REGULARIZED TRANSPORT WEIGHTS: A PRINCIPLED APPROACH FOR ACTIVE DOMAIN ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Distribution shift, where there is a mismatch between training and test distributions, is a fundamental challenge in modern machine learning. In this context, we consider the setting of active domain adaptation, where the learner has access to a labeled source dataset S, and aims to classify an unlabeled target dataset T. The learner is allowed to query the labels of a few points in T to mitigate the distribution shift between S and T, as in active learning. We propose a principled, task-agnostic method that jointly selects target queries and reweighs the source and queried examples. Our method tries to effectively use the existing source labels in order to balance coverage of the target distribution while mitigating the variance in the labels. We achieve this by minimizing a natural Regularized Transport Weighting (RTW) objective between the augmented dataset and the target T. We provide theoretical guarantees for our method, including a generalization bound and an approximation guarantee for our algorithm. Experiments on standard benchmarks show that our method is competitive with the state-of-the-art methods, and outperforms prior methods on some standard benchmarks– with only a few samples, we are able to train a classifier that significantly improves upon the accuracy achieved on the source dataset alone.

1 INTRODUCTION

A central challenge in machine learning is to transfer learned concepts to new unseen data and domains. A mismatch between the training and test distributions, such as covariate shift, can substantially reduce prediction accuracy. For example, a model trained on medical images from one hospital may perform poorly at another (Quiñonero-Candela et al., 2008; Mansour et al., 2009; Peng et al., 2019; Koh et al., 2021). It may be expensive to get labeled data from the target domain, but we often have access to data from related domains. This motivates *domain adaptation*, where the learner uses data from a source distribution $\mathcal{P}$, to learn predictors for the target distribution $\mathcal{Q}$.

Concretely, $\mathcal{P}$ and $\mathcal{Q}$ are distributions over $(x, y) \in \mathcal{X} \times \mathcal{Y}$ where $\mathcal{X}$ is the *feature* space and $\mathcal{Y}$ is the *label* space. We use $\mathcal{P}_x$ and $\mathcal{Q}_x$ to denote the feature marginals of $\mathcal{P}$ and $\mathcal{Q}$ respectively. We will use $\mathcal{H}$ to denote a class of hypothesis functions $h : \mathcal{X} \to \mathcal{Y}$ of interest. We consider the *covariate shift* setting between $\mathcal{P}$ and $\mathcal{Q}$, where the marginal distributions $\mathcal{P}_x$ and $\mathcal{Q}_x$ differ, while the conditional distribution $Y|x$ is the same for both $\mathcal{P}$ and $\mathcal{Q}$.

Domain adaptation is naturally limited by how different the source and target distributions are from one another; in regions of the covariate space where $\mathcal{P}$ and $\mathcal{Q}$ differ significantly, the source distribution $\mathcal{P}$ alone may provide insufficient information to make good predictions. A popular approach to tackle covariate shift is importance sampling (Sugiyama et al., 2007; Quiñonero-Candela et al., 2008). Importance sampling uses unlabeled data from the target distribution $\mathcal{Q}$ to reweigh source examples according to the target-to-source density ratio. Estimating importance weights can be difficult in high dimensions. Moreover even when estimated exactly, large importance weights can lead to high statistical error. These challenges motivate *active domain adaptation* methods, where the learner can query the labels of a small number of target points, in addition to unlabeled data from $\mathcal{Q}$ and labeled data from the source $\mathcal{P}$.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Active Domain Adaptation. Active domain adaptation captures a setting where target labels are expensive but not impossible to acquire. In the context of active domain adaptation, the question becomes:

Which target labels should we acquire and how should we combine them with the source data?

The ideal method should learn as much information about $\mathcal{Q}$ as possible from $\mathcal{P}$, and spend its query budget on parts of $\mathcal{Q}$ where we have insufficient information from the source $\mathcal{P}$.

We focus on *task-agnostic query methods*: for a fixed feature representation (and potentially a hypothesis class), queries are selected only using the features of the source and target datasets; label information is not used in determining the query points. This allows the selected points to be reused across multiple tasks using the same queries. This is motivated by scientific domains like astronomy, where the queried information is rich, expensive information (e.g., high-resolution spectra) that can be used to provide "labels" for several different tasks. Task-agnostic query methods also fall into the class of single-shot active domain adaptation methods that have been studied previously (e.g., de Mathelin et al., 2022; Mahmood et al., 2022).

1.1 Model and Our Contributions.

Formally, the learner is given an *unlabeled* target dataset $T \subset \mathcal{X}$ of i.i.d. samples from $\mathcal{Q}_x$, a labeled source dataset $(s_1, y_1), (s_2, y_2), \ldots, (s_n, y_n) \in \mathcal{X} \times \mathcal{Y}$ drawn from $\mathcal{P}$, and a query budget k. We use $S = \{s_1, \ldots, s_n\} \subset \mathcal{X}$ to denote the source dataset. A cost function $c : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_{\geq 0}$ specifies the distance or cost between points in the feature space $\mathcal{X}$ and encodes how much information s provides about t. A natural choice is the Euclidean distance $c(s,t) = \|s - t\|_2$ over $\mathcal{X} = \mathbb{R}^d$, or any metric over $\mathcal{X}$; in Section 3 we will use a choice that also depends on the hypothesis class $\mathcal{H}$ to get better bounds.

The learner is allowed to *query* the labels of k samples $A_k \subset T$. Moreover, the learner can also pick the distribution i.e., weights $w = (w_s : s \in S \cup A_k)$ over the labeled samples corresponding to $S \cup A_k$. The classifier is then trained on the distribution of labeled examples corresponding to $S \cup A_k$ and w. Our goal is to find the best k points to query along with the weights w to optimize the loss on the target dataset. To this end, we introduce Regularized Transport Weighting (RTW), which jointly selects target queries A_k and the weights on the source and queried examples $S \cup A_k$.

Regularized Transport Weighting (RTW). Our first theoretical contribution is to frame the problem of choosing queries from T as minimizing a natural *transport* objective. The objective trades off how well the weighted data represent the target against the statistical error from concentrating weight on too few samples.

Definition 1.1. Given source points S, target points T, select k points $A_k \subseteq T$ and weights $w = (w_s : s \in S \cup A_k)$ that

$$\min_{A_k \subseteq T, |A_k| \leq k} \mathrm{RTW}_\lambda(S \cup A_k), \tag{1}$$

where RTW_λ is the following regularized transport objective that jointly optimizes the weights $w = (w_s : s \in S \cup A_k)$ and the transport map $\pi = (\pi_{st}) \in [0,1]^{(S \cup A_k) \times T}$:

$$\mathrm{RTW}_\lambda(S \cup A_k) = \min_{\pi \geq 0,\, w} \underbrace{\sum_{s \in S \cup A_k} \sum_{t \in T} \pi_{st}\, c(s,t)}_{\text{bias}} + \lambda \underbrace{\sum_{s \in S \cup A_k} w_s^2}_{\text{regularization}} \tag{2}$$
$$\text{s.t.} \quad \forall t \in T, \sum_{s \in S \cup A_k} \pi_{st} = \frac{1}{|T|}, \quad \text{and} \quad \forall s \in S \cup A_k, \sum_{t \in T} \pi_{st} = w_s.$$

Once we select the set A_k to minimize the RTW objective, we then query the labels of A_k and train on $S \cup A_k$ using the weights w obtained from (2). The costs $c(s,t)$ measure some notions of dissimilarity, as discussed above. The objective function (2) captures two aspects of our problem: choosing the queries A_k, and also reweighting the augmented source points $S \cup A_k$ to be representative of T.

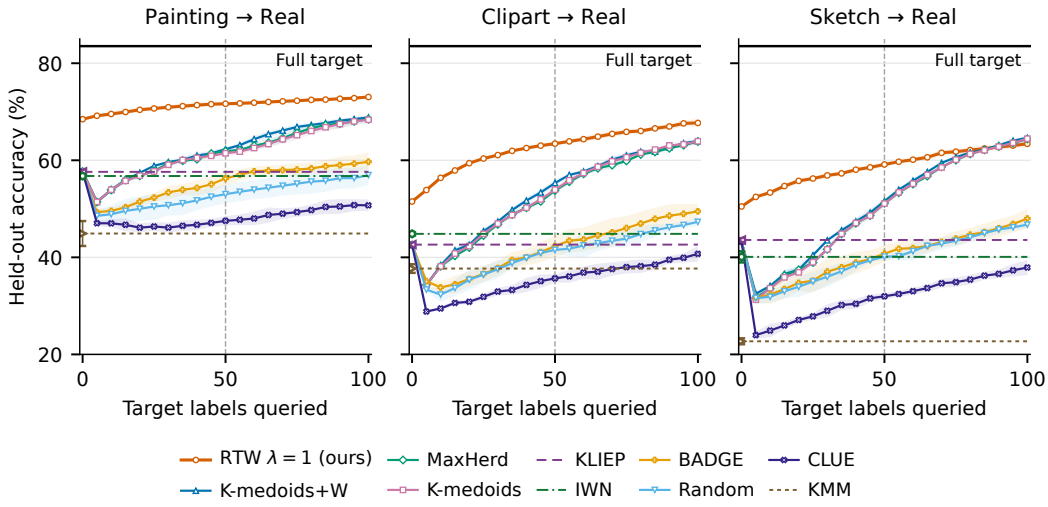

Figure 1: Results for three tasks on mini-DomainNet showing accuracy on the held-out evaluation set. RTW uses $\lambda = 1$. Full target denotes the accuracy of a classifier trained on the target query pool. Dashed horizontal baselines represent methods that do not query for additional labels. After $k = 50$ queries, our method improves accuracy by 14–21 percentage points over training on the source distribution. Our method outperforms other methods even at $k = 0$ due to source reweighting.

The two terms in the objective perform different roles. The first term in the objective is the optimal transport cost between the distribution w supported on $S \cup A_k$ and the uniform distribution over the target dataset T. This captures the bias in using the distribution on $S \cup A_k$ given by w to approximate the dataset T. However, this could concentrate too much weight on only a few points, which decreases the effective sample size and causes large statistical error. The second term is a regularization term that prevents this large statistical error.

Incorporating these two factors allows us to prove that a *reweighted empirical risk minimizer* for the (reweighted) augmented training set $S \cup A_k, w$ will generalize to the test distribution. See Theorem 3.1 and Theorem 3.2 in Section 3 for a formal statement that upper bounds the excess risk on the target distribution using an appropriate choice of the cost function c.

Algorithm with Provable Guarantees of Approximate Optimality. The generalization bound in Theorem 3.1 with the RTW objective gives statistical guarantees for our approach. However, the computational challenge of minimizing the RTW problem (1) efficiently remains. For a fixed set A_k of queries, computing the RTW objective is a convex optimization problem. The challenge is therefore to choose the query set A_k. Even in the case where $\lambda = 0$, this corresponds to the classic k-median problem, which is NP-hard in the worst-case Williamson & Shmoys (2011). In general, when $\lambda \neq 0$, the regularization term enforces some load balancing on the source weights. This is similar to the capacitated k-medians problem, which is algorithmically even more challenging than the standard uncapacitated version (Trinh, 2016). On the other hand, as $\lambda \to \infty$, this corresponds to assigning uniform weights across $S \cup A_k$. This is the setting considered by Mahmood et al. (2022), who design heuristics to minimize this objective.

Our main technical contribution is an efficient *polynomial time algorithm* that gives *provable guarantees for approximate optimality* for the RTW objective. Our key insight is to prove that the function (2) is *supermodular* for every choice of $\lambda \geq 0$. The unregularized objective when $\lambda = 0$ is supermodular since the transport map π just corresponds to the nearest neighbor assignment. When $\lambda > 0$ the transport map is no longer the nearest neighbor assignment, yet surprisingly the objective remains supermodular. To the best of our knowledge, this was not known for the regularized version of the problem. This supermodularity property allows us to prove that a simple greedy algorithm that adds to the query set A_k the point $\operatorname{argmin}_{t \in T} \mathrm{RTW}_\lambda(S \cup A_k \cup \{t\})$ achieves an approximate optimality guarantee. Hence the RTW problem (1) provides a natural objective that achieves two important goals simultaneously: it allows us to obtain statistical guarantees for active domain adaptation (through regularization), while also ensuring computational tractability (through supermodularity).

Validation. We implement our algorithm and validate it on standard domain adaptation benchmarks. On benchmarks, we see that our method is competitive with state-of-the-art methods (including those that use information about the labels to choose their queries), and even outperforms them on some datasets. Figure 1 shows three settings where our method gives a significant improvement over existing methods, and with $k = 50$ queries (out of a target query pool containing 52,252 samples) improves accuracy by 14–21 percentage points over training on the source alone.

2 Related Work

We describe some of the other related work in this section. Please see Appendix A for a more detailed overview of related work.

Discrepancy and domain adaptation. In *domain adaptation*, the learner is given access to labeled data from a source distribution $\mathcal{P}$ and unlabeled data from a target distribution $\mathcal{Q}$, with the goal of learning a model for $\mathcal{Q}$. This is similar to our problem when the number of label queries $k = 0$. Ben-David et al. (2010) give a generalization bound for a classifier learned on $\mathcal{P}$ when applied to $\mathcal{Q}$. They bound this via a notion of *discrepancy*, the $\mathcal{H}\Delta\mathcal{H}$ measure, which captures how different $\mathcal{P}$ is from $\mathcal{Q}$, with respect to the hypothesis class $\mathcal{H}$. Subsequent work defines a variety of other discrepancy measures that are tighter in certain regimes of the problem. These discrepancy measures also capture lower bounds for domain adaptation– when $\mathcal{Q}$ looks very different from $\mathcal{P}$ with respect to $\mathcal{H}$. The primary difference is that unlike standard domain adaptation, in active domain adaptation we are able to leverage additional label queries from Q, and many unlabeled samples from Q. These allow us to circumvent some of the fundamental bottlenecks in domain adaptation.

Active learning. In *active learning*, the goal of the learner is to learn a good hypothesis from a class $\mathcal{H}$ for a distribution $\mathcal{Q}$. The learner is given *unlabeled* samples from $\mathcal{Q}$, and can query the labels of a few points. This is similar to our problem when the learner is given an empty training set S. A line of work beginning with Dasgupta (2005) studies this problem in many settings, and characterizes the learnability of different hypothesis classes in a distribution-free setting (Hanneke, 2014). Our work differs from the standard viewpoint of active learning in a few ways. First, we have access to an additional training set S, that may not be representative of $\mathcal{Q}$. Thus our methods must adapt to and take advantage of this extra information. Second, we design *task agnostic* methods, that do not depend on the realized labels of the training set S or the queried points from T. This differs from traditional active learning methods, which usually depend on the labels.

Task-agnostic active domain adaptation methods. There are a few works that consider the problem of *active domain adaptation*. That is, they are given labeled samples from $\mathcal{P}$ and unlabeled samples from $\mathcal{Q}$. Their goal is to query the labels of a few representative samples from $\mathcal{Q}$ and use this to learn a good classifier for $\mathcal{Q}$ with respect to the hypothesis class $\mathcal{H}$. Our work focuses on designing *task agnostic* methods. That is, our methods choose which target points to query and find a reweighting of the augmented dataset, independently of the realized labels. This gives us a general way to augment the training set, which can be used to train a downstream model for any number of tasks. Prior work by Chattopadhyay et al. (2013a) designs a method to choose the queries and weights to minimize the *maximum mean discrepancy* to $\mathcal{Q}$. Bruns-Smith & Feller (2022) choose weights that balance worst-case bias against weight dispersion. de Mathelin et al. (2022) give a task-agnostic method based on k-medoids. Their objective is essentially the $\lambda = 0$ setting of our objective function. They give an algorithm to approximately minimize this objective via submodular maximization. Our method is similar, though establishing submodularity seems significantly less straightforward in the case where $\lambda \neq 0$. We provide more comparison to this method in the theory and experimental sections. The work of Mahmood et al. (2022) also provides a task-agnostic method that picks the next query based on the optimal transport cost; but they do not reweigh the distribution. Our contribution is to frame the RTW objective, which provably ensures generalization, while also being computationally tractable.

3 Regularized Reweighting Objective and Generalization Guarantees

In this section, we describe a general, discrepancy-based objective function for the selection of points that depends on the hypothesis class and loss function. The objective is motivated by a novel generalization guarantee. Recall that the end goal for selecting the query points and performing reweighting is to learn a good hypothesis function using the augmented dataset.

We begin with some new definitions and notation. From now on, we use index i for features and labels from the augmented dataset $S \cup A_k$ and j for the target dataset T. Let, for $h \in \mathcal{H}$, g_h be the function defined as $g_h(x) = \mathbb{E}_{Y \sim Y|x}[\ell(h(x), Y)]$. For a hypothesis class $\mathcal{H}$ and loss function ℓ, we define the following discrepancy measure between two distributions:

$$\mathrm{disc}_{\mathcal{H},\ell}(\mathcal{P}_x, \mathcal{Q}_x) := \sup_{h,h' \in \mathcal{H}} \left| \mathbb{E}_{\mathcal{P}_x}[g_h(X) - g_{h'}(X)] - \mathbb{E}_{\mathcal{Q}_x}[g_h(X) - g_{h'}(X)] \right|. \tag{3}$$

This measures the maximum error when estimating the difference in expected risks over $\mathcal{Q}_x$ of two hypotheses h, h', if we instead estimate it on a different covariate distribution $\mathcal{P}_x$. This is a fine-grained variant of the $\mathcal{Y}$-discrepancy in Mohri & Muñoz Medina (2012), replacing a single g_h by their pairwise differences, analogous to the classical notion of discrepancy (Mansour et al., 2009). It also equals the sum of the two directional transfer measures in Zhang et al. (2021).

Let ν_w be a distribution over the augmented source set $S_k = S \cup A_k$, where each $s_i \in S_k$ has weights $\nu_w(s_i) = w_i$ and label drawn from the labeling conditional distribution $Y_i \sim Y|s_i$. Let μ be the uniform empirical distribution over the target features T, and write $N = |T|$. We distinguish the realized weighted training risk from the target population risk:

$$\widehat{L}_{\nu_w}(h) = \sum_{s_i \in S_k} w_i \ell(h(s_i), Y_i), \qquad L_Q(h) = \mathbb{E}_{(X,Y)\sim\mathcal{Q}} \ell(h(X), Y) = \mathbb{E}_{\mathcal{Q}_x} g_h(X).$$

The pool conditional risk $L_\mu(h) = N^{-1} \sum_{t \in T} g_h(t)$ will serve as an intermediate quantity in the analysis. Let $\mathcal{F}_\ell = \{(x, y) \mapsto \ell(h(x), y) : h \in \mathcal{H}\}$ denote the loss-composed hypothesis class, and let $\mathrm{Pdim}(\mathcal{F}_\ell)$ be its pseudo-dimension. Our first result is a generalization bound for ERM on the realized weighted source distribution:

Theorem 3.1. *Suppose A_k and w are chosen using features only. Conditional on all feature locations, assume the labels are independent under the shared conditional distribution $Y \mid x$. Let $\hat{h} \in \arg\min_{h \in \mathcal{H}} \widehat{L}_{\nu_w}(h)$ be a minimizer of the weighted empirical risk on augmented source S_k. If $0 \le \ell \le 1$ and $\mathrm{Pdim}(\mathcal{F}_\ell) < \infty$, then there is a universal constant C such that, for every $\delta \in (0, 1)$, with conditional probability at least $1 - \delta$ over the labels,*

$$L_Q(\hat{h}) - \inf_{h \in \mathcal{H}} L_Q(h) \le \mathrm{disc}_{\mathcal{H},\ell}(\nu_w, \mathcal{Q}_x) + C\|w\|_2 \left(\sqrt{\mathrm{Pdim}(\mathcal{F}_\ell)} + \sqrt{\log \frac{2}{\delta}} \right). \tag{4}$$

The upper bound in equation (4) admits an intuitive decomposition: the first discrepancy term is a bias term, measuring the distortion in target population risk differences when using the weighted training features instead. The second term is a variance term, coming from the realistic modeling assumption that the label is not a deterministic function of the feature.

3.1 Regularized Transport Weighting

The discrepancy term can be upper bounded by a transportation problem. For $s, t \in \mathcal{X}$, define cost c to satisfy:

$$c(s, t) \ge \sup_{h,h' \in \mathcal{H}} |g_h(s) - g_{h'}(s) - g_h(t) + g_{h'}(t)|. \tag{5}$$

Intuitively, to compare the expected losses of hypotheses h and h' on point $t \in \mathcal{X}$, then $c(s, t)$ is the maximum amount of error we get if we compare their expected losses on s instead. Note here that if all functions g_h are L-Lipschitz and $\mathcal{X}$ is a metric space, then this cost can be taken to be $c(s, t) = 2L \cdot d(s, t)$.

The triangle inequality gives $\mathrm{disc}_{\mathcal{H},\ell}(\nu_w, \mathcal{Q}_x) \le \mathrm{disc}_{\mathcal{H},\ell}(\nu_w, \mu) + \mathrm{disc}_{\mathcal{H},\ell}(\mu, \mathcal{Q}_x)$. The second term is independent of the queries and weights. Bounding the first by transport cost and incorporating the variance term in Equation 4 gives the following optimization problem:

$$\min_{\pi, w \ge 0} \sum_{s \in S_k} \sum_{t \in T} \pi_{st} c(s,t) + \lambda_0 \|w\|_2, \tag{6}$$

where $\sum_{s \in S_k} \pi_{st} = \frac{1}{|T|}$ for all $t \in T$, and $w_s = \sum_{t \in T} \pi_{st}$ for all s.

By replacing the ℓ_2 regularization $\lambda_0 \|w\|_2$ with the *squared* ℓ_2 regularization term $\lambda \|w\|_2^2$, we obtain our RTW_λ objective presented in Equation 2. The squared ℓ_2 regularization has better properties (see Section 4), and an appropriate choice of λ also controls the excess population risk, up to a target-pool sampling term. This is formalized by the following theorem.

Theorem 3.2. *Under the assumptions of Theorem 3.1, suppose T consists of $N \ge 1$ i.i.d. samples from $\mathcal{Q}_x$. Let $S \cup A_k$ be nonempty and fix a cost c satisfying (5). For $\delta \in (0,1)$, take $\lambda_0 = C\left(\sqrt{\mathrm{Pdim}(\mathcal{F}_\ell)} + \sqrt{\log(4/\delta)}\right)$, with C as in Theorem 3.1. Let B be the minimum of (6), attained at (π_0^*, w_0^*) chosen using features only, and set $\lambda = \lambda_0/(2\|w_0^*\|_2)$. Let (π^*, w^*) be a minimizer of $\mathrm{RTW}_\lambda(S \cup A_k)$ chosen using features only, and let $\hat{h} \in \arg\min_{h \in \mathcal{H}} \widehat{L}_{\nu_{w^*}}(h)$. There is a universal constant C' such that, with probability at least $1 - \delta$ over the sampled features and labels,*

$$L_Q(\hat{h}) - \inf_{h \in \mathcal{H}} L_Q(h) \le 2\mathrm{RTW}_\lambda(S \cup A_k) + \frac{C'}{\sqrt{N}} \left(\sqrt{\mathrm{Pdim}(\mathcal{F}_\ell)} + \sqrt{\log \frac{4}{\delta}} \right).$$

The proof is in Appendix B.2.

For a specified L, the Wasserstein term uses only the feature locations and weights, without requiring the conditional risks themselves. The proofs of Theorems 3.1 and 3.2 appear in Appendix B.

4 Our Algorithm and Approximate Optimality Guarantees

In this section, we fix $\lambda \ge 0$ and study the computational aspects of the Regularized Transport Weighting problem in (1): given S, T, k, the costs c and $\lambda \ge 0$, find $A_k \subset T$ with $|A_k| = k$ that minimizes $\mathrm{RTW}_\lambda(S \cup A_k)$.

Recall that for a fixed set of query points $A_k \subset T$, the quantity $\mathrm{RTW}_\lambda(S \cup A_k)$ is the value of a convex program given in (2). The convex program is a regularized transport problem where the distribution supported on $S \cup A_k$ can be reweighted. This is related to the Wasserstein objective in (Mahmood et al., 2022); crucially however, we allow the distribution on the source to be reweighted (as opposed to uniform weights). While the *bias* term in our objective (2) captures the transport cost (or Wasserstein distance) between the target and the reweighted distribution w on $S \cup A_k$, the regularization term plays an important role in obtaining generalization in domain adaptation (e.g., with stochastic $Y|x$). The convex program (2) can be solved efficiently in polynomial time; moreover in Section 5 we describe a dual-based algorithm that is more efficient in practice.

The primary computational challenge is in identifying a good choice of A_k that minimizes objective (1). The problem is NP-hard even in the special case when $\lambda = 0$. The regularized transport weighting objective interpolates between two familiar regimes. When $\lambda = 0$, it reduces to k-median with S as pre-opened centers and A_k as the k additional centers, recovering k-medoids when $S = \varnothing$. For $\lambda \in (0, \infty)$, it is analogous to the more challenging capacitated k-median (Trinh, 2016), except it allows fractional assignments and incentivizes a spread load instead of constraining the max load. As $\lambda \to \infty$, the weights become uniform, and the objective converges to the uniformly weighted Wasserstein selection criterion of Mahmood et al. (2022).

4.1 Greedy Querying Algorithm

We propose a greedy algorithm (Algorithm (1)) for solving problem (1). We run for k iterations: at each iteration we select the point that brings down the objective RTW_λ the most, and add it to our set A_k. See Figure (2) to compare how our greedy RTW algorithm differs from the (greedy) k-medoids algorithm and random sampling in the way it picks the query points.

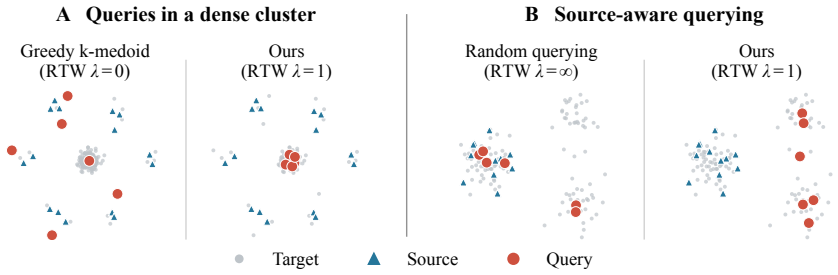

Figure 2: Illustration of our query selection method (Section 4) on two synthetic instances. When $\lambda = 0$ our algorithm reduces to the (greedy) k-medoids algorithm (de Mathelin et al., 2022). For intermediate values of λ, our algorithm greedily optimizes RTW_λ, overcoming the failures of both greedy k-medoids and random sampling.

Algorithm 1 Greedy Regularized Transport Weights

Inputs: source points $S \subseteq \mathcal{X}$, target points $T \subseteq \mathcal{X}$ and query budget k.
Hyperparameters: $\lambda \in \mathbb{R}_+$: regularization strength
Outputs: $A_k \subseteq T$: the set of points to query for labels, $w^* \in \Delta(S \cup A_k)$: weights of the augmented training set.

1: Initialize $A_0 = \emptyset$, $i = 0$.
2: **while** $i < k$ **do**
3: Compute $x_i^* = \mathrm{argmin}_{x \in T \setminus A_i} \mathrm{RTW}_\lambda(S \cup A_i \cup \{x\})$
4: $A_{i+1} \leftarrow A_i \cup \{x_i^*\}$
5: $i \leftarrow i + 1$
6: **end while**
7: Set w^* to be the optimum weights w for $\mathrm{RTW}_\lambda(S \cup A_k)$ defined in (2).
8: Output A_k, w^*.

4.2 Algorithmic Guarantees

Algorithm (1) outputs a set A_k in k iterations in polynomial time. Our main theoretical contribution is to show that Algorithm (1) outputs a set that is approximately optimal— it brings down the cost function by at least a constant times the best possible subset. The proof is given in Appendix C.2.

Theorem 4.1. *Suppose $\lambda \geq 0$, $S \neq \varnothing$, and $1 \leq k \leq |T|$ is an integer. Let $A_k^* = \mathrm{argmin}_{A \subseteq T, |A| = k} \mathrm{RTW}_\lambda(S \cup A)$. Then, Algorithm (1) given query budget k outputs a set $\hat{A}_k$ in polynomial time satisfying*

$$\mathrm{RTW}_\lambda(S) - \mathrm{RTW}_\lambda(S \cup \hat{A}_k) \geq \left(1 - \frac{1}{e}\right)(\mathrm{RTW}_\lambda(S) - \mathrm{RTW}_\lambda(S \cup A_k^*))$$

Moreover, for $\varepsilon, \eta \in (0, 1)$, suppose $\mathrm{RTW}_\lambda(S \cup A_k^*) \leq \eta \cdot \mathrm{RTW}_\lambda(S)$. Then our algorithm with query budget $k' = \min\{|T|, \lceil k \log(1/(\varepsilon\eta)) \rceil\}$ outputs $\hat{A}_{k'}$ such that

$$\mathrm{RTW}_\lambda(S \cup \hat{A}_{k'}) \leq (1 + \varepsilon) \cdot \eta \mathrm{RTW}_\lambda(S).$$

The crucial insight behind our analysis is the surprising *supermodularity* property of the function $\mathrm{RTW}_\lambda(S)$. For fixed S, the decrease $\mathrm{RTW}_\lambda(S) - \mathrm{RTW}_\lambda(S \cup A)$ is a monotone submodular function of A. The corresponding diminishing-returns inequality is: for every $S \subseteq S', x \notin S'$,

$$\mathrm{RTW}_\lambda(S) - \mathrm{RTW}_\lambda(S \cup \{x\}) \geq \mathrm{RTW}_\lambda(S') - \mathrm{RTW}_\lambda(S' \cup \{x\}).$$

See Theorem C.1 for the formal claim in Appendix C. While the supermodularity for $\lambda = 0$ is easy to verify (the transport map is just the nearest neighbor assignment), it becomes more challenging for $\lambda > 0$. The proof of the submodularity claim for the general regularized objective is new to the best of our knowledge.

The proof of submodularity involves a flow-based proof. A key change in perspective involves the function $f_S(q)$ for any non-negative vector of demands $q \in \mathbf{R}_{\geq 0}^T$, which corresponds to the optimal

regularized transport cost of matching the "target vector" q and allows us to better control the change when a new query point is added. We remark that the same proof also works when the regularizer $\lambda\|w\|_2^2$ is replaced by any separable sum $\lambda \sum_i \phi(w_i)$ where ϕ can be an arbitrary non-negative convex function.

5 Experiments

To evaluate our algorithm, we conducted experiments on two well known image datasets used for domain adaptation, OfficeHome-RSUT (Tan et al. (2020)) and mini-DomainNet (Zhou et al. (2021)).

RTW Implementation. Algorithm (1) involves solving multiple quadratic optimization problems subject to linear constraints. Our implementation solves these using gradient ascent on the unconstrained dual, which we found to be the most stable approach in our experiments. To further improve computational efficiency, we solve the optimization problems only for a randomly sampled subset of the candidate points. Further implementation details can be found in the appendix.

Baselines. We compare the performance of our proposed algorithm with that of other established techniques to mitigate the distribution shift between the source and the target. Broadly, we consider two types of baselines. Importance-weighting methods use labeled source samples and unlabeled target samples to estimate importance weights for the source data. A classifier is then trained on the labeled source samples using these weights. Active domain adaptation methods may additionally query the labels for a few selected target points, and use them during classifier training.

We consider three importance weighting baselines - **Kernel Mean Matching (KMM)** (Huang et al. (2006)), **KLIEP** (Sugiyama et al. (2007)) and **Importance Weighting Network (IWN)** (de Mathelin et al. (2022)). We consider five active domain adaptation methods - **K-medoids, K-medoids+W** (de Mathelin et al. (2022)), **MaxHerd** (Bae et al. (2025)), **BADGE** (Ash et al. (2020)) and **CLUE** (Prabhu et al. (2021)). de Mathelin et al. (2022) consider two weighting schemes - uniform weights which assign equal weight to all training samples, and balanced weights, which reweigh the points so that the original source and queried target samples contribute equally. In our experiments, we found that balanced reweighting performs best for all baselines, and so all reported results use balanced reweighting. For detailed descriptions of the baselines, see Appendix D.

5.1 Experimental Setup and Results

OfficeHome-RSUT. OfficeHome-RSUT, introduced by Tan et al. (2020), is a class-imbalanced variant of the original OfficeHome dataset (Venkateswara et al., 2017), an image dataset for domain adaptation. It consists of images from 3 domains - Real World, Clipart and Product. Each domain consists of images belonging to one of 65 object categories. We consider six tasks, corresponding to all possible source and target domains.

We use an ImageNet pre-trained ResNet-50 network (He et al., 2016) to extract 2,048-dimensional, ℓ_2-normalized embeddings of all the images in the dataset. The embeddings remain frozen throughout our experiments and we train only a linear classification head. For each method, we train a logistic-regression classifier on the labeled source samples together with any queried target samples, with the weighting prescribed by the method.

For each task, we generate five random partitions of the target dataset, using 80% of the samples as the acquisition pool and keeping 20% of the samples as a held-out evaluation set. All methods use the same partition, and each method is only allowed access to the samples in the acquisition pool. Each method is evaluated by the performance on the held-out samples. We set the query budget $K = 100$, and evaluate accuracy on the held-out set after every five queries.

Results: Figure (3) shows the accuracies evaluated at different query budgets on three selected tasks. At $K = 100$ queries, RTW achieves an accuracy of $76.73 \pm 2.61\%$ on Product$\rightarrow$ Real World and $45.29 \pm 2.47\%$ on Real World $\rightarrow$ Clipart, as compared to the best-performing baseline's accuracy of $75.86 \pm 3.45\%$ and $44.61 \pm 2.19\%$, achieved by K-medoids and BADGE respectively. In addition, RTW also outperforms other algorithms at lower query budgets. In particular, reweighting the source with our method outperforms existing importance weighting baselines even without additional queries. Additional results on all tasks can be found in Appendix D.

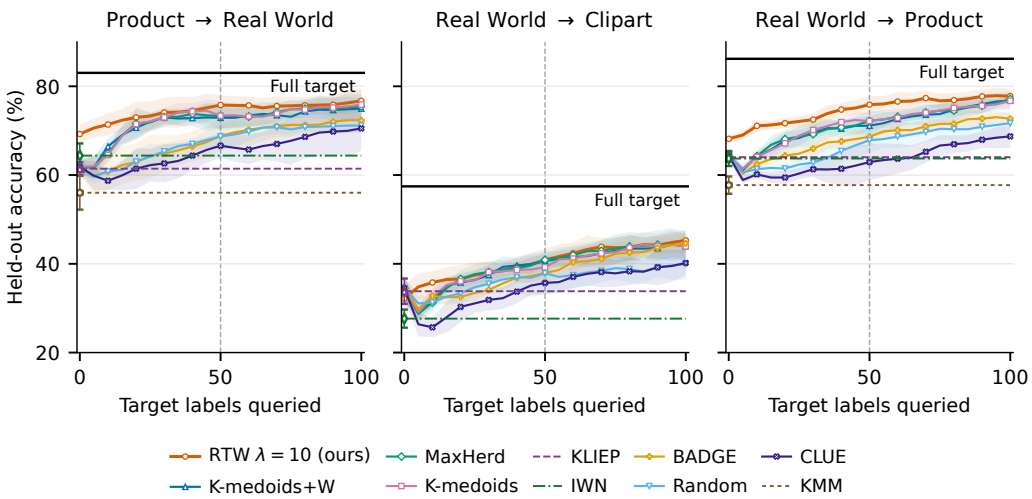

Figure 3: Results for three tasks on OfficeHome-RSUT. The full target line denotes the accuracy of the classifier trained on the target query pool. Dashed lines represent methods that do not query for additional labels. Our method outperforms existing methods, and with only a few additional queries we are able to improve the accuracy of the trained classifier significantly.

mini-DomainNet. mini-DomainNet is a subset of DomainNet containing 126 object categories across four domains introduced by Zhou et al. (2021): Clipart, Painting, Sketch, and Real. We evaluate three adaptation tasks: Clipart, Painting, and Sketch to Real. After removing duplicate images and samples with conflicting labels, we retain all source images. As in our OfficeHome-RSUT experiments, we construct five partitions of the target domain into a query pool using 80% of the images, and maintain a held-out evaluation set consisting of the remaining 20% of the images.

As in OfficeHome-RSUT, we use frozen, 2,048-dimensional, ℓ_2-normalized ResNet-50 ImageNet-V1 features, and train a weighted logistic regression on the labeled source samples augmented by the queried target samples. We set a query budget of $k = 100$, and compute accuracy on the evaluation set every five queries.

Results: Figure (1) shows our results on the mini-DomainNet dataset. At $K = 100$, RTW achieves $73.04 \pm 0.30\%$ on Painting→Real and $67.70 \pm 0.75\%$ on Clipart→Real, exceeding the strongest baseline, K-medoids+W ($68.83 \pm 0.43\%$ and $63.95 \pm 0.52\%$, respectively); its advantage is already present at $K = 0$ through source reweighting. On Sketch→Real, RTW reaches $63.40 \pm 0.62\%$, below K-medoids+W ($64.62 \pm 0.48\%$), K-medoids ($64.45 \pm 0.62\%$), and MaxHerd ($64.38 \pm 0.63\%$).

6 Conclusion

We provide a general and principled task-agnostic approach for active domain adaptation. Our notion of regularized transport weights (RTW) suggests new statistically tractable ways to leverage the geometry of the hypothesis class to compute weights between the training data and test dataset. We provide efficient algorithms to approximately minimize this objective, with provable guarantees for approximate optimality via the toolkit of submodular function maximization. This gives a simple and practical method that competes with the state of the art on a variety of applications.

AI Disclosure

In this work, we used generative AI tools in generating synthetic datasets, providing proof ideas for mathematical claims, experimental methodology, and implementing methods. We have not used generative AI tools for developing theoretical models, formulating mathematical claims, proposing and refining hypotheses, or interpreting results, and translation and qualitative data analysis are not applicable to this work. We have reviewed all AI-assisted work. All proofs have been verified and written by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Martin Anthony and Peter L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999.

Jordan T. Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. In *International Conference on Learning Representations*, 2020.

Pranjal Awasthi, Corinna Cortes, and Mehryar Mohri. Best-effort adaptation. *Annals of Mathematics and Artificial Intelligence*, 92(2):393–438, 2024. URL https://arxiv.org/abs/2305.05816.

Kyriakos Axiotis, Vincent Cohen-Addad, Monika Henzinger, Sammy Jerome, Vahab Mirrokni, David Saulpic, David Woodruff, and Michael Wunder. Data-efficient learning via clustering-based sensitivity sampling: Foundation models and beyond. In *Proceedings of ICML*, volume 235, pp. 2086–2107, 2024. URL https://proceedings.mlr.press/v235/axiotis24a.html.

Wonho Bae, Junhyug Noh, and Danica J. Sutherland. Generalized coverage for more robust low-budget active learning. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 318–334, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-73010-8.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79:151–175, 2010. doi: 10.1007/s10994-009-5152-4.

Dimitri P. Bertsekas. *Convex Optimization Theory*. Athena Scientific, 2009.

Alina Beygelzimer, Sanjoy Dasgupta, and John Langford. Importance weighted active learning. In *Proceedings of the 26th International Conference on Machine Learning*, 2009. doi: 10.1145/1553374.1553381.

Zalán Borsos, Mojmír Mutný, Marco Tagliasacchi, and Andreas Krause. Data summarization via bilevel optimization. *Journal of Machine Learning Research*, 25(73):1–53, 2024. URL https://jmlr.org/papers/v25/21-1132.html.

David A. Bruns-Smith and Avi Feller. Outcome assumptions and duality theory for balancing weights. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 11037–11055. PMLR, 2022. URL https://proceedings.mlr.press/v151/bruns-smith22a.html.

Rita Chattopadhyay, Wei Fan, Ian Davidson, Sethuraman Panchanathan, and Jieping Ye. Joint transfer and batch-mode active learning. In *Proceedings of the 30th International Conference on Machine Learning*, 2013a.

Rita Chattopadhyay, Zheng Wang, Wei Fan, Ian Davidson, Sethuraman Panchanathan, and Jieping Ye. Batch mode active sampling based on marginal probability distribution matching. *ACM Transactions on Knowledge Discovery from Data*, 7(3):13:1–13:25, 2013b. doi: 10.1145/2513092.2513094.

Sebastian Claici, Aude Genevay, and Justin Solomon. Wasserstein measure coresets. *arXiv preprint arXiv:1805.07412*, 2018.

Corinna Cortes, Mehryar Mohri, and Andrés Muñoz Medina. Adaptation based on generalized discrepancy. *Journal of Machine Learning Research*, 20(1):1–30, 2019. URL https://jmlr.org/papers/v20/15-192.html.

Sanjoy Dasgupta. Coarse sample complexity bounds for active learning. In Y. Weiss, B. Schölkopf, and J. Platt (eds.), *Advances in Neural Information Processing Systems*, volume 18. MIT Press, 2005. URL https://proceedings.neurips.cc/paper_files/paper/2005/file/6e82873a32b95af115de1c414a1849cb-Paper.pdf.

Antoine de Mathelin, Guillaume Richard, François Deheeger, Mathilde Mougeot, and Nicolas Vayatis. Adversarial weighting for domain adaptation in regression. In *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 49–56, 2021. doi: 10.1109/ICTAI52525.2021.00015.

Antoine de Mathelin, François Deheeger, Mathilde Mougeot, and Nicolas Vayatis. Discrepancy-based active learning for domain adaptation. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=p98WJxUC3Ca.

Antoine de Mathelin, Francois Deheeger, Mathilde Mougeot, and Nicolas Vayatis. Fast and accurate importance weighting for correcting sample bias. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2022, Grenoble, France, September 19–23, 2022, Proceedings, Part I*, pp. 659–674, Berlin, Heidelberg, 2022. Springer-Verlag. ISBN 978-3-031-26386-6. doi: 10.1007/978-3-031-26387-3_40. URL https://doi.org/10.1007/978-3-031-26387-3_40.

Raaz Dwivedi and Lester Mackey. Kernel thinning. In *Proceedings of the 34th Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pp. 1753, 2021. URL https://proceedings.mlr.press/v134/dwivedi21a.html.

Bo Fu, Zhangjie Cao, Jianmin Wang, and Mingsheng Long. Transferable query selection for active domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7272–7281, 2021.

Steve Hanneke. Theory of disagreement-based active learning. *Found. Trends Mach. Learn.*, 7(2–3): 131–309, June 2014. ISSN 1935-8237. doi: 10.1561/2200000037. URL https://doi.org/10.1561/2200000037.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Jiayuan Huang, Arthur Gretton, Karsten M. Borgwardt, Bernhard Schölkopf, and Alex J. Smola. Correcting sample selection bias by unlabeled data. In *Advances in Neural Information Processing Systems*, 2006.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton Earnshaw, Imran Haque, Sara M Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. Wilds: A benchmark of in-the-wild distribution shifts. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5637–5664. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/koh21a.html.

Seiichi Kuroki, Nontawat Charoenphakdee, Han Bao, Junya Honda, Issei Sato, and Masashi Sugiyama. Unsupervised domain adaptation based on source-guided discrepancy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 4122–4129, 2019. doi: 10.1609/aaai.v33i01.33014122.

Changsheng Li, Handong Ma, Zhao Kang, Ye Yuan, Xiao-Yu Zhang, and Guoren Wang. On deep unsupervised active learning. In *Proceedings of IJCAI*, pp. 2626–2632, 2020. doi: 10.24963/ijcai.2020/364.

Ziang Liu, Xue Jiang, Hanbin Luo, Weili Fang, Jiajing Liu, and Dongrui Wu. Pool-based unsupervised active learning for regression using iterative representativeness-diversity maximization (iRDM). *arXiv preprint arXiv:2003.07658*, 2020. URL https://arxiv.org/abs/2003.07658.

Rafid Mahmood, Sanja Fidler, and Marc T. Law. Low-budget active learning via wasserstein distance: An integer programming approach. In *International Conference on Learning Representations*, 2022.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. In *Conference on Learning Theory*, 2009.

Mehryar Mohri and Andrés Muñoz Medina. New analysis and algorithm for learning with drifting distributions. In *Algorithmic Learning Theory*, pp. 124–138, 2012. doi: 10.1007/978-3-642-34106-9_13.

Alexander Munteanu, Chris Schwiegelshohn, Christian Sohler, and David P. Woodruff. On coresets for logistic regression. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://arxiv.org/abs/1805.08571.

George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 14:265–294, 1978. doi: 10.1007/BF01588971.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Viraj Prabhu, Arjun Chandrasekaran, Kate Saenko, and Judy Hoffman. Active domain adaptation via clustering uncertainty-weighted embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8505–8514, 2021. URL https://openaccess.thecvf.com/content/ICCV2021/html/Prabhu_Active_Domain_Adaptation_via_Clustering_Uncertainty-Weighted_Embeddings_ICCV_2021_paper.html.

Joaquin Quiñonero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D. Lawrence (eds.). *Dataset Shift in Machine Learning*. Neural Information Processing series. MIT Press, Cambridge, MA, 2008. ISBN 978-0-262-17005-5.

Ievgen Redko, Amaury Habrard, and Marc Sebban. Theoretical analysis of domain adaptation with optimal transport. In *Machine Learning and Knowledge Discovery in Databases*, pp. 737–753, 2017. doi: 10.1007/978-3-319-71246-8_45. URL https://arxiv.org/abs/1610.04420.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018.

Changjian Shui, Fan Zhou, Christian Gagné, and Boyu Wang. Deep active learning: Unified and principled method for query and training. In *Proceedings of AISTATS*, volume 108, pp. 1308–1318, 2020. URL https://proceedings.mlr.press/v108/shui20a.html.

Jong-Chyi Su, Yi-Hsuan Tsai, Kihyuk Sohn, Buyu Liu, Subhransu Maji, and Manmohan Chandraker. Active adversarial domain adaptation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 739–748, 2020.

Masashi Sugiyama, Shinichi Nakajima, Hisashi Kashima, Paul von Bünau, and Motoaki Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Proceedings of the 21st International Conference on Neural Information Processing Systems*, NIPS'07, pp. 1433–1440, Red Hook, NY, USA, 2007. Curran Associates Inc. ISBN 9781605603520.

Shuhan Tan, Xingchao Peng, and Kate Saenko. Class-imbalanced domain adaptation: An empirical odyssey. In *Computer Vision – ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part I*, pp. 585–602, Berlin, Heidelberg, 2020. Springer-Verlag. ISBN 978-3-030-66414-5. doi: 10.1007/978-3-030-66415-2_38. URL https://doi.org/10.1007/978-3-030-66415-2_38.

Khoa Trinh. A survey of algorithms for capacitated k-median problems. Technical report, Department of Computer Science, University of Maryland, 2016. URL https://www.cs.umd.edu/sites/default/files/scholarly_papers/Trinh.pdf.

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5385–5394, 2017. doi: 10.1109/CVPR.2017.572.

Tom J. Viering, Jesse H. Krijthe, and Marco Loog. Nuclear discrepancy for single-shot batch active learning. *Machine Learning*, 2019.

David P. Williamson and David B. Shmoys. *The Design of Approximation Algorithms*. Cambridge University Press, New York, NY, USA, 2011. ISBN 978-0-521-19527-0.

Binhui Xie, Longhui Yuan, Shuang Li, Chi Harold Liu, Xinjing Cheng, and Guoren Wang. Active learning for domain adaptation: An energy-based approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 8708–8716, 2022. doi: 10.1609/aaai.v36i8.20850.

Mixue Xie, Shuang Li, Rui Zhang, and Chi Harold Liu. Dirichlet-based uncertainty calibration for active domain adaptation. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2302.13824.

Haoyun Yin, Yixuan Qiu, and Xiao Wang. Wasserstein coreset via sinkhorn loss. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=DrMCDS88IL.

Yao-Liang Yu and Csaba Szepesvári. Analysis of kernel mean matching under covariate shift. In *Proceedings of the 29th International Conference on Machine Learning*, 2012. URL https://icml.cc/2012/papers/330.pdf.

Guojun Zhang, Han Zhao, Yaoliang Yu, and Pascal Poupart. Quantifying and improving transferability in domain generalization. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://papers.nips.cc/paper/2021/hash/5adaacd4531b78ff8b5cedfe3f4d5212-Abstract.html.

Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael I. Jordan. Bridging theory and algorithm for domain adaptation. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7404–7413, 2019. URL https://proceedings.mlr.press/v97/zhang19i.html.

Yuchen Zhang, Mingsheng Long, Jianmin Wang, and Michael I. Jordan. On localized discrepancy for domain adaptation. *arXiv preprint arXiv:2008.06242*, 2020. URL https://arxiv.org/abs/2008.06242.

Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain adaptive ensemble learning. *IEEE Transactions on Image Processing*, 30:8008–8018, 2021. doi: 10.1109/TIP.2021.3112012.

CONTENTS

1 Introduction **1**

1.1 Model and Our Contributions. 2

2 Related Work **4**

3 Regularized Reweighting Objective and Generalization Guarantees **5**

3.1 Regularized Transport Weighting 5

4 Our Algorithm and Approximate Optimality Guarantees **6**

4.1 Greedy querying algorithm 6

4.2 Algorithmic Guarantees 7

5 Experiments **8**

5.1 Experimental Setup and Results 8

6 Conclusion **9**

A Extended Related Work **14**

B Proofs of the Risk Bounds **16**

B.1 Proof of Theorem 3.1 16

B.2 Risk Bound via Squared Regularization 17

C Proofs from Section 4 **18**

C.1 Supermodularity Proof 18

C.2 Greedy approximation guarantee 21

D Additional Experimental Details **21**

A EXTENDED RELATED WORK

Discrepancy and domain adaptation. Ben-David et al. (2010) use the $\mathcal{H}\Delta\mathcal{H}$ divergence to obtain a domain-adaptation generalization bound. Mansour et al. (2009) extend the analysis to general losses and motivate source reweighting. Source-guided discrepancy fixes one hypothesis in the classical pairwise discrepancy to a source-risk minimizer, using source labels without requiring target labels (Kuroki et al., 2019). Localized discrepancy instead restricts attention to hypotheses with small source risk (Zhang et al., 2020). Generalized discrepancy allows the source weights to depend on the hypothesis rather than fixing one weighting for the entire class (Cortes et al., 2019). For multiclass adaptation, Zhang et al. (2019) introduce margin disparity discrepancy and use it to derive an adversarial learning algorithm. Connecting discrepancy to label acquisition, de Mathelin et al. (2022) use a label-consistent localized class to obtain single-shot active domain adaptation bounds and a geometric K-medoids query rule.

Redko et al. (2017) obtain Wasserstein-based adaptation bounds, including a setting with both source and target labels. The $\mathcal{Y}$-discrepancy of Mohri & Muñoz Medina (2012) measures the largest absolute risk difference over a hypothesis class between two joint input–label distributions, allowing stochastic labels and general bounded losses. Zhang et al. (2021) instead compare excess risks; un-

der covariate shift and a common hypothesis class, our conditional-risk discrepancy equals the sum of their two directional transfer measures.

Balancing weights. Kernel mean matching corrects sample-selection bias by matching weighted source and target feature means, without estimating either density (Huang et al., 2006). Yu & Szepesvári (2012) give high-probability bounds for this estimator under regularity conditions on the regression function and kernel. Bruns-Smith & Feller (2022) balance squared IPM bias against weight dispersion; for a uniform empirical source, the dispersion term is a squared ℓ_2 penalty. Their weights are placed on a fixed support, while our support $S \cup A_k$ changes with the selected target batch.

Weights can also be learned together with a predictor: WANN uses adversarial training with a $\mathcal{Y}$-discrepancy objective to learn a regressor and weights over labeled source and target samples (de Mathelin et al., 2021). Best-Effort Adaptation provides discrepancy-based bounds that hold uniformly over the weights used to combine labeled source and target data (Awasthi et al., 2024). Both take the labeled target sample as given; we also choose which target labels to acquire. Weights play a different role in importance-weighted active learning (IWAL): inverse query-probability weights correct the bias introduced by sequential active sampling (Beygelzimer et al., 2009), whereas our weights are chosen by distribution matching for a single-shot batch.

Active domain adaptation. JO-TAL jointly reweights source samples and selects a target batch through an MMD-based convex relaxation (Chattopadhyay et al., 2013a). We additionally optimize the queried samples' training weights and penalize weight concentration.

AADA combines adversarial adaptation with acquisition scores based on predictive uncertainty and domain-discriminator outputs (Su et al., 2020). TQS uses committee disagreement, uncertainty, and domainness, with randomized selection for diversity (Fu et al., 2021). CLUE incorporates diversity through uncertainty-weighted clustering of target embeddings (Prabhu et al., 2021). EADA uses energy-based scores to account for both domain shift and instance uncertainty (Xie et al., 2022). To address uncertainty miscalibration under shift, DUC uses Dirichlet predictions and a two-round selection rule (Xie et al., 2023). Our acquisition rule instead uses only feature geometry once the representation is fixed.

Batch active learning. Chattopadhyay et al. (2013b) select batches by matching the marginal distributions of labeled and unlabeled data. Viering et al. (2019) analyze single-shot selection through MMD and discrepancy bounds and introduce nuclear discrepancy under an optimistic probabilistic analysis. Shui et al. (2020) use a Wasserstein-based objective while alternating neural-network training and batch querying.

The core-set approach of Sener & Savarese (2018) selects batches by geometric coverage in a learned representation. BADGE combines uncertainty and diversity by selecting diverse gradient embeddings (Ash et al., 2020). iRDM selects regression samples without labels by balancing representativeness and diversity (Liu et al., 2020). DUAL learns an encoder–decoder representation and selects samples that preserve input patterns and cluster structure without supervision (Li et al., 2020). We combine feature-only batch selection with adaptation from a source sample and regularized training weights.

Coresets and distribution matching. For logistic regression, Munteanu et al. (2018) construct sensitivity-sampling coresets under a data-complexity condition, approximating an already labeled training objective. Borsos et al. (2024) formulate weighted coreset construction as cardinality-constrained bilevel optimization of a model-training objective. Axiotis et al. (2024) combine clustering and sensitivity sampling to approximate average loss under Hölder continuity in an embedding. Our acquisition objective uses feature geometry rather than observed training losses, since queries must be selected before their labels are available.

Distributional coresets approximate a measure rather than a particular training objective. Kernel thinning compresses a sample while controlling worst-case integration error over an RKHS (Dwivedi & Mackey, 2021). Wasserstein measure coresets use transport distance to control families of downstream objectives (Claici et al., 2018). Yin et al. (2025) optimize support locations with a Sinkhorn loss; their entropic transport regularization differs from our penalty on training weights. Closer to

our query-selection problem, Mahmood et al. (2022) select uniformly weighted batches by discrete Wasserstein minimization via integer programming. We instead learn regularized weights over both source and queried target points.

B Proofs of the Risk Bounds

B.1 Proof of Theorem 3.1

Proof. Condition on the feature locations in S and T, and on the feature-only choice of A_k and w. All probabilities and expectations over training labels below are conditional on these features and choices. Write $S_k = S \cup A_k$ and $w_i = \nu_w(s_i)$ for $s_i \in S_k$, so $\sum_i w_i = 1$. The training loss is $\widehat{L}_{\nu_w}(h) = \sum_i w_i \ell(h(s_i), Y_i)$, while the population risk is $L_Q(h) = \mathbb{E}_{Q_x} g_h(X)$. Conditional on the features, the labels Y_i are independent with the shared conditional law $Y \mid s_i$. Write $Z_h = \sum_i w_i\{\ell(h(s_i), Y_i) - g_h(s_i)\}$. For any $h, h' \in \mathcal{H}$, the definition of discrepancy and the triangle inequality give

$$\left|[L_Q(h) - L_Q(h')] - [\widehat{L}_{\nu_w}(h) - \widehat{L}_{\nu_w}(h')]\right| \le \mathrm{disc}_{\mathcal{H},\ell}(\nu_w, \mathcal{Q}_x) + 2 \sup_{h \in \mathcal{H}} |Z_h|.$$

We first bound the noise term uniformly over $\mathcal{H}$.

Symmetrization. Let Y_i' be independent copies of Y_i from the same conditional distributions, independent across i, and let σ_i be independent Rademacher random variables. Since $g_h(s_i) = \mathbb{E}_{Y_i'} \ell(h(s_i), Y_i')$, using Jensen's inequality and triangle inequality, we have

$$\begin{aligned}
\mathbb{E}_Y \sup_h |Z_h| &\le \mathbb{E}_{Y,Y'} \sup_h \left| \sum_i w_i \{\ell(h(s_i), Y_i) - \ell(h(s_i), Y_i')\} \right| \\
&= \mathbb{E}_{Y,Y',\sigma} \sup_h \left| \sum_i w_i \sigma_i \{\ell(h(s_i), Y_i) - \ell(h(s_i), Y_i')\} \right| \\
&\le 2\mathbb{E}_{Y,\sigma} \sup_h \left| \sum_i w_i \sigma_i \ell(h(s_i), Y_i) \right|.
\end{aligned}$$

Weighted complexity bound. Conditional on the labels, consider the stochastic process $R_f = \sum_i w_i \sigma_i f(s_i, Y_i)$ indexed by functions $f \in \mathcal{F}_\ell$, and define the probability measure

$$Q_w = \frac{1}{\|w\|_2^2} \sum_i w_i^2 \delta_{(s_i, Y_i)}.$$

Define $\|f - f'\|_{L_2(Q_w)} = \left(\mathbb{E}_{(x,y)\sim Q_w}[(f(x,y) - f'(x,y))^2]\right)^{1/2}$. For every $s \in \mathbb{R}$,

$$\begin{aligned}
\mathbb{E}_\sigma \exp\{s(R_f - R_{f'})\} &= \prod_i \mathbb{E}_{\sigma_i} \exp\{s w_i \sigma_i [f(s_i, Y_i) - f'(s_i, Y_i)]\} \\
&= \prod_i \cosh(s w_i [f(s_i, Y_i) - f'(s_i, Y_i)]) \\
&\le \exp\left\{ \frac{s^2}{2} \sum_i w_i^2 [f(s_i, Y_i) - f'(s_i, Y_i)]^2 \right\} \\
&= \exp\left\{ \frac{s^2 \|w\|_2^2}{2} \|f - f'\|_{L_2(Q_w)}^2 \right\}.
\end{aligned}$$

The first equality uses independence; the second uses the equal probabilities of the two signs and $\cosh v = (e^v + e^{-v})/2$. The last equality follows from the definition of Q_w, giving the sub-Gaussian increment bound. The inequality $\cosh v \le e^{v^2/2}$ follows by comparing power series: since $(2k)! \ge 2^k k!$,

$$\cosh v = \sum_{k=0}^{\infty} \frac{v^{2k}}{(2k)!} \le \sum_{k=0}^{\infty} \frac{v^{2k}}{2^k k!} = e^{v^2/2}.$$

Let $\mathcal{N}(\varepsilon, \mathcal{F}_\ell, L_2(Q))$ be the covering number at radius ε under $L_2(Q)$. The pseudo-dimension covering bound for a class in $[0,1]$ gives, for every probability measure Q,

$$\log \mathcal{N}(\varepsilon, \mathcal{F}_\ell, L_2(Q)) \le C_0 \operatorname{Pdim}(\mathcal{F}_\ell) \log \frac{C_0}{\varepsilon}, \qquad 0 < \varepsilon \le 1,$$

where C_0 is universal (Anthony & Bartlett, 1999). A cover of $\mathcal{F}_\ell$ and its negation covers $\mathcal{F}_\ell \cup (-\mathcal{F}_\ell)$ with at most twice as many balls. Applying the Dudley entropy integral to this signed class therefore gives

$$\begin{aligned}\mathbb{E}_\sigma \sup_{f \in \mathcal{F}_\ell} |R_f| &\le C_1 \|w\|_2 \left[1 + \int_0^1 \sqrt{\log\left(2\mathcal{N}(\varepsilon, \mathcal{F}_\ell, L_2(Q_w))\right)}\, d\varepsilon\right] \\ &\le C_2 \|w\|_2 \left(\sqrt{\operatorname{Pdim}(\mathcal{F}_\ell)} + 1\right).\end{aligned}$$

The constant term accounts for the coarse scale of the signed class, whose $L_2(Q_w)$-diameter is at most two. The integral is finite because $\int_0^1 \sqrt{\log(C_0/\varepsilon)}\, d\varepsilon < \infty$. This bound holds for every realization of the labels, so symmetrization implies

$$\mathbb{E}_Y \sup_h |Z_h| \le 2C_2 \|w\|_2 \left(\sqrt{\operatorname{Pdim}(\mathcal{F}_\ell)} + 1\right).$$

Concentration over labels. Changing one label Y_i changes $\sup_h |Z_h|$ by at most w_i, since the loss lies in $[0,1]$. McDiarmid's inequality consequently gives

$$\mathbb{P}\left\{\sup_h |Z_h| - \mathbb{E}_Y \sup_h |Z_h| \ge s\right\} \le \exp\left(-\frac{2s^2}{\|w\|_2^2}\right).$$

Taking $s = \|w\|_2 \sqrt{\frac{1}{2}\log(1/\delta)}$ bounds the centered deviation with probability at least $1-\delta$. Since $\sqrt{\log(2/\delta)}$ absorbs both the constant term and $\sqrt{\log(1/\delta)}$, we obtain, for a universal C,

$$\sup_h |Z_h| \le C \|w\|_2 \left(\sqrt{\operatorname{Pdim}(\mathcal{F}_\ell)} + \sqrt{\log \frac{2}{\delta}}\right).$$

For every $h \in \mathcal{H}$, the initial pairwise bound and the optimality of $\hat{h}$ give

$$\begin{aligned}L_Q(\hat{h}) - L_Q(h) &\le \widehat{L}_{\nu_w}(\hat{h}) - \widehat{L}_{\nu_w}(h) + \operatorname{disc}_{\mathcal{H},\ell}(\nu_w, \mathcal{Q}_x) + 2 \sup_{h' \in \mathcal{H}} |Z_{h'}| \\ &\le \operatorname{disc}_{\mathcal{H},\ell}(\nu_w, \mathcal{Q}_x) + 2 \sup_{h' \in \mathcal{H}} |Z_{h'}|.\end{aligned}$$

Taking the supremum over h on the left and absorbing the factor of two into the universal constant proves the theorem; no target-risk minimizer need exist. □

B.2 Risk Bound via Squared Regularization

Proof of Theorem 3.2. Use B, (π_0^*, w_0^*), and $\lambda = \lambda_0/(2\|w_0^*\|_2)$ as in the statement. For every feasible (π, w), subtracting the unsquared objective from the squared objective and substituting this choice of λ gives

$$\lambda \|w\|_2^2 - \lambda_0 \|w\|_2 = \lambda\big(\|w\|_2^2 - 2\|w_0^*\|_2 \|w\|_2\big).$$

Adding the constant $\lambda \|w_0^*\|_2^2$ completes the square:

$$\lambda\big(\|w\|_2^2 + \|w_0^*\|_2^2 - 2\|w_0^*\|_2 \|w\|_2\big) = \lambda\big(\|w\|_2 - \|w_0^*\|_2\big)^2 \ge 0.$$

The two objectives therefore satisfy

$$\underbrace{\Big(\sum_{s,t} \pi_{st} c(s,t) + \lambda \|w\|_2^2\Big)}_{(2)} + \underbrace{\lambda \|w_0^*\|_2^2}_{\text{constant}}$$

$$= \underbrace{\Big(\sum_{s,t} \pi_{st} c(s,t) + \lambda_0 \|w\|_2\Big)}_{(6)} + \underbrace{\lambda\big(\|w\|_2 - \|w_0^*\|_2\big)^2}_{\ge 0}.$$

The unsquared objective is at least B, and the remaining square is nonnegative. At (π_0^*, w_0^*), these terms equal B and zero, respectively. Since the added constant does not depend on (π, w), this pair also minimizes the squared objective. Every squared minimizer attains the same shifted value B, so it must also attain the unsquared minimum B. Since the transport cost is nonnegative,

$$\mathrm{RTW}_\lambda(S_k) = \sum_{s,t} \pi^*_{0,st} c(s,t) + \frac{\lambda_0}{2}\|w_0^*\|_2 \quad \Longrightarrow \quad \mathrm{RTW}_\lambda(S_k) \le B \le 2\mathrm{RTW}_\lambda(S_k).$$

Let (π^*, w^*) and $\hat{h}$ be as in the statement, and let $S_k = S \cup A_k$. The cost condition (5) and the transport marginal constraints give $\mathrm{disc}_{\mathcal{H},\ell}(\nu_{w^*}, \mu) \le \sum_{s,t} \pi^*_{st} c(s,t)$. Theorem 3.1, applied with failure probability $\delta/2$, and the triangle inequality for discrepancy therefore give, with conditional probability at least $1 - \delta/2$,

$$\begin{aligned} L_Q(\hat{h}) - \inf_{h \in \mathcal{H}} L_Q(h) &\le \sum_{s,t} \pi^*_{st} c(s,t) + \lambda_0 \|w^*\|_2 + \mathrm{disc}_{\mathcal{H},\ell}(\mu, \mathcal{Q}_x) \\ &= B + \mathrm{disc}_{\mathcal{H},\ell}(\mu, \mathcal{Q}_x) \\ &\le 2\mathrm{RTW}_\lambda(S_k) + \mathrm{disc}_{\mathcal{H},\ell}(\mu, \mathcal{Q}_x). \end{aligned}$$

It remains to bound the discrepancy between the target pool and population. By definition, $\mathrm{disc}_{\mathcal{H},\ell}(\mu, \mathcal{Q}_x) \le 2 \sup_h |L_\mu(h) - L_Q(h)|$. Introduce auxiliary labels $\widetilde{Y}_j \sim \mathcal{Q}_{Y|X=t_j}$ independently conditional on T, and independent Rademacher signs σ_j. Symmetrization over the i.i.d. target features, followed by Jensen's inequality over the auxiliary labels, yields

$$\begin{aligned} \mathbb{E}_T \sup_h |L_\mu(h) - L_Q(h)| &\le 2\mathbb{E}_{T,\sigma} \sup_h \left| \frac{1}{N} \sum_{j=1}^N \sigma_j g_h(t_j) \right| \\ &\le 2\mathbb{E}_{T,\widetilde{Y},\sigma} \sup_h \left| \frac{1}{N} \sum_{j=1}^N \sigma_j \ell(h(t_j), \widetilde{Y}_j) \right| \\ &\le \frac{2C_2}{\sqrt{N}} \left(\sqrt{\mathrm{Pdim}(\mathcal{F}_\ell)} + 1 \right), \end{aligned}$$

where the last step is the weighted complexity bound from the proof of Theorem 3.1, with uniform weights $1/N$. Changing one target feature changes $\sup_h |L_\mu(h) - L_Q(h)|$ by at most $1/N$. McDiarmid's inequality thus gives, for a universal constant C', with probability at least $1 - \delta/2$ over T,

$$\mathrm{disc}_{\mathcal{H},\ell}(\mu, \mathcal{Q}_x) \le \frac{C'}{\sqrt{N}} \left(\sqrt{\mathrm{Pdim}(\mathcal{F}_\ell)} + \sqrt{\log \frac{4}{\delta}} \right).$$

The label bound holds conditionally for every feature realization. Integrating this bound and taking a union bound with the feature event proves the stated probability $1 - \delta$; the two events need not be independent. □

C Proofs from Section 4

In this section we prove the results in Section 4 i.e. Theorem 4.1. Theorem 4.1 follows from the fact that RTW_λ is a *supermodular* function.

Theorem C.1. *For every $S \subseteq S'$, $x \notin S'$, we have*

$$\mathrm{RTW}_\lambda(S) - \mathrm{RTW}_\lambda(S \cup \{x\}) \ge \mathrm{RTW}_\lambda(S') - \mathrm{RTW}_\lambda(S' \cup \{x\}).$$

C.1 Supermodularity Proof

We first prove Theorem C.1. We in fact show that a more general function is supermodular. Let $T \subseteq \mathcal{X}$ be the target set of points, and $q \in \mathbb{R}^T_{\ge 0}$ be a vector over T with nonnegative entries. For

any finite nonempty set of points S, define the function $f_S : \mathbb{R}^T_{\geq 0} \to \mathbb{R}$ as

$$f_S(q) = \min_{\pi_{st} \geq 0, w_s} \sum_{s \in S} \sum_{t \in T} \pi_{st} c(s,t) + \lambda \sum_{s \in S} w_s^2$$
$$\text{subject to } \sum_{s \in S} \pi_{st} = q_t \forall t \in T,$$
$$\sum_{t \in T} \pi_{st} = w_s, \forall s \in S$$

where $c(s,t)$ is the cost between points $s \in S$ and $t \in T$ as defined in equation (2), and $\lambda > 0$. We then have that $\mathrm{RTW}_\lambda(S) = f_S(\mu)$ where $\mu \in \mathbb{R}^T_{>0}$ is such that $\mu_t = \frac{1}{|T|}$.

We prove that $f_S(q)$ is supermodular for any value of q.

Theorem C.2. *For all $S \subset S'$ and point $x \notin S'$,*

$$f_S(q) - f_{S \cup \{x\}}(q) \geq f_{S'}(q) - f_{S' \cup \{x\}}(q)$$

for all $q \in \mathbb{R}^T_{>0}$.

We will use the following lemmas.

Lemma C.3. *For a set S and a point $x \notin S$,*

$$f_{S \cup \{x\}}(q) = \min_{0 \leq \delta \leq q} f_S(q - \delta) + \sum_{t \in T} c(x,t) \delta_t + \lambda \left(\sum_{t \in T} \delta_t \right)^2$$

for all $q \in \mathbb{R}^T_{\geq 0}$.

Lemma C.4. *The function f_S is differentiable for all $q > 0$. Furthermore, for sets S, S' such that $S \subset S'$,*

$$\frac{\partial f_S}{\partial q_t}(q) \geq \frac{\partial f_{S'}}{\partial q_t}(q).$$

for all $t \in T$, wherever f_S is differentiable.

We first prove Theorem C.2 assuming Lemma C.3 and Lemma C.4.

Proof of Theorem C.2. Using Lemma C.3, we can write

$$\begin{aligned} f_S(q) - f_{S \cup \{x\}}(q) &= \max_{0 \leq \delta \leq q} f_S(q) - f_S(q - \delta) - \sum_{t \in T} \delta_t c(x,t) - \lambda \left(\sum_{t \in T} \delta_t \right)^2 \\ &= \max_{0 \leq \delta \leq q} \sum_{t \in T} \int_0^1 \delta_t \frac{\partial f_S}{\partial q_t}(q - \tau\delta) \mathrm{d}\tau - \sum_{t \in T} \delta_t c(x,t) - \lambda \left(\sum_{t \in T} \delta_t \right)^2 \\ &= \max_{0 \leq \delta \leq q} \Delta_S(q, \delta) \end{aligned}$$

where $\Delta_S(q,\delta) = \sum_{t \in T} \int_0^1 \delta_t \frac{\partial f_S}{\partial q_t}(q - \tau\delta) \, \mathrm{d}\, \tau - \sum_{t \in T} \delta_t c(x,t) - \lambda \left(\sum_{j \in T} \delta_j \right)^2$. If $q - \delta$ has zero coordinates, the integral is understood as the limit as its upper endpoint tends to 1; continuity of f_S gives the same identity.

By Lemma C.4, for any δ, we have that

$$\begin{aligned} &\sum_{t \in T} \int_0^1 \delta_t \frac{\partial f_S}{\partial q_t}(q - \tau\delta) \, \mathrm{d}\, \tau \geq \sum_{t \in T} \int_0^1 \delta_t \frac{\partial f_{S'}}{\partial q_t}(q - \tau\delta) \, \mathrm{d}\, \tau \\ \implies &\Delta_S(q,\delta) \geq \Delta_{S'}(q,\delta) \\ \implies &\max_{0 \leq \delta \leq q} \Delta_S(q,\delta) \geq \max_{0 \leq \delta \leq q} \Delta_{S'}(q,\delta) \\ \implies &f_S(q) - f_{S \cup \{x\}}(q) \geq f_{S'}(q) - f_{S' \cup \{x\}}(q). \end{aligned}$$

□

We now give the proof of the helper lemmas.

Proof of Lemma C.3. We can write the optimization problem for $S \cup \{x\}$ as

$$f_{S\cup\{x\}}(q) = \min_{\pi_{st},\pi_{xt}\geq 0,\, w,w_x} \sum_{s\in S}\sum_{t\in T}\pi_{st}c(s,t) + \lambda\sum_{s\in S} w_s^2 + \sum_{t\in T}\pi_{xt}c(x,t) + \lambda w_x^2$$
$$\text{s.t. } \forall t \in T, \sum_{s\in S}\pi_{st} + \pi_{xt} = q_t, \forall s \in S, \sum_{t\in T}\pi_{st} = w_s, \text{ and } \sum_{t\in T}\pi_{xt} = w_x.$$

Introduce an auxiliary variable $\delta \in \mathbb{R}^T$ satisfying $\delta_t = q_t - \sum_{s\in S}\pi_{st}$ for all $t \in T$. Then, $\pi_{xt} = \delta_t$ for all $t \in T$, and we can write an equivalent optimization problem:

$$f_{S\cup\{x\}}(q) = \min_{0\leq\delta\leq q}\min_{\pi_{st}\geq 0,\, w_s} \sum_{s\in S}\sum_{t\in T}\pi_{st}c(s,t) + \lambda\sum_{s\in S} w_s^2 + \sum_{t\in T}\delta_t c(x,t) + \lambda\left(\sum_{t\in T}\delta_t\right)^2$$
$$\text{s.t. } \forall t \in T, \sum_{s\in S}\pi_{st} = q_t - \delta_t, \forall s \in S, \sum_{t\in T}\pi_{st} = w_s.$$

Finally, the internal optimization problem over π, w can be written in the form required. □

Proof of Lemma C.4. The objective function is strictly convex in w, and this ensures that the optimal weights are unique. Thus, Danskin's theorem (Bertsekas, 2009) implies that f_S is differentiable.

Further, we can write the derivative using the KKT conditions as

$$\frac{\partial f_S}{\partial q_t}(q) = \min_{\ell\in S} c(\ell,t) + 2\lambda\cdot w_\ell^*(S)$$

where $w_\ell^*(S)$ is the optimal assignment for the w variables. From the KKT conditions, we also have, for each $t \in T$,

$$\pi_{st}^*(S) > 0 \implies c(s,t) + 2\lambda\cdot w_s^*(S) = \min_{\ell\in S} c(\ell,t) + 2\lambda\cdot w_\ell^*(S) \tag{7}$$

For $S' \supset S$, we get

$$\frac{\partial f_{S'}}{\partial q_t}(q) = \min_{\ell\in S'} c(\ell,t) + 2\lambda\cdot w_\ell^*(S') \tag{8}$$
$$\leq \min_{\ell\in S} c(\ell,t) + 2\lambda\cdot w_\ell^*(S'). \tag{9}$$

We will now show that $w_\ell^*(S') \leq w_\ell^*(S)$, for all $\ell \in S$, from which the result follows.

Let $A = \{\ell \in S : w_\ell^*(S') > w_\ell^*(S)\}$, and assume to the contrary that A is non-empty. Then, since $\sum_{s\in A} w_s^*(S') > \sum_{s\in A} w_s^*(S)$, there exists $t \in T, \ell \in A$ and $k \in S \setminus A$ such that $\pi_{\ell t}^*(S') > \pi_{\ell t}^*(S)$ and $\pi_{kt}^*(S') < \pi_{kt}^*(S)$. This implies that $\pi_{\ell t}^*(S') > 0$ and $\pi_{kt}^*(S) > 0$, and equation (7) implies that

$$c(\ell,t) + 2\lambda\cdot w_\ell^*(S') \leq c(k,t) + 2\lambda\cdot w_k^*(S') \text{ and } c(\ell,t) + 2\lambda\cdot w_\ell^*(S) \geq c(k,t) + 2\lambda\cdot w_k^*(S).$$

Since $\ell \in A$, $w_\ell^*(S') > w_\ell^*(S)$. Combined with the above, we have $w_k^*(S') > w_k^*(S)$. This is a contradiction, since $k \notin A$, we must have $w_k^*(S') \leq w_k^*(S)$.

Hence, we must have that $w_\ell^*(S') \leq w_\ell^*(S)$, for all $\ell \in S$. Using equation (8), we get

$$\frac{\partial f_{S'}}{\partial q_t}(q) \leq \min_{\ell\in S} c(\ell,t) + 2\lambda\cdot w_\ell^*(S')$$
$$\leq \min_{\ell\in S} c(\ell,t) + 2\lambda\cdot w_\ell^*(S)$$
$$= \frac{\partial f_S}{\partial q_t}(q).$$

□

Task	K	RTW ($\lambda = 10$)	K-medoids+W	MaxHerd	K-medoids	BADGE	CLUE	Random
Clipart → Product	50	67.66 ± 1.05	$\mathbf{69.07 \pm 1.17}$	$\underline{68.41 \pm 0.88}$	68.31 ± 2.40	58.89 ± 2.12	50.38 ± 3.84	59.24 ± 1.80
	100	72.85 ± 1.35	$\mathbf{74.66 \pm 1.03}$	$\underline{73.90 \pm 1.86}$	73.50 ± 1.59	68.46 ± 1.93	58.49 ± 3.32	66.20 ± 1.98
Clipart → Real World	50	67.89 ± 2.83	$\underline{68.05 \pm 2.17}$	$\mathbf{68.13 \pm 2.50}$	67.33 ± 2.70	61.99 ± 3.42	53.94 ± 2.94	61.35 ± 3.32
	100	71.39 ± 1.55	$\mathbf{72.75 \pm 1.82}$	71.55 ± 2.04	$\underline{71.79 \pm 2.21}$	68.84 ± 2.60	61.35 ± 2.25	66.45 ± 2.49
Product → Clipart	50	$\mathbf{39.12 \pm 3.18}$	$\underline{39.02 \pm 3.17}$	37.84 ± 1.06	37.65 ± 3.07	36.18 ± 3.13	33.33 ± 2.08	36.27 ± 3.47
	100	$\underline{43.53 \pm 2.41}$	$\mathbf{44.02 \pm 2.78}$	42.16 ± 2.38	41.76 ± 2.18	40.88 ± 4.62	38.73 ± 2.59	39.80 ± 3.49
Product → Real World	50	$\mathbf{75.78 \pm 2.15}$	72.91 ± 2.44	73.31 ± 2.83	$\underline{73.39 \pm 3.23}$	68.84 ± 3.24	66.61 ± 2.72	68.69 ± 2.95
	100	$\mathbf{76.73 \pm 2.61}$	74.98 ± 2.46	75.46 ± 2.73	$\underline{75.86 \pm 3.45}$	72.35 ± 2.65	70.52 ± 5.00	71.16 ± 1.63
Real World → Clipart	50	$\mathbf{40.88 \pm 2.61}$	40.59 ± 2.15	$\underline{40.88 \pm 1.41}$	39.22 ± 2.38	37.94 ± 1.98	35.69 ± 2.58	37.84 ± 4.68
	100	$\mathbf{45.29 \pm 2.47}$	44.02 ± 2.89	44.12 ± 3.16	43.92 ± 2.85	$\underline{44.61 \pm 2.19}$	40.20 ± 2.94	40.00 ± 3.61
Real World → Product	50	$\mathbf{75.87 \pm 1.94}$	71.13 ± 2.20	72.19 ± 2.38	$\underline{72.19 \pm 1.78}$	68.61 ± 2.24	62.92 ± 4.01	67.86 ± 2.80
	100	$\mathbf{77.78 \pm 1.55}$	76.78 ± 2.10	$\underline{77.13 \pm 0.57}$	76.73 ± 1.60	72.70 ± 1.11	68.72 ± 2.58	71.69 ± 2.72

Table 1: Held-out target accuracies for active domain adaptation methods on all OfficeHome-RSUT tasks, at two query budgets $K = 50$ and $K = 100$. The values displayed are mean $\pm$ standard deviation across five random 80/20 acquisition pool and evaluation splits. The best values in each row are indicated in bold, and the second best values are underlined.

C.2 Greedy approximation guarantee

Proof of Theorem 4.1. Consider the function $g(A) := \mathrm{RTW}_\lambda(S) - \mathrm{RTW}_\lambda(S \cup A)$. By Theorem C.1, g is monotone and submodular. The first claim therefore follows directly from the cardinality-constrained greedy guarantee of Nemhauser et al. (1978).

For the second claim, consider greedily adding another queried point to $S \cup \hat{A}_i$ on each step i, and suppose the current objective is still larger than the value given by A_k^*, that is,

$$\mathrm{RTW}_\lambda(S \cup \hat{A}_i) \geq \mathrm{RTW}_\lambda(S \cup A_k^*).$$

By the greedy property and diminishing returns from adding each point (supermodularity), the next greedy step reduces this cost difference by at least $1/k$. Formally, we have

$$\mathrm{RTW}_\lambda(S \cup \hat{A}_{i+1}) - \mathrm{RTW}_\lambda(S \cup A_k^*) \leq \left(1 - \frac{1}{k}\right)\left(\mathrm{RTW}_\lambda(S \cup \hat{A}_i) - \mathrm{RTW}_\lambda(S \cup A_k^*)\right).$$

If the current cost is already at most $\mathrm{RTW}_\lambda(S \cup A_k^*)$, the desired guarantee holds by monotonicity. Otherwise, iterating from $\hat{A}_0 = \varnothing$ gives

$$\mathrm{RTW}_\lambda(S \cup \hat{A}_{k'}) \leq \mathrm{RTW}_\lambda(S \cup A_k^*) + e^{-k'/k}\left(\mathrm{RTW}_\lambda(S) - \mathrm{RTW}_\lambda(S \cup A_k^*)\right).$$

If $k' = |T|$, all target points are selected, so monotonicity gives $\mathrm{RTW}_\lambda(S \cup \hat{A}_{k'}) \leq \mathrm{RTW}_\lambda(S \cup A_k^*) \leq \eta \mathrm{RTW}_\lambda(S)$. Otherwise, $k' = \lceil k \log(1/(\varepsilon\eta)) \rceil$, and the last term is at most $\varepsilon\eta \mathrm{RTW}_\lambda(S)$. Combining this with $\mathrm{RTW}_\lambda(S \cup A_k^*) \leq \eta \mathrm{RTW}_\lambda(S)$ proves the second claim, including when the optimal cost is zero. □

D Additional Experimental Details

In this section we present the full experimental results, and mention some additional implementation details.

Additional Implementation Details. To speed up our algorithm, we use several computational techniques. Since we solve a sequence of optimization problems with an increasing number of variables, we warm-start each problem using the solution to the preceding problem. For example, when computing $\mathrm{RTW}_\lambda(S \cup \{x\})$, we initialize the variables corresponding to S using the solution obtained while computing $\mathrm{RTW}_\lambda(S)$. Additionally, at each iteration, we first screen the candidate points by solving a one-dimensional optimization problem for each candidate to obtain a dual lower bound. We then solve the full optimization problem only for the candidates that cannot be ruled out by these lower bounds. Finally, we solve multiple optimization problems in parallel batches on a GPU.

Task	RTW ($\lambda = 10$)	KMM	KLIEP	IWN
Clipart → Product	$\mathbf{49.02 \pm 2.08}$	30.63 ± 1.96	$\underline{46.55 \pm 1.43}$	43.17 ± 2.24
Clipart → Real World	$\mathbf{53.15 \pm 0.96}$	25.82 ± 1.81	$\underline{50.36 \pm 1.11}$	45.90 ± 2.81
Product → Clipart	$\underline{31.57 \pm 2.34}$	13.92 ± 2.61	$\mathbf{32.06 \pm 3.61}$	26.86 ± 2.26
Product → Real World	$\mathbf{69.24 \pm 2.13}$	56.02 ± 3.81	61.43 ± 1.45	$\underline{64.38 \pm 2.74}$
Real World → Clipart	$\underline{32.16 \pm 2.99}$	15.20 ± 1.47	$\mathbf{33.82 \pm 2.86}$	27.65 ± 2.04
Real World → Product	$\mathbf{68.16 \pm 1.00}$	57.73 ± 1.95	$\underline{64.03 \pm 1.03}$	63.73 ± 1.69

Table 2: Held-out target accuracies for importance-weighting baselines on OfficeHome-RSUT tasks. The values displayed are mean ± standard deviation across five random 80/20 acquisition pool and evaluation splits. Each method trains on the labeled source samples using its own learned reweighting. The best values in each row are indicated in bold, and the second best values are underlined.

Task	K	RTW ($\lambda = 1$)	K-medoids+W	MaxHerd	K-medoids	BADGE	CLUE	Random
Clipart → Real	50	$\mathbf{63.43 \pm 0.46}$	$\underline{55.32 \pm 0.66}$	53.76 ± 0.55	53.94 ± 0.51	42.30 ± 2.31	35.67 ± 0.86	41.54 ± 0.97
	100	$\mathbf{67.70 \pm 0.75}$	$\underline{63.95 \pm 0.52}$	63.86 ± 0.52	63.92 ± 0.65	49.48 ± 1.53	40.72 ± 1.63	47.32 ± 0.84
Painting → Real	50	$\mathbf{71.65 \pm 0.37}$	$\underline{62.29 \pm 0.37}$	61.89 ± 0.68	61.45 ± 0.64	56.27 ± 1.58	47.52 ± 0.95	53.01 ± 1.90
	100	$\mathbf{73.04 \pm 0.30}$	$\underline{68.83 \pm 0.43}$	68.47 ± 0.75	68.36 ± 0.54	59.69 ± 1.92	50.72 ± 1.22	56.94 ± 2.09
Sketch → Real	50	$\mathbf{59.14 \pm 0.42}$	$\underline{51.57 \pm 0.34}$	51.16 ± 0.81	51.07 ± 0.64	40.78 ± 1.72	31.98 ± 0.64	40.15 ± 0.77
	100	63.40 ± 0.62	$\mathbf{64.62 \pm 0.48}$	64.38 ± 0.63	$\underline{64.45 \pm 0.62}$	48.00 ± 1.98	37.90 ± 1.55	46.73 ± 0.67

Table 3: Held-out target accuracies for active domain adaptation methods on all three mini-DomainNet tasks, at query budgets $K = 50$ and $K = 100$. The values displayed are mean ± standard deviation across five random 80/20 acquisition-pool and evaluation partitions. The best value in each row is bold, and the second best is underlined.

Baselines. As described in Section 5, our baselines can broadly be divided into importance-weighting baselines and active domain adaptation baselines. We consider three importance-weighting baselines. **Kernel Mean Matching (KMM)** (Huang et al. (2006)) estimates the importance weights by minimizing the MMD between the (reweighted) source and the target dataset. **KLIEP** (Sugiyama et al. (2007)) computes importance weights by minimizing the KL divergence between the train and test distributions. **Importance Weighting Network (IWN)** (de Mathelin et al. (2022)) uses a neural network to learn the importance weights.

We considered five active domain adaptation methods. de Mathelin et al. (2022) use **K-medoids** to select the points to query. They also introduce **K-medoids+W**, a weighted version of k-medoids that uses the source labels to assign larger weights to points closer to the margin. Note that **K-medoids**, with nearest-neighbor reweighting, corresponds to our method with $\lambda = 0$. **MaxHerd** (Bae et al. (2025)) greedily queries points to maximize the kernel-based coverage of the target samples. **BADGE** (Ash et al. (2020)) uses the gradient embeddings to pick points that are uncertain and diverse. **CLUE** (Prabhu et al. (2021)) uses uncertainty-weighted clustering to pick target points that are both uncertain and diverse. In addition to these, we also consider **Random**, which queries points uniformly at random from the target.

OfficeHome-RSUT. Table (1) shows accuracy of all active domain adaptation methods on all OfficeHome-RSUT tasks, evaluated at two query budgets. Our method outperforms or is competitive against other active domain adaptation baselines. Table (2) shows accuracy of all the importance weighting methods, and compares it to the performance of our reweighting at no additional queries. RTW outperforms or is competitive with all other importance-weighting methods.

Figure (4) shows the accuracy of different methods on three additional tasks. Our RTW algorithm is competitive with other baselines at $K = 100$, and outperforms baselines for smaller query budgets.

mini-DomainNet. Table (3) reports query-method accuracies on all three adaptation tasks at $K = 50$ and $K = 100$; Table (4) compares source reweighting without additional queries.

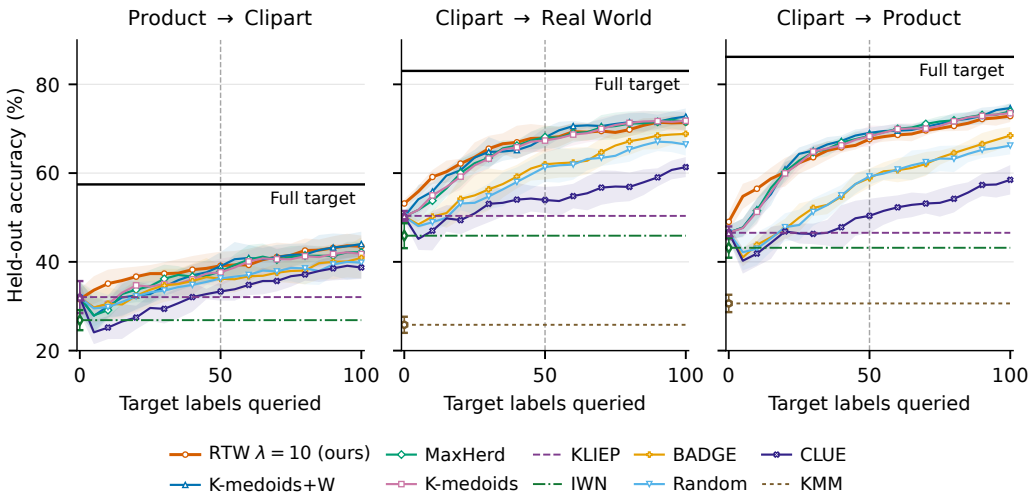

Figure 4: Results for three additional tasks on OfficeHome-RSUT. The full target line denotes the accuracy of the classifier trained on the target query pool. Dashed lines represent methods that do not query for additional labels.

Task	RTW ($\lambda = 1$)	KMM	KLIEP	IWN
Clipart → Real	$\mathbf{51.50 \pm 0.16}$	37.69 ± 0.88	42.62 ± 0.18	$\underline{44.84 \pm 0.28}$
Painting → Real	$\mathbf{68.47 \pm 0.26}$	44.91 ± 2.57	$\underline{57.62 \pm 0.28}$	56.76 ± 0.52
Sketch → Real	$\mathbf{50.48 \pm 0.22}$	22.70 ± 0.61	$\underline{43.58 \pm 0.38}$	40.10 ± 1.15

Table 4: Held-out target accuracies (%) for importance-weighting methods on mini-DomainNet at $K = 0$. Values are mean $\pm$ sample standard deviation across random 80/20 partitions of the target into acquisition pool and held-out evaluation set. Each method trains only on labeled source samples using its learned weights. The best value in each row is bold, and the second best is underlined.