6-7

汎化性向上のための相補的学習機構に基づくアンサンブル学習

日大生産工(院)　○町田和明
日大生産工　　山下安雄

1. はじめに

機械学習の分野において大規模な学習モデルを構築する際には、計算量の増大や汎化性が問題となってくる。一般に、機械学習における学習とは、データに内在する規則性を獲得していること、あるいは、同じ生成構造を持つ未学習データに対して、学習により構築された仮説(ネットワーク)との当てはまりが良くなるように行っていく。このとき、未学習データに対して仮説の当てはまりが良いとき、その仮説は汎化性が良いといわれる。

近年、この汎化性向上のためのアプローチとして、同一タスクに対して複数の仮説を生成し、それらの出力を統合することで最終仮説を求めるアンサンブル学習法が提案されている。アンサンブル学習の著名な手法としては、Bagging や Boosting がある。Boosting は一般に Bagging よりも高い精度が得られることが知られている。しかし、ノイズの多く乗ったデータに対して Boosting は精度が低下する一方、Bagging はノイズに対して頑健であるという特性がある。このように、それぞれの手法には利点と欠点の双方が存在し、データの状態によって結果が影響してしまう。そのため、双方の欠点を補う形の新たな学習機構の構築が有効であると考えられる。こうした考えのもと、Bagging と Boosting を統合させて最終的な分類精度を向上させようという研究報告もある。

本研究では、まず、UCI のデータセットを用いて、データ集合の状態が学習に与える影響ついて述べる。そのあと、汎化性向上のための Boosting アルゴリズムを基にした相補的学習アルゴリズムを提案する。

2. AdaBoost アルゴリズム

AdaBoost は Boosting の中で最も広く利用されているアルゴリズムである。AdaBoost は、与えられた学習アルゴリズムを実行して、一回のラウンドで一つの仮説を生成する。このとき、ラウンドの繰り返し回数は前もって決定されているものとする。また、各訓練データに対して、重要度を反映した重みを設定する。各学習時に用いられる重み係数は、それ以前に学習された仮説の性能に依存してデータ点ごとに計算される。そして、それら各ラウンドにより生成された仮説の予測を重み付き多数決によって結合し、最終的な予測を得る。AdaBoost をはじめとする Boosting と Bagging の原理的な違いは、仮説を逐次的に学習するか否かという点で、AdaBoost は前者にあたる。

具体的には、初回を除き、各ラウンドでは前ラウンドの仮説が誤分類したデータ点の重み係数を増加させ、正しく分類したデータ点の重み係数は変えない。つまり、逐次的に学習される仮説では、それ以前の仮説で誤分類されたデータ点が強調される学習機構であるといえる(Fig.1、Fig.2)。

Ensemble Learning based on Complementary Learning Framework for Improving Generalization Property

Kazuaki MACHIDA and Yasuo YAMASHITA

[入力] : $(\boldsymbol{x}_1,\ \mathrm{t}_1,),\ldots\ldots,(\boldsymbol{x}_\mathrm{N},\ \mathrm{t}_\mathrm{N},)$;

$\boldsymbol{x}_i \in \boldsymbol{X},\ \mathrm{t}_\mathrm{N}\ (\mathrm{t}_\mathrm{n} \in \{0,1\})$

1. n = 1,.....,N のデータの重み係数 $\{\mathrm{w}_\mathrm{n}\}$ を、
   $\mathrm{w}_\mathrm{n}^{(1)} = 1/\mathrm{N}$ に初期化する。
2. m = 1,.....M について以下を繰り返す :

(a) 仮説 $\mathrm{y}_\mathrm{m}(\boldsymbol{x})$ を、次の重み付けされた誤差関数を最小化するように訓練データにフィットさせる。

$$\mathrm{J}_\mathrm{m} = \sum_{\mathrm{n}=1}^{\mathrm{N}} \mathrm{w}_\mathrm{n}^{(\mathrm{m})} \mathrm{I}(\mathrm{y}_\mathrm{m}(\boldsymbol{x}_\mathrm{n}) \neq \mathrm{t}_\mathrm{n})$$

なお、$\mathrm{I}(\mathrm{y}_\mathrm{m}(\boldsymbol{x}_\mathrm{n}) \neq \mathrm{t}_\mathrm{n})$ は、指示関数であり、$\mathrm{y}_\mathrm{m}(\boldsymbol{x}_\mathrm{n}) \neq \mathrm{t}_\mathrm{n}$ のときには1であり、それ以外は0である。

(b) 次の値を計算する。

$$\epsilon_\mathrm{m} = \frac{\sum_{\mathrm{n}=1}^{\mathrm{N}} \mathrm{w}_\mathrm{n}^{(\mathrm{m})} \mathrm{I}(\mathrm{y}_\mathrm{m}(\boldsymbol{x}_\mathrm{n}) \neq \mathrm{t}_\mathrm{n})}{\sum_{\mathrm{n}=1}^{\mathrm{N}} \mathrm{w}_\mathrm{n}^{(\mathrm{m})}}$$

これを用いて、次の量を求める。

$$\alpha_\mathrm{m} = \ln\left\{\frac{1-\epsilon_\mathrm{m}}{\epsilon_\mathrm{m}}\right\}$$

(c) データ点の重み係数を以下の式で更新する。

$$\mathrm{w}_\mathrm{n}^{(\mathrm{m}+1)} = \mathrm{w}_\mathrm{n}^{(\mathrm{m})} \exp\{\alpha_\mathrm{m} \mathrm{I}(\mathrm{y}_\mathrm{m}(\boldsymbol{x}_\mathrm{n}) \neq \mathrm{t}_\mathrm{n})\}$$

3. 以下の式で、最終仮説の予測を構成する。

$$\mathrm{Y}_\mathrm{M}(\boldsymbol{x}) = \mathrm{sign}\left(\sum_{\mathrm{m}=1}^{\mathrm{M}} \alpha_\mathrm{m} \mathrm{y}_\mathrm{m}(\boldsymbol{x})\right)$$

Fig.1　AdaBoost の学習アルゴリズム

3. データ集合が学習に与える影響

1. はじめに で述べたように、データ集合の状態(ノイズ事例や例外的な事例の有無、分布)によって、学習アルゴリズムの効力に差が出る。AdaBoost では、誤認識したものに対して重みを付加して学習していくため、ノイズや例外的な事例、つまり、herd example に特化した仮説が生成される可能性を持つ。このように、データ集合の状態を知ることは、機械学習を行う上で避けては通れない問題であると考える。

そこで、実際に UCI のデータセットを用いて、訓練用のデータ集合の状態を変えて実験を試みた。このデータセットは、複数の属性値から年収が$50k/yr より多い(class A)か少ない(classB)かを予測する 2 クラス分類問題である。訓練用データとテスト用データの集合の大きさはそれぞれ 10000 と 5000 にした。この訓練用データ集合を用いて BP ネットワークで学習させた結果が Fig.3 である。横軸は学習回数(単位はエポック)、縦軸(主軸 ; 左側)は教師信号とネットワークの出力との正誤数([教:classA, ネ;classA⇒TRUE1],[教:classB, ネ;classB⇒TRUE1],[教:classA, ネ;classB⇒FALSE1], [教:classB,ネ;classA⇒FALSE2])、縦軸(第 2 軸 ; 右側)は訓練データにおける認識率をそれぞれ示している。

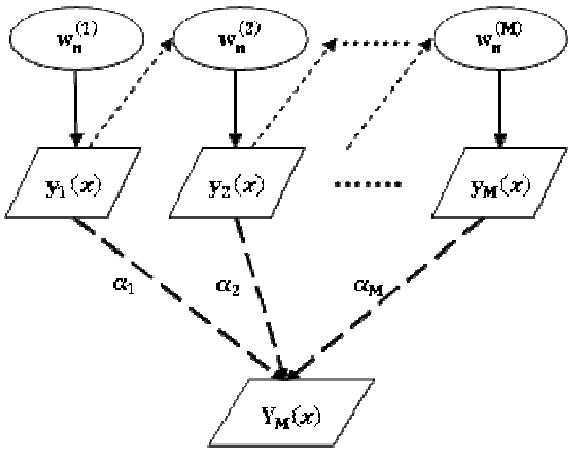

Fig.2　Boosting によるアンサンブル学習過程

これより、学習が終了した時点での訓練データに対する認識率は 85%程度であり、また、学習の初期段階に比べ少しずつ認識率が向上してきている様子が見て取れる。これは TRUE1～FALSE2 の各グループに分けられたものが、学習と共に FALSE 側に属していたデータが TRUE 側へと移ったためである。つまり、学習アルゴリズムによって少しずつパターンの分類精度が向上しているといえる。

しかしながら、ここで注目したいのは、ここで学習したデータ集合には偏りがあるということである。これは、Fig.3 で明らかなように、TRUE2 だけが他と比べて大きな集合になっているからである。このように、データ集合の大きさに偏りがあると、集合の大きな方に重みを置いた形で学習されてしまう可能性がある。そこで、今回この偏りによる差異を検討するた

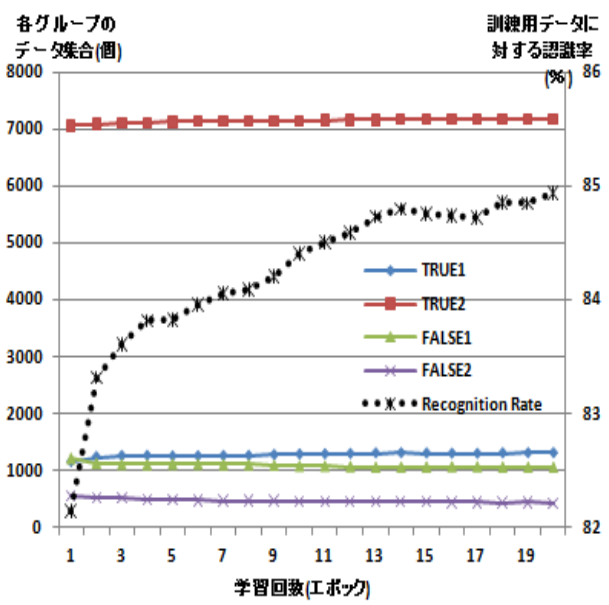

Fig.3　学習過程とデータ集合の分布

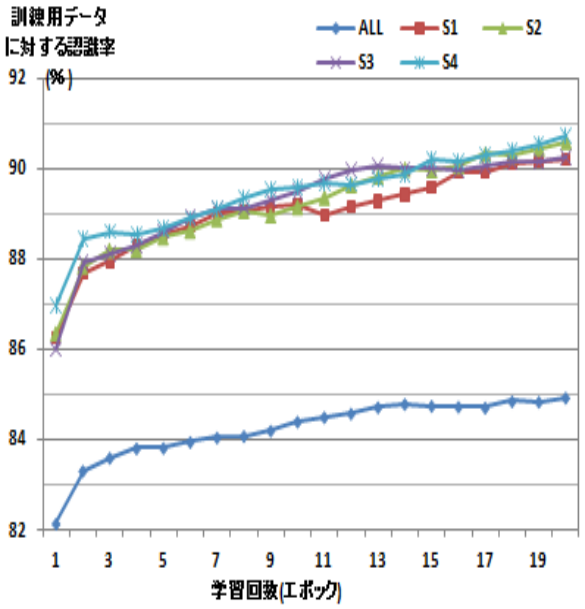

Fig.4　データ集合の相違による認識率

めに、2 クラスの各々のデータ集合の大きさを等しくして学習させた。このとき、データ集合の大きなクラスの方のデータを 4 分割(少々の重複あり)したものの一つと残りのクラスとで新たな訓練用データ集合(部分集合:S_i)を生成した(Fig.5)。その結果が Fig.4 である。これより、データに偏りがある集合(ALL)より、偏りのない集合(S_i)の方が認識率が高いという結果が得られた。

次に、Fig.4 でそれぞれ学習したものについて、今度はテスト用データを用いて認識を行った(Table 1)。ここから、クラスの分布に偏りがないように生成したデータ集合においては、テストデータに対する認識率にかなりばらつきがあった。これは、テスト用データ集合も訓練用データ集合と同様に偏った分布をしているため、部分集合である均一にしたデータで学習した方は多くのパターンに対応できない可能性がある。しかしながら、これら複数個生成された部分集合にも得意とする特定のパターンは学習されていると考えられ、そのそれぞれの良いところを統合(アンサンブル学習)していくことにより、分類精度の向上が図れるものと考える。

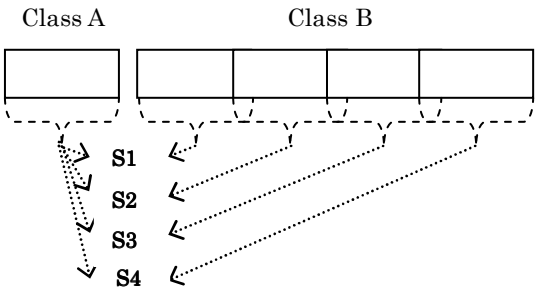

Fig.5　訓練用部分集合の生成

4. 相補的学習機構

ここでは、3. データ集合が学習に与える影響 で示した訓練用データ集合を偏りがないように生成してから学習する手法を用いた、汎化性向上のための学習機構について提案する。

Fig.6 に示すように、まず偏りのない訓練データ集合を生成する。そして、それら部分集合に対して仮説を生成し、Bagging のような手順で仮説を統合する。このとき得られた統合した仮説と、個々の仮説を比較し、最終仮説に好影響を与えている、つまり、両者の仮説の一致度が高い集合を取り出し、それを AdaBoost に与える初期集合とする。このとき、もし AdaBoost の各ラウンドで集合のリサンプリングが必要になった場合は、二番目に好影響だったものを取り出してくる。

Table 1　テスト用データに対する認識率

		認識率 (%)	
		訓練用データ	テスト用データ
訓練データ集合	ALL	84.94	82.51
	S1	90.21	33.58
	S2	90.61	36.54
	S3	90.25	44.74
	S4	90.73	82.04

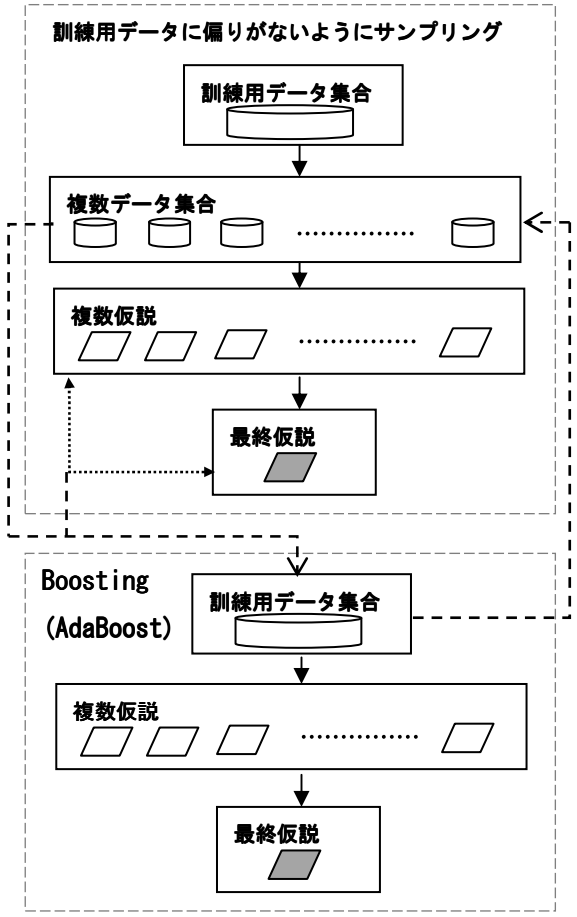

Fig.6　相補的学習機構

これにより、極端に偏ったデータなどがAdaBoostで過度に重み付けられるという状況が軽減されるのではないかと思われる。尚、更に、計算量の問題等も考慮に入れ、ラウンドごとに学習の結果を見て、あまり結果に影響しない属性から順に削除していき、学習させる次元の縮約を行うのも有効であると考える。

5. まとめと今後の課題

今回、データ集合の状態が学習に様々な影響を与えるということを述べ、そして、そのデータ集合の状態にあまり左右されないようにするための相補的学習機構を提案した。この中で、データ集合の状態が学習に影響を与えることから、今回はUCIの一つのデータセット集合だけで議論したが、今後は別のデータセット集合についても取り扱う必要があると考えられる。また、相補的学習機構についても、汎化性を向上させるために、今回提案した方法でどれほど優れた結果が出るか確かめてみなければならない。更に、BaggingやBoostingの従来法とも比較していく必要がる。

以上より、これまで議論してきた内容をより深く研究し、今後益々必要となりつつあるデータマイニングによる知識マネジメントなどへの応用に繋げていきたいと考える。

「参考文献」

1） 町田和明，山下安雄，アンサンブル学習を用いたデータマイニングによる意思決定支援，日本大学生産工学部 (第41回)学術講演会マネジメント部会，(2008)，pp.77-80.

2） 土屋成光，藤吉弘亘，Boost 学習に基づく特徴量の貢献度を用いた特徴選択手法, 画像の認識・理解シンポジウム(MIRU)，(2008).

3）G. Martínez-Muñoz，D. Hernández-Lobato and D. Suárez.，“An Analysis of Ensemble Pruning Techniques Based on Ordered Aggregation”，IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.31, No.2, (2009)，pp.245-259.

4） 中村雅之，野宮浩揮，上原邦昭，重み更新規則の修正による Boosting アルゴリズムの改善について，情報処理学会論文誌，Vol.45, No.3，(2004)，pp.1001-1013.

5） R. Kohavi and B. Becker，UCI Machine Learning Repository : Data Sets , http://archive.ics.uci.edu/ml/datasets.html

6） C. M. Bishop，“Pattern Recognition and Machine Learning”，Springer，(2006).