 a を中心とする $9(= 3 \times 3)$ 点のうち 1 が m 点以上の場合 $a = 1$, m 点未満の場合 $a = 0$ とする.
   (m: 9 以下の指定する自然数)
   なお 9 点を $25(= 5 \times 5)$ 点と変えて, 大きな範囲を基に補正を行うこともできる.
3. 工程 2 の m を変えながら M 回繰り返す.
   M: 指定する自然数 (10 ～ 20 程度)

すなわちこの補正アルゴリズム 1 によって, 画像を 1 本の太い曲線に近づけることになる. 次に太い 1 本の曲線を細くする, 以下の補正アルゴリズム 2 を施す.

4.2 補正アルゴリズム 2

1. 画像を pbm データとして保存する. 順に補正を行うが, 各点の元の値を α とし, 補正後の値を β とする. $(\alpha, \beta \in \{0, 1\})$
2. 各行 (横方向), 両端については $\beta = \alpha$.
3. その他の点については
   (1) $\alpha = 0$ のとき,
   (i) 並びが $1\alpha1$ ならば $\beta = 1$,
   (ii) それ以外ならば $\beta = 0$,
   とする.
   (2) $\alpha = 1$ のとき,
   (i) $0\alpha0, 1\alpha1$ ならば $\beta = 1$,
   (ii) $0\alpha11\cdots1, 1\cdots11\alpha0$ ならば,

$$\begin{cases} 1 \text{ の個数が } L \text{ 個より大のとき} \beta = 0, \\ 1 \text{ の個数が } L \text{ 個以下のとき} \beta = 1, \end{cases}$$

   とする.
4. すべての行に対して工程 $2 \sim 3$ を行う.
5. 工程 $2 \sim 4$ を列 (縦方向) で行う.
6. 横方向, 縦方向の補正を交互に複数回繰り返す.

L の値によって最終的な曲線は定まる. また L の値に依存するが, 横縦方向を 1 セットと考え, 約 30 セット以上繰り返すと, 画像はほぼ一定の状態に落ち着く.

4.3 画像補正を施した例

推定法のシミュレーションで使用したデータの結果を使用する. なお $L = 5$ とし, 30 セットの繰り返しを行った.

群の設定： $y \geq x^3 - x \to \{0\}$
$y < x^3 - x \to \{1\}$

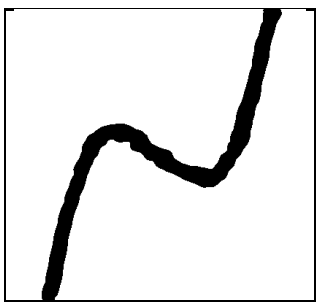


図 3: 補正 1 後

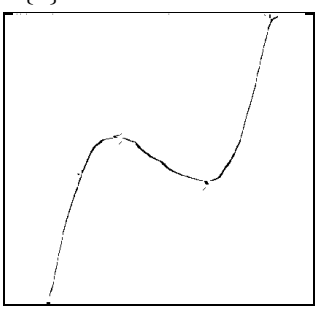


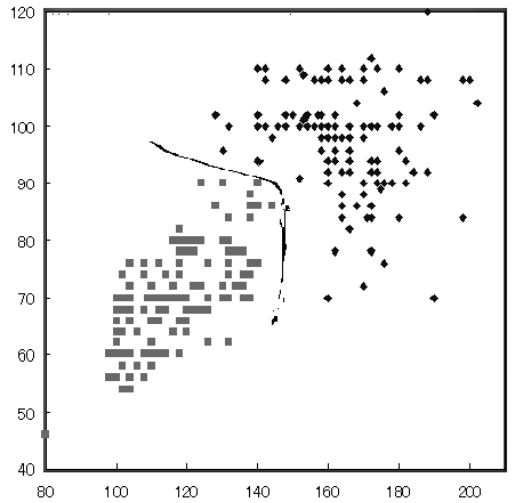
図 4: 補正 2 後

図 3 は補正 1 を m の値を変えながら, 15 回行ったものである. 図 4 は, 図 3 に補正アルゴリズム 2 を適用した結果で, $L = 5$ であることから縦, 横共に 5 ピクセル以下となる.

5 実データへの応用

実データへの応用として, 最高血圧値, 最低血圧値に基づいて血圧疾患の判定が行われた健康診断のデータを用いた. 正常 (0 値), 異常 (1 値), それぞれ 130 人ずつとなるように抽出した.

以下横軸が最高血圧値, 縦軸が最低血圧値であり, 推定アルゴリズムと補正アルゴリズム 1, 2 を施した結果とデータを重ね合わせた図である.



WHO の血圧判定基準によると, 最高血圧値 140mmHg, 最低血圧値 90mmHg からが高血圧とみなされる. 上の結果より, 用いたデータはこの基準に基づいて血圧異常の判定を行っていると推定できる.

6 まとめ

バギング法と簡単な画像処理を用いた, 2 次元非線形判別曲線を推定するひとつの手法を提案した.

シミュレーションデータの場合, 事前に設定したデータ値の境界線とほぼ同じ曲線 (特に円の場合においても) が推定可能であった. 実データへの応用では, シミュレーションと比べデータ数が少なく, データのばらつきに偏りがあるにも関わらず, ある程度の判別曲線の推定が得られた. したがって, 本研究で提案した手法は 2 次元データの非線形な判別曲線の推定に有効な手法のひとつであると思われる. 今後の課題として, 少数データに関する適応法なども検討したい.