データマイニングにおける統計的外れ値検出

竹内 純一* 山西 健司†

1 まえがき

近年，統計学，機械学習 (machine learning)，ニューラルネットなどの応用の場として，データマイニング (data mining) が盛んに研究されている．データマイニングとはデータを採掘するという意味で，膨大なデータの中から，自明でない有用な情報を取り出すことを指す．これは，様々な分野において計算機に蓄えられるデータ量が近年飛躍的に増大したために，膨大なデータを十分に活かし切れなくなっていることによる．

外れ値/異常値の検出は，古くから統計学の問題の一つであるが，データマイニングの文脈でも重要な問題になっている．その応用範囲は，クレジットカード不正利用や携帯電話のなりすまし利用などの詐欺 (fraud) の検出やネットワーク侵入 (intrusion) 検出，保険金請求データの例外事象 (rare event) 検出などに渡っている．

従来，外れ値検出の手法として，統計学における検定手法に基づく方法があったが [1, 3]，多変数の場合に計算上の困難を伴う，カテゴリカル変数と連続値データを同時に扱えない，オンライン処理できない等の問題があった．また，データマイニングや機械学習の分野では，ニューラルネット等を用いる教師あり学習に基づくものがいくつか提案されているが [2, 4, 7, 10]，教師無しの状況に適応できない，外れ値の統計的意味が不明瞭等の問題があった．

筆者らは，以上の問題点を克服し，オンラインかつ教師無しで統計的外れ値を検出するツール **SmartSifter** を開発している [11, 14, 12] (sifter は「ふるい」の意)．本稿では，SmartSifter の原理を簡単に説明し，これをネットワーク不正侵入検出に適用した事例を紹介する．

*たけうち じゅんいち．NEC 情報通信メディア研究本部．
†やまにし けんじ．同上．

2 SmartSifter の概要

SmartSifter は，基本的に統計的外れ値検出を行うアルゴリズムであり，データが従う確率分布を学習 (推定) し，学習された分布に基づいてスコア (外れ値の度合い) を計算する．ただし，多量のデータを処理できるように，基本的にオンラインで動作し，計算時間がデータ数について線形になるように設計されている．ここでいうスコアは，大きいほど統計的に珍しい値であることを示す指標である．

表 1 に出力例を示す．これは，SmartSifter によって (service,duration,src_bytes,dst_bytes) という四つの属性をもつデータにスコアを付け，スコアの高い順にソートし，上位 10 件を載せたものである．ラベルのうち 'normal' でないものは，実際に異常値であったことを表している (スコア付けにはラベルを用いていないことに注意)．詳しくは 3 節を参照．

今，$\mathbf{x}$ で d 次元の実数値ベクトルからなるデータを表し，$\mathbf{y}$ で離散値に値を取る属性ベクトルからなるデータを表す．これらの同時確率密度は $p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x}|\mathbf{y})p(\mathbf{y})$ と分解できる．ここで，$p(\mathbf{x}|\mathbf{y})$ は，$\mathbf{y}$ が与えられたもとでの $\mathbf{x}$ の条件付き確率密度を表す．SmartSifter の学習ステップでは，上記の二つの確率密度を，有限個のパラメータで指定されるモデル (パラメトリックモデル) で表現し，そのパラメータを，データ $(\mathbf{x}, \mathbf{y})$ が一つ入力されるごとに更新していく．このときの重要な特徴は，過去のデータの影響が徐々に小さくな

順位	service	duration	src_bytes	dst_bytes	スコア/r^2	ラベル
1	http	5158	89581520	7028652	11.1	normal.
2	http	107	0	0	9.45	normal.
3	http	5046	0	5134218	9.26	normal.
4	smtp	0	224	2776333	8.65	normal.
5	http	0	0	17520	8.60	normal.
6	http	0	0	20595	8.48	normal.
7	ftp	5051	5135678	0	8.16	**warezclient.**
8	ftp	5051	5135678	0	8.12	**warezclient.**
9	http	0	45908	7300	7.91	**back.**
10	http	0	54540	8314	7.88	**back.**

表 1: SmartSifter の出力例

るような更新を行うことである．このことを「過去のデータを忘却していく」と表す．これによって非定常なデータ列にも対応することが出来る．

離散値データ $\mathbf{y}$ の生成モデルとしてはヒストグラム密度を用いる．以下のように，もとの $\mathbf{y}$ の値域をクラスタリングすることにより，推定すべきパラメータの数を減らす．すなわち，クラスタの数を m とし，j 番目のクラスタを A_j と書く．$u = (u_1, ..., u_m)$ を m 次元のベクトルで $\sum_j u_j = 1$, $u_j \geq 0$ を満たすとする．このときヒストグラム密度が表す確率分布を，$\mathbf{y} \in A_j$ のとき $p(\mathbf{y}|u) = u_j/|A_j|$ と定める．

$\mathbf{y}$ が与えられたもとでの $\mathbf{x}$ の確率分布を表すモデルにはガウス密度 (正規分布の密度関数) を有限個重ね合わせて作られる**ガウス混合モデル**を用いる．これは比較的少ないパラメータで複雑な形の密度関数を表現できるモデルである．ここでは，$\mathbf{y}$ と $\mathbf{y}'$ が同一のクラスタに属するならば $p(\mathbf{x}|\mathbf{y}) = p(\mathbf{x}|\mathbf{y}')$ とする．従って，$\mathbf{y} \in A_j$ に対して $p(\mathbf{x}|\mathbf{y}) = p_j(\mathbf{x}|\theta^{(j)})$ と書ける $(j = 1, \cdots, m)$．ここで $\theta^{(j)}$ はガウス混合モデルを指定するパラメータである．今，j 番目のクラスタに関する確率密度が k_j 個のガウス密度の混合であるとすると，それは

$$p_j(\mathbf{x}|\theta^{(j)}) = \sum_{i=1}^{k_j} c_i^{(j)} g(\mathbf{x}|\mu_i^{(j)}, \Lambda_i^{(j)})$$

と表せる．ただし，$g(\mathbf{x}|\mu_i^{(j)}, \Lambda_i^{(j)})$ は平均値が $\mu_i^{(j)}$(d 次元ベクトル)，分散共分散行列が $\Lambda_i^{(j)}$(d 元正定行列) である d 次元ガウス密度を表す．また，$c_i^{(j)}$ は実数で $\sum_{i=1}^{k_j} c_i^{(j)} = 1$ かつ $c_i^{(j)} \geq 0$ を満たすとし，$\theta^{(j)} = (c_1^{(j)}, ..., c_{k_j}^{(j)}, \mu_1^{(j)}, ..., \mu_{k_j}^{(j)}, \Lambda_1^{(j)}, ..., \Lambda_{k_j}^{(j)})$ とおいている．

これらの確率モデルの学習には，ヒストグラムモデル用には SDLE(Sequentially Discounting Laplace Estimating) アルゴリズムを，ガウス混合モデル用には SDEM(Sequentially Discounting EM) アルゴリズムをそれぞれ用いる．いずれのアルゴリズムも，先に述べた忘却機能をもつところに特徴がある．また，過去のデータを蓄えておく必要がないという意味でオンライン性をもったアルゴリズムになっている．図 1 は処理の流れを表している．ここで，$p^{(t)}$ を t 番目のデータまで処理した結果得られた確率密度とする．t 番目の入力データのスコアには，$p^{(t-1)}$ と $p^{(t)}$ の間の Hellinger 距離 (例えば [13]) を用いる．これはデータ $(\mathbf{x}_t, \mathbf{y}_t)$ によってモデルを更新するときの変化量を表す．つまり学習の結果，モデルを大きく変化させたデータ程スコアが大きいと見なすのである．SDLE アルゴリズムと SDEM アルゴリズムの詳細は [11, 14, 12] を参照して頂きたいが，後

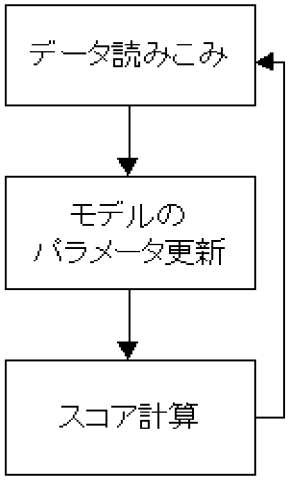

図 1: SmartSifter の処理の流れ

者を一般的な形で紹介しておく．

今，パラメトリックモデル $p(w,z|\theta)$(確率密度)において，w は観測出来ない隠れ変数であるとする．ガウス混合モデルの場合，$\mathbf{x}$ がどの正規分布から発生したのかを示す変数に相当する．z の確率密度は $p(z|\theta) = \int p(w,z|\theta)dw$ である．ここでの我々の目標は，$z_1, z_2, ...$ とデータが次々に与えられる状況下で，オンラインでパラメータ θ を推定することである．

今，r を $0 < r < 1$ を満たす定数 (0 に近いとする) とし，$p(w|z,\theta) \stackrel{\text{def}}{=} p(w,z|\theta)/p(z|\theta)$ と置く．以下に SDEM アルゴリズムを与える．

初期化: $\theta^{(0)}$ を適当な値にセットして，$s := 1$, $\tilde{f}_0(w^0,\theta) \equiv 1$ とする．

E-step: データ z_s を読み込む．$\theta^{(s-1)}$ が与えられたもとで以下の関数を定義する．

$$\tilde{f}_s(w^s,\theta) := (1-r)\tilde{f}_{s-1}(w^{s-1},\theta) + r\int p(w|z_s,\theta^{(s-1)})\log p(w,z_s|\theta)dw$$

M-step: 推定値 $\theta^{(s)}$ を

$$\theta^{(s)} := \arg\max_{\theta} \tilde{f}_s(w^s,\theta).$$

で計算し，$s := s+1$ として E-step に戻る．

このアルゴリズムは，インクリメンタル EM アルゴリズム [9] の変形とみなせるが，オンラインの状況で一つのデータに対し一度だけイタレーションを行うところに特徴があり，これは高速性に寄与している．

$\theta^{(s)}$ は，時点 t での統計量

$$r(1-r)^{s-t}\int p(w|z_t,\theta^{(t-1)})\log p(w,z_t|\theta)dw$$

の (t についての) 総和を最大にする θ である．ここで，ある固定した t についての項は，s が大きくなるにつれて $1-r$ の指数オーダで寄与が小さくなり，忘却が行われる．

一般の SDEM アルゴリズムでは，関数 $\tilde{f}_s$ を記憶するためにデータ数に比例する大きさのメモリが必要だが，モデルが指数型分布族やその有限混合である場合 (例えばガウス混合分布) は，十分統計量を用いることで，有限のメモリで済むように変更出来る．[11, 14, 12] ではそうした形のアルゴリズムを採用している．

3 ネットワーク侵入検出

SmartSifter をネットワーク侵入検出問題に適用した例を示す．データとして KDD Cup 1999 で使われたもの [5] を用いた．本実験の目的はラベルを教師情報として用いずに，オンラインでできるだけ多くの侵入データを検出することである．KDD Cup 1999 では，教師つき侵入検出問題が扱われたため，データのラベルが学習段階で利用されたが，本実験ではラベルは SmartSifter の性能の評価時にしか用いていない．

各データは 34 個の連続値属性と 7 個のカテゴリカル属性からなる．各々にはラベル (22 種類: normal, back, buffer_overflow, ftp_write, etc.) がつけられている．ここで 'normal' 以外の全てのラベルは「攻撃」を表す．実験では全属性のうち (service, duration, src_bytes, dst_bytes) の 4 つを用いた．これらを用いた理由は最も基本的な属性

であり，他の多くはこれらを人為的に合成したものだからである．上記4属性のうち‘service’だけがカテゴリカル属性で，その他は全て連続値属性である．変数serviceは，http, smtp, finger,... 等の35の値を取る．これを{http, smtp, ftp, ftp_data, その他}に手動でクラスタリングした．また，連続値属性はいずれも0付近に集中する傾向があるため，$y = \log(x + 0.1)$ なる変換を施した．

元のデータセットは4,898,431個のデータ中，3,925,651件の攻撃(80.1%)を含んでいた．この割合は外れ値検出としては多すぎるので，属性*logged_in*が正であるものだけを取り出すことにより976,157件のデータ(うち攻撃は3,377件(0.35%))からなる部分集合SFを生成した．ここで*logged_in*が正であるデータを「**侵入**」とよぶ．さらにSFから約50万件のデータをランダムサンプリングにより取りだしてSF50というデータセットを作成し，これにSmartSifterを適用した．

SF50の最初の30,000件は学習のみに用いてスコアを付けなかった．SF50はサンプリングのとり方を変えることにより2通り用意した．一方に含まれる侵入の数は1,687件，もう一方は1,685件であった．表1に，SmartSifterが付けたスコアの高い順にデータをソートした場合に，上位‘top ratio’%のデータの中に含まれていた侵入の数と，全侵入数に対するその割合(カバー率)を示した．この表から，

top ratio	含まれていた侵入数 (カバー率)	
1%	922 (55%)	968 (57%)
3%	1282 (76%)	1260 (75%)
5%	1391 (82%)	1284 (76%)
10%	1617 (96%)	1576 (94%)

表 2: **SF50の中に見付かった侵入データの数**

例えば全侵入のおよそ80%を検出するために，ランダムに抽出した場合は全体の80%のデータを調べる必要があるのに対し，SmartSifterによるスコアで優先順位を付けて調べれば，全体の5%のデータを調べれば済むことが示唆される．この値はBurge and Shawe-Taylorの方法[3]を遥かに凌駕するものである[12]．計算時間はPentinumIII550MHzのマシンを用いておよそ1500秒であり，実用的な値である．

4 おわりに

SmartSifterは不審な医療行為の検出にも適用できる．実際，筆者らはCSIRO(豪国立研究機関)との共同研究[11]を通じて豪健康保険委員会HIC(Health Insurance Comission)提供の国民健康保険データからSmartSifterによって不審な医療サービスデータを検出することに成功した．医療サービスデータには様々な医療検査の記録があり，検出されたデータの中には一見して偏った医療検査を行っているものが多かった．このような不審医療行為の検出は150件の医療機関，2万人の医師，400万人の患者のデータに渡って行われた．HICのWarwick博士はSmartSifterが国民健康保険の運営計画を考える上で役に立つであろうと述べている．

SmartSifterの応用分野は多岐にわたると期待できる．金融，損保，通信，製造業での不正検出はもちろんのこと，例えば流通業界や広告業界などでも，トレンド検出や最先端の流行の検出などにSmartSifterが適用できる可能性がある．

参考文献

[1] Barnett and Lewis, *Outliers in Statistical Data*, John Wiley & Sons, 1994.

[2] Bonchi, Giannotti, Mainetto, and Pedeschi, “A classification-based methodology for planning audit strategies in fraud detection”, *Proc. of KDD-99*, pp. 175-184, 1999.

[3] Burge and Shawe-Taylor, “Detecting cellular fraud using adaptive prototypes”, *Proc. of AI Approaches to Fraud Detection and Risk Management*, pp. 175-184, 1997.

[4] Fawcett and Provost, “Combining data mining and machine learning for effective fraud detection”, *Proc. of AI Approaches to Fraud Detection and Risk Management*, pp. 14-19, 1997.

[5] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[6] Knorr and Ng, "Algorithms for mining distance-based outliers in large datasets", *Proc. of the 24th VLDB Conference*, pp. 392–403, 1998.

[7] Lee, Stolfo and Mok, "Mining in a data-flow environment: experience in network intrusion detection", *Proc. of KDD-99*, pp. 114–124, 1999.

[8] Y. Moreau and J. Vandewalle, Detection of mobile phone fraud using supervised neural networks: a first prototype, ftp://ftp.esat.kuleuven.ac.jp/pub/SISTA/moreau/reports/icann97_TR97-44.ps.

[9] Neal and Hinton, A view of the EM algorithm that justifies incremental, sparse, and other variants, ftp://ftp.cs.toronto.edu/pub/radford/emk.pdf, 1997.

[10] Rosset, Murad, Neumann, Idan and Pinkas, Discovery of fraud rules for telecommunications-challenges and solutions, in *Proc. of KDD-99*, pp. 409-413, 1999.

[11] Yamanishi, Takeuchi, Williams and Milne, "On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms", *Proc. of KDD2000*, pp. 320-324, 2000.

[12] 竹内，山西，"外れ値検出エンジン SmartSifter の実験的評価", **第 23 回情報理論とその応用シンポジウム予稿集**, pp. 419-422, 2000.

[13] 竹内 啓編, **統計学辞典**, 東洋経済新報社, 1989.

[14] 山西，竹内, "オンライン忘却型学習アルゴリズムを用いた統計的外れ値検出", **第 3 回情報論的学習理論ワークショップ予稿集**, pp. 227-232, 2000.