2007 年情報論的学習理論ワークショップ
2007 Workshop on Information-Based Induction Sciences (IBIS2007)
Tokyo, Japan, November 5 - 7, 2007.

密度比推定を用いた特異点検出手法
Novelty Detection by Densitiy Ratio Estimation

比戸将平* 坪井祐太* 鹿島久嗣* 杉山将†
Shohei Hido Yuta Tsuboi Hisashi Kashima Masashi Sugiyama

Abstract: Novelty detection is a problem of finding irregular data in the test set based on the training set consisting of regular data. We propose a new statistical approach to novelty detection. Our key idea is that we do not directly model the training and test density functions, but we only estimate the ratio of the densities (i.e., importance). This allows us to avoid solving an unnecessarily difficlut problem of density estimation and therefore our approach is expected to have better performance. Indeed, simulations with benchmark data sets show that the proposed approach is promising in novelty detection.

Keywords: novelty detection, density estimation, one-class classification, importance, covariate shift

1 はじめに

特異点検出 (novelty detection) とは，正常なデータから成る訓練データ集合を用いて，検証データ集合に含まれる異質なデータ (特異点) を発見する問題である．その応用は極めて多岐に渡り，例えば，ネットワークの不正侵入検出，ニュース文書の新規トピック検出，機械システムの異常検知などが挙げられる．状況や設定によっては外れ値検出 (outlier detection) や異常値検出 (anomaly detection) と呼ばれることもあり，その重要性から，統計学や学習の分野では古くから特異点検出に関する数多くの研究が行われてきた [5].

特異点に関する前提知識を持たない場合，この問題はクラスタリング等と同様，教師無し学習に分類される．一般に，それらデータのみに基づく問題は，真の分布の確率密度関数 $p(\boldsymbol{x})$ を知ることができれば容易に解ける．特異点検出であれば，$p(\boldsymbol{x})$ の値が小さい領域に存在する検証データを探すことに相当する．このように正確な確率密度推定 (density estimation) は全ての教師無し学習問題にとって極めて有用である [9]．しかしながら，密度推定を有意な精度で行えるのは，データが低次元かつ単純な分布に従う場合に限られ，実世界のデータは高次元かつ異種混合であるため，これらの仮定が成り立つことはほとんど無い．

密度関数の推定を避けた特異点検出手法としては One-class SVM [9] や Support Vector Data Description [14] などが存在する．それらの手法は密度そのものではなく定められた割合の訓練データを含む体積最小の高密度領域を，カーネル関数によって射影された特徴空間において推定する．そして学習した高密度領域の境界の外に存在するデータは特異点と判定される．これらの手法の性能はカーネルパラメータの値に依存するが，これまでのところカーネルパラメータの適切な決定法は知られていないようである．

本論文では，サンプリングで用いられる「重要度 (importance)」の考え方を，特異点検出に応用することで，新しい特異点検出の手法を提案する．具体的には，訓練データと検証データの 2 つのデータ集合が与えられているような場合に，2 つの確率密度の比を用いて特異点検出を行う方法を提案する．直感的には，特異点においては訓練データの確率密度よりも検証データの確率密度のほうが大きい ($p_{tr}(\boldsymbol{x}) < p_{te}(\boldsymbol{x})$) ため，この密度比を特異スコアとすることにより，検証データに含まれる特異点を検出できると考えられる．しかしながら，前述したように，特異スコアの分母と分子に現れる確率密度関数を直接推定するのは困難であるため，このスコアを，密度推定を避けながら直接的に求めることが必要となる．そこで我々は，近年非常に精力的に研究されている，共変量シフト (covariate shift) [10, 12] と呼ばれる，訓練

*日本アイー・ビー・エム (株) 東京基礎研究所, 〒 242-8502 神奈川県大和市下鶴間 1623-14. e-mail:{hido,yutat,hkashima}@jp.ibm.com
Tokyo Research Laboratory, IBM Research, 1623-14 Shimotsuruma, Yamato-shi, Kanagawa, 242-8502 Japan.
†東京工業大学大学院情報理工学研究科, 〒 152-8552 東京都目黒区大岡山 2-12-1-W8-74. e-mail:sugi@cs.titech.ac.jp
Department of Computer Science, Tokyo Institute of Technology, 2-12-1-W8-74, O-okayama, Meguro-ku, Tokyo, 152-8552 Japan.

データと検証データの入力分布に違いが生じているような場合の教師付き学習問題における適応手法を用いる．そこでは各点における分布 $p_{tr}(\boldsymbol{x})$ と $p_{te}(\boldsymbol{x})$ の比を訓練データの重要度として重み付け補正を行うことにより，分布の違いをうまく吸収し適切な学習を行えることが知られているが，この重要度を，密度推定を避けながら直接的に求めるアプローチがいくつか提案されている．例えば，KMM(kernel mean matching) [6] では，ガウス基底によって張られる特徴空間において，2 つのデータ集合のサンプル平均の差が最小となる重要度を導き出す．また最新の研究として，杉山らはこの密度比を直接推定することにより，共変量シフトに良く適応した学習を行うアルゴリズム KLIEP(Kullback-Leibler Importance Estimation Procedure) を提案している [13].

KLIEP には交差検定によってモデルを選択できるという大きな利点があることから，本論文では KLIEP アルゴリズムを採用し，重要度と同様に特異スコアを推定する手法を提案する．これは困難な確率密度推定を直接行うことなく，特異点検出に十分でありつつより容易な問題を解いていることに相当する．

最後に人工データとベンチマークデータを用いた実験によって，我々の手法は特異スコアを適切に評価し，他の手法と比較しても優れた検出力を持つことを示す．

2 特異スコアの推定手法

本節では，重要度を用いた特異点検出法を定式化するとともに，特異スコアの推定手法である KLIEP [13] の概要を述べる．

2.1 問題設定

$\mathcal{D}$ ($\subset \mathbb{R}^d$) を入力データ領域として，訓練入力分布の密度関数 $p_{tr}(\boldsymbol{x})$ に従がって i.i.d でサンプリングされた訓練データ集合 $\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ と，同様に検証入力分布の密度関数 $p_{te}(\boldsymbol{x})$ から i.i.d に生成された検証データ集合 $\{\boldsymbol{x}_i^{te}\}_{i=1}^{n_{te}}$ を与えられたとする．ここで全ての $\boldsymbol{x} \in \mathcal{D}$ において $p_{tr}(\boldsymbol{x}) > 0$ と $p_{te}(\boldsymbol{x}) > 0$ が成り立つとする．

一般に，特異点は訓練データがあまり現れない領域に存在するデータと考えられる．よって，訓練入力データ分布 $p_{tr}(\boldsymbol{x})$ を求め，その値が小さい領域に存在する検証データを特異点と判断するのが自然である．しかしながら，訓練データ集合 $\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ を与えられた時にその密度関数 $p_{tr}(\boldsymbol{x})$ を正確に求めることは，正規分布などの単純な分布に従わない高次元なデータにおいて非常に困難であることが知られている [5].

そこで我々は，$\boldsymbol{x}$ における訓練入力データ分布と検証入力データ分布の密度関数の比:

$$w(\boldsymbol{x}) = \frac{p_{tr}(\boldsymbol{x})}{p_{te}(\boldsymbol{x})}$$

を特異スコアと考えることにする[1]．この場合，特異点である可能性が高いほど $w(\boldsymbol{x})$ の値は小さいことに注意する．実世界においては，真の特異点として与えられるデータが無いため，特異点であると判断する基準となる閾値を定めることは実用上あまり重要ではない．そこで本論文では，閾値に寄らない評価方法として，特異スコア $w(\boldsymbol{x})$ の小さい順に特異点らしいと判断し，その有意性を AUC(Area Under ROC Curves) の値で評価する [2].

本論文で扱う特異点検出問題を整理する．$\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ と $\{\boldsymbol{x}_i^{te}\}_{i=1}^{n_{te}}$ に基づいて $w(\boldsymbol{x})$ を学習し，それを用いて各検証データサンプル $\boldsymbol{x}_j^{te}$ における特異スコア $w(\boldsymbol{x}_i^{te})$ を評価する．ここで必要とされるのは，$p_{tr}(\boldsymbol{x})$ と $p_{te}(\boldsymbol{x})$ を直接推定するという極めて困難な問題を回避しつつ，効率よく密度比 $w(\boldsymbol{x})$ のみを求める手法である．

2.2 KLIEP アルゴリズム

特異スコア $w(\boldsymbol{x})$ を以下の線形モデルによりモデル化する．

$$\widehat{w}(\boldsymbol{x}) = \sum_{\ell=1}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x}), \tag{1}$$

$\{\alpha_\ell\}_{\ell=1}^b$ が学習すべきパラメータであり，$\{\varphi_\ell(\boldsymbol{x})\}_{\ell=1}^b$ は次の条件を満たす基底関数である:

$$\varphi_\ell(\boldsymbol{x}) \geq 0 \text{ for all } \boldsymbol{x} \in \mathcal{D} \text{ and for } \ell = 1, 2, \ldots, b.$$

本論文では基底関数 $\{\varphi_\ell(\boldsymbol{x})\}_{\ell=1}^b$ として $\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ と $\{\boldsymbol{x}_i^{te}\}_{i=1}^{n_{te}}$ に基づいたカーネルを用いるが，そのモデル選択の詳細については 2.3 節で説明する．

学習されたモデル $\widehat{w}(\boldsymbol{x})$ を用いると，訓練入力データ分布 $p_{tr}(\boldsymbol{x})$ を次式で推定できる:

$$\widehat{p}_{tr}(\boldsymbol{x}) = \widehat{w}(\boldsymbol{x}) p_{te}(\boldsymbol{x}).$$

モデル (式 (1)) のパラメータ $\{\alpha_\ell\}_{\ell=1}^b$ は次式の $p_{tr}(\boldsymbol{x})$ から $\widehat{p}_{tr}(\boldsymbol{x})$ への Kullback-Leibler 情報量を最小化するように定める:

$$KL[p_{tr}(\boldsymbol{x}) \| \widehat{p}_{tr}(\boldsymbol{x})] = \int_{\mathcal{D}} p_{tr}(\boldsymbol{x}) \log \frac{p_{tr}(\boldsymbol{x})}{\widehat{w}(\boldsymbol{x}) p_{te}(\boldsymbol{x})} d\boldsymbol{x}$$
$$= \int_{\mathcal{D}} p_{tr}(\boldsymbol{x}) \log \frac{p_{tr}(\boldsymbol{x})}{p_{te}(\boldsymbol{x})} d\boldsymbol{x} - \int_{\mathcal{D}} p_{tr}(\boldsymbol{x}) \log \widehat{w}(\boldsymbol{x}) d\boldsymbol{x}.$$

[1] この密度比は共変量シフト下の重要度の定義の逆数となっており，KLIEP を提案した元の論文 [13] においてその推定手法は iKLIEP(inverse KLIEP) と記述されている．しかしここでは表記を簡単にするため，それらを区別せず特異点検出向けの KLIEP として説明することにする．iKLIEP を採用する理由は，モデル選択において分子側のデータが多いことが好ましい点にある．特異点検出問題においては通常，訓練入力データ数 n_{tr} が検証入力データ数 n_{te} に比べ大きい．

Input: $m = \{\varphi_\ell(\boldsymbol{x})\}_{\ell=1}^{b}$, $\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$, $\{\boldsymbol{x}_i^{te}\}_{i=1}^{n_{te}}$
Output: $\widehat{w}(\boldsymbol{x})$

$\boldsymbol{A}_{j,\ell} \longleftarrow \varphi_\ell(\boldsymbol{x}_j^{tr})$ for $j = 1, 2, \ldots, n_{tr}$ and $\ell = 1, 2, \ldots, b$;
$\boldsymbol{b}_\ell \longleftarrow \frac{1}{n_{te}} \sum_{i=1}^{n_{te}} \varphi_\ell(\boldsymbol{x}_i^{te})$
Initialize $\boldsymbol{\alpha}$ ($> \boldsymbol{0}$) and ε ($0 < \varepsilon \ll 1$);
Repeat until convergence
  $\boldsymbol{\alpha} \longleftarrow \boldsymbol{\alpha} + \varepsilon \boldsymbol{A}^\top (\boldsymbol{1}./\boldsymbol{A}\boldsymbol{\alpha})$; % Gradient ascent
  $\boldsymbol{\alpha} \longleftarrow \boldsymbol{\alpha} + (1 - \boldsymbol{b}^\top \boldsymbol{\alpha}) \boldsymbol{b} / (\boldsymbol{b}^\top \boldsymbol{b})$;
  $\boldsymbol{\alpha} \longleftarrow \max(\boldsymbol{0}, \boldsymbol{\alpha})$;
  $\boldsymbol{\alpha} \longleftarrow \boldsymbol{\alpha} / (\boldsymbol{b}^\top \boldsymbol{\alpha})$;
end
$\widehat{w}(\boldsymbol{x}) \longleftarrow \sum_{\ell=1}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x})$;

図 1: KLIEP メインコード

Input: $\mathcal{M} = \{m | m = \{\varphi_\ell^m(\boldsymbol{x})\}_{\ell=1}^{b_m}\}$, $\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$, $\{\boldsymbol{x}_i^{te}\}_{i=1}^{n_{te}}$
Output: $\widehat{w}(\boldsymbol{x})$

Split $\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ into R disjoint subsets $\{\mathcal{X}_r\}_{r=1}^{R}$;
for each model $m \in \mathcal{M}$
  for each split $r = 1, 2, \ldots, R$
    $\widehat{w}_r(\boldsymbol{x}) \longleftarrow$ KLIEP$(m, \{\boldsymbol{x}_i^{te}\}_{i=1}^{n_{te}}, \{\mathcal{X}_j\}_{j \neq r})$;
    $\widehat{J}_r(m) \longleftarrow \frac{1}{|\mathcal{X}_r|} \sum_{\boldsymbol{x} \in \mathcal{X}_r} \log \widehat{w}_r(\boldsymbol{x})$;
  end
  $\widehat{J}(m) \longleftarrow \frac{1}{R} \sum_{r=1}^{R} \widehat{J}_r(m)$;
end
$\widehat{m} \longleftarrow \mathrm{argmax}_{m \in \mathcal{M}} \widehat{J}(m)$;
$\widehat{w}(\boldsymbol{x}) \longleftarrow$ KLIEP$(\widehat{m}, \{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}, \{\boldsymbol{x}_i^{te}\}_{i=1}^{n_{te}})$;

図 2: LCV 法によるモデル選択

最後の式の第 1 項は $\{\alpha_\ell\}_{\ell=1}^{b}$ と無関係であるため無視し，第 2 項のみに重点を置き，J と定義する:

$$\begin{aligned} J &= \int_{\mathcal{D}} p_{tr}(\boldsymbol{x}) \log \widehat{w}(\boldsymbol{x}) d\boldsymbol{x} \approx \frac{1}{n_{tr}} \sum_{j=1}^{n_{tr}} \log \widehat{w}(\boldsymbol{x}_j^{tr}) \\ &= \frac{1}{n_{tr}} \sum_{j=1}^{n_{tr}} \log \left(\sum_{\ell=1}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x}_j^{tr}) \right). \end{aligned}$$

ここで，真の訓練データ分布に関する期待値を，訓練データ集合 $\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ の経験分布に関する期待値で近似している．この J がパラメータ $\{\alpha_\ell\}_{\ell=1}^{b}$ に関して最大化すべき目的関数であり，上に凸である [1]．この目的関数は訓練データ集合 $\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ のみを含んでいることに留意する．つまり検証データ集合 $\{\boldsymbol{x}_i^{te}\}_{i=1}^{n_{te}}$ はここまで利用していないが，以下に示すように制約条件の中で用いる．

$\widehat{w}(\boldsymbol{x})$ は特異スコア $w(\boldsymbol{x})$ の推定値であり，密度比としての定義から非負である．それゆえ，全ての $\boldsymbol{x} \in \mathcal{D}$ に対して $\widehat{w}(\boldsymbol{x}) \geq 0$ を保証しなければならない．この条件は最適化問題に以下の制約を入れることで満たされる:

$$\alpha_\ell \geq 0 \text{ for } \ell = 1, 2, \ldots, b.$$

この非負制約に加え，$\widehat{p}_{tr}(\boldsymbol{x})$ ($= \widehat{w}(\boldsymbol{x}) p_{te}(\boldsymbol{x})$) が確率密度関数となるために次の等式を満たす必要がある:

$$\begin{aligned} 1 &= \int_{\mathcal{D}} \widehat{p}_{tr}(\boldsymbol{x}) d\boldsymbol{x} = \int_{\mathcal{D}} \widehat{w}(\boldsymbol{x}) p_{te}(\boldsymbol{x}) d\boldsymbol{x} \\ &\approx \frac{1}{n_{te}} \sum_{i=1}^{n_{te}} \widehat{w}(\boldsymbol{x}_i^{te}) = \frac{1}{n_{te}} \sum_{i=1}^{n_{te}} \sum_{\ell=1}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x}_i^{te}). \quad (2) \end{aligned}$$

ここでも，真の検証データ分布に関する期待値を，検証データ集合 $\{\boldsymbol{x}_i^{te}\}_{i=1}^{n_{te}}$ の経験分布に関する期待値で近似している．近似誤差により，通常この正規化制約 (式 (2)) は完全には保証されない．しかし，特異スコアのスケールは特異点の検出力に無関係であるので，実際上大きな問題にはならないと考えられる．

以上から得られた最適化規準をここにまとめる．

$$\begin{aligned} \max_{\{\alpha_\ell\}_{\ell=1}^{b}} & \sum_{j=1}^{n_{tr}} \log \left(\sum_{\ell=1}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x}_j^{tr}) \right) \\ \text{subject to } & \sum_{i=1}^{n_{te}} \sum_{\ell=1}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x}_i^{te}) = n_{te} \\ & \text{and } \alpha_1, \alpha_2, \ldots, \alpha_b \geq 0. \end{aligned}$$

この最適化問題は凸であり，単純な最急上昇法 (gradient ascent) と制約充足との繰り返しによって最適解を計算可能である[2]．さらに，求められる最適解 $\{\widehat{\alpha}_\ell\}_{\ell=1}^{b}$ は疎になりやすいため，検証時の計算コストは低く抑えられる [1]．以上の手法をまとめて，*Kullback-Leibler Importance Estimation Procedure* (KLIEP) と呼ぶ [13]．KLIEP の仮想コードを図 1 に示した．

2.3 尤度交差判定によるモデル選択

KLIEP による特異点検出の性能は，基底関数 $\{\varphi_\ell(\boldsymbol{x})\}_{\ell=1}^{b}$ の選択に依存する．本節ではそれらをデータサンプルからどのように適切に選ぶかを述べる．

密度推定における標準的なモデル選択手法の 1 つに，尤度交差検定 (LCV: Likelihood Cross Validation) がある [3]．まず一般のサンプル集合 $\{\boldsymbol{x}_k\}_{k=1}^{n}$ からその確率密度関数 $p(\boldsymbol{x})$ を推定する場合を考える．何らかのモデル選択パラメータが与えられた場合，LCV 法では $\{\boldsymbol{x}_k\}_{k=1}^{n}$ の部分集合を用いて密度推定を行い，その結果 $\widehat{p}(\boldsymbol{x})$ の尤度を，推定に用いられなかった残りの部分集合から見積

[2] 必要に応じて，ペナルティ項を目的関数に加えるか，最適解の値に上界を設定することで問題を正則化できる．

もる．この作業をさまざまなモデルに対して繰り返し，尤度最大となるモデルを選択する．このとき，LCV 法は $p(\boldsymbol{x})$ から $\widehat{p}(\boldsymbol{x})$ までの Kullback-Leibler 距離を最小化するモデルを選ぶことに対応している．

同様にして，KLIEP にも LCV 法を適用する方法を説明する．KLIEP は J の値 (式 (2)) の最大化に基づいているため，J が最大となるようなモデルを選ぶことが適切である．ここで密度推定における LCV 法の考え方を応用することにより，J に含まれる $p_{tr}(\boldsymbol{x})$ に関する期待値は次のように数値的に近似できる．まず訓練データ集合 $\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ を，R 個の重なりのない部分集合 $\{\mathcal{X}_r^{tr}\}_{r=1}^{R}$ に分割する．そして推定特異スコア $\widehat{w}_r(\boldsymbol{x})$ を $\{\mathcal{X}_j^{tr}\}_{j\neq r}$ と $\{\boldsymbol{x}_i^{te}\}_{i=1}^{n_{te}}$ から求め，$\mathcal{X}_r^{tr}$ を以下のように用いて J の値を近似する:

$$\widehat{J}_r = \frac{1}{|\mathcal{X}_r^{tr}|} \sum_{\boldsymbol{x}\in\mathcal{X}_r^{tr}} \log \widehat{w}_r(\boldsymbol{x}).$$

この過程を $r = 1, 2, \ldots, R$ について繰り返し，$\widehat{J}_r$ の全ての r に関する平均値 $\widehat{J}$ が最大となるモデルを選択する．LCV 法の仮想コードを図 2 に示す．この他に，明示的なモデル候補ではなくモデルの条件を与え，その範囲内で J の値を LCV 法で推定しながら何らかの最適化を行うアルゴリズムも考えられる．

通常の交差検定における潜在的な弱点の 1 つとして，データ集合を分割することにより，サンプル数が減少し信頼性が低下してしまうことが挙げられる．しかし，検証データでなく訓練データが分割されることと，特異点検出問題では通常は十分な数の訓練データが与えられることから，上記の交差検定法ではサンプル数不足の問題は発生しないと考えられる．

この LCV 法による交差検定を用いることで，モデル候補の集合の中から最も良いものが選択されることが期待される．モデル候補としては，訓練データ集合 $\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ を中心にしたガウシアンカーネルを用いることにする:

$$\widehat{w}(\boldsymbol{x}) = \sum_{\ell=1}^{n_{tr}} \alpha_\ell K_\sigma(\boldsymbol{x}, \boldsymbol{x}_\ell^{tr}).$$

ここで $K_\sigma(\boldsymbol{x}, \boldsymbol{x}')$ は幅 σ をパラメータに持つガウシアンカーネルとする:

$$K_\sigma(\boldsymbol{x}, \boldsymbol{x}') = \exp\left\{-\frac{\|\boldsymbol{x}-\boldsymbol{x}'\|^2}{2\sigma^2}\right\}. \tag{3}$$

検証データ集合 $\{\boldsymbol{x}_i^{te}\}_{i=1}^{n_{te}}$ でなく訓練データ集合 $\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ をカーネル中心に選ぶ理由を以下に述べる．定義より，特異スコア $w(\boldsymbol{x})$ は訓練データ密度 $p_{tr}(\boldsymbol{x})$ が大きく検証入力データ密度 $p_{te}(\boldsymbol{x})$ が小さい領域において大きな値を取る．反対に，訓練データ密度 $p_{tr}(\boldsymbol{x})$ が小さく検証データ密度 $p_{te}(\boldsymbol{x})$ が大きければ $w(\boldsymbol{x})$ は 0 に近づく．ガウシアンカーネルモデルで関数を近似する場合，対象の関数が大きな値を取る領域周辺においてカーネルをより多く必要とすると考えられる．一方，関数の値が 0 に近い領域は少ないカーネルによる近似で十分である．この経験則に従って，訓練データ集合 $\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ をカーネル中心とすることによって訓練データ密度 $p_{tr}(\boldsymbol{x})$ が大きい領域により多くのカーネルを配置することにする．

あるいは，$\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ と $\{\boldsymbol{x}_i^{te}\}_{i=1}^{n_{te}}$ の両方に $(n_{te}+n_{tr})$ 個のカーネル中心を置くことも考えられる．しかしながら，$w(\boldsymbol{x})$ の値が小さな領域に多く存在する $\{\boldsymbol{x}_i^{te}\}_{i=1}^{n_{te}}$ を用いることは，その効果に比べて計算量増加の負担が大きい．さらに，特異点検出では通常 n_{tr} が大きいので，訓練データ集合 $\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ 全てをカーネル中心とすることも非常に計算量が大きい．そこで，計算量を低減するために，$\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ の部分集合のみをカーネル中心とすることにする:

$$\widehat{w}(\boldsymbol{x}) = \sum_{\ell=1}^{b} \alpha_\ell K_\sigma(\boldsymbol{x}, \boldsymbol{c}_\ell). \tag{4}$$

$\boldsymbol{c}_\ell$ は b $(\leq n_{tr})$ を定数として $\{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}}$ からランダムに b 個選んだ代表点である．この後の実験においては，代表点の数 b を n_{tr} より小さい値に置き換え，カーネル幅 σ は上記の尤度交差検定法によって最適なものを選ぶ．

3 KLIEP による特異スコアの推定例

実例を用い，KLIEP を用いた特異スコアの推定の様子を示す．用いるのは図 3 に表した単純な 1 次元の分布 2 種類である．各分布から 100 個ずつデータを生成し，それぞれ訓練用と検証用のデータ集合とした．訓練データ集合は単純なガウシアン $\mathcal{N}(0, 2)$ にしたがっている一方，検証データ集合には $\mathcal{N}(10, 2)$ という中心の異なるガウシアンが混在している．ガウシアンカーネルの幅は $\{0.5, 1, 2.5, 5, 10, 25, 50\}$ を候補とする．図 4 は真の J の値と，LCV 法による推定値 $\hat{J}$ を表している．実際に KLIEP を用いて推定した特異スコアの結果を図 5 に示した．

これによると，$\boldsymbol{x} = 0$ 周辺に比べ，$\boldsymbol{x} = 10$ 周辺の特異スコアの値が正しく小さくなっていることがわかる．特に，LCV 法によって求められた $\sigma = 5$ の場合は真の値 $w(\boldsymbol{x})$ を表す曲線に近い，高精度の推定値が得られていることがわかる．一方 $\sigma = 0.5$ と $\sigma = 50$ のプロットから，カーネルの幅が小さいと推定値が一部不安定になり，幅が大きいと全体が平滑化されることがわかる．このように，LCV 法によって適切なカーネル幅を選択できることは提案手法の大きな利点と考えられる．

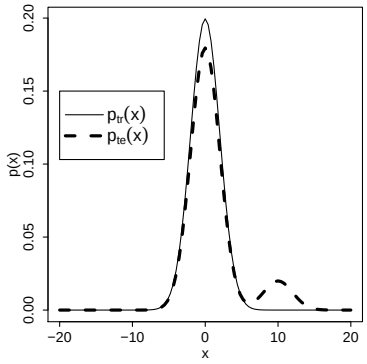

図 3: 特異点を含むデータセット例

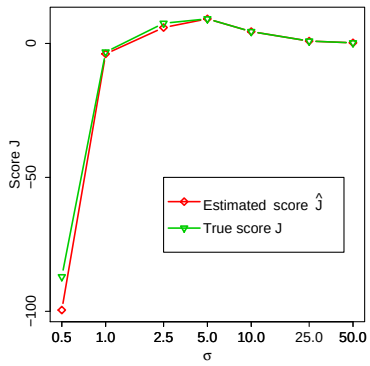

図 4: LCV 法によるモデル選択

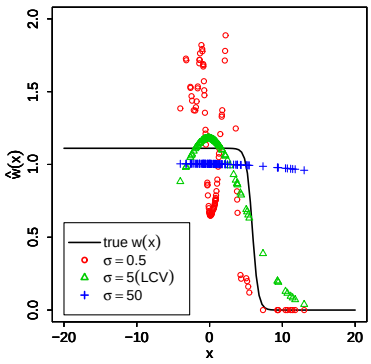

図 5: KLIEP による特異スコア推定

4 関連研究

本節では，既存の密度比推定手法，及び特異点検出手法と KLIEP の関連について述べる．

4.1 Kernel Mean Matching

Kernel mean matching(KMM) は密度推定を避けながら直接的に訓練データ集合の重要度を予測する手法である [6]．KMM の基本的な考え方は，$p_{tr}(\boldsymbol{x})$ と $\widehat{w}(\boldsymbol{x})p_{te}(\boldsymbol{x})$ から得られた 2 つのサンプル集合に，ある特徴空間 $\mathcal{F}$ における非線形変換 f を適用した上の平均値の 2 乗誤差の上限を最小化するような $\widehat{w}(\boldsymbol{x})$ を求めることである．

$$\min_{w(\boldsymbol{x})} \sup_{f:f\in\mathcal{F},\|f\|_{\mathcal{F}}=1} \left\| \mathop{\mathbb{E}}_{\boldsymbol{x}^{tr}}[f(\boldsymbol{x}^{tr})] - \mathop{\mathbb{E}}_{\boldsymbol{x}^{te}}[w(\boldsymbol{x}^{te})f(\boldsymbol{x}^{te})] \right\|_{\mathcal{F}}^2$$
$$\text{subject to } \mathop{\mathbb{E}}_{\boldsymbol{x}^{te}}[w(\boldsymbol{x}^{te})] = 1$$
$$\text{and } w(\boldsymbol{x}) \geq 0.$$

この問題の解は，$\mathcal{F}$ が普遍再生核ヒルベルト空間 [11] である場合に真の重要度と一致することが示されている [6]．ガウシアンカーネル (式 (3)) は普遍再生核ヒルベルト空間を導くことが知られており，上記の最小化問題の経験近似は次の 2 次計画問題として表現される．

$$\min_{\{w_i\}_{i=1}^{n_{te}}} \left[\frac{1}{2} \sum_{i,i'=1}^{n_{te}} w_i w_{i'} K_\sigma(\boldsymbol{x}_i^{te}, \boldsymbol{x}_{i'}^{te}) - \sum_{i=1}^{n_{te}} w_i \kappa_i \right]$$
$$\text{subject to } \left| \sum_{i=1}^{n_{te}} w_i - n_{te} \right| \leq n_{te}\epsilon$$
$$\text{and } 0 \leq w_1, w_2, \ldots, w_{n_{te}} \leq B.$$

ここで，$\kappa_i = \sum_{j=1}^{n_{te}} K_\sigma(\boldsymbol{x}_i^{tr}, \boldsymbol{x}_j^{te})$ を用いた．調節すべきパラメータは $B\ (\geq 0)$ と $\epsilon\ (\geq 0)$ である．この問題の解 $\{\widehat{w}_i\}_{i=1}^{n_{tr}}$ は訓練入力データ点 $\{\boldsymbol{x}_i^{tr}\}_{i=1}^{n_{tr}}$ における重要度の推定値となっている．

KMM は密度推定を必要としないため，高次元においても有効に働くと思われる．しかしながら，重要度推定が入力訓練データ点においてのみ与えられるため，性能に大きく影響を与える 3 つのパラメータ B，ϵ，σ は交差検定 (CV) などの簡単な方法では最適化できない．それゆえ，交差検定の枠組みで KMM を最適化するには訓練データ集合以外を扱う拡張が必要であるが，現在のところ未解決問題であると思われる．

4.2 One-class Support Vector Machine

教師なし (One-class) であるデータ集合 $\{\boldsymbol{x}_k\}_{k=1}^{n}$ $(= \{\boldsymbol{x}_j^{tr}\}_{j=1}^{n_{tr}} \cup \{\boldsymbol{x}_i^{te}\}_{i=1}^{n_{te}})$ を用い，特徴空間においてある割合 $1-\nu$ 以上のデータを内包する単純な部分領域 S を推定するために，Schölkopf らは One-class SVM を提案した [9]．この問題においても，入力データの真の密度関数を得られれば，部分領域 S を求めることは容易である．しかしここで注目すべきは，密度関数の形が未知であっても，境界さえ探し出せば部分領域 S は導出できることである．この考えに基づき，One-class SVM では割合 $1-\nu$ のデータを含むような領域の中で体積が最小となるような境界を特徴空間の超平面として求める．

One-class SVM の学習は，以下の 2 次計画問題へと帰着することができる．

$$\min_{\{w_k\}_{k=1}^{n}} \frac{1}{2} \sum_{k,k'=1}^{n} w_k w_{k'} K_\sigma(\boldsymbol{x}_k^{tr}, \boldsymbol{x}_{k'}^{tr})$$
$$\text{subject to } \sum_{k=1}^{n} w_k = 1$$
$$\text{and } 0 \leq w_1, w_2, \ldots, w_n \leq \frac{1}{\nu n}.$$

ここで，ν は領域 S 外に位置するデータ割合の上限と，サポートベクトルとなるのデータの割合の下限の両方を表すパラメータである[3]．この問題の解に基づいて，各データ点は超平面よりも原点から近い側にあれば特異点もしくは外れ値と判断される．しかしながら，特異スコアを考えた場合，One-class SVM は明示的にはそのような連続値を出力しない．超平面からの距離を特異スコアとして評価する拡張が考えられるが，それらが有効かどうかは明らかではない．

5 実験

本節では，人工データとベンチマークデータを用いた実験を通して提案法の性能を既存手法と比較し，さらに手書き文字識別問題に対する応用例を示す．比較対象のアルゴリズムとしては，KLIEP と同じく重要度推定手法である KMM [6] と，特異点検出において用いられている One-class SVM [9] を採用する．アルゴリズムは全て統計解析システム R [7] における実装を利用し，実験も全て R 上で行った．

5.1 人工データ

まず簡単な人工データに対する特異点の検出力を示す．2 種類の人工データセットは 2 次元のデータとし，それぞれの次元の値は正規分布から生成する．正常データは両次元とも平均 0，分散 1 の標準正規分布 $\mathcal{N}(0,1)$ に従う．一方，片方の次元を平均をずらした分布 $\mathcal{N}(5,1)$，もしくは分散を大きくした分布 $\mathcal{N}(0,3)$ に変更したものを異常データとする．それぞれ 100 個のサンプルから成る訓練データ集合と検証データ集合全体において，異常データの割合が $\rho = \{0.05, 0.1, 0.15\}$ となる数の異常データを検証データ集合に加えた．例えば $\rho = 0.05$ の場合，検証データ集合は 10 個の異常データを含んでいる．

モデル候補となるガウシアンカーネルの幅として，$\sigma = \{0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50\}$ の 8 種類を用いて実験を行った．ただし，KLIEP は $0.01 \leq \sigma \leq 50$ の範囲で黄金分割法を用いながら LCV 法を適用し，最適と思われるモデル候補を決定した．KMM では文献 [6] に従い $B = 1000$，$\epsilon = (\sqrt{n_{te}} - 1)/\sqrt{n_{te}}$ と固定した．また，One-class SVM にも訓練データ集合中で高密度領域の外にあると判断するデータ割合を調整するパラメータ ν が存在する．ここでは実際の異常データ割合 ρ を ν が上回った場合と下回った場合の性能を確かめるために，$\nu = 0.1$ に固定する．それぞれのアルゴリズムは検証データ集合における特異スコアを，AUC (Area Under ROC Curves) の値で評価した．AUC による評価は，特異スコアのような順序付けが，2 つの集合を少ない誤りで有意に切り分けられるかを判定する Wilcoxon の順位和検定と等価であり [2]，異常データに対する各アルゴリズムの検出力を評価できる．値が 1.0 に近いほど，異常データの特異スコア $w(\boldsymbol{x})$ を正常データよりも正しく小さく推定できることを意味する．

表 1: 人工データセットにおける AUC 値評価

異常データ		KLIEP	KMM		One-class SVM	
分布	ρ	LCV	Best	Worst	Best	Worst
$\mathcal{N}(5,1)$	0.05	1.000	0.999	0.940	1.000	0.593
	0.1	1.000	1.000	0.936	1.000	0.379
	0.15	1.000	1.000	0.946	1.000	0.487
$\mathcal{N}(0,3)$	0.05	0.962	0.957	0.852	0.938	0.593
	0.1	0.773	0.769	0.745	0.754	0.546
	0.15	0.818	0.808	0.756	0.790	0.274
AUC 平均値		0.925	0.922	0.862	0.914	0.479

これらの人工データセットのランダム生成と実験を 100 回ずつ行った結果を表 1 にまとめた．各行はそれぞれ異常データの分布と割合の組み合わせに対応する．各セルの数値は，AUC の平均値を示している．KLIEP の列には，LCV 法によって適切に選択されたモデルによる AUC 値のみを示した．KMM と One-class SVM はモデル選択を行わないため，カーネル幅 σ を変化させ実験を行った後，結果として平均値が最大と最小となったカーネル幅における値を Best と Worst の列に並べた．この図によると，KLIEP や KMM の Best における AUC 値は One-class SVM の Best とほぼ等しく，これら重要度推定の手法が特異点検出に有効であることが示された．One-class SVM はカーネル幅 σ や異常データ割合 ρ によって大きく性能がばらつく事がわかる．特に，訓練データ集合に含まれる異常データの割合が想定より大きい ($\nu < \rho$) 場合においては，AUC 値が 0.5 を下回っており，これは求められた特異スコアと真の特異スコアの大小が逆転してしまっていることを意味する．KMM も，Best においては良い性能を示しているが，そのモデルを前もって選択できないため，最悪の場合は Worst のような性能となってしまう．それらと比較して，KLIEP では安定して高い性能を示すモデルが LCV 法によって自動的に選ばれていることがわかる．実験全体の平均値を比べても KMM と One-class SVM の Best をわずかに上回っており，KLIEP の有用性が明らかとなった．

[3]興味深いことに，One-class SVM の最適化問題は，KMM の最適化問題において $\kappa_i = 0\,(i = 1, 2, \ldots, n_{tr},\ \epsilon = 0,\ B = \frac{1}{\nu n})$ とおいたものと一致する．

表 2: ベンチマークデータセットにおける AUC 値評価

ベンチマーク		KLIEP	KMM		One-class SVM	
データ	ρ	LCV	Best	Worst	Best	Worst
heart	0.05	0.799	0.725	0.512	0.862	0.329
	0.1	0.836	0.742	0.524	0.524	0.149
	0.15	0.838	0.785	0.524	0.524	0.166
breast	0.05	0.744	0.597	0.538	0.605	0.415
	0.1	0.536	0.625	0.529	0.571	0.569
	0.15	0.747	0.548	0.419	0.611	0.510
diabetis	0.05	0.655	0.828	0.620	0.719	0.000
	0.1	0.764	0.844	0.615	0.252	0.000
	0.15	0.722	0.846	0.669	0.690	0.000
AUC 平均値		0.738	0.727	0.550	0.595	0.237

5.2 ベンチマークデータセット

続いて標準的なベンチマークデータセットを用いて評価を行う．今回は Rätsch's Real Data Collection [8] からデータセットを 3 つ選んだ．各データセットには 2 クラスから成る訓練データ集合と検証データ集合の組み合わせが 100 通り用意されているが，今回はそこから 100 回分をランダムに選んで実験に用いた．ここではクラスラベル $+1$ を正常データ，クラスラベル -1 を異常データとして扱う．訓練データ集合の異常データは全て削除し，検証データ集合は人工データと同様に異常データ割合が $\rho = \{0.05, 0.1, 0.15\}$ となるよう異常データ数を調整し，クラスラベル無しで学習に用いた．その他のカーネル幅などの設定は 5.1 節と同一である．

各設定において 30 回行った実験の結果を表 2 にまとめた．これによれば，人工データセットと比べ KMM と One-class SVM における Best と Worst の差が大きくなっており，やはりカーネル幅 σ の値によって検出力が変動することがわかる．特に One-class SVM は heart データセットや diabetis データセットにおいてやはり特異スコアの大小が逆転し AUC 値が 0.5 よりも低い．一方，KLIEP は diabetis データセットにおいて KMM の Best を下回ったものの，いずれの Worst も大きく上回っており，やはりモデル選択によってロバストな検出力を実現している．さらに全データセットにおける平均でも，KLIEP は他の 2 手法の Best の結果を上回っており，実世界のデータにおいてもその優位性が示された．

5.3 USPS データセット

USPS は米国郵政公社 (U.S. Postal Service) において封筒からスキャンされた手書き数字のデータセットであり，パターン識別手法の評価実験においてベンチマークデータとして良く用いられている [4]．各画像データサンプルは縦 16 ピクセル，横 16 ピクセルの計 256 変数から成り，各変数はグレースケールで-1 から 1 までのピクセル濃度を値に持っており，それぞれの画像には，どの数字を表すかという 0-9 のクラスラベルが割り振られている．このデータセットのタスクとしては，学習データ集合として与えられた 7291 サンプルから数字を判別するモデルを学習し，テストデータ集合 2007 サンプルの数字を予測するものがある．しかしながら，テストデータ集合にはノイズのような画像やクラスの付け間違いと思われるデータが含まれており，その判別は容易ではないことが知られている．そこで，ここでは特異点検出の問題として USPS 学習データ集合を訓練データとして学習し，検証データとする USPS テストデータ集合中の異常なデータを検出する実験を行う．さらに，訓練データと検証データを入れ替えることで，テストデータ集合から見て学習データ集合に含まれ，学習に悪影響を及ぼしていると考えられる特異なデータも見つける．ここでは [9] と同様，数字ラベルを示す 10 次元の属性をデータに加えた．この工夫により，画像の形として異常なものだけでなく，数値ラベルの間違いと思われる特異なサンプルを検出することができる．この実験においても KLIEP は LCV 法を用いて最適なカーネル幅を選択している．訓練データ集合ではカーネル中心となる代表点の数を $b = 1000$ とし，ランダムに選んだ．

実験結果を図 6 と図 7 に示した．それぞれ，特異スコアの最も高かったサンプル 5 個を左から順に並べている．右下の数字は各サンプルに割り振られている数字ラベルである．図 6 を見ると，これら USPS テストデータ集合に含まれる特異なサンプルは人間の目で見ても数字ラベルの判断が難しいことがわかる．このように KLIEP では，LCV 法を用いて適切にモデル選択を行うことにより，実データに含まれる特異点の検出に成功している．一方，図 7 と図 6 とを見比べると明らかなように，USPS 学習データ集合は特異なサンプルであっても数値が見分

図 6: USPS テストデータ集合に含まれる特異なデータ

図 7: USPS 学習データ集合に含まれる特異なデータ

けられる，標準的な数字の形からかけ離れたデータの少ない良質なデータであることが判る．

この結果より，KLIEP を用いた特異点検出は，交差検定によるモデル選択の利点を生かすことにより，実問題において有効であることが分かった．

6 まとめ

我々は特異点検出問題をより容易な密度比推定問題に変換し，特異スコアを精度良く効率的に求める手法を提案した．

本論文では，訓練入力に比べて検証入力の密度関数が大きいデータ点は特異点であるという自然な仮定に基づいて特異スコアを計算する．しかしながら一般に特異点検出を含む教師無し学習において重要な確率密度関数は，推定が難しいことが知られている．特に，データが高次元であったりガウシアンなどの単純な確率分布に従わない場合に精度良く推定することは非常に困難である．一方，それら密度関数の比率を，共変量シフトに良く適応した学習を行うために有用な重要度 (importance) として直接推定するアルゴリズム KLIEP が提案されている [13]．本論文では，この KLIEP を応用した効率の良い特異スコアの計算手法を導入した．KLIEP には，尤度交差検定によってモデルを選択できるという大きな利点があり，密度比を精度良く求めることで高い特異点検出力を示すモデルを安定して得ることができる．

簡単な例おいて，KLIEP による密度比の推定が特異点検出問題に応用できることを示した．また，カーネル幅が性能に大きく影響するため，モデル選択法を備えていることが極めて重要な利点となることが明らかとなった．人工データとベンチマークデータを用いた実験によって，我々の手法は特異スコアを正しく評価し，他の手法と比較しても優れた検出力を持つことを示した．手書き数字データセットに対して適用したところ，KLIEP は数字として異常な画像データを正しく発見することに成功した．

参考文献

[1] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, 2004.

[2] A. P. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997.

[3] W. Hardle, M. Müller, S. Sperlich, and A. Werwatz. Nonparametric and semiparametric models. *Springer Series in Statistics*, 2004.

[4] T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning*. Springer, 2001.

[5] V. Hodge and J. Austin. A survey of outlier detection methodologies. *Artificial Intelligence Review*, 22(2):85–126, 2004.

[6] J. Huang, A. Smola, A. Gretton, K. Borgwardt, and B. Schölkopf. Correcting sample selection bias by unlabeled data. In *Advances in Neural Information Processing Systems*, volume 19. The MIT Press, Cambridge, MA, 2007.

[7] R Development Core Team. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria, 2005.

[8] G. Rätsch, T. Onoda, and K. R. Müller. Soft margins for AdaBoost. *Machine Learning*, 42(3):287–320, 2001.

[9] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7):1443–1471, 2001.

[10] H. Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.

[11] I. Steinwart. The influence of the kernel on the consistency of support vector machines. *Journal of Machine Learning Research*, 2:67–93, 2001.

[12] M. Sugiyama, M. Krauledat, and K.-R. Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8:985–1005, 2007.

[13] M. Sugiyama, S. Nakajima, H. Kashima, P. von Bünau, and M. Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. Technical report, TR07-0003, Department of Computer Science, Tokyo Institute of Technology, Tokyo, Japan, 2007.

[14] D. M. J. Tax and R. P. W. Duin. Support vector data description. *Machine Learning*, 54(1):45–66, 2004.