Robust domain-adaptive discriminant analysis

Kouw, Wouter; Loog, Marco

DOI
10.1016/j.patrec.2021.05.005

Publication date
2021

Document Version
Final published version

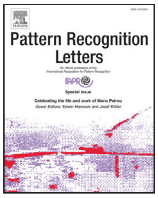
Published in
Pattern Recognition Letters

Pattern Recognition Letters 148 (2021) 107–113

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec



Robust domain-adaptive discriminant analysis[☆]

Wouter M. Kouw[a,b,*], Marco Loog[b,c]

[a] Department of Electrical Engineering, Eindhoven University of Technology, Groene Loper 3, 5612 AE Eindhoven, the Netherlands
[b] Department of Intelligent Systems, Delft University of Technology, Van Moerik Broekmanweg 6, 2628 XE Delft, the Netherlands
[c] Datalogisk Institut, University of Copenhagen, Universitetsparken 5, DK-2100 Copenhagen Ø, Denmark

ARTICLE INFO

Article history:
Received 7 September 2019
Revised 26 March 2021
Accepted 3 May 2021
Available online 20 May 2021

MSC:
41A05
41A10
65D05
65D17

Keywords:
Domain adaptation
Robust estimator
Discriminant analysis
Transduction

ABSTRACT

Consider a domain-adaptive supervised learning setting, where a classifier learns from labeled data in a source domain and unlabeled data in a target domain to predict the corresponding target labels. If the classifier's assumption on the relationship between domains (e.g. covariate shift, common subspace, etc.) is valid, then it will usually outperform a non-adaptive source classifier. If its assumption is invalid, it can perform substantially *worse*. Validating assumptions on domain relationships is not possible without target labels. We argue that, in order to make domain-adaptive classifiers more practical, it is necessary to focus on robustness; robust in the sense that an adaptive classifier will still perform at least as well as a non-adaptive classifier without having to rely on the validity of strong assumptions. With this objective in mind, we derive a conservative parameter estimation technique, which is transductive in the sense of Vapnik and Chervonenkis, and show for discriminant analysis that the new estimator is guaranteed to achieve a lower risk on the given target samples compared to the source classifier. Experiments on problems with geographical sampling bias indicate that our parameter estimator performs well.

1. Introduction

Generalization in supervised learning relies on the fact that future samples originate from the same underlying data-generating distribution as the ones used for training. However, this is not the case in settings where data is collected from different locations, different measurement instruments are used or there is only access to biased data [25]. In these situations the labeled data does not represent the distribution of interest. This problem setting is referred to as a *domain adaptation* setting, where the distribution of the labeled data is called the *source domain* and the distribution of interest is called the *target domain* [3,15]. Most often, data in the target domain is not labeled and adapting a source domain classifier, i.e., changing predictions to suit the target domain, is the only means by which one can make accurate predictions. Unfortunately, depending on the domain dissimilarity, adaptive classifiers can easily perform *worse* than non-adaptive ones. We formulate a conservative adaptive classifier that always performs at least as well as the non-adaptive one.[1]

In the general setting, domains can be arbitrarily different, which means generalization will be extremely difficult. However, there are cases where the problem setting is more structured: in the *covariate shift* setting, the marginal data distributions differ but the posterior distributions are equal [5,9,28]. In such cases, a correctly specified adaptive classifier will converge to the same solution as the target classifier [9]. One way to carry out adaptation is by weighing each source sample by how important it is under the target distribution and training on the importance-weighted labeled source data. However, such a classifier can perform poorly when applied to settings where the covariate shift assumption is false, i.e., where the posterior distributions from both domains are not equal [8,19]. In that case, one often observes that a few samples are given large weights and all other samples are given near-zero weights, which greatly reduces the effective sample size [23, Chapter 8]. Sensitivity to domain relationship assumptions is not restricted to covariate shift. Another adaptive algorithm, Transfer Component Analysis (TCA), assumes the existence of a latent representation common to both domains. When that does not hold, mapping both source and target data onto transfer components

[☆] Handle by Associate Editor Francesco Tortorella.

[*] Corresponding author.
E-mail address: w.m.kouw@tue.nl (W.M. Kouw).

[1] A shortened, preliminary version was accepted for S+SSPR [16]. The current version offers a significant extension with a clearer exposition, additional technical details and references, more experiments, and a comprehensive analysis and discussion.

https://doi.org/10.1016/j.patrec.2021.05.005
0167-8655/© 2021 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/)

will result in mixing of the class-conditional distributions and performance will deteriorate [24].

Since the validity of the aforementioned assumptions is difficult – if not impossible – to check, it is of interest to design robust classifiers. Robustness to uncertainty is often achieved through minimax optimization [17]. An example of a robust adaptive classifier is Robust Covariate Shift Adjustment (RCSA), which first maximizes risk with respect to the importance-weights and subsequently minimizes risk with respect to the classifier parameters [32]. It attempts to account for estimation errors in importance-weights. Another example is the Robust Bias-Aware (RBA) classifier, which plays a game between a risk minimizing target classifier and a risk maximizing target posterior distribution [19]. The adversary is constrained to pick posteriors that match the moments of the source distribution statistics, to avoid posterior probabilities that result in degenerate classifiers (e.g. assign all posterior probabilities to 1). Matching moments means that RBA classifiers lose predictive power in areas of feature space where the source distribution has limited support. Note that both robust methods still rely on assuming covariate shift.

Our main contribution is a parameter estimator that produces estimates with a risk that is always *lower or equal* to the risk of the source classifier, with respect to the given target samples. It does so without making domain relationship assumptions such as covariate shift but by constructing a specific type of risk that can be considered transductive in the sense originally defined by by Vapnik and Chervonenkis [see 30]. Furthermore, we show that in the case of discriminant analysis, the estimator will produce *strictly* smaller risks on the target data. To the best of our knowledge, such performance guarantees compared to the source classifier have not been shown before.

The paper is outlined as follows: Section 3 presents the formulation of our method, with discriminant analysis in Section 4. Section 5.1 shows experiments on two data sets involving geographical sampling bias, indicating that our estimator consistently performs among the best. We conclude with limitations and a discussion in Section 6. To start with, the next section briefly introduces the specific domain adaptation setting that we consider and comments on the transductive nature of our particular approach.

2. Domain adaptation and transduction

A *domain* is defined here as a particular joint probability distribution over a D-dimensional input space $\mathcal{X} \subseteq \mathbb{R}^D$ and a K-dimensional output space of one-hot vectors $\mathcal{Y} = \left\{ b \in \{0,1\}^K : \sum_k b_k = 1 \right\}$ [15]. Let $\mathcal{S}$ mark a *source* domain, with n samples $x = (x_1, \dots, x_n)$ with corresponding labels $y = (y_1, \dots, y_n) \in \mathcal{Y}^n$ drawn from the source domain's joint distribution. Similarly, let $\mathcal{T}$ mark a *target* domain, with m samples $z = (z_1, \dots, z_m)$ with corresponding labels $u = (u_1, \dots, u_m)$ drawn from the target domain's joint distribution. The target labels u are unknown at training time and the goal is to predict them, using only the unlabeled target samples z and the labeled source samples (x, y).

2.1. The meaning of transduction

Given that the primary performance measure in this work is specifically the risk on the unlabeled data of the target domain that is available to us, our objective is essentially transductive [see 15]. This is in line with the original definition of transduction as proposed by Vapnik and Chervonenkis [see 30].

It should be pointed out that, confusingly, what is referred to as transductive for most transfer learning and domain adaptation methods, just means that there is labeled data available for the source but not for the target domain [see also 15]. The classifiers considered in papers such as [1,10,13], like most papers in our review work [15], do not focus on the unlabeled samples in the target domain in particular and are actually *not* transductive in the sense of Vapnik and Chervonenkis [see also 15]. Works like [27,29] exploit graph methods that do not have a ready out-of-sample extension and are therefore transductive in the sense of Vapnik and Chervonenkis. As Section 3 shows, our method focuses particularly on the risk obtained on the given target data and is, as such, transductive. As it turns out, it is specifically this approach that can provide us with performance guarantees, where other techniques cannot.

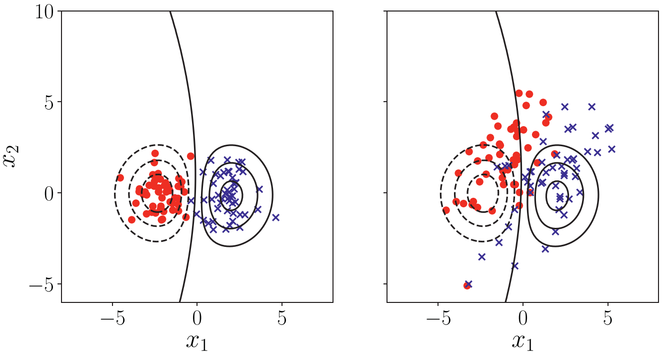


Fig. 1. Example domain adaptation setting. (Left) Labeled source domain data, (right) labeled target domain data. Black lines show a classifier trained on source data, applied to source data (left) and target data (right).

We should note that, typically, our target classifiers can still be used for classifying new and unseen target domain samples. That is, they can also be used for inductive inference. This is especially the case if the samples from the target domain can be considered representative of that domain. In that case, the performance on those particular target domain instances can equally well be interpreted as a regular empirical risk, used in standard empirical risk minimization [26,31]. Just as in the supervised learning setting, it is then assumed that having a small empirical risk carries over to a small generalization error and that the classifier can be successfully employed inductively.

As a final remark, we like to state that the benefits of transduction over induction, or vice versa, are not always easily identified. Especially because in many settings, inductive classifiers can be used for transduction and the other way around. Refer to Chapter 25 in [6] for further views and considerations.

2.2. Example

Fig. 1 visualizes some concepts used throughout the paper. On the left is shown samples from the source domain, labeled as points (red) versus crosses (blue). These were drawn from isotropic Gaussians centered at [−2, 0] and [+2, 0], respectively. The black lines are a contour plot of the posterior probabilities of a classifier trained on the source data. On the right is shown data from the target domain, as well as the source classifier applied to the target data. These target samples were drawn from two Gaussian distributions, both with covariance matrix [3, 2; 2, 4] but one with a mean of [−1, 2] and one with a mean of [+2, 1]. The source and target domains are therefore related to each other through an affine transformation. Note that the source classifier does not fit the target data well.

3. Robust estimator for target domain

In the following, we present the construction of our estimator. First, we discuss the risk of the classifier in the target domain. Secondly, we compare the target risk of a proposal classifier with the target risk incurred by the source classifier and thirdly, we assume a worst-case labeling for the given target samples.

3.1. Target risk

The empirical risk of a classifier in the source domain is computed as the average loss with respect to source samples (x, y):

$$\hat{R}\big(h \,|\, x, y\big) = \frac{1}{n} \sum_{i=1}^{n} \ell(h \,|\, x_i, y_i)\,, \tag{1}$$

where h is the classification function mapping input to labels and ℓ is a loss function comparing the classifier's prediction $h(x_i)$ with the true label y_i at training time. Since the classification error, or $0-1$ loss, cannot be directly optimized over, it is customary to choose surrogate loss functions, such as the quadratic loss $(h(x_i) - y_i)^2$ [11]. The *source classifier* is the classifier found by minimizing the empirical risk with respect to source samples:

$$\hat{h}^{\mathcal{S}} = \underset{h \in \mathcal{H}}{\arg\min}\ \hat{R}(h \,|\, x, y)\,, \tag{2}$$

where $\mathcal{H}$ refers to the hypothesis space.

Since the source classifier does not incorporate any part of the target domain, it is essentially entirely naive of it. But source domains are chosen for a reason – often because they are the most similar data available – and source classifiers are subsequently regarded as the best alternative for classifying the target domain. To evaluate $\hat{h}^{\mathcal{S}}$ in the target domain, the risk of the classifier with respect to target samples (z, u), is computed:

$$\hat{R}\big(\hat{h}^{\mathcal{S}} \,|\, z, u\big) = \frac{1}{m} \sum_{j=1}^{m} \ell\big(\hat{h}^{\mathcal{S}} \,|\, z_j, u_j\big)\,. \tag{3}$$

We argue that adaptive classifiers should never perform worse than source classifiers. In other words, they should never achieve a larger target risk.

3.2. Contrast

We formalize the desire to never achieve a larger target risk by directly comparing the target risk of a potential alternative classifier with the target risk of the source classifier. If we subtract the target risk of the source classifier, then we can argue that the resulting function should never be positive:

$$\hat{R}\big(h \,|\, z, u\big) \ - \ \hat{R}\big(\hat{h}^{\mathcal{S}} \,|\, z, u\big) \tag{4}$$

If this contrast between risk functions is used as a minimization objective, i.e., $\hat{h} = \min_h \ \hat{R}(h \,|\, z, u) \ - \ \hat{R}(\hat{h}^{\mathcal{S}} \,|\, z, u)$, then the target risk of the resulting classifier is bounded above by the risk of the source classifier: $\hat{R}(\hat{h} \,|\, z, u) \leq \hat{R}(\hat{h}^{\mathcal{S}} \,|\, z, u)$. Equality occurs when the source classifier is recovered: $\hat{h} = \hat{h}^{\mathcal{S}}$. Classifiers that lead to larger target risks are not valid outcomes of this minimization procedure.

3.3. Robustness

Eq. (4) still relies on target labels u, which are unknown during training. Instead of u we use a worst-case labeling, achieved by *maximizing* risk with respect to a hypothetical labeling q. For any classifier h, the risk with respect to this worst-case labeling will always be larger than the risk with respect to the true target labeling:

$$\hat{R}(h \,|\, z, u) \leq \max_{q} \ \hat{R}(h \,|\, z, q)\,. \tag{5}$$

Maximizing over a set of discrete labels is a combinatorial problem and, unfortunately, this one is computationally expensive. To avoid this, we apply a relaxation by considering a soft labeling, $q_{jk} = p(u_j = k \,|\, z_j)$. This means that q_j is a vector of K elements that sum to 1. In other words, a point on a $K-1$ simplex, Δ_{K-1}. For m samples, the Cartesian product of m simplices is taken: $\Delta_{K-1} \times \Delta_{K-1} \times \cdots = \Delta_{K-1}^{m}$. By optimizing with respect to a worst-case labeling, the estimator will be more robust to uncertainty over target labels [17].

3.4. Target Contrastive Pessimistic risk

Combining the contrast between risk functions from (4) with the worst-case labeling q from (5) produces the following risk function:

$$\hat{R}^{\text{TCP}}\big(h \,|\, \hat{h}^{\mathcal{S}}, z, q\big) = \frac{1}{m} \sum_{j=1}^{m} \ell\big(h \,|\, z_j, q_j\big) - \ell\big(\hat{h}^{\mathcal{S}} \,|\, z_j, q_j\big)\,. \tag{6}$$

We refer to the risk in Eq. (6) as the Target Contrastive Pessimistic risk (TCP). Minimizing with respect to a classifier h and maximizing with respect to a hypothetical labeling q, produces the new TCP target classifier:

$$\hat{h}^{\mathcal{T}} = \underset{h \in \mathcal{H}}{\arg\min} \ \max_{q \in \Delta_{K-1}^{m}} \hat{R}^{\text{TCP}}\big(h \,|\, \hat{h}^{\mathcal{S}}, z, q\big)\,. \tag{7}$$

Note that the TCP risk only considers the performance on the target domain. More precisely, it considers the performance on the given samples from the target domain and is, in this sense, a transductive approach [12,30]. It is different from the risk formulations in [19,32], and those mentioned in Section 2, because those incorporate performance on the source domain as well. Our formulation focuses purely on the performance *gain* we can achieve over the source classifier, in terms of target risk.

3.5. Optimization

If the loss function ℓ is restricted to be globally convex and the hypothesis space $\mathcal{H}$ to be a convex set, then the TCP risk will be globally convex with respect to h and there will be a unique optimum for h. The TCP risk is linear with respect to q and the optimum need not be unique for q. But the *combined* minimax objective will be globally convex-linear, which guarantees the existence of a saddle point, i.e., a unique optimum with respect to both h and q [7].

Finding this saddle point can be done through first performing a gradient descent step according to the partial derivative with respect to h, followed by a gradient ascent step according to the partial derivative with respect to q. However, this last step causes the updated q to leave the simplex. In order to enforce the constraint, the updated q is projected back onto the simplex. The projection, $\mathcal{P}$, maps a point outside the simplex, a, to the point, b, that is the closest point on the simplex in terms of Euclidean distance: $\mathcal{P}(a) = \arg\min_{b \in \Delta} \|a - b\|_2$ [22]. Unfortunately, the projection step complicates the computation of the step size, which we replace by a learning rate α^t, decreasing over iterations t. This results in the overall update:

$$q^{t+1} \leftarrow \mathcal{P}(q^t + \alpha^t \nabla q^t)\,. \tag{8}$$

A gradient descent-ascent procedure for globally convex-linear objectives is guaranteed to converge to a saddle point (c.f. proposition 4.4 and corollary 4.5 of [7]).

4. Discriminant analysis

Interestingly, for classical discriminant analysis (DA), it can be shown that the TCP risk produces parameter estimates with *strictly* smaller risks than that of the source classifier. Discriminant analysis models the data from each class with a Gaussian distribution, weighted proportional to a class prior: $\pi_k \ \mathcal{N}(x \,|\, \mu_k, \Sigma_k)$ [11]. We use the following shorthand notation for the parameters: $\theta_k =$

(π_k, μ_k, Σ_k). The model is expressed as an empirical risk minimization formulation by taking the negative log-likelihood as a loss function, $\ell(\theta \,|\, x, y) = \sum_k^K -y_k \log[\pi_k \, \mathcal{N}(x \,|\, \mu_k, \Sigma_k)]$.

4.1. Quadratic discriminant analysis

If each class is modeled with a separate covariance matrix, the resulting classifier is a quadratic function of the difference in means and covariances, and is hence called *quadratic discriminant analysis* (QDA). For target data z and probabilistic labels q, the loss is formulated as:

$$\ell_{\text{QDA}}(\theta \,|\, z_j, q_j) = \sum_{k=1}^{K} -q_{jk} \log[\pi_k \, \mathcal{N}(z_j \,|\, \mu_k, \Sigma_k)] \, . \tag{9}$$

Note that the loss is now expressed in terms of classifier parameters θ, as opposed to the classifier h. Plugging the loss from (9) into (6), the TCP-QDA risk becomes:

$$\begin{aligned} \hat{R}^{\text{TCP}}_{\text{QDA}}(\theta \,|\, \hat{\theta}^{\mathcal{S}}, z, q) &= \frac{1}{m} \sum_{j=1}^{m} \ \ell_{\text{QDA}}(\theta \,|\, z_j, q_j) - \ell_{\text{QDA}}(\hat{\theta}^{\mathcal{S}} \,|\, z_j, q_j) \\ &= \frac{1}{m} \sum_{j=1}^{m} \sum_{k=1}^{K} -q_{jk} \log \frac{\pi_k \mathcal{N}(z_j \,|\, \mu_k, \Sigma_k)}{\hat{\pi}_k^{\mathcal{S}} \mathcal{N}(z_j \,|\, \hat{\mu}_k^{\mathcal{S}}, \hat{\Sigma}_k^{\mathcal{S}})} \, , \end{aligned} \tag{10}$$

where the estimate itself is:

$$\hat{\theta}^{\mathcal{T}} = \underset{\theta \in \Theta}{\arg\min} \ \max_{q \in \Delta_{K-1}^{m}} \hat{R}^{\text{TCP}}_{\text{QDA}}(\theta \,|\, \hat{\theta}^{\mathcal{S}}, z, q) \, . \tag{11}$$

Minimization with respect to θ has a closed-form solution for discriminant analysis models. For each class, the parameter estimates are:

$$\pi_k = \frac{1}{m} \sum_{j=1}^{m} q_{jk} \, , \tag{12}$$

$$\mu_k = \Big(\sum_{j=1}^{m} q_{jk} \Big)^{-1} \sum_{j=1}^{m} q_{jk} z_j \, , \tag{13}$$

$$\Sigma_k = \Big(\sum_{j=1}^{m} q_{jk} \Big)^{-1} \sum_{j=1}^{m} q_{jk} (z_j - \mu_k)^{\top} (z_j - \mu_k) \, . \tag{14}$$

Keeping θ fixed, the gradient with respect to q_{jk} is:

$$\frac{\partial}{\partial q_{jk}} \hat{R}^{\text{TCP}}_{\text{QDA}}\big(\theta \,|\, \hat{\theta}^{\mathcal{S}}, z, q\big) = -\frac{1}{m} \log \frac{\pi_k \mathcal{N}(z_j \,|\, \mu_k, \Sigma_k)}{\hat{\pi}_k^{\mathcal{S}} \mathcal{N}(z_j \,|\, \hat{\mu}_k^{\mathcal{S}}, \hat{\Sigma}_k^{\mathcal{S}})} \, . \tag{15}$$

4.2. Example

Fig. 2 visualizes the difference between the source classifier and our TCP-QDA classifier. On the left is shown the source classifier applied to the target data from Section 2.2. On the right is shown the TCP-QDA classifier applied to the same data. Note that it has shifted upwards to better fit the target samples, achieving a smaller risk than the source classifier.

4.3. Regularization

One of the properties of a discriminant analysis model is that it requires the estimated covariance matrix Σ_k to be non-singular. It is possible for the maximizer over q in TCP-QDA to assign less samples than dimensions to one of the classes, causing the covariance matrix for that class to be singular. To prevent this, we regularize its estimation by enforcing a lower bound on the eigenvalues of the estimated covariance matrix.

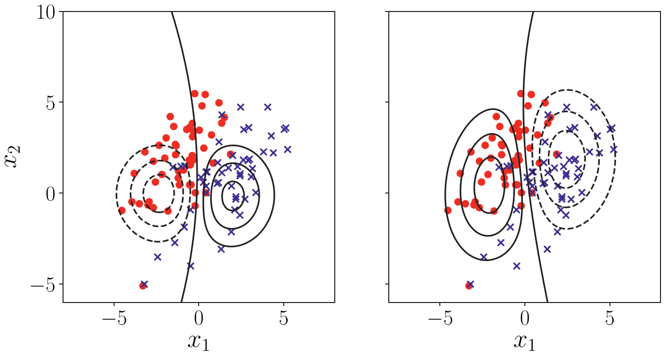


Fig. 2. Example of difference between source Quadratic Discriminant Analysis (left, $\hat{\theta}^{\mathcal{S}}$) and Target Contrastive Pessimistic - Quadratic Discriminant Analysis (right, $\hat{\theta}^{\mathcal{T}}$) on the target domain data from Section 2.2.

4.4. Linear discriminant analysis

If the model is constrained to share a covariance matrix between classes, the resulting classifier is a linear function of the difference in means and is hence termed linear discriminant analysis (LDA). This constraint is imposed through the weighted sum over class covariance matrices $\Sigma = \sum_k^K \pi_k \Sigma_k$.

4.5. Performance guarantee

For the discriminant analysis model, the TCP parameter estimator obtains a *strictly* smaller risk. In other words, this parameter estimator is guaranteed to improve its performance – on the given target samples, and in terms of risk – over the source classifier. This is the first domain adaptation parameter estimator for which such a guarantee can be provided.

Theorem 1. *For a continuous target distribution, with more samples than features for every class, the empirical target risk, with respect to discriminant analysis, of TCP estimated parameters* $\hat{\theta}^{\mathcal{T}}$ *is, almost surely, strictly smaller than that of the source parameters* $\hat{\theta}^{\mathcal{S}}$*:*

$$\hat{R}_{\text{QDA}}\big(\hat{\theta}^{\mathcal{T}} \,|\, z, u\big) \ < \ \hat{R}_{\text{QDA}}\big(\hat{\theta}^{\mathcal{S}} \,|\, z, u\big) \tag{16}$$

The reader is referred to Appendix A for the proof. It follows similar steps as a guarantee for discriminant analysis in semi-supervised learning [20]. Note that as long as the same amount of regularization is added to both the source and the TCP estimator, the strictly smaller risk also holds for a regularized model.

5. Experiments

We see the TCP risk formulation from Section 3, together with Theorem 1, as our main contributions. Of course, it is still of interest to see how other approaches compare to ours. We compare[2] the performance of our classifiers with that of some well-known and robust domain-adaptive classifiers. We implemented Transfer Component Analysis (TCA) [24], Kernel Mean Matching (KMM) [14], Robust Covariate Shift Adjustment (RCSA) [32] and the Robust Bias-Aware (RBA) classifier [19]. TCA and KMM make explicit assumptions: TCA assumes that there are latent factors on which the data can be projected such that the distributions are more similar, while the original properties such as class separability are preserved. We trained a logistic regressor to the source data mapped onto the transfer components. KMM assumes that the posterior distributions in each domain are equal and that the support of the target distribution is contained within the support of the source

[2] Code is available at https://github.com/wmkouw/tcpr

Table 1
WeatherAUS data set. AUC for all pairwise combinations of domains (D='Darwin', P='Perth', B='Brisbane' and M='Melbourne').

$\mathcal{S}$	D	D	D	P	P	B	P	B	M	B	M	M	
$\mathcal{T}$	P	B	M	B	M	M	D	D	D	P	P	B	avg
S-LDA	0.650	0.700	0.672	0.783	0.732	0.565	0.862	0.819	0.919	0.789	0.879	0.903	0.773
S-QDA	0.681	0.857	0.642	0.914	0.940	0.881	0.950	0.937	0.955	0.898	**0.929**	0.959	0.879
TCA	0.825	0.856	0.718	0.838	0.72	0.628	0.842	0.856	0.845	0.834	0.808	0.662	0.786
KMM	0.778	0.704	0.556	0.766	0.705	0.691	0.827	0.717	0.768	0.612	0.517	0.505	0.679
RCSA	0.837	**0.895**	**0.769**	0.841	0.759	0.726	0.858	0.872	0.878	0.813	0.851	0.851	0.829
RBA	**0.844**	0.884	0.764	0.843	0.756	0.741	0.86	0.874	0.878	0.818	0.844	0.839	0.829
TCP-LDA	0.833	0.886	0.749	0.853	0.738	0.733	0.858	0.869	0.875	0.828	0.838	0.859	0.827
TCP-QDA	0.710	0.886	0.760	**0.932**	**0.946**	**0.903**	**0.965**	**0.95**	**0.969**	**0.905**	0.908	**0.964**	**0.900**

distribution. We trained both a weighted logistic regressor and a weighted least-squares classifier using the importance-weights estimated by KMM. We report the best performing of the two, namely least-squares. RCSA also assumes equal posterior distributions, but employs worst-case importance-weight estimation to be robust to weight estimation errors. We used the authors' implementation, which trains a weighted support vector machine using the estimated worst-case weights. RBA assumes that the moments of the source classifier's predictions match that of the target classifier. In our implementation, only the first moments are constrained to match. As baselines, we included a non-adaptive linear (S-LDA) and quadratic (S-QDA) discriminant analysis model trained on the source domain.

All target samples are given - unlabeled - to the adaptive classifiers. The classifiers make predictions for those given target samples and their performance is evaluated with respect to those target samples' true labels. Performance is measured in terms of Area Under the ROC-curve (AUC). All methods are trained using L^2-regularization. Since there is no labeled target data available for validation, we set the regularization parameter to a small value, namely 0.01.

5.1. Data sets

We performed a set of experiments on two data sets that are geographically split into domains. In the first problem, the goal is to predict whether it will rain the following day, based on 22 features including wind speed, humidity, and sunshine (data set is part of the R package Rattle [33]). The measurements are taken over a period of 200 days from Australian weather stations located in Darwin, Perth, Brisbane, and Melbourne. Each station can be considered a domain because the feature spaces are equal but the underlying data-generating distributions are different. For instance, the average temperature is several degrees higher in Darwin than in Melbourne.

The second data set is from the UCI machine learning repository [18]. The goal is to predict heart disease in patients from 4 different hospitals. These are located in Hungary (294 patients), Switzerland (123 patients), California (200 patients) and Ohio (303 patients). Each hospital can be considered a domain because patients are measured on the same clinical features but the local patient populations differ. For example, patients in Hungary are on average younger than patients from Switzerland (48 versus 55 years). Heart disease is predicted from 13 clinical features such as age, sex, cholesterol level and chest pain type. Both data sets are preprocessed by first imputing missing values with zeros and then z-scoring each feature.

5.2. Results

Table 1 compares the AUCs of various classifiers on the WeatherAUS data set. All combinations of using one station as the source domain and another station as the target domain, are taken. Firstly, as a collective, the robust methods (TCP-QDA, TCP-LDA, RBA, RCSA) rather consistently outperform the non-robust methods (TCA, KMM, S-LDA, S-QDA), though it certainly is not the case that every robust method outperforms every non-robust one. Also, there is one exception where S-QDA actually performs best of all. Secondly, RCSA outperforms KMM in all cases, indicating that it is either difficult to estimate appropriate importance weights or that it is difficult to train the importance-weighted classifier given KMM's weights. Thirdly, in eight out of twelve cases TCP-LDA outperforms S-LDA. TCP-QDA is better than S-QDA in eleven of the twelve. Lastly, S-LDA occasionally outperform the non-TCP, adaptive classifiers, where this most notably happens in the three cases when $\mathcal{S} = \text{M}$. For S-QDA this happens in all cases except for $\mathcal{S} = \text{D}$. When $\mathcal{S} = \text{M}$ and $\mathcal{T} = \text{P}$, we find that S-QDA performs best overall. Particularly where S-LDA is concerned, these results indicate that adaptation strategies can also be detrimental to performance.

Table 2 lists AUCs of each classifier in the heart disease data set. Overall, the AUC's are lower here, indicating that these settings are more difficult than those of the weather stations. Firstly, TCP-LDA generally outperforms TCP-QDA here, indicating that most problem settings are linearly separable and the additional flexibility of QDA is not helpful. Secondly, the differences in performance between S-LDA and S-QDA and their TCP versions is clearly less appreciable. In most cases the differences seem insignificant. Exceptions occur when $\mathcal{S} = \text{S}$ and $\mathcal{T} = \text{O}$, in which case the original methods actually perform clearly better and when $\mathcal{S} = \text{S}$ and $\mathcal{T} = \text{H}$, in which case the TCP adaptations do so. Thirdly, RCSA does not always outperform KMM, but since both KMM and RCSA perform worse than chance on a few occasions, it does seem that the assumption of equivalent posterior distributions is invalid in many cases. Fourthly, TCA's performance also varies around chance level, which means that it is difficult to recover a common latent representation here. Lastly, S-LDA and S-QDA outperform the adaptive classifiers on a number of occasions again.

6. Discussion

Although, by construction, the TCP classifiers are never worse than the source classifier in terms of empirical *risk*, they will not automatically lead to improvements in the error rate. This is due to the fact that a surrogate loss function is used during training: the classifier that minimizes the surrogate loss need not be the classifier that minimizes the 0/1-loss [2,4,21]. Similar performance guarantees as we have given with respect to empirical risk, cannot be given with respect to classification error, because the $0 - 1$ loss cannot be directly optimized.

Although our TCP estimator is guaranteed to never perform worse than the source classifier, it may not perform well if the source classifier is a poor choice to begin with. Of course, if no decent source classifiers can be formed, then one can wonder whether any kind of adaptation will be able to construct a satisfactory target classifier, unless particularly reliable assumptions can be made.

Table 2
Heart disease data set. AUC for all pairwise combinations of domains (O='Ohio', H='Hungary', S='Switzerland' and C='California').

$\mathcal{S}$	O	O	O	H	H	S	H	S	C	S	C	C	
$\mathcal{T}$	H	S	C	S	C	C	O	O	O	H	H	S	avg
S-LDA	**0.866**	0.674	**0.658**	0.671	**0.726**	0.527	0.866	0.500	0.831	0.559	**0.883**	0.440	0.683
S-QDA	0.829	0.674	0.503	0.660	0.668	0.484	0.840	0.500	0.811	0.502	0.834	0.452	0.647
TCA	0.674	0.597	0.500	0.453	0.466	0.530	0.544	0.439	0.693	0.408	0.661	**0.572**	0.545
KMM	0.709	0.591	0.460	0.503	0.568	0.552	0.742	0.302	0.294	0.345	0.290	0.508	0.489
RCSA	0.646	0.667	0.572	0.641	0.483	0.459	0.749	**0.626**	0.651	0.685	0.647	0.343	0.597
RBA	0.502	0.670	0.430	0.636	0.423	**0.582**	0.556	0.366	0.523	0.396	0.597	0.412	0.508
TCP-LDA	0.864	**0.675**	0.653	**0.673**	0.725	0.555	**0.867**	0.424	**0.831**	**0.717**	0.882	0.447	**0.693**
TCP-QDA	0.822	**0.675**	0.500	0.661	0.660	0.432	0.841	0.422	0.813	0.565	0.847	0.414	0.638

Given that reliable assumptions *can* be made, our TCP estimator could still be useful. Rather than the original supervised source classifier, one can, in principle, use any adaptive classifier in combination with TCP parameter estimation. In that case, the TCP parameter estimator would still retain its guarantee to not perform worse that the classifier it is compared against, which in this case is the adaptive classifier. Potentially, this may of course lead to even better parameter estimates. A wide range of standard classifiers that rely on the optimization of a convex loss can be incorporated, such as least-squares or support vector machines, meaning that TCP could be combined with many adaptive classifiers. Nonconvex losses, as widely employed in this era of deep learning, are a challenge and, as yet, it is an open and interesting research question to what extent our theoretical results can be salvaged in that setting.

Another possible extension to the current estimator is to use multiple source domains. Perhaps our TCP estimator could produce better estimates than the *best* source estimates. One could envision contrasting the proposal classifier with the classifier producing the lowest risk from among a set of source classifiers, each trained on its own source domain. Finding the best one from among the set of source classifiers would require an additional minimization step over source domains, which would increase the computational cost. Selecting a subset of source domains in advance, could limit this increase in cost and make such an approach feasible.

7. Conclusion

We have designed a risk minimization formulation for a domain-adaptive classifier whose performance, in terms of empirical target risk, is always at least as good as that of the non-adaptive source classifier, without making assumptions on the relationship between domains. This is something that no other method can guarantee. Furthermore, for the discriminant analysis case, its performance is always strictly better. As demonstrated, our Target Contrastive Pessimistic discriminant analysis model performs consistently strong among other robust classifiers.

Declaration of Competing Interest

The authors state that they hold no conflict of interests.

Acknowlgedgment

A word of thanks goes out to the two anonymous reviewers whose feedback helped us improve the presentation of our work. We gladly acknowledge their constructive remarks and comments.

Appendix A

Proof of Theorem 1. Let $\{(x_i, y_i)\}_{i=1}^n$ be a data set of size n drawn *iid* from a continuous distribution defined over input space $\mathcal{X} \subseteq \mathbb{R}^D$ and output space $\mathcal{Y} = \big\{\{0,1\}^K : \sum_k y_k = 1, y \in \mathcal{Y}\big\}$. Similarly, let $\{(z_j, u_j)\}_{j=1}^m$ be a data set of size m, drawn *iid* from another continuous distribution defined over $\mathcal{X} \times \mathcal{Y}$. Consider a discriminant analysis model parameterized with $\theta = (\pi_1, \dots, \pi_K, \mu_1, \dots, \mu_K, \Sigma_1, \dots, \Sigma_K)$ with empirical risk defined by:

$$\hat{R}_{\text{QDA}}(\theta \mid x, y) = \frac{1}{m} \sum_{i=1}^{m} \sum_{k=1}^{K} -y_{ik} \log[\pi_k \, \mathcal{N}(x_i \mid \mu_k, \Sigma_k)] \, . \tag{A.1}$$

The sample covariance matrix, Σ_k, is required to be non-singular, which is guaranteed when there are more unique samples than features for every class. Let $\hat{\theta}^{\mathcal{S}}$ be the parameters estimated on labeled source data:

$$\hat{\theta}^{\mathcal{S}} = \underset{\theta \in \Theta}{\arg\min} \; \hat{R}_{\text{QDA}}\big(\theta \mid x, y\big) \, . \tag{A.2}$$

and let $(\hat{\theta}^{\mathcal{T}}, q^*)$ be the parameters and worst-case labeling estimated by mini-maximizing the Target Contrastive Pessimistic risk:

$$\hat{\theta}^{\mathcal{T}}, q^* = \underset{\theta \in \Theta}{\arg\min} \; \underset{q \in \Delta_{K-1}^m}{\arg\max} \; \hat{R}_{\text{QDA}}\big(\theta \mid z, q\big) - \hat{R}_{\text{QDA}}\big(\hat{\theta}^{\mathcal{S}} \mid z, q\big) \, . \tag{A.3}$$

Firstly, keeping q fixed, the minimization over the contrast between the target risk of the proposal parameters θ and the source parameters $\hat{\theta}^{\mathcal{S}}$ is upper bounded by 0, because both sets of parameters are elements of the same parameter space, $\theta, \hat{\theta}^{\mathcal{S}} \in \Theta$:

$$\min_{\theta \in \Theta} \; \hat{R}_{\text{QDA}}\big(\theta \mid z, q\big) - \hat{R}_{\text{QDA}}\big(\hat{\theta}^{\mathcal{S}} \mid z, q\big) \leq 0 \, , \tag{A.4}$$

for all choices of q. Since θ can always be set to $\hat{\theta}^{\mathcal{S}}$, values for θ that would result in a larger target risk than that of $\hat{\theta}^{\mathcal{S}}$ are not valid minimizers of the contrast. Considering that the contrast is upper bounded for any labeling q, it is also upper bounded by 0 for the worst-case labeling:

$$\min_{\theta \in \Theta} \hat{R}_{\text{QDA}}\big(\theta \mid z, q^*\big) - \hat{R}_{\text{QDA}}\big(\hat{\theta}^{\mathcal{S}} \mid z, q^*\big) \leq 0 \, , \tag{A.5}$$

and since $\hat{\theta}^{\mathcal{T}}$ is the minimizer of the left-hand side of (A.5):

$$\hat{R}_{\text{QDA}}\big(\hat{\theta}^{\mathcal{T}} \mid z, q^*\big) - \hat{R}_{\text{QDA}}\big(\hat{\theta}^{\mathcal{S}} \mid z, q^*\big) \leq 0 \, . \tag{A.6}$$

Secondly, keeping θ fixed, the empirical risk with respect to the true labeling u is always less than or equal to the empirical risk with respect to the worst-case labeling:

$$\begin{aligned} \hat{R}_{\text{QDA}}(\theta \mid z, u) \; &- \; \hat{R}_{\text{QDA}}(\hat{\theta}^{\mathcal{S}} \mid z, u) \\ &\leq \max_{q \in \Delta_{K-1}^m} \; \hat{R}_{\text{QDA}}(\theta \mid z, q) - \hat{R}_{\text{QDA}}(\hat{\theta}^{\mathcal{S}} \mid z, q) \, . \end{aligned} \tag{A.7}$$

Since q^* is the maximizer for $\hat{\theta}^{\mathcal{T}}$ as parameters, we can write:

$$\begin{aligned} \hat{R}_{\text{QDA}}(\hat{\theta}^{\mathcal{T}} \mid z, u) \; &- \; \hat{R}_{\text{QDA}}(\hat{\theta}^{\mathcal{S}} \mid z, u) \\ &\leq \hat{R}_{\text{QDA}}(\hat{\theta}^{\mathcal{T}} \mid z, q^*) - \hat{R}_{\text{QDA}}(\hat{\theta}^{\mathcal{S}} \mid z, q^*) \, . \end{aligned} \tag{A.8}$$

Combining Inequalities A.6 and A.8 gives:

$$\hat{R}_{\text{QDA}}\big(\hat{\theta}^{\mathcal{T}} \mid z, u\big) - \hat{R}_{\text{QDA}}\big(\hat{\theta}^{\mathcal{S}} \mid z, u\big) \leq \; 0 \, . \tag{A.9}$$

Bringing the second term on the left-handside to the right-handside shows that the target risk of the TCP estimate is always less than or equal to the target risk of the source classifier's:

$$\hat{R}_{\text{QDA}}\big(\hat{\theta}^{\mathcal{T}} \,|\, z, u\big) \;\leq\; \hat{R}_{\text{QDA}}\big(\hat{\theta}^{\mathcal{S}} \,|\, z, u\big) \,. \tag{A.10}$$

Equality in (A.10) occurs with probability 0, which can be shown through the parameter estimators. The total mean for the source classifier consists of the weighted combination of the class means, resulting in the overall source sample average:

$$\begin{aligned}\hat{\mu}^{\mathcal{S}} &= \sum_{k=1}^{K} \hat{\pi}_k^{\mathcal{S}} \; \hat{\mu}_k^{\mathcal{S}} \\ &= \sum_{k=1}^{K} \frac{\sum_i^n y_{ik}}{n} \left[\frac{1}{\sum_i^n y_{ik}} \sum_{i=1}^{n} y_{ik} x_i \right] \\ &= \frac{1}{n} \sum_{i=1}^{n} x_i \,. \end{aligned} \tag{A.11}$$

The total mean for the TCP-QDA estimator is similarly defined, resulting in the overall target sample average:

$$\begin{aligned}\hat{\mu}^{\mathcal{T}} &= \sum_{k=1}^{K} \hat{\pi}_k^{\mathcal{T}} \; \hat{\mu}_k^{\mathcal{T}} \\ &= \sum_{k=1}^{K} \frac{\sum_j^m q^*_{jk}}{m} \left[\frac{1}{\sum_j^m q^*_{jk}} \sum_{j=1}^{m} q^*_{jk} z_j \right] \\ &= \sum_{k=1}^{K} \frac{1}{m} \sum_{j=1}^{m} q^*_{jk} z_j \\ &= \frac{1}{m} \sum_{j=1}^{m} z_j \,. \end{aligned} \tag{A.12}$$

Note that since q^* consists of probabilities, the sum over classes $\sum_k^K q^*_{jk}$ is 1, for every sample j. Equal risks for these parameter sets, $\hat{R}_{\text{QDA}}\big(\hat{\theta}^{\mathcal{T}} \,|\, z, u\big) = \hat{R}_{\text{QDA}}(\hat{\theta}^{\mathcal{S}} \,|\, z, u)$, implies equality of the total means, $\hat{\mu}^{\mathcal{T}} = \hat{\mu}^{\mathcal{S}}$. By Eqs. A.11 and A.12, equal total means imply equal sample averages: $\frac{1}{m}\sum_j^m z_j = \frac{1}{n}\sum_i^n x_i$. Given a set of source samples, drawing a set of target samples such that their averages are *exactly equal*, constitutes a single event under a probability density function. By definition, single events under continuous distributions have probability 0. Therefore, a strictly smaller risk occurs almost surely:

$$\hat{R}_{\text{QDA}}\big(\hat{\theta}^{\mathcal{T}} \,|\, z, u\big) \;<\; \hat{R}_{\text{QDA}}\big(\hat{\theta}^{\mathcal{S}} \,|\, z, u\big) \,. \tag{A.13}$$

□

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.patrec.2021.05.005.

References

[1] A. Arnold, R. Nallapati, W.W. Cohen, A comparative study of methods for transductive transfer learning, in: IEEE International Conference on Data Mining Workshops, 2007, pp. 77–82.

[2] P.L. Bartlett, M.I. Jordan, J.D. McAuliffe, Convexity, classification, and risk bounds, J. Am. Stat. Assoc. 101 (2006) 138–156.

[3] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, J.W. Vaughan, A theory of learning from different domains, Mach. Learn. 79 (2010) 151–175.

[4] S. Ben-David, D. Loker, N. Srebro, K. Sridharan, Minimizing the misclassification error rate using a surrogate convex loss, in: International Conference on Machine Learning, 2012, pp. 83–90.

[5] S. Bickel, M. Brückner, T. Scheffer, Discriminative learning under covariate shift, J. Mach. Learn. Res. 10 (2009) 2137–2155.

[6] O. Chapelle, B. Scholkopf, A. Zien, Semi-Supervised Learning, MIT Press, 2006.

[7] A. Cherukuri, B. Gharesifard, J. Cortes, Saddle-point dynamics: conditions for asymptotic stability of saddle points, SIAM J. Control Optim. 55 (2017) 486–511.

[8] C. Cortes, M. Mohri, Domain adaptation and sample bias correction theory and algorithm for regression, Theor. Comput. Sci. 519 (2014) 103–126.

[9] C. Cortes, M. Mohri, M. Riley, A. Rostamizadeh, Sample selection bias correction theory, in: Algorithmic Learning Theory, 2008, pp. 38–53.

[10] N. Farajidavar, T.E. de Campos, J. Kittler, Adaptive transductive transfer machine, in: British Machine Vision Conference, 2014, pp. 1–12.

[11] J. Friedman, T. Hastie, R. Tibshirani, The Elements of Statistical Learning, Springer, 2001. volume 1

[12] A. Gammerman, V. Vovk, V. Vapnik, Learning by transduction, in: Conference on Uncertainty in Artificial Intelligence, 1998, pp. 148–155.

[13] Q. Gu, J. Zhou, Learning the shared subspace for multi-task clustering and transductive transfer classification, in: IEEE International Conference on Data Mining, 2009, pp. 159–168.

[14] J. Huang, A.J. Smola, A. Gretton, K.M. Borgwardt, B. Schölkopf, et al., Correcting sample selection bias by unlabeled data, in: Advances in Neural Information Processing Systems, 2007, pp. 601–608.

[15] W.M. Kouw, M. Loog, A review of domain adaptation without target labels, IEEE Trans. Pattern Anal. Mach.Intell. 43 (2021) 766–785.

[16] W.M. Kouw, M. Loog, Target robust discriminant analysis, in: IAPR Joint International Workshops on Statistical techniques in Pattern Recognition and Structural and Syntactic Pattern Recognition, 2021, p. accepted.

[17] E.L. Lehmann, G. Casella, Theory of Point Estimation, Springer, 2006.

[18] M. Lichman, UCI machine learning repository, 2013,. http://archive.ics.uci.edu/ml.

[19] A. Liu, B. Ziebart, Robust classification under sample selection bias, in: Advances in Neural Information Processing Systems, 2014, pp. 37–45.

[20] M. Loog, Contrastive pessimistic likelihood estimation for semi-supervised classification, IEEE Trans. Pattern Anal. Mach.Intell. 38 (2016) 462–475.

[21] M. Loog, J.H. Krijthe, A.C. Jensen, On measuring and quantifying performance: error rates, surrogate loss, and an example in semi-supervised learning, in: Handbook of Pattern Recognition and Computer Vision, World Scientific, 2016, pp. 53–68.

[22] N. Maculan, G.G. De Paula Jr, A linear-time median-finding algorithm for projecting a vector on the simplex of $\mathbb{R}^n$, Oper. Res. Lett. 8 (1989) 219–222.

[23] A.B. Owen, Monte Carlo theory, methods and examples, 2013,. https://statweb.stanford.edu/~owen/mc/.

[24] S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain adaptation via transfer component analysis, IEEE Trans. Neural Netw. 22 (2011) 199–210.

[25] J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, N.D. Lawrence, Dataset Shift in Machine Learning, MIT Press, 2009.

[26] B. Schölkopf, A.J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and beyond, MIT press, 2002.

[27] O. Sener, H.O. Song, A. Saxena, S. Savarese, Unsupervised transductive domain adaptation, arXiv preprint arXiv:1602.03534 (2016).

[28] H. Shimodaira, Improving predictive inference under covariate shift by weighting the log-likelihood function, J. Stat. Plan. Inference 90 (2000) 227–244.

[29] L. Shu, L.J. Latecki, Transductive domain adaptation with affinity learning, in: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, 2015, pp. 1903–1906.

[30] V. Vapnik, Estimation of Dependences based on Empirical Data, Springer, 1982.

[31] V. Vapnik, Principles of risk minimization for learning theory, in: Advances in neural information processing systems, 1992, pp. 831–838.

[32] J. Wen, C.N. Yu, R. Greiner, Robust learning under uncertain test distributions: Relating covariate shift to model misspecification, in: International Conference on Machine Learning, 2014, pp. 631–639.

[33] G.J. Williams, Data Mining with Rattle and R: The Art of Excavating Data for Knowledge Discovery, Springer, 2011. Use R!