

ワーカーの類似性を考慮したクラウドソーシングデータからの学習

Clustering Crowds

梶野 洸^{*1} 坪井 祐太^{*2} 鹿島 久嗣^{*1}
 Hiroshi Kajino Yuta Tsuboi Hisashi Kashima

^{*1}東京大学大学院情報理工学系研究科

Graduate School of Information Science and Technology, The University of Tokyo

^{*2}日本アイ・ビー・エム株式会社 東京基礎研究所

IBM Research - Tokyo

Crowdsourcing is often used to create a data set for supervised learning. Because the quality of the data set heavily depends on the abilities of workers, many methods have been proposed to learn a classifier from the data set. From some observations, we notice that workers form clusters according to their abilities. To utilize such knowledge, we propose a method to jointly learn a classifier and clusters of workers from a crowd-generated data set. The proposed method has two advantages. One is that it realizes robust estimation of a classifier because the proposed method utilizes clusters of workers whereas the other methods do not. The other is that we can obtain clusters of workers, which help us analyze the properties of the workers. Experimental results on synthetic and real data sets indicate that the proposed method can estimate a classifier robustly.

1. 序論

クラウドソーシングとは、不特定多数のワーカーに仕事を依頼することができる仕組みである。例えば、教師あり学習に必要な教師データのラベルを作成する際に用いられる。人の知能が必要なタスクを大量に処理可能だという利点がある一方、クラウドソーシングで得られたデータ（以下、クラウドデータと呼ぶ）の品質が不明だという欠点が挙げられる。ワーカーは匿名化されており、その能力が未知であることがこの問題の1つの原因だと考えられる。このような教師データを活用するために、クラウドデータの品質を考慮した学習問題が研究されている。これは、クラウドデータを入力として、目標識別器という真のラベルを予測する問題である。

この問題に対する手法は主に2通り検討されている。一方は間接法で、クラウドデータから真のラベルを推定した後、推定したラベルを用いて目標識別器を学習するという手法である。もう一方は直接法で、クラウドデータから直接目標識別器を学習するという手法である。間接法で真のラベルを推定するために、まず1つのデータに対して複数のワーカーにラベル付けを依頼して冗長性を確保する手法がとられる [Sheng 08]。その冗長なラベルから統計的手法を用いて真のラベルを推定する。特にワーカーのラベル付けの過程をモデル化して、その推論を通じて真のラベルを推定する手法が一般的である [Dawid 79, Welinder 10]。推定された真のラベルを用いて、通常の教師あり学習の手法で目標識別器を学習する。一方直接法では、ワーカーのラベル付けのモデルに目標識別器を取り込むことで直接目標識別器を学習する [Raykar 10, 梶野 12]。

以上のように、いずれの手法でもワーカーのモデル化が行われており、識別器の性能向上のためにモデル化の工夫が重要だと考えられる。ワーカーの様々な性質をモデルに取り込むことで、識別器の性能向上や匿名化されたワーカーの性質の理解が期待される。本論文ではワーカーの類似性に注目し、識別器統合モデル [梶野 12] にその性質を取り込んだクラスタ化識別器統合法（以下、CPC法）を提案する。

まずワーカーの類似性について議論する。最も単純な例はスパムワーカーである。スパムワーカーとは、すべてのデータに同じラベルをつけたり、ランダムなラベルを返すワーカーである。スパムワーカーはある程度類似した性質を持つと考えられ、それに注目するとワーカーをスパムワーカーとその他に分けられると期待される。他の例として、Welinderら [Welinder 10]の実験結果が挙げられる。Welinderらは、クラウドデータから真のラベルを推定する問題を扱い、各ワーカーが潜在空間上で識別器としてモデル化されているのが特徴である。その手法を実データに対して適用した時に得られた各ワーカーの識別器のパラメータを見ると、ワーカーがクラスタを形成していることが確認された (Welinderらの論文の図6参照)。以上より、ワーカーはその能力の類似性に注目することでワーカーのクラスタリングできることが示唆される。そのような情報を利用することで目標識別器の性能の向上が期待できる。

クラスタ化識別器統合法は、ワーカーのクラスタ構造と目標識別器を同時に推定する手法である。クラスタリング正則化項を導入することでワーカーのクラスタ構造の推定が可能となる。クラスタリング正則化項は凸関数であるため、パラメータ推定問題は凸最適化問題として定式化される。ワーカーのクラスタリングを考えた手法は本研究が初めてである。人工データと実データを用いて提案手法の性質や性能を確認する。人工データを用いた実験により、提案手法が既存手法と比べて安定した性能を示すことを確認し、実データを用いた実験により、提案手法の性能が既存手法に勝ることを示す。なお本論文の詳細は [Kajino 13] を参照されたい。

2. 問題設定

本論文で用いる記号と問題を定義する。 I 個のデータが与えられ、各データ $i \in \{1, \dots, I\} (= \mathcal{I})$ に対して D 次元特徴ベクトル $\mathbf{x}_i \in \mathbb{R}^D$ が与えられる。その集合を $\mathcal{X} = \{\mathbf{x}_i \mid i \in \mathcal{I}\}$ とする。データ i に対して真のラベル $y_i \in \{0, 1\}$ が存在すると仮定する（ただし観測はされない）。このデータに J 人のワーカーがラベルを付ける。ワーカー $j \in \{1, \dots, J\} (= \mathcal{J})$ がラベルを

付けたデータ集合を $\mathcal{I}_j \subseteq \mathcal{I}$ とし、ワーカー j は各データ $i \in \mathcal{I}_j$ に対してラベル $y_{ij} \in \{0, 1\}$ を付ける (j をワーカー ID と呼ぶ)。このラベルは品質が未知で $y_{ij} = y_i$ となるとは限らない。ワーカー j から得られたラベル集合を $\mathcal{Y}_j = \{y_{ij} \mid i \in \mathcal{I}_j\}$ とし、クラウドソーシングで得られたラベル集合を $\mathcal{Y} = \bigcup_{j=1}^J \mathcal{Y}_j$ とし、これらのラベルをクラウドラベルと呼ぶ。同様にデータ i に対してラベルを付けたワーカー ID の集合を $\mathcal{J}_i \subseteq \mathcal{J}$ とし、データ i に付けられたラベル集合を $\mathcal{Y}_i = \{y_{ij} \mid j \in \mathcal{J}_i\}$ とする。本論文では、訓練データ $(\mathcal{X}, \mathcal{Y})$ を入力とし、真のラベルを予測する識別器 $f: \mathbb{R}^D \rightarrow \{0, 1\}$ (目標識別器と呼ぶ) を出力とする問題を考える。

3. 提案手法

本章では、識別器統合法 [梶野 12] を紹介した後、それを拡張した提案手法を紹介する。識別器統合法との違いは、クラスタリング正則化項を導入して似たワーカー同士をクラスタリングする点にある。

3.1 識別器統合法 (PC 法) の紹介

目標識別器はロジスティック回帰モデルを用いて

$$p(y_i = 1 \mid \mathbf{x}_i, \mathbf{w}_0) = \sigma(\mathbf{w}_0^\top \mathbf{x}_i) = (1 + e^{-\mathbf{w}_0^\top \mathbf{x}_i})^{-1}, \quad (1)$$

とモデル化される。パラメタ $\mathbf{w}_0 \in \mathbb{R}^D$ には事前分布 $p(\mathbf{w}_0 \mid \eta) = \mathcal{N}(\mathbf{0}_D, \eta^{-1} \mathbf{I}_D)$ が与えられている。ここで、 $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ を平均 $\boldsymbol{\mu} \in \mathbb{R}^D$ 、分散共分散行列 $\Sigma \in \mathbb{R}^{D \times D}$ の正規分布とし、 $\mathbf{0}_D$ を D 次元零ベクトル、 \mathbf{I}_D を D 次元単位行列とし、 $\eta > 0$ をハイパーパラメタとした。各ワーカー $j \in \mathcal{J}$ も同様にパラメタ $\mathbf{w}_j \in \mathbb{R}^D$ のロジスティック回帰モデルを用いて $p(y_{ij} = 1 \mid \mathbf{x}_i, \mathbf{w}_j) = \sigma(\mathbf{w}_j^\top \mathbf{x}_i)$ とモデル化される。これらの識別器を個別識別器と呼ぶ。個別識別器は、目標識別器と

$$\mathbf{w}_j - \mathbf{w}_0 \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(\mathbf{0}_D, \lambda^{-1} \mathbf{I}_D),$$

のように関係しているとする。ここで $\lambda > 0$ をハイパーパラメタとする。

パラメタ $\mathbf{w}_0, \mathbf{W} = \{\mathbf{w}_j \mid j \in \mathcal{J}\}$ の推定は MAP 推定で行う。パラメタの MAP 推定量は

$$\min_{\mathbf{w}_0, \mathbf{W}} L(\mathbf{W}) + \frac{\lambda}{2} \sum_{j=1}^J \|\mathbf{w}_j - \mathbf{w}_0\|^2 + \frac{\eta}{2} \|\mathbf{w}_0\|^2, \quad (2)$$

という凸最適化問題を解くことで得られる。ここで $L(\mathbf{W}) = -\sum_{j=1}^J \sum_{i \in \mathcal{I}_j} [y_{ij} \log \sigma(\mathbf{w}_j^\top \mathbf{x}_i) + (1 - y_{ij}) \log(1 - \sigma(\mathbf{w}_j^\top \mathbf{x}_i))]$ とおいた。また $\|\cdot\|$ は l_2 ノルムとする。

3.2 クラスタ化識別器統合法 (CPC 法)

前節で紹介した識別器統合法を拡張することでクラスタ化識別器統合法を提案する。ワーカー間の類似性を発見するために、クラスタリング損失項 [Hocking 11] を正則化項として用いる。クラスタリング損失項は

$$\Omega(\mathbf{w}_0, \mathbf{W}) = \sum_{(j,k) \in \mathcal{K}} m_{jk} \|\mathbf{w}_j - \mathbf{w}_k\|, \quad (3)$$

と書ける。ここで $\mathcal{K} = \{(j, k) \mid 0 \leq j < k \leq J\}$ とし、 $m_{jk} > 0$ をハイパーパラメタとする*1。式 (2) の第二項を Ω で置き換えることで、

$$\min_{\mathbf{w}_0, \mathbf{W}} L(\mathbf{W}) + \mu \Omega(\mathbf{w}_0, \mathbf{W}) + \frac{\eta}{2} \|\mathbf{w}_0\|^2, \quad (4)$$

*1 実験ではすべての $(j, k) \in \mathcal{K}$ に対して $m_{jk} = 1/J$ とした。

という、ワーカー間の類似性を考慮したクラスタ化識別器統合法のパラメタ推定問題が得られる。ここで $\mu > 0$ をハイパーパラメタとする。識別器統合法と同様に、正則化項 Ω によって、目標識別器と個別識別器が関係付けられるため、この最適化問題を解くことで目標識別器が得られる。また μ を徐々に大きくすることで、式 (3) のいくつかの項が 0 となり、ワーカーらがクラスタリングされていく。これは Group Lasso と似た原理に基づく。ここで、ワーカー j とワーカー k が同じクラスタであるとは $\mathbf{w}_j = \mathbf{w}_k$ となることと定義した。

3.3 アルゴリズム

ADMM [Gabay 76] を元にして、凸最適化問題 (4) を効率的に解くアルゴリズムを提案する。まず最適化問題 (4) を書き換えて ADMM を適用できる形に変形する。はじめに、各 $(j, k) \in \mathcal{K}$ に対して $\mathbf{u}_{jk} = \mathbf{w}_j - \mathbf{w}_k$ という変数を導入して、微分不可能な項を分離可能にする。 $\mathbf{U} = \{\mathbf{u}_{jk}\}_{(j,k) \in \mathcal{K}}$ と置くと、最適化問題 (4) は

$$\min_{\mathbf{w}_0, \mathbf{W}, \mathbf{U}} L(\mathbf{W}) + \mu \sum_{(j,k) \in \mathcal{K}} m_{jk} \|\mathbf{u}_{jk}\| + \frac{\eta}{2} \|\mathbf{w}_0\|^2 \quad (5)$$

$$\text{s.t. } \mathbf{u}_{jk} = \mathbf{w}_j - \mathbf{w}_k, \quad \forall (j, k) \in \mathcal{K}.$$

という同値な最適化問題に変形される。制約付き最適化問題 (5) に対して定義される拡張ラグランジュ関数 \mathcal{L}_ρ は

$$\begin{aligned} \mathcal{L}_\rho(\mathbf{w}_0, \mathbf{W}, \mathbf{U}; \Phi) \\ = L(\mathbf{W}) + \mu \sum_{(j,k) \in \mathcal{K}} m_{jk} \|\mathbf{u}_{jk}\| + \frac{\eta}{2} \|\mathbf{w}_0\|^2 \\ + \sum_{(j,k) \in \mathcal{K}} \phi_{jk}^\top (\mathbf{u}_{jk} - (\mathbf{w}_j - \mathbf{w}_k)) \\ + \frac{\rho}{2} \sum_{(j,k) \in \mathcal{K}} \|\mathbf{u}_{jk} - (\mathbf{w}_j - \mathbf{w}_k)\|^2, \end{aligned}$$

と書ける。ここで $\Phi = \{\phi_{jk} \in \mathbb{R}^D\}_{(j,k) \in \mathcal{K}}$ をラグランジュ乗数の集合とし、 $\rho > 0$ を正の定数とする。提案アルゴリズムでは、この関数に対する $(\mathbf{w}_0, \mathbf{W})$, \mathbf{U} , Φ それぞれに関する 3 つの更新を収束まで繰り返す。詳細な更新則を次に述べる。 t (≥ 0) を更新回数とする。

(i) $(\mathbf{w}_0^{(t+1)}, \mathbf{W}^{(t+1)})$ に関する更新

$\mathbf{U}^{(t)}$ と $\Phi^{(t)}$ が得られたとき、 $(\mathbf{w}_0^{(t+1)}, \mathbf{W}^{(t+1)})$ を

$$(\mathbf{w}_0^{(t+1)}, \mathbf{W}^{(t+1)}) = \arg \min_{\mathbf{w}_0, \mathbf{W}} \mathcal{L}_\rho(\mathbf{w}_0, \mathbf{W}, \mathbf{U}^{(t)}; \Phi^{(t)}), \quad (6)$$

と更新する。 \mathcal{L}_ρ は \mathbf{w}_0 と \mathbf{W} に関して凸関数かつ微分可能であるため、この最適化は準ニュートン法の一種の L-BFGS 法を用いて行う。

(ii) $\mathbf{U}^{(t+1)}$ に関する更新

$(\mathbf{w}_0^{(t+1)}, \mathbf{W}^{(t+1)})$ と $\Phi^{(t)}$ が得られたとき、 $\mathbf{U}^{(t+1)}$ を

$$\mathbf{U}^{(t+1)} = \arg \min_{\mathbf{U}} \mathcal{L}_\rho(\mathbf{w}_0^{(t+1)}, \mathbf{W}^{(t+1)}, \mathbf{U}; \Phi^{(t)}), \quad (7)$$

と更新する。目的関数は \mathbf{U} に関して分離可能であるため、 $\mathbf{u}_{jk}^{(t+1)}$ はそれぞれ独立に計算可能である。 $\mathbf{v}_{jk} = \rho(\mathbf{w}_j^{(t+1)} - \mathbf{w}_k^{(t+1)}) - \phi_{jk}^{(t)}$ と置くと更新則は $\mathbf{u}_{jk}^{(t+1)} = \max\left(0, \frac{\|\mathbf{v}_{jk}\| - \mu m_{jk}}{\rho \|\mathbf{v}_{jk}\|}\right) \mathbf{v}_{jk}$ と書ける。

(iii) $\Phi^{(t+1)}$ に関する更新

ADMM の更新則に従うと、 $\Phi^{(t+1)}$ をそれぞれ

$$\phi_{jk}^{(t+1)} = \phi_{jk}^{(t)} + \rho(\mathbf{u}_{jk}^{(t+1)} - (\mathbf{w}_j^{(t+1)} - \mathbf{w}_k^{(t+1)})), \quad (8)$$

と更新する。

4. 実験

人工データ, 実データを用いてクラスタ化識別器統合法の性質, 性能を評価する。人工データを用いて CPC 法の性質を確認し, 実データを用いてその性能を評価する。

4.1 対抗手法

対抗手法として, 梶野ら [梶野 13] が用いた多数決法 (MV 法), 単純学習法 (AOC 法), 潜在ラベル法 (LC 法), 識別器統合法 (PC 法) を用いた。

4.2 人工データを用いた実験

人工データを用いて CPC 法の利点および欠点を考察する。

4.2.1 データセット

スパムデータセットと能力差データセットの 2 種類のデータを用いた。いずれも能力の高いワーカーのクラスタ $\mathcal{J}_g (\subseteq \mathcal{J})$ と能力の低いワーカーのクラスタ $\mathcal{J}_b (= \mathcal{J} \setminus \mathcal{J}_g)$ の 2 種類のクラスタを考える。スパムデータセットでは, 能力の高いワーカー群と, ランダムにラベルを返すスパムワーカー群を模倣し, 能力差データセットでは, 能力の高いワーカー群と, 能力は低いがスパムワーカーよりは良いワーカー群を模倣した。

いずれも次に説明するモデルにしたがって生成した。各ワーカー $j \in \mathcal{J}$ は能力パラメタ $p_j \in [0, 1]$ と識別器パラメタ $w_j \in \mathbb{R}^D$ でモデル化され,

$$y_{ij} = \begin{cases} \mathbf{1}_{w_j^\top x_i > 0} & \text{with probability } p_j, \\ 1 - \mathbf{1}_{w_j^\top x_i > 0} & \text{with probability } 1 - p_j, \end{cases}$$

とラベル付けを行うとする。ここで $\mathbf{1}_{\text{condition}}$ は condition が成り立てば 1, そうでなければ 0 を返す。このようにモデル化するワーカーに対して, 同じパラメタを持つワーカーをクラスタと定める。2 つのデータセットのパラメタは表 1*2 のように定めた。能力差データセットでは, 識別器パラメタを用いて能力差を模倣する。能力の低いワーカー群の識別器パラメタ $w_{\mathcal{J}_b}$ は, 一部の要素 (次元) の値は w_0 と同じであるが, それ以外の要素の値は 0 となる。例えば, $w_{\mathcal{J}_g} = [1 \ 1]^\top$ としたとき, $w_{\mathcal{J}_b} = [1 \ 0]^\top$ のようにする。これは, 能力の低いワーカーは 2 次元目の差異に気づかないという設定である。

テストセットとして 10,000 個のデータを生成し, 真のラベルを $y_i = \mathbf{1}_{w_0^\top x_i > 0}$ のように生成した。訓練データおよびテストデータでは, 特徴ベクトル $\{x_i\}_{i \in \mathcal{I}}$ は一様分布 $\mathcal{U}([-10, 10]^D)$ から独立に生成し, またすべてのワーカーがすべてのデータにラベルをつけたと仮定した。

4.2.2 実験設定及び実験結果

実験に用いたデータセットのパラメタを表 2 に示す。ここで $r = |\mathcal{J}_b|/J$ とし, $\mathbf{1}_D \in \mathbb{R}^D$ を各要素が 1 となるベクトルとした。PC 法は高次元特徴空間に弱いこと, CPC 法はその点を改善した手法であることを実験的に示す。実験ではデータの生成および学習を 50 回繰り返して ROC 曲線の AUC の平均を計算して性能を測った。比較の簡単のためにすべての手法で真の識別器に対する正則化パラメタを $\eta = 1.0$ と固定した。

はじめに特徴空間の次元が $D = 2$ の場合の結果を示す (図 1)。 $r < 0.5$ の場合どの手法も似た結果を示すが, $r > 0.5$ の場合 AUC は $\text{CPC} = \text{PC} = \text{AOC} > \text{LC} = \text{MV}$ の順に高かった。次に特徴空間の次元が $D = 10$ の場合の結果を示す (図 2)。この場合, AUC は $\text{LC} = \text{MV} > \text{CPC} = \text{PC} > \text{AOC}$ の順に高かった。

*2 $p_{\mathcal{J}_b} = p_j (j \in \mathcal{J}_b)$ のような記法を用いることにする。

表 3: 実データにおける性能比較。

	適合率	再現率	F 値
CPC 法	0.647	0.716	0.680
PC 法	0.637	0.721	0.677
LC 法	0.625	0.732	0.675
AOC 法	0.680	0.670	0.675
MV 法	0.686	0.651	0.668

この実験結果より, PC 法および CPC 法は高次元の場合に性能が悪くなるのがわかる。高次元の場合は個別識別器を推定するのが難しくなるからだと考えられる。一方で, CPC 法は PC 法よりも一貫して性能が良く, 特に能力差データセットでの実験 (図 2 右) では, CPC 法も PC 法も MV 法と同程度の性能で, かつ LC 法よりも高い性能を示すことがわかる。以上より, CPC 法は高次元における PC 法の問題を改善した手法であるといえる。

4.3 実データを用いた実験

実データは梶野ら [梶野 13] と同様のものを用いた。

4.3.1 実験設定及び実験結果

前節で紹介した訓練データ, テストデータを用いて性能を評価した。評価尺度は適合率, 再現率, F 値を用いた。いずれの手法もハイパーパラメタは F 値が最大となるものとした。

実験結果を表 3 に示す。まず F 値の結果より, 提案手法は既存手法よりも高い性能を示すことがわかる。また PC 法と AOC 法と比較をすると, 適合率は $\text{AOC} > \text{CPC} > \text{PC}$ の順に高く, 再現率は $\text{PC} > \text{CPC} > \text{AOC}$ の順に高かった。この結果は, CPC 法は PC 法と AOC 法の中間的な強さの正則化を行っていることを反映しており, その定式化によって適合率と再現率のトレードオフが改善されたと考えられる。

またワーカーのクラスタ系統樹を図 3 に示す。 $\mu < 200$ のとき w_0 が他のクラスタに合流しないことが観察される。この結果よりワーカーの能力が多岐にわたっていることが推察される。また, ワーカー 30 は μ を大きくしても他のクラスタと合流しなかったため外れ値ワーカーだと予想される。実際他のワーカーは表 3 の結果と似た適合率と再現率であったのに対し, ワーカー 30 は低い適合率と高い再現率 ($\mu = 10$ でそれぞれ 0.454, 0.857) であった。このことから, CPC 法を用いてワーカーの性質を解析できる可能性が示唆される。

5. 結論

目標識別器とワーカーのクラスタ構造を同時に推定する手法であるクラスタ化識別器統合法を提案し, 人工データを用いた実験により提案手法の利点, 欠点を考察し, 実データを用いた実験により, 提案手法の優位性について確認した。

謝辞

本研究の一部は, 内閣府最先端研究開発プログラム (FIRST) 「超巨大データベース時代に向けた最高速データベースエンジンの開発と当該エンジンを核とする戦略的サービスの実証・評価」の助成を受けたものである。

参考文献

[Dawid 79] Dawid, A. P. and Skene, A. M.: Maximum likelihood estimation of observer error-rates using the EM

表 1: 人工データを生成するパラメタ.

	p_j		w_j	
	\mathcal{J}_g	\mathcal{J}_b	\mathcal{J}_g	\mathcal{J}_b
スパム	$p_{\mathcal{J}_g}$	0.5	w_0	
能力差	$p_{\mathcal{J}}$		w_0	$w_{\mathcal{J}_b}$

表 2: 実験設定. 各列は各実験で用いたデータセットのパラメタを示す.

データ	D	J	I	r	$w_{\mathcal{J}_g}$	$w_{\mathcal{J}_b}$	$p_{\mathcal{J}_g}$	$p_{\mathcal{J}_b}$	結果
スパム	2	10	10	$0, 0.1, \dots, 1.0$	$[1 \ 0]^T$		0.8	0.5	図 1 左
能力差	2	10	10	$0, 0.1, \dots, 1.0$	$[1 \ 1]^T$	$[1 \ 0]^T$	0.8		図 1 右
スパム	10	10	10	$0, 0.1, \dots, 1.0$	$[\mathbf{1}_5^T \ \mathbf{0}_5^T]^T$		0.8	0.5	図 2 左
能力差	10	10	10	$0, 0.1, \dots, 1.0$	$[\mathbf{1}_5^T \ \mathbf{1}_5^T]^T$	$[\mathbf{1}_5^T \ \mathbf{0}_5^T]^T$	0.8		図 2 右

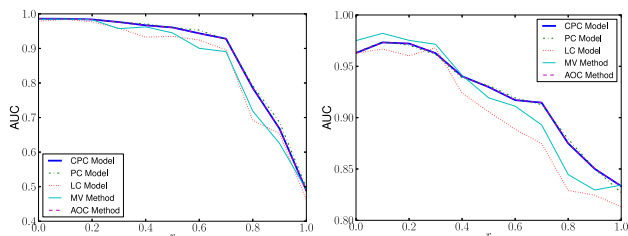


図 1: 2次元データにおける AUC の比較. 左図がスパムデータセット, 右図が能力差データセットでの結果である.

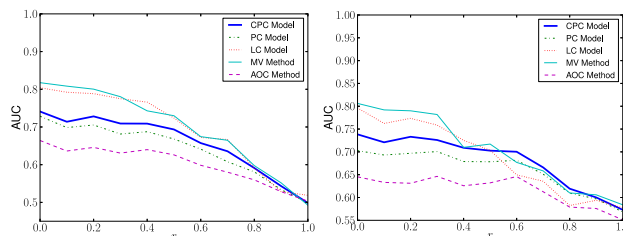


図 2: 10次元データにおける AUC の比較. 左図がスパムデータセット, 右図が能力差データセットでの結果である.

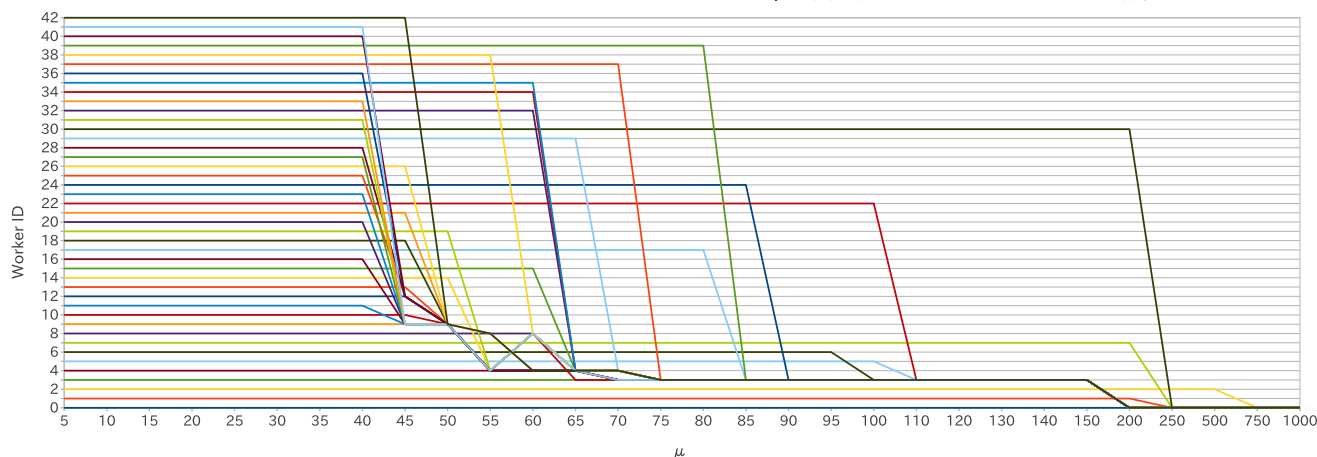


図 3: 実データでのクラスタリング結果. x 軸が μ , y 軸がワーカー ID に対応する. 真の識別器は ID=0 に対応する. $w_j = w_k$ となった時, ワーカー j の線がワーカー k の線と合流する. つまり同じクラスに属するワーカーの線が合流する.

algorithm, *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, Vol. 28, No. 1, pp. 20–28 (1979)

[Gabay 76] Gabay, D. and Mercier, B.: A dual algorithm for the solution of nonlinear variational problems via finite element approximation, *Computers & Mathematics with Applications*, Vol. 2, No. 1, pp. 17–40 (1976)

[Hocking 11] Hocking, T. D., Joulin, A., Bach, F., and Vert, J.-P.: Clusterpath: an algorithm for clustering using convex fusion penalties, in *Proceedings of the 28th International Conference on Machine Learning*, pp. 745–752 (2011)

[Kajino 13] Kajino, H., Tsuboi, Y., and Kashima, H.: Clustering Crowds, in *Proceedings of the 27th AAAI Conference on Artificial Intelligence (AAAI-13)* (2013)

[Raykar 10] Raykar, V. C., Yu, S., Zhao, L. H., Florin, C., Bogoni, L., and Moy, L.: Learning from crowds, *Journal of Machine Learning Research*, Vol. 11, pp. 1297–1322 (2010)

[Sheng 08] Sheng, V. S., Provost, F., and Ipeirotis, P. G.: Get another label? Improving data quality and data mining using multiple, noisy labelers, in *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 614–622 (2008)

[Welinder 10] Welinder, P., Branson, S., Belongie, S., and Perona, P.: The multidimensional wisdom of crowds, in *Advances in Neural Information Processing Systems 23*, pp. 2424–2432 (2010)

[梶野 12] 梶野 洸, 鹿島 久嗣: クラウドソーシングを用いた教師付き学習の凸最適化による定式化, *人工知能学会論文誌*, Vol. 27, No. 3, pp. 133–142 (2012)

[梶野 13] 梶野 洸, 坪井 祐太, 佐藤 一誠, 鹿島 久嗣: エキスパートによる訓練データとクラウドソーシングで作成した訓練データからの教師付き学習, *人工知能学会論文誌*, Vol. 28, No. 3, pp. 243–248 (2013)