DEIM Forum 2022 C31-3

データ分布のクラスタ構造適合による転移学習

GongLinghua† 中村 篤祥††

† 北海道大学大学院情報科学院 〒 060-0814 札幌市北区北 14 条西 9 丁目
†† 北海道大学大学院情報科学研究院 〒 060-0814 札幌市北区北 14 条西 9 丁目
E-mail: †gong@ist.hokudai.ac.jp, ††atsu@ist.hokudai.ac.jp

あらまし クラスラベル付きソースデータから属性データ分布が少し異なるクラスラベル無しターゲットデータのクラスラベルを予測する問題において、データ分布のクラスタ構造適合による転移学習法を提案する。提案法では、ソースデータに対してクラスタリングを行い、そのクラスタ構造が、分布が少し異なるターゲットデータのクラスタ構造と合うように別の空間へ射影し、射影された空間においてソースデータのクラスラベルを利用してターゲットデータのクラスラベルを予測する。実データを用いた実験によれば、クラス毎の分布が合うように射影する従来法より高い精度の予測結果が得られた。

キーワード 転移学習，クラスタリング，画像分類，ドメイン適応

1 はじめに

機械学習において、ラベル付データを用いて教師あり学習を行うことが、モデルの精度を確保するためには有効である。しかし、ラベル付データを得るには、通常高いコストを必要とする。転移学習を用いれば、類似な領域の知識を目標領域に適用して、ラベル付データへの依存を減らすことが可能である。例えば、画像認識用ロボットには、様々な応用があり、応用毎に画像にラベルを付けることは不可能である。また、同じ応用でもラベル付訓練データを用いて事前にモデルを訓練すれば、訓練データとの照明や角度などの違いにより、実際の応用場面ではモデルの精度が下がることもある。転移学習は、訓練時と応用時の状況に違いがあっても、低いコストでモデルの認識精度を保つための手法である。

転移学習法の一種として、ドメイン適応 (Domain Adaption) という方法がある [1]。ドメイン適応の目標は、ソースデータとターゲットデータを類似な確率分布を持つ同じ特徴空間で表現することである。そうすることにより、ソースデータを用いて訓練したモデルは、ターゲットデータにも適用できる。本稿では、ソースデータとターゲットデータが次元が同じ特徴空間に属し、少し異なる確率分布を持つ 2 種類のデータの場合を扱う。ただし、ソースデータはラベル付データであり、ターゲットデータはラベル無しデータとする。

従来法では、ソースデータとターゲットデータの確率分布として、特徴空間における確率分布、またそれをラベルで条件付けた確率分布を考え、それらの確率分布の平均ベクトル違いを最小化するように、ソースデータとターゲットデータを新し特徴空間に射影する [4,5,7,8]。

本稿では、データ分布のクラスタ構造適合による転移法を提案する。ソースデータとターゲットデータの確率分布として、特徴空間における確率分布とそれのクラスタラベルで条件付けた条件付き確率分布を用い、それらの確率分布の平均ベクトルの違いを最小化するように、両方のデータを別の空間へ射影する。射影された空間においてターゲットデータのクラスラベルを予測する。クラスタリング方法として、1 クラスタから始める G-means 法 [2] を用いることにより、クラスタ数を指定せず安定した結果を得ることができる。

本稿の構成は以下の通りである。第 2 節では関連研究を紹介する。第 3 節では本稿の問題設定を行う。第 4 節では提案手法について述べる。第 4.1 節ではクラスタリングの手法、第 4.2 では転移学習の提案手法を説明する。第 5 節では提案法の有効性を検証するための実験と結果を示す。第 6 節では実験結果により、従来法と比べて提案法の精度が良かった理由を考察する。第 7 節では結論を述べる。

2 関連研究

本研究は、クラスタリングを用いたドメイン適応法に関するものである。クラスタリングの手法としては k-means 法が、様々な応用で使われている。しかし、本研究ではクラスタの数を適切に設定できることが重要であるため、自動的にクラスタ数を決めるクラスタリング法 [2,9,10] が望ましい。提案法では G-means を用いた。

ドメイン適応の研究では、ソースデータとターゲットデータの確率分布の違いが小さくなるような特徴空間へ射影することにより、転移学習を行う。Transfer Component Analysis (TCA) [5] は、ソースデータとターゲットデータの特徴空間における確率分布が合うような射影を求める。Joint Distribution Adaption(JDA) [4] は、ソースデータとターゲットデータの特徴空間における確率分布とそれのラベルで条件付けた確率分布が共に合うような、射影を求める。Balanced Distribution Adaption(BDA) [12] と Manifold Embedded Distribution Alignment(MEDA) [11] は、JDA において、二つの分布を重み付けしてバランスをとった目標関数を用いて、転移学習を行う。これら 4 つのドメイン適応法（TCA、JDA、BDA、

MEDA）において、二つの確率分布の違いは、Maximum Mean Discrepancy(MMD) [6] を用いて、データの平均値の距離で測っている。MMD は標本を利用し、二つの母集団が同じ分布に従うかどうかを検証する方法である。

3 問題設定

ラベル付きソースデータ $\mathbf{D}_s = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), ..., (\mathbf{x}_{n_s}, y_{n_s})\} \subseteq \mathcal{X} \times \mathcal{Y}$ と、ラベルなしのターゲットデータは $\mathbf{D}_{t,\mathbf{X}} = \{\mathbf{x}_{n_s+1}, \mathbf{x}_{n_s+2}, ..., \mathbf{x}_{n_s+n_t}\} \subseteq \mathcal{X}$ が与えられて、ターゲットデータのラベル $\{y_{n_s+1}, y_{n_s+2}, ..., y_{n_s+n_t}\} \subseteq \mathcal{Y}$ を予測する問題を考える。ただし、$\mathcal{X} \subseteq \mathbb{R}^d, \mathcal{Y} \subseteq \mathbb{R}$ とする。ソースデータを確率変数 (X_s, Y_s)、ターゲットデータを確率変数 (X_t, Y_t) で表したとき、X_s, X_t の分布 P_{X_s}, P_{X_t} 及び X_s, X_t で条件付けた Y_s, Y_t の条件付き分布 $P_{Y_s|X_s}, P_{Y_t|X_t}$ は異なると仮定する。つまり $P_{X_s} \neq P_{X_t}, P_{Y_s|X_s} \neq P_{Y_t|X_t}$ とする。ソースデータの特徴データの集合は $\mathbf{D}_{s,\mathbf{X}} = \{\mathbf{x}_1, ..., \mathbf{x}_{n_s}\}$ で表す。学習の目標は、ソースとターゲットの特徴データの確率分布と特徴データで条件を付けたクラスの条件付き確率分布が等しくなるような空間に射影する $d \times l$ 行列 $\mathbf{A}$ を見つけることである (l<d)。つまり $P_{\mathbf{A}^\mathrm{T}X_s} \approx P_{\mathbf{A}^\mathrm{T}X_t}$ と $P_{Y_s|\mathbf{A}^\mathrm{T}X_s} \approx P_{Y_t|\mathbf{A}^\mathrm{T}X_t}$ が成り立つような $\mathbf{A}$ を求めることである。

4 提案手法

従来法では $P_{\mathbf{A}^\mathrm{T}X_s} \approx P_{\mathbf{A}^\mathrm{T}X_t}, P_{Y_s|\mathbf{A}^\mathrm{T}X_s} \approx P_{Y_t|\mathbf{A}^\mathrm{T}X_t}$ が成り立つような $\mathbf{A}$ を見つけるために、ソースデータ $\mathbf{A}^\mathrm{T}\mathbf{x}_i (i \in 1, ..., n_s)$ の平均ベクトルとターゲットデータ $\mathbf{A}^\mathrm{T}\mathbf{x}_j (j \in n_s+1, ..., n_s+n_t)$ の平均ベクトルの距離と、クラスで条件付けたそれらの平均ベクトルの距離が小さくなるような $\mathbf{A}$ を求める。しかし、1 つのクラスが 2 つ以上のクラスタから成っている場合や、クラスの分布が重っている場合にこの方法では良い射影 $\mathbf{A}$ を見つけることは難しい。

そこで、本論文の提案手法は、「ソースデータの各々のクラスタに対し、射影された空間における平均ベクトルと同じクラスタに属すると推定されるターゲットデータの平均ベクトルの距離を最小化する」ことを目標として転移学習を行う。手順としては、まず特徴空間 $\mathcal{X}$ においてソースデータをクラスタリングして、ソースデータにクラスタラベルを付ける。次に、同じクラスタラベルを持つソースとターゲットデータの平均ベクトルの距離が最小化するように、二つの操作「ターゲットデータのクラスタラベル推定」と「新しい空間への射影」を交互に行う。以下第 4.1 節ではクラスタリング、第 4.2 節では転移学習を説明する。

4.1 クラスタリング

データのクラスタリングは提案手法の中でもかなり重要な部分である。クラスタリングの結果が空間におけるデータの分布を正しく表しているかどうかは、その後の転移学習の質に影響する。

よく用いられるクラスタリング法は、k-means である。k-means は、ユーザーが設定したクラスタの数 k に基づいて、k 個のクラスタをランダムに初期化して、EM アルゴリズムを用いてクラスタリングを行う。クラスタ数 k の値の選択は、クラスタリングの結果が空間におけるデータの分布に合うかどうかに影響する。また、同じ k の値でもランダムな初期化すると、クラスタリングの結果にランダム性が生じる。k の値の選択とランダムな初期化のいずれかがクラスタリング結果の品質に影響を与え、ひいては転移学習の効果に影響する。

提案手法では、クラスタリング手法として、G-mean [2] を用いる。G-means は各クラスタがガウス分布に従うと仮定するクラスタリング手法であり、自動的にクラスタ数 k を決定できる。提案法では、1 クラスタから始める G-means を使う。検定の順番を固定することにより G-means でのクラスタリング結果は、入力データ $\mathbf{D}_{s,\mathbf{X}}$ と唯一のパラメーター α（Anderson-Darling 検定法の有意水準）の値のみに依存する。

用いる G-means のアルゴリズムを Algotithm 1 に示す。初期状態では、クラスタ数 n_c を 1 に設定する。中心 $\mathbf{c}$ の各クラスタ $\mathbf{D}$ に対し、以下の式 (1) によって、新しクラスタ中心 $\mathbf{c}_1$ と $\mathbf{c}_2$ の初期位置を計算する。

$$\mathbf{c}_1 = \mathbf{c} + \mathbf{p}\sqrt{2\lambda/\pi}, \quad \mathbf{c}_2 = \mathbf{c} - \mathbf{p}\sqrt{2\lambda/\pi} \qquad (1)$$

ただし、$\mathbf{p}$ は $\mathbf{D}$ に対応するデータ行列の第一主成分であり, λ は対応する固有値である。

$\mathbf{c}_1$ と $\mathbf{c}_2$ を初期クラスタの中心として、2-means でクラスタリングを行う。新しいクラスタ中心を $\mathbf{c}'_1$ と $\mathbf{c}'_2$ とする。また Anderson-Darling 検定法 [3] を利用し、有意水準 α で、中心 $\mathbf{c}$ のクラスタが正規分布に従うかを検定する。Anderson - Darling 法は一次元のデータに適用されるので、すべてのデータ $\mathbf{x}_i \in \mathbf{D}$ は以下の式 (2) によって、$\mathbf{c}'_2$ から $\mathbf{c}'_1$ へのベクトル $\mathbf{v} = \mathbf{c}'_1 - \mathbf{c}'_2$ に射影して、一次元のデータ x'_i にしてから検定する。

$$x'_i = < \mathbf{x}_i, \mathbf{v} > / ||\mathbf{v}||^2 \qquad (2)$$

クラスタが正規分布に従う場合は、クラスタ中心 $\mathbf{c}$ を保持し、クラスタ中心 $\mathbf{c}'_1$ と $\mathbf{c}'_2$ を破棄する。従わない場合は、中心 $\mathbf{c}'_1$ と $\mathbf{c}'_2$ を保持し、クラスタ中心 $\mathbf{c}$ を破棄する。すべてのクラスタが統計検定で正規分布に従うことが判定されたら、G-means が止まる。データ $\mathbf{x}_i$ のクラスタラベルを c_i とし、取得したソースデータのクラスタラベル集合を $\mathbf{D}_{s,C} = \{c_1, c_2, ..., c_{n_s}\}$ で表す。

4.2 転移学習

4.2.1 転移学習の目標

ソースデータとターゲットデータの確率分布として、特徴データ $\mathbf{x}$ の確率分布と特徴データで条件を付けたクラスの条件付き確率分布を考える。従来法の転移学習の目標はこの二種類の確率分布がほぼ等しくなるように射影することである。つまり、$P_{\mathbf{A}^\mathrm{T}X_s} \approx P_{\mathbf{A}^\mathrm{T}X_t}$ と $P_{Y_s|\mathbf{A}^\mathrm{T}X_s} \approx P_{Y_t|\mathbf{A}^\mathrm{T}X_t}$ となるような $\mathbf{A}$ を求める。ただし、$P_{Y_s|\mathbf{A}^\mathrm{T}X_s}$ は直接推定できない。JDA [4] の方法では、仮定 $P_{Y_s} = P_{Y_t}$ と転移学習目標 $P_{\mathbf{A}^\mathrm{T}X_s} \approx P_{\mathbf{A}^\mathrm{T}X_t}$ より、$P_{\mathbf{A}^\mathrm{T}X|Y} = \frac{P_{\mathbf{A}^\mathrm{T}X} P_{Y|\mathbf{A}^\mathrm{T}X}}{P_Y}$ を用いて、

Algorithm 1 Algorithm 1: G-means

input: $\mathbf{D}_{s,\mathbf{X}}$: Source Data;$\alpha$: Significance level of Anderson-Darling test.

output: $\mathbf{D}_{s,C}$: clustering labels of source data.

begin : Set n_c to 1; Set $\mathbf{D}^1$ to $\mathbf{D}_{s,\mathbf{X}}$.
for $j = 1$ to n_c **do**
  Calculate the center $\mathbf{c}$ of $\mathbf{D}^j$.
  According to Eq.(1), calculate centers $\mathbf{c}_1$ and $\mathbf{c}_2$.
  Use $\mathbf{c}_1$ and $\mathbf{c}_2$ as initial centers to do 2-means clustering on data $\mathbf{D}^j$. Get subsets $\mathbf{D}_1^j$, $\mathbf{D}_2^j$ and new centers $\mathbf{c}_1'$, $\mathbf{c}_2'$ after clustering.
  According to Eq.(2), calculate data $\mathbf{D}'$ by projecting $\mathbf{x} \in \mathbf{D}^j$ on vector $\mathbf{v} = \mathbf{c}_1' - \mathbf{c}_2'$.
  Use Anderson-Darling test to figure out if data $\mathbf{D}'$ follows a normal distribution.
  if $\mathbf{D}'$ dose not follow a normal distribution **then**
    Set n_c to n_c+1; Use the centers of sets $\mathbf{D}^1, ..., \mathbf{D}^{j-1}$,$\mathbf{D}_1^j$, $\mathbf{D}_2^j$, $\mathbf{D}^{j+1}, ..., \mathbf{D}^{n_c-1}$ as initial centers to do k-means clustering on data $\mathbf{D}_{s,\mathbf{X}}$. Refresh $\mathbf{D}^1, ..., \mathbf{D}^{n_c}$ as the new cluster list. Set the next j to 1.
  end if
end for
Store the clustering labels of $\mathbf{D}_{s,X}$ as $\mathbf{D}_{s,C}$.
return $\mathbf{D}_{s,C}$.

$P_{Y_s|\mathbf{A}^{\mathrm{T}}X_s} \approx P_{Y_t|\mathbf{A}^{\mathrm{T}}X_t}$ を $P_{\mathbf{A}^{\mathrm{T}}X_s|Y_s} \approx P_{\mathbf{A}^{\mathrm{T}}X_t|Y_t}$ に置き換える。

提案手法の転移学習では、ソースデータ X_s、ターゲットデータ X_t のクラスタラベルを表す確率変数を C_s, C_t とすれば、$P_{Y_s|\mathbf{A}^{\mathrm{T}}X_s} \approx P_{Y_t|\mathbf{A}^{\mathrm{T}}X_t}$ の代わりに $P_{C_s|\mathbf{A}^{\mathrm{T}}X_s} \approx P_{C_t|\mathbf{A}^{\mathrm{T}}X_t}$ なるような $\mathbf{A}$ を求める。同様に、仮定 $P_{C_s} = P_{C_t}$ と転移学習目標 $P_{\mathbf{A}^{\mathrm{T}}X_s} \approx P_{\mathbf{A}^{\mathrm{T}}X_t}$ より、$P_{\mathbf{A}^{\mathrm{T}}X|C} = \frac{P_{\mathbf{A}^{\mathrm{T}}X} P_{C|\mathbf{A}^{\mathrm{T}}X}}{P_C}$ を用いて、$P_{C_s|\mathbf{A}^{\mathrm{T}}X_s} \approx P_{C_t|\mathbf{A}^{\mathrm{T}}X_t}$ を $P_{\mathbf{A}^{\mathrm{T}}X_s|C_s} \approx P_{\mathbf{A}^{\mathrm{T}}X_t|C_{\mathrm{t}}}$ に置き換える。提案手法の転移学習の目標は $P_{\mathbf{A}^{\mathrm{T}}X_s} \approx P_{\mathbf{A}^{\mathrm{T}}X_t}$ と $P_{\mathbf{A}^{\mathrm{T}}X_s|C_s} \approx P_{\mathbf{A}^{\mathrm{T}}X_t|C_{\mathrm{t}}}$ となる。

4.2.2 特徴データの確率分布に関する目的関数

目標関数を構築するために、二つのデータセットの確率分布の差を数値化する必要がある。線形写像による転移学習を構築するために、ここでは MMD [6] に基づいて、データセットの一次モーメント（平均）の距離で差を数値化する TCA [5] を用いる。つまり、ソースとターゲットのデータの平均ベクトルの距離で $P_{\mathbf{A}^{\mathrm{T}}X_{\mathrm{s}}}$ と $P_{\mathbf{A}^{\mathrm{T}}X_t}$ の差を測る。計算式は：

$$\left\| \frac{1}{n_s} \sum_{i=1}^{n_s} \mathbf{A}^{\mathrm{T}}\mathbf{x}_i - \frac{1}{n_t} \sum_{j=n_s+1}^{n_s+n_t} \mathbf{A}^{\mathrm{T}}\mathbf{x}_j \right\|^2 = trace\left(\mathbf{A}^{\mathrm{T}}\mathbf{X}\mathbf{M}_0\mathbf{X}^{\mathrm{T}}\mathbf{A}\right) \tag{3}$$

ここで $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_{n_s+n_t}]$、$\mathbf{M}_0$ は行列であり、ij 成分 $(\mathbf{M}_0)_{ij}$ は以下のように定義される。

$$(\mathbf{M}_0)_{ij} = \begin{cases} \frac{1}{n_s n_s} & (\mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_s) \\ \frac{1}{n_t n_t} & (\mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_t) \\ \frac{-1}{n_s n_t}, & otherwise \end{cases} \tag{4}$$

式（3）の値を最小化するように、最適な A を計算する。

4.2.3 クラスタラベル条件付き確率分布に関する目的関数

この節では $P_{\mathbf{A}^{\mathrm{T}}X_s|C_s}$ と $P_{\mathbf{A}^{\mathrm{T}}X_t|C_t}$ の違いを最小化する方法を説明する。同様に、MMD の方法に基づき、差を数値化する。条件付き確率分布の差を、各 $c \in \{1, ..., n_c\}$ に対する $P_{\mathbf{A}^{\mathrm{T}}X|C=c}$ の差の総和と考える。そこで、同じクラスタラベルを持つソースとターゲットデータの平均ベクトルの差を計算する。射影された空間で、ソースデータ $\mathcal{D}_{s,\mathbf{A}^{\mathrm{T}}\mathbf{X},C} = \{(A^{\mathrm{T}}\mathbf{x}_1, c_1), (A^{\mathrm{T}}\mathbf{x}_2, c_2), ..., (A^{\mathrm{T}}\mathbf{x}_{n_s}, c_{n_s})\}$ を利用して kNN 分類器を用いて、射影されたターゲットデータ $\mathbf{x}_j$ の予測クラスタラベル $\hat{c}_j (j = n_s+1, ..., n_s+n_t)$ を求める。同じクラスタに属するソースとターゲットの特徴データの平均ベクトルの距離の総和は、次の式で計算する。

$$\sum_{c=1}^{n_c} \left\| \left(\frac{1}{n_s^{(c)}} \sum_{\substack{i:c_i=c \\ 1 \leqq i \leqq n_s}} \mathbf{A}^{\mathrm{T}}\mathbf{x}_i - \frac{1}{n_t^{(c)}} \sum_{\substack{j:\hat{c}_j=c \\ n_s+1 \leqq j \leqq n_s+n_t}} \mathbf{A}^{\mathrm{T}}\mathbf{x}_j \right) \right\|^2 = trace\left(\mathbf{A}^{\mathrm{T}}\mathbf{X}\mathbf{M}_c\mathbf{X}^{\mathrm{T}}\mathbf{A}\right) \tag{5}$$

ここで $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_{n_s+n_t}]$、$\mathbf{M}_c$ は行列であり、以下のように定義される。

$$(\mathbf{M}_c)_{ij} = \begin{cases} \frac{1}{n_s^{(c_i)} n_s^{(c_j)}}, & (x_i, x_j \in \mathcal{D}_{s,\mathbf{X}}) \\ \frac{1}{n_t^{(\hat{c}_i)} n_t^{(\hat{c}_j)}}, & (x_i, x_j \in \mathcal{D}_{t,\mathbf{X}}) \\ \frac{-1}{n_s^{(c)} n_t^{(\hat{c}_j)}}, & (x_i \in \mathcal{D}_{s,\mathbf{X}}, x_j \in \mathcal{D}_{t,\mathbf{X}}) \\ \frac{-1}{n_s^{(\hat{c}_i)} n_t^{(c)}}, & (x_i \in \mathcal{D}_{t,\mathbf{X}}, x_j \in \mathcal{D}_{s,\mathbf{X}}) \\ 0, & otherwise \end{cases} \tag{6}$$

ただし、$n_s^{(c)} = \left|\mathbf{D}_s^{(c)}\right|$、$\mathbf{D}_s^{(c)} = \{\mathbf{x}_i : (\mathbf{x}_i \in \mathbf{D}_{s,\mathbf{X}}) \wedge (c_i = c)\}$、$n_t^{(c)} = \left|\mathbf{D}_t^{(c)}\right|$、$\mathbf{D}_t^{(c)} = \{\mathbf{x}_j : (\mathbf{x}_j \in \mathbf{D}_{t,\mathbf{X}}) \wedge (\hat{c}_j = c)\}$ である。(5) 式を最小化する $\mathbf{A}$ を求める。

ターゲットデータのクラスタラベル予測の精度が、転移学習の目的関数値に影響を与える。逆に転移学習の目的関数値が、ターゲットデータのクラスタラベル予測の精度にも影響を与える。そこで、ターゲットデータのクラスタラベル予測と転移学習の目的関数値を交互に最適化することを繰り返すことにより、局所解を求めることができる。実験では、転移学習の目標関数値の変化率が 0.001 以下になるまで、射影行列 $\mathbf{A}$ の更新とターゲットデータのクラスタラベル予測を交互に繰り返す。

4.2.4 最　適　化

提案手法では、ソースとターゲットデータの特徴データの確率分布とクラスタラベルで条件を付けたそれらの条件付き確率分布での差を同時に最小化する。式（3）と（5）を組み合わせて、提案手法の最適化は以下の式（7）で表される。$\mathrm{c} = 0$ の場合は、ソースとターゲットの特徴データの平均ベクトルの距離に関する目的関数であり、$c \in 1, ..., n_c$ の部分は、各クラスタの平均ベクトルの距離に関する目的関数である。

Algorithm 2 Algotithm 2: Transfer Learning

input: $\mathbf{X} = [\mathbf{x}_1, .., \mathbf{x}_{n_s+n_t}]$: Matrix of source and target data; $\mathbf{D}_{s,Y} = \{y_1, ..., y_{n_s}\}$: Set of class labels of source data; $\mathbf{D}_{s,C} = \{c_1, c_2, ..., c_{n_s}\}$: Set of clustering labels of source data; l: Dimension number of transferred space; λ: regularization parameter; k: parameter of nearest neighbor classifier.

output: $\mathbf{A}$: Adaption matrix; $\mathbf{D}_{s,\mathbf{A}^{\mathrm{T}}\mathbf{X}}$: Set of transferred source feature data; $\mathbf{D}_{t,\mathbf{A}^{\mathrm{T}}\mathbf{X}}$: Set of transferred target feature data; $\mathbf{D}_{t,\hat{Y}} = \{\hat{y}_{n_s+1}, ..., \hat{y}_{n_s+n_t}\}$: Set of predicted class labels of target data.

1: **begin** : Set MMD matrix $\mathbf{M}_0$ by Eq.(4), $\{\mathbf{M}_c := 0\}_{c=1}^{n_c}$.
2: **while** not convergence **do**
3: Do eigendecomposition of $(\mathbf{XHX}^{\mathrm{T}})^{-1}(\mathbf{X}\sum_{c=0}^{C}\mathbf{M}_c+\lambda\mathbf{I})$ to get eigenvalues and eigenvectors.
4: Select l smallest eigenvalues and their eigenvectors. Construct the adaption matrix $\mathbf{A}$ where column vectors are those l eigenvectors, and calculate $\mathbf{D}_{s,\mathbf{A}^{\mathrm{T}}\mathbf{X}} = \{\mathbf{A}^{\mathrm{T}}\mathbf{x} | \mathbf{x} \in \mathbf{D}_{s,\mathbf{X}}\}$ and $\mathbf{D}_{t,\mathbf{A}^{\mathrm{T}}\mathbf{X}} = \{\mathbf{A}^{\mathrm{T}}\mathbf{x} | \mathbf{x} \in \mathbf{D}_{t,\mathbf{X}}\}$.
5: For each $\mathbf{A}^{\mathrm{T}}\mathbf{x}_i \in \mathbf{D}_{t,\mathbf{A}^{\mathrm{T}}\mathbf{X}}$ predict its clustering label $\hat{c}_i$ by kNN using $(\mathbf{D}_{s,\mathbf{A}^{\mathrm{T}}\mathbf{X}}, \mathbf{D}_{s,C})$.
6: According to Eq.(6), update $\{\mathbf{M}_c\}_{c=1}^{n_c}$ by $\mathbf{D}_{s,C}$ and $\mathbf{D}_{t,\hat{C}} = \{\hat{c}_{n_s+1}, ..., \hat{c}_{n_s+n_t}\}$.
7: **end while**
8: For each $\mathbf{A}^{\mathrm{T}}\mathbf{x}_i \in \mathbf{D}_{t,\mathbf{A}^{\mathrm{T}}\mathbf{X}}$ predict its class label $\hat{y}_t$ by kNN using $(\mathbf{D}_{s,\mathbf{A}^{\mathrm{T}}\mathbf{X}}, \mathbf{D}_{s,Y})$. Let $\mathbf{D}_{t,\hat{Y}} = \{\hat{y}_{n_s+1}, ..., \hat{y}_{n_s+n_t}\}$.
9: **return** Adaption matrix $\mathbf{A}$; Embedded feature data $\mathbf{D}_{s,\mathbf{A}^{\mathrm{T}}\mathbf{X}}$ and $\mathbf{D}_{t,\mathbf{A}^{\mathrm{T}}\mathbf{X}}$; Predicted target label $\mathbf{D}_{t,\hat{Y}}$.

$$\min_{\mathbf{A}:\mathbf{A}^{\mathrm{T}}\mathbf{XHX}^{\mathrm{T}}\mathbf{A}=\mathbf{I}} \sum_{c=0}^{n_c} trace\left(\mathbf{A}^{\mathrm{T}}\mathbf{XM}_c\mathbf{X}^{\mathrm{T}}\mathbf{A}\right) + \lambda\|\mathbf{A}\|_F^2 \quad (7)$$

ここで $\lambda\|\mathbf{A}\|_F^2$ は正則化項であり、λ は正則化パラメーターである。実験では、$\lambda = 1$ とした。$\|\mathbf{A}\|_F^2$ は $\mathbf{A}$ のフロベニウスノルムであって、計算式は $\|\mathbf{A}\|_F^2 = \sum_{i=1}^{d}\sum_{j=1}^{l}\|a_{ij}\|^2$ である。制約 $\mathbf{A}^{\mathrm{T}}\mathbf{XHX}^{\mathrm{T}}\mathbf{A} = \mathbf{I}$ は射影後の分散を保つために付けられている。$\mathbf{H}$ は中心行列（centering matrix）であり、$\mathbf{I} - \frac{1}{n_s+n_t}\mathbf{1}$ で計算する。$\mathbf{I}$ はサイズは $(n_s + n_t) \times (n_s + n_t)$ の単位行列である。$\mathbf{1}$ は全ての要素が 1 の行列、そのサイズは $(n_s + n_t) \times (n_s + n_t)$ である。

最適化はラグランジュの未定乗数法によって、ラグランジュ乗数から成る対角行列 $\mathbf{\Phi} = diag(\phi_1, ..., \phi_k) \in \mathbb{R}^{l*l}$ を設定する。式（7）からラグランジュ関数に変換すると、式（8）のような式になる。

$$\begin{aligned} L =& trace(\mathbf{A}^{\mathrm{T}}(\mathbf{X}\sum_{c=0}^{C}\mathbf{M}_c\mathbf{X}^{\mathrm{T}} + \lambda\mathbf{I})\mathbf{A}) \\ &+ trace((\mathbf{I} - \mathbf{A}^{\mathrm{T}}\mathbf{XHX}^{\mathrm{T}}\mathbf{A})\mathbf{\Phi}) \end{aligned} \quad (8)$$

$\frac{\partial L}{\partial \mathbf{A}} = 0$ を解くと、式（9）が求まる。式（9）より、$\mathbf{A}$ は行列 $(\mathbf{X}H\mathbf{X}^{\mathrm{T}})^{-1}(\mathbf{X}\sum_{c=0}^{C}\mathbf{M}_c + \lambda\mathbf{I})$ の l 個の最小の固有値に対応する固有ベクトルから成る行列である。

$$(\mathbf{X}\sum_{c=0}^{C}\mathbf{M}_c\mathbf{X}^{\mathrm{T}} + \lambda\mathbf{I})\mathbf{A} = \mathbf{XHX}^{\mathrm{T}}\mathbf{A}\Phi \quad (9)$$

提案手法の転移学習部分の手順を、Algorithm 2 に示す。

5 実　　験

この節では、提案手法の性能を検証するために行った実験結果について説明する。

5.1 実験データ

5.1.1 手書き数字

使用するデータセットは手書き数字のデータセットで、MNIST と USPS である [13,14]。データセット MNIST は 60000 件の画像データが含まれており、データセット USPS は 7291 件の画像データが含まれている。

MNIST のデータ量は、USPS に比べて 8 倍以上である。そこで実験では、MNIST からデータの一部をランダムにサンプリングして、その部分を利用する。0 から 9 までの 10 種類の数字があるので、データセットの元々の構成要素の比率を考慮して、数字の種類ごとに 800 件のデータを選んで、合計 8000 件のサブセットを作った。実験では、データを $d = 256$ 次元の特徴空間から $l = 30$ 次元の特徴空間に射影する。

5.1.2 物品の写真

使用するデータセットは物品の写真（Coil）である [15]。20 種類の対象物をそれぞれ 5 度ずつ角度をずらして撮影し、合計 1440 枚の写真を撮影したものである。画像のデータを処理して、長さが 1024 のベクトルに変換して、1440 件の画像が含まれるデータセットを作成している。

実験のために、データセットを二つのサブセット（Coil_A と Coil_B) に分割した。角度が [0°, 85°]∪[180°,265°] の写真がセット Coil_A に所属して、角度が [90°, 175°]∪[270°, 355°] の写真がセット Coil_B に所属する。実験では、データを $d = 1024$ 次元の特徴空間から $l = 100$ 次元の特徴空間に射影する。

5.1.3 人　の　顔

使用するデータセットは人の顔の写真（Face）である [16]。10 人の顔を [0°, 180°] の角度で一人一人 5 度ずつ角度をずらして撮影し、合計 370 枚の写真を撮影し以下の処理をしたものである。画像をミラーリングして、画像の数を元の 2 倍に増やした後、画像のデータを処理して、長さが 1024 のベクトルに変換している。740 件の画像がデータセットに含まれている。

実験のために、データセットを二つのサブセット、(Face_A と Face_B) に分割した。角度が [5°, 45°]∪[140°,180°] の写真がセット Face_A に所属して、角度が [50°, 135°] の写真がセット Face_B に所属する。実験では、データを $d = 1024$ 次元の特徴空間から $l = 100$ 次元の特徴空間に射影する。

5.2 比較の手法

アルゴリズムの精度は、ターゲットデータのクラスラベル予測の精度で表す。l 次元空間に射影されたソースデータのクラスラベルを利用して、$k = 11$ の kNN 分類器を用いて、l 次元空間に射影されたターゲットデータのクラスラベルを予測する。

実験データ	JDA（$\mathbf{A}^*$）	kNN	TCA	JDA	提案手法
U to M	75.95%	15.32%	53.63%	65.32%	**70.85**%
M to U	82.36%	13.78%	60.95%	72.85%	**75.75**%
C_A to C_B	80.97%	78.47%	78.33%	76.11%	**82.36**%
C_B to C_A	79.44%	75.97%	76.52%	75.83%	**84.86**%
F_A to F_B	63.68%	62.89%	61.58%	53.15%	**63.63**%
F_B to F_A	57.5%	55.83%	58.61%	55.83%	**59.44**%

表 1: 従来法との精度比較（$\alpha = 0.01$）。実験データ「A to B」の A はソースデータであり、B はターゲットデータである。データセットの文字 U,M,C,F はそれぞれ USPS,MNIST,Coil,Face を表す。

実験データ	$\alpha = 0.01$	$\alpha = 0.005$	$\alpha = 0.001$	$\alpha = 0.0005$
U to M	70.85%	71.23%	69.79%	**72.91**%
(クラスタ数)	(104)	(97)	(93)	(82)
M to U	**75.75**%	74.17%	74.02%	71.59%
(クラスタ数)	(114)	(110)	(102)	(93)

表 2: 異なる有意水準の比較。実験データ「A to B」の意味は表 1 と同様。

従来法は以下の四つである。

kNN: 転移学習を行わず、元の特徴空間でターゲットデータを分類する手法。

TCA: ソースデータとターゲットデータの特徴データの確率分布の違いを最小化するように、射影する手法。

JDA: 特徴データの確率分布とクラスで条件を付けた条件付き確率分布が合うように射影する手法。（ベースライン）

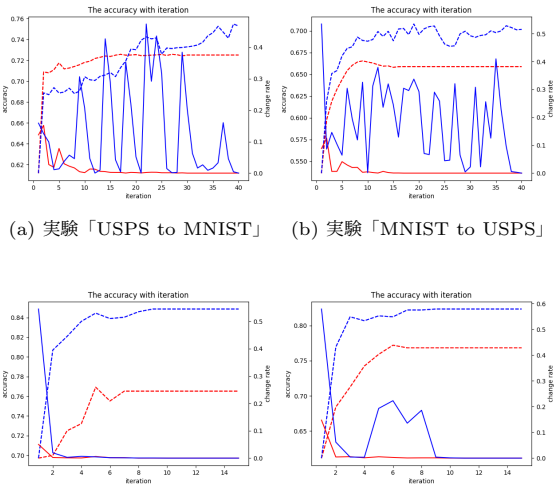
JDA($\mathbf{A}^*$): ソースとターゲット両方のラベル付きデータを入力として、JDA の最適解の $\mathbf{A}^*$ を求める方法。この方法は本来使うことのできない情報（ターゲットデータのラベル）も用いていることに注意されたい。

G-means の有意水準 α の値がクラスタリング結果のクラスタ数に影響する。提案法の精度へのクラスタ数の影響を調べるために、G-means の有意水準 α を変化させて実験した。

5.3 実験結果

JDA（$\mathbf{A}^*$）、kNN、TCA、JDA と提案手法の分類精度を、表 1 に示す。MNIST はサンプリングすることにより、ランダム性が入ることを考慮し、「USPS to MNIST」と「MNIST to USPS」の精度は、5 回のサンプリングに対する平均値を示している。すべての実験において、提案手法の精度は JDA の精度よりも高い。特に「Coil」の実験では、提案手法の精度は JDA の最適解（本来は与えられないターゲットデータのクラスラベルまで与えた解）の精度より高くなっている。また、「Face」と「Coil」は kNN の分類精度と TCA の分類精度が高いことから、ソースデータとターゲットデータの確率分布の類似度が高いと推測できる。このような場合でも、提案手法はさらに精度を向上させている。

注目すべきは提案手法は収束が JDA と比べて遅いことである。収束が遅いとは、目的関数値の変化率が 0.001 より小さくなると、転移学習が止まるように実験を行ったが、停止するまでの繰り返しが多いということである。図 1 は一つのサンプリングに対する実験 (「USPS to MNIST」、「USPS to MNIST」、「Coil_A to Coil_B」、「Coil_B to Coil_A」) のラベル予測精度と目標関数値の変化率を示す。図 1 の手書き数字の実験では、提案法は 40 回 (「USPS to MNIST」) と 39 回 (「MNIST to USPS」) の繰り返しで収束したが、実験の平均では、収束のために 59.8 回 (「USPS to MNIST」) と 64.4 回 (「MNIST to USPS」) の繰り返しを必要とした。それに対し、JDA の手書き数字の実験では、平均値は 17.0 回 (「USPS to MNIST」) と 23.4 回 (「MNIST to USPS」) の繰り返しで停止した。部品の写真の実験と顔の写真の実験においても、目標関数値の変化率よる停止は、提案法の方が JDA より遅い。



(a) 実験「USPS to MNIST」　(b) 実験「MNIST to USPS」

(c) 実験「Coil_A to Coil_B」　(d) 実験「Coil_B to Coil_A」

図 1: 繰り返し回数による精度変化のグラフ。赤い破線は JDA の精度であり、青い破線は提案法の精度である。赤い実線は JDA の目標関数の変化率であり、青い実線は提案法の目標関数の変化率である。実験「USPS to MNIST」と「MNIST to USPS」は複数回行ったが、その中の一回の精度の変化を示す。

G-means の有意水準 α の値を変えて、分類の精度を比較した結果を表 2 で示す。ここでも、MNIST をサンプリングすることにより、ランダム性が入るので、示された精度は 5 回の実験の平均値である。α の値が小さいほど、クラスタの数が少なくなる。予想に反して、クラスタの数が減ってもターゲットデータの分類精度は上がらなかった。つまり、有意水準 α とクラスタの数の変化は、精度に顕著な変化を起さなかった。

6 考　　察

この節では手書き数字データの「USPS to MNIST」の実験結果の分析を通じて、提案法と従来法を比較して、提案法の方が精度が良かった理由を考察する。この実験データに対して JDA の場合は精度が 66.98% であり、提案法の場合は精度が 73.23% である。「射影した空間におけるソースとターゲットのデータ分布の違い」と「転移学習の結果の特性」二つの視点か

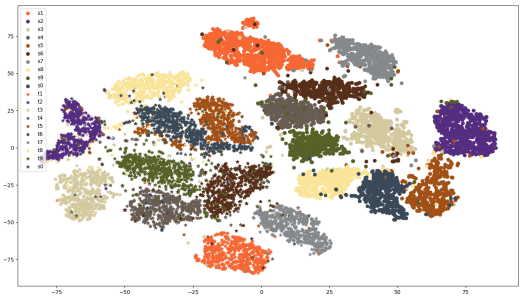
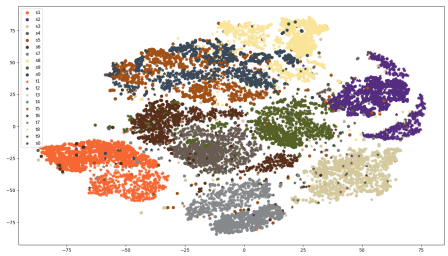
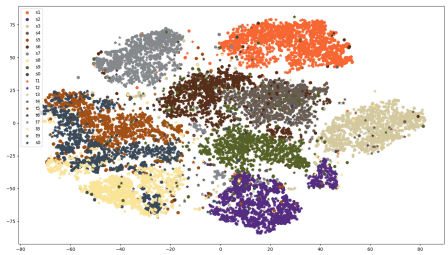


図 2: 元の特徴空間におけるソースとターゲットの特徴データの分布。丸の点はソースデータを表して、星印の点はターゲットデータを表す。ターゲットデータは $\mathbf{D}_{t,Y}$ を用いて、クラスを表示する。



(a) JDA で射影した特徴空間



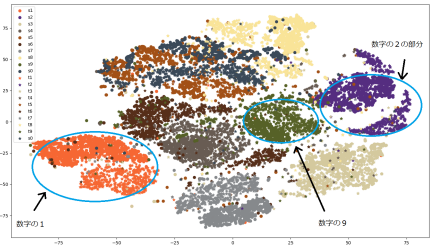
(b) 提案法で射影した特徴空間

図 3: 転移学習で射影した特徴空間におけるソースとターゲットの特徴データの分布。丸の点はソースデータを表して、星印の点はターゲットデータを表す。ターゲットデータは $\mathbf{D}_{t,Y}$ を用いて、クラスを表示する。

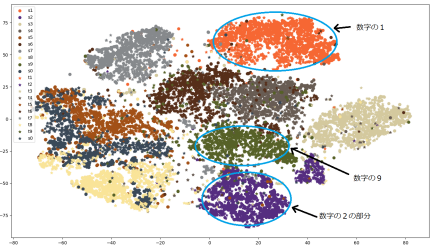
ら、JDA と提案法を比較して、提案法の特性を分析する。

6.1 射影した空間におけるソースとターゲットのデータ分布の違い

まず転移学習の効果について、データ分布を可視化にして画像を通じて分析する。元の特徴空間におけるデータの分布を図 2 に示す。転移学習で求める新たな特徴空間における特徴データの分布は、図 3 で示す。転移学習で求めたソースとターゲットの特徴データは 30 次元であるが、t-distributed Stochastic Neighbor Embedding(t-SNE) [17] で二次元に落として表示する。t-SNE は非線形次元削減の手法であり、データを二次元ま



(a) JDA で射影した特徴空間



(b) 提案法で射影した特徴空間

図 4: 転移学習で射影した特徴空間におけるソースとターゲットの特徴データの分布。丸の点はソースデータを表して、星印の点はターゲットデータを表す。

たは三次元に配置する際に、高い確率で距離の近いデータを近傍に、距離の遠いデータを遠方に配置する手法である。

図 2 によって、元の特徴空間で、ソースデータとターゲットデータは明確に分けられている。右上がソースデータ、左下がターゲットデータである。JDA と提案法により射影された空間でのデータの分布の画像 (図 3) によって、この二つの方法は、特徴空間におけるソースデータとターゲットデータの分布の違いを減少させることが分かる。しかし、JDA の結果と比べて、提案法で得られた結果は、同じクラスのソースとターゲットのデータの分布がより近いだけではなく、重なっているところも多い。例えば図 4 で提示する三箇所は、射影された空間において同じクラスのソースとターゲットのデータが JDA では分けられているが、提案法では重なっている箇所である。

6.2 転移学習の結果の特性

提案法では、あるクラスタラベルを付けたソースデータと同じクラスタラベルに予測されたターゲットデータには共通の特徴があることが確認できる。この節では数字「8」を例として、分析する。まずソースデータにおいて、数字「8」が 9 割以上占めるクラスタを対象にして、それらのクラスタに属すターゲットの数とラベル予測結果の統計を表 3 に示す。

またソースデータにおいて、数字「8」が 9 割以上占めるクラスタの中に、「クラスタ 83」、「クラスタ 1」、「クラスタ 19」の三つのクラスタを例として、同じクラスタに属するソースデータとターゲットデータの共通特徴を分析する。この三つのクラスタ各々に対し、典型的なデータ二つを図 5 に示す。各クラスタ内のデータは、同じ特徴を持っており、異なるクラスタ

データ数（合計）	JDA のみ正解した数	提案法のみ正解した数
939	57	106

表 3: ソースデータにおいて、数字「8」が 9 割以上占めるクラスタに属するターゲットデータの統計

(a) クラスタ 83　(b) クラスタ 1　(c) クラスタ 19

図 5: 「クラスタ 83」、「クラスタ 1」と「クラスタ 19」のソースデータ。「クラスタ 83」の中のデータは特徴「上の半分と下の半分の大きさは同じ」を持つデータである。「クラスタ 1」の中のデータは特徴「上の半分は下の半分より大きい」を持つデータである。「クラスタ 19」の中のデータは特徴「上の半分が離れている」を持つデータである。

(a) クラスタ 83　(b) クラスタ 1　(c) クラスタ 19

図 6: 「クラスタ 83」、「クラスタ 1」と「クラスタ 19」のターゲットデータ

に属すデータは、異なる特徴を持っていることが確認できる。その三つのクラスタラベルに予測されたターゲットデータの中で、クラスタ毎に典型的なデータ二つを図 6 に示す。図 5 と図 6 の比較により、同じクラスタのソースとターゲットデータは、同じ形状的な特徴を持つと言える。つまり、提案法では、クラスが同じでも異なる特徴を持つソースデータに異なるクラスタラベルを付けて、またそれらと対応する特徴を持つターゲットデータに対応するクラスタラベルを予測している。

この三つのクラスタにおいて、提案法では正しくラベルが予測され、JDA で誤って予測されたデータを図 7 に示す。この二つのデータはそれぞれ「クラスタ 19」、「クラスタ 1」の特徴を持つので、提案法で対応するクラスタに分類され、ラベルは正しく「8」と予測される。また JDA で正しくラベルが予測されて、提案法では誤って予測されたデータを図 8 に示す。この三つのデータは、すべて提案法で「クラスタ 19」に分類され、ラベル「8」と予測された。原因は、この三つの数字「4」が、「クラスタ 19」の数字「8」と同じ特徴を持つことによる。

上述の分析により、提案法はクラスタ構造を合わせることにより、同じ特徴を持つデータ集合を合わせていることが確認できた。また、クラスタ構造を捉えて正解するようになったデータ数は、クラス情報を使わないことによって不正解になったデータ数より多かった。よって、「ソースデータとターゲットデータの共通のクラスタ構造を捉えることによって、実験における精度を向上させた」と言える。

7 結　　論

本稿は G-means クラスタリング法を用いたデータ分布のクラスタ構造適合による転移学習法を提案し、クラス構造を適合

(a)　(b)

図 7: 提案法で正しくラベルを予測されて、JDA で誤って予測されたデータ。(a) は提案法で「クラスタ 19」に予測されて、JDA で数字「2」に予測されるデータである。(b) は提案法で「クラスタ 1」に予測されて、JDA で数字「9」に予測されるデータである。

(a)　(b)　(c)

図 8: JDA で正しくラベルを予測されて、提案法で誤って予測されたデータ。(a),(b),(c) は提案法で「クラスタ 19」に予測されて、数字「8」に間違えて予測された。

する従来法より精度が良いことを実験により示した。また実験結果の分析により、「ソースデータとターゲットデータの共通クラスタ構造を捉えて、ソースデータとターゲットデータの分布がより近くなる射影を導いている」ことを確認した。

今後の研究において、方向は 2 つある。一つ目は、より転移学習に適したクラスタリング手法を見つけることである。 クラスタリングの結果は、転移学習の効果に大きく影響する。したがって、より適したクラスタリング手法は転移学習の結果をより向上させると考える。二つ目は、提案手法が適応可能な範囲の明確化である。提案法はソースデータとターゲットデータの共通クラスタ構造を仮定している。提案法を応用するために、適用範囲の明確化は今後の課題である。

文　　献

[1] Pan S J, Yang Q. A survey on transfer learning[J]. IEEE Transactions on knowledge and data engineering, 2009, 22(10): 1345-1359.

[2] HAMERLY, Greg; ELKAN, Charles. Learning the k in k-means. Advances in neural information processing systems, 2003, 16: 281-288.

[3] Anderson, Theodore W., and Donald A. Darling. "Asymptotic theory of certain" goodness of fit" criteria based on stochastic processes." The annals of mathematical statistics (1952): 193-212.

[4] LONG, Mingsheng, et al. Transfer feature learning with joint distribution adaptation. In: Proceedings of the IEEE international conference on computer vision. 2013. p. 2200-2207.

[5] PAN, Sinno Jialin, et al. Domain adaptation via transfer component analysis. IEEE transactions on neural networks, 2010, 22.2: 199-210.

[6] Gretton, Arthur, et al. "A kernel method for the two-sample-problem." Advances in neural information processing systems 19 (2006): 513-520.

[7] Satpal, Sandeepkumar, and Sunita Sarawagi. "Domain adaptation of conditional probability models via feature subsetting." European Conference on Principles of Data Mining and Knowledge Discovery. Springer, Berlin, Heidelberg, 2007.

[8] Hou, Cheng-An, et al. "Unsupervised domain adaptation

with label and structural consistency." IEEE Transactions on Image Processing 25.12 (2016): 5552-5562.

[9] Pelleg, Dan, and Andrew W. Moore. "X-means: Extending k-means with efficient estimation of the number of clusters." Icml. Vol. 1. 2000.

[10] Bischof, Horst, Aleš Leonardis, and Alexander Selb. "MDL principle for robust vector quantisation." Pattern Analysis & Applications 2.1 (1999): 59-72.

[11] Wang, Jindong, et al. "Visual domain adaptation with manifold embedded distribution alignment." Proceedings of the 26th ACM international conference on Multimedia. 2018.

[12] Wang, Jindong, et al. "Balanced distribution adaptation for transfer learning." 2017 IEEE international conference on data mining (ICDM). IEEE, 2017.

[13] LeCun, Yann, et al. "Gradient-based learning applied to document recognition." Proceedings of the IEEE 86.11 (1998): 2278-2324.

[14] Hull, Jonathan J. "A database for handwritten text recognition research." IEEE Transactions on pattern analysis and machine intelligence 16.5 (1994): 550-554.

[15] Nene, Sameer A., Shree K. Nayar, and Hiroshi Murase. "Columbia object image library (coil-100)." (1996).

[16] The National Cheng Kung University Face Database, http://www.datatang.com/data/14866 (accessed March 10, 2020).

[17] Van der Maaten, Laurens, and Geoffrey Hinton. "Visualizing data using t-SNE." Journal of machine learning research 9.11 (2008).