Active Online Learning with Hidden Shifting Domains

Yining Chen
Stanford University

Haipeng Luo
University of Southern California

Tengyu Ma
Stanford University

Chicheng Zhang
University of Arizona

Abstract

Online machine learning systems need to adapt to domain shifts. Meanwhile, acquiring label at every timestep is expensive. Motivated by these two challenges, we propose a surprisingly simple algorithm that adaptively balances its regret and its number of label queries in settings where the data streams are from a mixture of hidden domains. For online linear regression with oblivious adversaries, we provide a *tight* tradeoff that depends on the durations and dimensionalities of the hidden domains. Our algorithm can adaptively deal with interleaving spans of inputs from different domains. We also generalize our results to non-linear regression for hypothesis classes with bounded eluder dimension and adaptive adversaries. Experiments on synthetic and realistic datasets demonstrate that our algorithm achieves lower regret than uniform queries and greedy queries with equal labeling budget.

1 Introduction

In statistical learning, model performance often significantly drops when the testing distribution drifts away from the training distribution (Torralba and Efros, 2011; Recht et al., 2018, 2019; Engstrom et al., 2020). Online learning addresses worst-case domain shift by assuming the data is given by an adversary (Hazan, 2016). However, practical deployments of fully-online learning systems have been somewhat limited, because labels are expensive to obtain; see (Strickland, 2018) for an example in fake news detection. A label budget linear in time is too much of a luxury.

Cesa-Bianchi et al. (2004b) study label-efficient online learning for prediction with expert advice. Their algorithm queries the label of every example with a fixed probability, which, as they show, achieves minimax-optimal regret and query complexity for this problem. However, querying with uniform probability does not take into account the algorithm's uncertainty on each individual example, and thus can be suboptimal when the problem has certain favorable structures. For example, a sequence of online news may come from the mixture of a few topics or trends, and some news topics may require more samples to categorize well compared to others.

We aim to improve label-efficiency in online learning by exploiting hidden domain structures in the data. We assume that each input is from one of m unknown and potentially overlapping domains (e.g. news topics). For each input, the learner makes a prediction, incurs a loss, and decides whether to query its label. The regret of the learner is defined as the difference between its cumulative loss and that of the best fixed predictor in hindsight. Our goal is to trade off between regret and query complexity: given a fixed label budget, we hope to incur the smallest regret possible.

In statistical learning, domains usually refer to data distributions. One commonly-studied type of domain shift is *covariate shift* (see e.g. Sugiyama et al., 2007), where the conditional distribution $P(y|x)$ is fixed, but the marginal distribution $P(x)$ changes across domains. In online learning, however, the inputs are usually not independently drawn from a distribution, and can even be adversarially chosen from a certain support. We thus propose to study *support shift* in online learning as the natural counterpart of covariate shift in statistical learning. Specifically, we assume that inputs from each domain $u \in [m]$ have an unknown support $\mathcal{X}_u$ and an unknown total duration T_u, and inputs from different domains are interleaved as a stream fed to the online learner.

Support shift is common for high-dimensional vision / language datasets where domain supports have little overlap. As a motivating example, consider an online regression task of sentiment prediction for twitter feeds. Here, a domain is a (not necessarily contiguous) subsequence of tweets around a certain topic. Different topics contain disjoint keywords. As the hot topics change over time, we likely receive inputs from different domains for varying time periods.

Similar to covariate shift, we assume realizability, i.e., there exists a predictor that is Bayes optimal across all the do-

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

mains. Realizability is a reasonable assumption in modern machine learning for two reasons. First, high-dimensional features are usually of high quality; for example, adding only one additional output layer to pre-trained embeddings such as BERT obtain state-of-the-art results on many language tasks (Devlin et al., 2018). Second, models are often overparameterized (Zhang et al., 2016). Thus, the model can rely on different features in different domains. In the sentiment regression task, the model can combine positive or negative words in all domains to predict well. In statistical learning, domain adaptation methods assuming that a single model can perform well on different domains (Ganin et al., 2016) indeed have been empirically successful on high-dimensional datasets.

Under this setup, we propose QuFUR (Query in the Face of Uncertainty for Regression), a surprisingly simple query scheme based on uncertainty quantification. We start with online linear regression from $\mathbb{R}^d$ to $\mathbb{R}$ with an oblivious adversary. With additional regularity conditions, we provide the following regret guarantee of QuFUR with label budget B: for *any* m and *any* partition of $[T]$ into domains $I_1, \ldots, I_m$, with T_u being the number of examples from domain I_u and d_u being the dimension of the space spanned by these examples, the regret of is QuFUR is $\tilde{O}((\sum_{u=1}^{m} \sqrt{d_u T_u})^2 / B)$ (Theorem 2).[1]

When choosing $m = 1$ and $I_1 = [T]$, we see that the regret of QuFUR is at most $\tilde{O}(dT/B)$, matching minimax lower bounds (Theorem 7) in this setting. The advantage of QuFUR's adaptive regret guarantees becomes significant when the domains have heterogeneous time spans and dimensions: $(\sum_{u=1}^{m} \sqrt{d_u T_u})^2$ can be substantially less than dT when the T_u/d_u's are heterogeneous across different u's. For example, if $m = 2$, $d_1 = d$, $d_2 = 1$, $T_1 = d$, and $T_2 = T - d$, then the resulting regret bound is of order $O(T + d^2)$ which can be much smaller than dT when $1 \ll d \ll T$. Using standard online-to-batch conversion (Cesa-Bianchi et al., 2004a), we also obtain novel results in batch active learning for regression (Theorem 12). Furthermore, we also define a stronger notion of minimax optimality, namely *hidden domain minimax optimality*, and show that QuFUR is optimal in this sense (Theorem 3), for a wide range of domain structure specifications.

We generalize our results to online regression with general hypothesis classes against an adaptive adversary. We obtain a similar regret-query complexity tradeoff, where the analogue of d_u is (roughly) the *eluder dimension* (Russo and Van Roy, 2013) of the hypotheses class with respect to the support of domain u (Theorem 4).

Experimentally, we show that our algorithm outperforms the baselines of uniform and greedy query strategies, on a synthetic dataset and three high-dimensional language and image datasets with realistic support shifts. Our code is available online at `https://github.com/cynnjjs/online_active_AISTATS`.

[1]Throughout this paper, $[n]$ denotes the set $\{1, \ldots, n\}$; notations $\tilde{O}$ and $\tilde{\Omega}$ hide logarithmic factors.

2 Related works

Active learning. We refer the readers to Balcan et al. (2009); Hanneke (2014); Dasgupta et al. (2008); Beygelzimer et al. (2010) and the references therein for background on active learning. For classification, a line of works (Dasgupta and Hsu, 2008; Minsker, 2012; Kpotufe et al., 2015; Locatelli et al., 2017) performs hierarchical sampling for nonparametric active learning. The main idea is to maintain a hierarchical partitioning over the instance domain (either a pre-defined dyadic partition or a pre-clustering over the data), and performs adaptive label querying with partition-dependent probabilities. For regression, many works (Fedorov and Hackl, 2012; Chaudhuri et al., 2015) study the utility of active learning for maximum likelihood estimation in the realizable setting. Recent works also study active linear regression in nonrealizable (Drineas et al., 2006; Derezinski et al., 2018; Dereziński and Warmuth, 2018; Sabato and Munos, 2014) and heteroscedastic (Chaudhuri et al., 2017; Fontaine et al., 2019) settings. These works do not consider domain structures except for Sabato and Munos (2014), who propose a domain-aware stratified sampling scheme. Their algorithm needs to know the domain partition a priori, whose quality is crucial to ensure good performance.

Active learning for domain adaptation. The empirical works of Rai et al. (2010); Saha et al. (2011); Xiao and Guo (2013) study stream-based active learning when inputs comes from pre-specified source and target distributions. Su et al. (2020) combine domain adversarial neural network (DANN) with active learning, where the discriminator in DANN serves as a density ratio estimator that guides active sampling. In contrast, our algorithm handles multiple domains, does not assume iid-ness for inputs from a domain, and does not require knowledge of which domain the inputs come from.

Active online learning. Earlier works on selective sampling when iid data arrive in a stream and a label querying decision has to be made after seeing each example (Cohn et al., 1994; Dasgupta et al., 2008; Hanneke, 2011) implicitly provide online regret and label complexity guarantees. Works on worst-cast analysis of selective sampling for linear classification (Cesa-Bianchi et al., 2006) provide regret guarantees similar to that of popular online linear classification algorithms such as Perceptron and Winnow, but their label complexity guarantees are runtime-dependent and therefore cannot be easily converted to a guarantee that only involves problem parameters defined apriori. Subsequent works (Cesa-Bianchi et al., 2009; Dekel et al., 2010; Cavallanti et al., 2011; Agarwal, 2013) study the setting where there is a parametric model on $P(y|x, \theta)$ with unknown parameter θ, and the x's shown can be adversarial. Under those assumptions, they obtain regret and query complexity guarantees dependent on the fraction of examples with low margins. Yang (2011) gives a worst-case analysis of active online learning for classification with drifting distributions, under the assumption that the Bayes optimal classifier is in the learner's hypothesis class. In contrast, our work gives adaptive regret guarantees in terms of the hidden domain structure in the data, and focuses on regression instead of classification.

KWIK model. In the KWIK model (Li et al., 2011), at each time step, the algorithm is asked to either abstain from prediction and query the label, or predict an output with at most ϵ error. In contrast, in our setting, the learner's goal is to minimize its cumulative regret, as opposed to making pointwise-accurate predictions. Cesa-Bianchi et al. (2009) study linear regression in the KWIK model, and propose the BBQ sampling rule; our work can be seen as analyzing a variant of BBQ and showing its adaptivity to domain structures. Szita and Szepesvári (2011) propose an algorithm that works in an agnostic setting, where the error guarantee at every round depends on the agnosticity of the problem. A relaxed KWIK model that allows a prespecified number of mistakes has been studied in Sayedi et al. (2010); Zhang and Chaudhuri (2016).

Adaptive/Switching Regret. Adaptive regret (Hazan and Seshadhri, 2007; Daniely et al., 2015) is the excessive loss of an online algorithm compared to the locally optimal solution over any continuous timespan. Our algorithm can be interpreted as being competitive with the locally optimal solution on every domain, even if the timespans of the domains are not continuous, which is closer to the concept of switching regret with long-term memory studied in e.g, (Bousquet and Warmuth, 2002; Zheng et al., 2019). Switching regret bounds typically have a polynomial dependence on the number of domain switches, which does not appear in our bounds. However, the above works allow target concept to shift over time, whereas our bounds require realizability and thus compete with a fixed optimal concept. Overall, we achieve a stronger form of guarantee under more assumptions.

Online linear regression. Literature on fully-supervised online linear regression has a long history (Vovk, 2001; Azoury and Warmuth, 2001). As is implicit in Cesa-Bianchi et al. (2004b), we can reduce from fully-supervised online regression to active online regression by querying uniformly randomly with a fixed probability. Combining this reduction with existing online linear regression algorithms (Hazan et al., 2007), we get $\tilde{O}(dT/B)$ regret with $O(B)$ queries for any $B \leq T$. Our bound matches this in the realizable and oblivious setting when there is one domain, and is potentially much better with more domain structures.

3 Setup and Preliminaries

3.1 Setup

Active online regression with domain structure. Let $\mathcal{F} = \{f : \mathcal{X} \to [-1, 1]\}$ be a hypothesis class. We consider a realizable setting where $y_t = f^*(x_t) + \xi_t$ for some $f^* \in \mathcal{F}$ and random noise ξ_t. The adversary decides $f^* \in \mathcal{F}$ before the interaction starts, and ξ_t's are independent zero-mean sub-Gaussian random variables with variance proxy η^2.

The example sequence $\{x_t\}_{t=1}^T$ has the following domain structure unknown to the learner: $[T]$ can be partitioned into m disjoint nonempty subsets $\{I_u\}_{u=1}^m$, where for each u, $|I_u| = T_u$, and $\{x_t\}_{t \in I_u}$ lie in a subspace of dimension d_u.

The interaction between the learner and the adversary follows the protocol below.

For each $t = 1, \ldots, T$:

1. Example x_t is revealed to the learner.
2. The learner predicts $\hat{y}_t = \hat{f}_t(x_t)$ using predictor $\hat{f}_t : \mathcal{X} \to [-1, 1]$, incurring loss $(\hat{y}_t - y_t)^2$.
3. The learner sets a query indicator $q_t \in \{0, 1\}$. If $q_t = 1$, y_t is revealed.

The performance of the learner is measured by its number of queries $Q = \sum_{t=1}^T q_t$, and its regret $R = \sum_{t=1}^T (\hat{y}_t - f^*(x_t))^2$. By our realizability assumption, our notion of regret coincides with the one usually used in online learning when expectations are taken; see Appendix D. Our goal is to design a learner that has low regret R subject to a budget constraint: $Q \leq B$, for some fixed budget B.

Oblivious vs. adaptive adversary. In the *oblivious* setting, the adversary decides the sequence $\{x_t\}_{t=1}^T$. In the *adaptive* setting, the adversary can choose x_t depending on the history $H_{t-1} = \{x_{1:t-1}, \hat{f}_{1:t-1}, \xi_{1:t-1}\}$.

Miscellaneous notations. For a vector $v \in \mathbb{R}^d$ and a positive semi-definite matrix $M \in \mathbb{R}^{d \times d}$, define $\|v\|_M := \sqrt{v^\top M v}$. For vectors $\{z_t\}_{t=1}^T \subseteq \mathbb{R}^l$, and $S = \{i_1, \ldots, i_n\} \subseteq [T]$, denote by Z_S the $n \times l$ matrix whose rows are $z_{i_1}^\top, \ldots, z_{i_n}^\top$. Define $\mathrm{clip}(z) := \min(1, \max(-1, z))$ and $\tilde{\eta} := \max\{1, \eta\}$. For a set of vectors S, define $\mathrm{span}(S)$ as the linear subspace spanned by S.

3.2 Baselines for linear regression

We first study linear regression with an oblivious adversary, and then generalize to the non-linear case with an adaptive adversary in Section 5. For now, hypothesis class $\mathcal{F}$ is $\{\langle x, \theta\rangle : \theta \in \mathbb{R}^d, \|\theta\|_2 \le C\}$. Let the ground truth hypothesis be $f^*(x) = \langle \theta^*, x\rangle$, where $\theta^* \in \mathbb{R}^d$, and input space $\mathcal{X}$ be a subset of $\{x \in \mathbb{R}^d : \|x\|_2 \le 1, \langle x, \theta^*\rangle \le 1\}$.[2]

Uniform querying is minimax-optimal with no domain structure. As a starter, consider an algorithm that queries every label and predicts using the regularized least squares estimator $\hat{\theta}_t = \operatorname{argmin}_\theta \sum_{i=1}^{t-1} (\langle \theta, x_i\rangle - y_i)^2 + \lambda\|\theta\|^2$, where $\lambda = 1/C^2$. It is well-known from (Vovk, 2001; Azoury and Warmuth, 2001) that (a variant of) this fully-supervised algorithm achieves $R = \tilde{O}(\tilde{\eta}^2 d)$ with $Q = T$. Consider an active learning extension of the above algorithm that queries labels independently with probability B/T, and predicts with the regularized least squared estimator computed based on all queried examples $\hat{\theta}_t = \operatorname{argmin}_\theta \sum_{i\in[t-1], q_i=1} (\langle \theta, x_i\rangle - y_i)^2 + \lambda\|\theta\|^2$. We show that the above active online regression strategy achieves $\mathbb{E}[R] = \tilde{O}(\tilde{\eta}^2 dT/B)$ with $\mathbb{E}[Q] = B$ in Appendix A.4. As shown in Theorem 7, this tradeoff is minimax optimal if $\tilde{\eta}$ is a constant. Although this guarantee is optimal in the worst case, one major weakness is that it is too pessimistic: as we will see next, when the data has certain hidden domain structure, the learner can achieve substantially better regret guarantees than the worst case if given access to auxiliary domain information.

Oracle baseline when domain structure is known. Suppose the learner is given the following piece of knowledge from an oracle: there are m domains; for each u in $[m]$, there are a total of T_u examples from domain u from a subspace of $\mathbb{R}^d$ of dimension d_u. In addition, for every t, the learner is given the index of the domain to which example x_t belongs. In this setting, the learner can combine the aforementioned regularized least squares linear predictor with the following domain-aware querying scheme: for any example in domain u, the learner queries its label independently with probability $\mu_u \in (0, 1]$. Within domain u, the learner incurs $O(\mu_u T_u)$ queries and $\tilde{O}(\tilde{\eta}^2 d_u/\mu_u)$ regret. Summing over all m domains, its achieves a label complexity of $O(\sum_{u=1}^m \mu_u T_u)$ and a regret bound of $\tilde{O}(\tilde{\eta}^2 \sum_{u=1}^m d_u/\mu_u)$. This motivates the following optimization problem:

$$\min_\mu \sum_{u=1}^m d_u/\mu_u, \text{ s.t. } \sum_{u=1}^m \mu_u T_u \le B, \mu_u \in [0, 1], \forall u \in [m].$$

i.e., we choose domain-dependent query probabilities that minimize the learner's total regret guarantee, subject to its query complexity being controlled by B. When $B \le (\sum_{u=1}^m \sqrt{d_u T_u}) \min_u \sqrt{T_u/d_u}$, the optimal μ_u is $\sqrt{d_u/T_u} \cdot \frac{B}{\sum_{u=1}^m \sqrt{d_u T_u}}$, i.e. μ_u is proportional to $\sqrt{d_u/T_u}$.[3] This yields a regret guarantee of $O(\tilde{\eta}^2(\sum_u \sqrt{d_u T_u})^2/B)$.

Algorithm 1 Query in the Face of Uncertainty for Regression (QuFUR(α))

Require: Total dimension d, time horizon T, θ^*'s norm bound C, noise level η, parameter α.

- $M \leftarrow \frac{1}{C^2} I$, queried dataset $\mathcal{Q} \leftarrow \emptyset$.
- **for** $t = 1$ to T **do**
  - Compute regularized least squares solution $\hat{\theta}_t \leftarrow M^{-1} X_\mathcal{Q}^\top Y_\mathcal{Q}$.
  - Let $\hat{f}_t(x) = \text{clip}(\langle \hat{\theta}_t, x\rangle)$ be the predictor at time t, and predict $\hat{y}_t \leftarrow \hat{f}_t(x_t)$.
  - Uncertainty estimate $\Delta_t \leftarrow \tilde{\eta}^2 \min\{1, \|x_t\|^2_{M^{-1}}\}$.
  - With probability $\min\{1, \alpha\Delta_t\}$, set $q_t \leftarrow 1$; otherwise set $q_t \leftarrow 0$.
  - **if** $q_t = 1$ **then**
    - Query y_t. $M \leftarrow M + x_t x_t^\top$, $\mathcal{Q} \leftarrow \mathcal{Q} \bigcup \{t\}$.

Although this strategy can sometimes achieve much smaller regret than uniform querying (as we discuss in Section 1, $(\sum_u \sqrt{d_u T_u})^2$ could be substantially smaller than dT), it has two clear drawbacks: first, it is not clear if this guarantee is always no worse than uniform querying, especially when $\sum_{u=1}^m d_u \gg d$; second, the domain memberships of examples are rarely known in practice. In the next section, we develop algorithms that match the performance this domain-aware query scheme without these drawbacks.

4 Active online learning for linear regression: algorithms, analysis, and matching lower bounds

We start by presenting a parameterized algorithm in Section 4.1, where the parameter α has a natural cost interpretation. We then present a fixed-budget variant of it in Section 4.2. Section 4.3 shows that our algorithm is minimax-optimal under a wide range of domain structure specifications.

4.1 Main Algorithm: Query in the Face of Uncertainty for Regression (QuFUR)

We propose QuFUR (Query in the Face of Uncertainty for Regression), shown in Algorithm 1. At each time step t, the algorithm first computes $\hat{\theta}_t$, a regularized least squares estimator on the labeled data obtained so far, then predict using $\hat{f}_t(x) = \text{clip}(\langle \hat{\theta}_t, x\rangle)$. It makes a label query with probability proportional to Δ_t, a high-probability upper bound of the instantaneous regret $(\hat{y}_t - \langle \theta^*, x_t\rangle)^2$ (see Lemma 1 for details), which can also be interpreted as an uncertainty measure of x_t. Intuitively, when the algorithm is already confident about the current prediction, it saves its labeling budget for learning from less certain inputs in the fu-

[2] The constraint $\|x\|_2 \le 1$ can be relaxed by only increasing the logarithmic terms in the regret and query complexity guarantees.

[3] For larger budget $B > \sum_{u=1}^m \sqrt{d_u T_u} \cdot \min_u \sqrt{T_u/d_u}$, there exists a threshold τ such that the optimal solution is $\mu_u = 1$ for $\{u \in [m] : \sqrt{d_u/T_u} \ge \tau\}$, and $\mu_u \propto \sqrt{d_u/T_u}$ for $\{u \in [m] : \sqrt{d_u/T_u} < \tau\}$; see Appendix G.

ture. More formally, $\Delta_t := \tilde{\eta}^2 \min(1, \|x_t\|^2_{M_t^{-1}})$, where $M_t = \lambda I + \sum_{i \in \mathcal{Q}_t} x_i x_i^\top$, and $\mathcal{Q}_t$ is the set of labeled examples seen up to time step $t-1$. QuFUR queries the label y_t with probability $\min\{1, \alpha\Delta_t\}$, where α is a parameter that trades off between query complexity and regret.

Perhaps surprisingly, the simple query strategy of QuFUR can leverage hidden domain structure, as shown by the following theorem.

Theorem 1. *Suppose the example sequence $\{x_t\}_{t=1}^T$ has the following structure: $[T]$ can be partitioned into m disjoint nonempty subsets $\{I_u\}_{u=1}^m$, where for each u, $|I_u| = T_u$, and $\{x_t\}_{t \in I_u}$ lie in a subspace of dimension d_u. If Algorithm 1 receives as inputs dimension d, time horizon T, norm bound C, noise level η, and parameter α, then, with probability $1-\delta$:*
1. Its query complexity is

$$Q = \tilde{O}\left(\sum_{u=1}^m \min\left\{T_u, \tilde{\eta}\sqrt{\alpha d_u T_u}\right\} + 1\right).$$

2. Its regret is $R = \tilde{O}\left(\sum_{u=1}^m \max\{\tilde{\eta}^2 d_u, \tilde{\eta}\sqrt{d_u T_u/\alpha}\}\right)$.

The proof of the theorem is deferred to Section A.1. For better intuition, we focus on the regime of $\alpha \in \left[\frac{1}{\tilde{\eta}^2}\left(\frac{1}{\sum_u \sqrt{d_u T_u}}\right)^2, \frac{1}{\tilde{\eta}^2}\min_{u \in [m]} \frac{T_u}{d_u}\right]$, where the bounds become $Q = \tilde{O}(\tilde{\eta} \cdot \sqrt{\alpha}\sum_u \sqrt{d_u T_u})$, and $R = \tilde{O}(\tilde{\eta} \cdot \sum_u \sqrt{d_u T_u}/\sqrt{\alpha})$. We make a series of remarks below for this range of α:

Novel notion of adaptive regret. The above tradeoff is novel in that it holds for *any* meaningful domain partitions. Our proof actually shows that for any (not necessarily contiguous) subsequence $I \subseteq [T]$, QuFUR ensures $Q = \tilde{O}(\tilde{\eta} \cdot \sqrt{d_I |I|} \cdot \sqrt{\alpha})$ and $R = \tilde{O}(\tilde{\eta}\sqrt{d_I |I|})/\sqrt{\alpha}$ within I, where d_I is the dimension of $\mathrm{span}(\{x_t : t \in I\})$. This type of guarantee is different from the adaptive regret guarantees provided by e.g. Hazan and Seshadhri (2007), where the regret guarantee is only with respect to continuous intervals. However, note that the results in Hazan and Seshadhri (2007) do not require realizability.

Matching uniform querying baseline and minimax optimality. Our tradeoff is never worse than the uniform querying baseline; this can be seen by applying the theorem with the trivial partition $\{[T]\}$, yielding $Q = \tilde{O}(\tilde{\eta}\sqrt{\alpha d T})$ and $R = \tilde{O}(\tilde{\eta}\sqrt{dT/\alpha})$. Therefore, same as the uniform query baseline, this guarantee is also minimax optimal for constant η, in light of Theorem 7 in Appendix B.2.

Matching oracle baseline and domain structure-aware minimax optimality. QuFUR matches the domain-aware oracle baseline discussed in 3.2 even without prior knowledge of domain structure. Furthermore, as we will see, we show in Theorem 3 that in a wide range of problem specifications, this baseline, as well as QuFUR, is minimax-optimal in our problem formulation with domain structure.

Algorithm 2 Fixed-Budget QuFUR

Require: Total dimension d, time horizon T, label budget B, θ^*'s norm bound C, noise level η.

- $k \leftarrow \lceil 3\log_2 T \rceil$.
- **for** $i = 0$ to k **do**
  - Parameter $\alpha_i \leftarrow 2^i/T^2$.
- Initialize $M \leftarrow \frac{1}{C^2} I$, $\mathcal{Q} \leftarrow \emptyset$.
- **for** $t = 1$ to T **do**
  - Compute regularized least squares solution $\hat{\theta}_t \leftarrow M^{-1} X_{\mathcal{Q}}^\top Y_{\mathcal{Q}}$.
  - Let $\hat{f}_t(x) = \mathrm{clip}(\langle \hat{\theta}_t, x\rangle)$ be the predictor at time t, and predict $\hat{y}_t \leftarrow \hat{f}_t(x_t)$.
  - Uncertainty estimate $\Delta_t \leftarrow \tilde{\eta}^2 \min\{1, \|x_t\|^2_{M^{-1}}\}$.
  - **for** $i = 0$ to k **do**
    - Set $q_t^i = 0$.
    - **if** $\sum_{j=1}^{t-1} q_j^i < \lfloor B/k \rfloor$ **then**
      - With probability $\min\{1, \alpha_i \Delta_t\}$, set $q_t^i = 1$.
  - **if** $\sum_{i=0}^k q_t^i > 0$ **then**
    - Query y_t. $M \leftarrow M + x_t x_t^\top$, $\mathcal{Q} \leftarrow \mathcal{Q} \bigcup \{t\}$.

Fixed-cost-ratio interpretation. The tradeoff in Theorem 1 can be interpreted in a fixed-cost-ratio formulation. Suppose a practitioner decides that the cost ratio between 1 unit of loss and 1 label query is $c : 1$. The performance of the algorithm is then measured by its total cost $cR + Q$. Theorem 1 shows that QuFUR(α) balances the cost incurred by prediction and the cost incurred by label queries, in that $Q \approx \alpha R$. We show in Appendix C that QuFUR with input $\alpha = c$ achieves near-optimal total cost, for a wide range of domain structure parameters.

Dependence on η. Our query complexity and regret bounds have a dependence on $\tilde{\eta} = \max(\eta, 1)$. Similar dependence also appears in online least-squares regression literature (Vovk, 2001; Azoury and Warmuth, 2001).

Running time and extension to kernel regression. The most computationally intensive operation in QuFUR is calculating Δ_t for each time, leading to a total time complexity of $O(Td^2)$ (since the update of M^{-1} can be done in $O(d^2)$ via the Sherman-Morrison formula). For high dimensional problems with $d \gg n$, we can kernelise Algorithm 1 following an approach similar to Valko et al. (2013), which has a time complexity of $O(TQ^2 k)$, assuming that evaluating the kernel function takes $O(k)$ time. See Appendix F.

4.2 QuFUR with a fixed label budget

The label complexity bound in Theorem 1 involve parameters $\{(d_u, T_u)\}_{u=1}^m$, which may be unknown in advance. In many practical settings, the learner is given a label budget B. For such settings, we propose a fixed-budget version of QuFUR, Algorithm 2, that takes B as input, and achieves near-optimal regret bound subject to the budget constraint, under a wide range of domain structure specifications.

Algorithm 2 is a master algorithm that aggregates over $k = O(\log T)$ copies of QuFUR (α). Each copy uses a different value of α from an exponentially increasing grid $\{2^i/T^2 : i = 0, \ldots, k\}$. The grid ensures that each copy still has label budget $\lfloor B/k \rfloor = \tilde{\Omega}(B)$, and there is always a copy that takes full advantage of its budget to achieve low regret. The algorithm queries whenever one of the copies issues a query, and predicts using a model learned on all historical labeled data. A copy can no longer query when its budget is exhausted. In the realizable setting, the regret of the master algorithm is no worse that of the copy running on a parameter α_i that make $\tilde{\Theta}(B)$ queries when run on its own; this insight yields the following theorem.

Theorem 2. *Suppose the example sequence $\{x_t\}_{t=1}^T$ has the following structure: $[T]$ can be partitioned into m disjoint nonempty subsets $\{I_u\}_{u=1}^m$, where for each u, $|I_u| = T_u$, and $\{x_t\}_{t \in I_u}$ lie in a subspace of dimension d_u. Also suppose $B \le \tilde{O}\left(\sum_u \sqrt{d_u T_u} \min_{u \in [m]} \sqrt{T_u/d_u}\right)$. If Algorithm 2 receives as inputs dimension d, time horizon T, label budget B, norm bound C, and noise level η, then:*
1. Its query complexity Q is at most B.
2. With probability $1-\delta$, its regret is

$$R = \tilde{O}\left(\tilde{\eta}^2 \big(\sum_u \sqrt{d_u T_u}\big)^2 / B\right).$$

The proof of the theorem is deferred to Appendix A.2. We now compare the guarantee of QuFUR with the oracle baseline in Section 3.2: for any budget $B \le \tilde{O}(\sum_u \sqrt{d_u T_u} \min_u \sqrt{T_u/d_u})$, Fixed-Budget QuFUR achieves a regret guarantee no worse than that of domain-aware uniform sampling, while being agnostic to $\{(d_u, T_u)\}_{u=1}^m$ and the domain memberships of the examples. For larger budget $B > \tilde{O}(\sum_u \sqrt{d_u T_u} \min_u \sqrt{T_u/d_u})$, QuFUR still matches the oracle baseline; we defer the discussion to Appendix G.

4.3 Lower bound

Our development so far establishes domain structure-aware regret upper bounds $R = \tilde{O}(\tilde{\eta}^2(\sum_u \sqrt{d_u T_u})^2/B)$, achieved by Fixed-Budget QuFUR and domain-aware uniform sampling baseline (the latter requires extra knowledge about the domain structure and domain membership of each example, whereas the former does not). In this section, we study optimality properties of the above upper bounds. Specifically, we show via Theorem 3 that they are tight up to logarithmic factors, for a wide range of domain structure specifications. Its proof can be found in Appendix B.1.

Theorem 3. *For any noise level $\eta \ge 1$, set of positive integers $\{(d_u, T_u)\}_{u=1}^m$ and integer B that satisfy*

$$d_u \le T_u, \forall u \in [m], \quad \sum_{u=1}^m d_u \le d, \qquad B \ge \sum_{u=1}^m d_u, \tag{1}$$

there exists an oblivious adversary such that:
1. It uses a ground truth linear predictor $\theta^\star \in \mathbb{R}^d$ such that $\|\theta^*\|_2 \le \sqrt{d}$, and for all t, $\left|\langle \theta^*, x_t \rangle\right| \le 1$; in addition, the noises $\{\xi_t\}_{t=1}^T$ are sub-Gaussian with variance proxy η^2.
2. It shows example sequence $\{x_t\}_{t=1}^T \subset \{x : \|x\|_2 \le 1\}$, such that $[T]$ can be partitioned into m disjoint nonempty subsets $\{I_u\}_{u=1}^m$, where for each u, $|I_u| = T_u$, and $\{x_t\}_{t \in I_u}$ lie in a subspace of dimension d_u.
3. Any online active learning algorithm $\mathcal{A}$ with label budget B has regret $\Omega((\sum_{u=1}^m \sqrt{d_u T_u})^2/B)$.

The above theorem is a domain structure-aware refinement of the $\Omega(dT/B)$ minimax lower bound (Theorem 7 in Appendix B.2), in that it further constrains the adversary to present sequences of examples with domain structure parameterized by $\{(d_u, T_u)\}_{u=1}^m$. In fact, the $\Omega(dT/B)$ minimax lower bound is a special case of the lower bound of Theorem 3, by taking $m = 1, d_1 = d$, and $T_1 = T$.

To discuss the tightness of the upper and lower bounds we obtained so far in more detail, we first set up some useful notations. Denote by $\mathrm{E} = \mathrm{E}(\{(d_u, T_u)\}_{u=1}^m)$ the set of oblivious adversaries that shows example sequences with domain structures specified by $\{(d_u, T_u)\}_{u=1}^m$. Additionally, denote by $\mathrm{A}(B)$ the set of online active learning algorithms that uses a label budget of B. Finally, for an algorithm $\mathcal{A}$ and an oblivious adversary $\mathcal{E}$, define $\mathrm{R}(\mathcal{A}, \mathcal{E})$ as the expected regret of $\mathcal{A}$ in the environment induced by $\mathcal{E}$. Theorem 3 shows that for all $\{(d_u, T_u)\}_{u=1}^m$ and B such that Eq. (1) holds, we have

$$\min_{\mathcal{A} \in \mathrm{A}(B)} \max_{\mathcal{E} \in \mathrm{E}} \mathrm{R}(\mathcal{A}, \mathcal{E}) \ge \Omega\left(\left(\sum_{u=1}^m \sqrt{d_u T_u}\right)^2 / B\right).$$

On the other hand, Theorem 2 says for all $\{(d_u, T_u)\}_{u=1}^m$ and $B \le \tilde{O}\left(\sum_u \sqrt{d_u T_u} \min_{u \in [m]} \sqrt{T_u/d_u}\right)$, we have

$$\max_{\mathcal{E} \in \mathrm{E}} \mathrm{R}\left(\text{QuFUR}(B), \mathcal{E}\right) \le \tilde{O}\left(\left(\sum_{u=1}^m \sqrt{d_u T_u}\right)^2 / B\right).$$

This shows that, for a wide range of domain structure specifications $\left\{(d_u, T_u)\right\}_{u=1}^m$ and budgets B (i.e., $\sum_{u=1}^m d_u \le B = \tilde{O}(\sum_u \sqrt{d_u T_u} \min_{u \in [m]} \sqrt{T_u/d_u})$), the regret guarantee of Fixed-Budget QuFUR is optimal; furthermore, the algorithm requires no knowledge on the domain structure. We call this property of Fixed-Budget QuFUR its *hidden-domain minimax optimality*.

5 Extensions to realizable non-linear regression with an adaptive adversary

QuFUR's design principle, namely querying with probability proportional to uncertainty estimates of unlabeled data, can be easily generalized to deal with other active online learning problems. We demonstrate this by generalizing QuFUR to non-linear regression with adaptive adversaries, using the concept of eluder dimension from Russo and Van Roy (2013).

In this section, we relax the assumption in Section 4 that domain structure is fixed before interaction starts — we allow each input and its domain membership to depend on past history. Formally, we require the domain partition $\left\{I_u : u \in [m]\right\}$ to be *admissible*, defined as:

Definition 1. *The partition $\left\{I_u : u \in [m]\right\}$ is admissible, if the domain membership of the t-th example, $u_t \in [m]$ depends on the interaction history up to $t-1$ and unlabeled example x_t; formally, u_t is $\sigma(H_{t-1}, x_t)$-measurable.*

Domain complexity measure. Analogous to the dimension of the support in linear regression, we use $d'_u = \dim_u^E(\mathcal{F}, 1/T_u^2)$, the *eluder dimension* of $\mathcal{F}$ with respect to domain $u \in [m]$ with support $\mathcal{X}_u$, to measure the complexity of a domain, formally defined below.

Definition 2. *An input $x \in \mathcal{X}$ is ϵ-dependent on a set of inputs $\{x_i\}_{i=1}^n \subseteq \mathcal{X}$ with respect to $\mathcal{F}$ if for all $f_1, f_2 \in \mathcal{F}$, $\sqrt{\sum_{i=1}^n (f_1(x_i) - f_2(x_i))^2} \le \epsilon$ implies $f_1(x) - f_2(x) \le \epsilon$. x is ϵ-independent of $\{x_i\}_{i=1}^n$ with respect to $\mathcal{F}$ if it is not ϵ-dependent on the latter.*

Definition 3. *The ϵ-eluder dimension of $\mathcal{F}$ with respect to support $\mathcal{X}_u$, $\dim_u^E(\mathcal{F}, \epsilon)$, is defined as the length of the longest sequence of elements in $\mathcal{X}_u$ such that for some $\epsilon' > \epsilon$, every element is ϵ'-independent of its predecessors.*

The above domain-dependent eluder dimension notion captures how effective the potential value of acquiring a new label can be estimated from labeled examples in domain u.[4]

The Algorithm. The master algorithm, Algorithm 4 in Appendix A.3, runs $O(\log T)$ copies of Algorithm 3. At round t, Algorithm 3 predicts using the empirical square loss minimizer $\hat{f}_t$ based on all previously queried examples.

[4] Appendix D in Russo and Van Roy (2013) gives upper bounds of eluder dimensions for common function classes.

Algorithm 3 QuFUR(α) for Nonlinear Regression

Require: Hypothesis set $\mathcal{F}$, time horizon T, parameters α, δ, η.

Labeled dataset $\mathcal{Q} \leftarrow \emptyset$.
for $t = 1$ to T **do**
  Find $\hat{f}_t \leftarrow \mathrm{argmin}_{f \in \mathcal{F}} \sum_{i \in \mathcal{Q}} (f(x_i) - y_i)^2$.
  Predict $\hat{f}_t(x_t)$.
  Define confidence set

$$\mathcal{F}_t \leftarrow \left\{ f \in \mathcal{F} : \sum_{i \in \mathcal{Q}} (f(x_i) - \hat{f}_t(x_i))^2 \le \beta_{|\mathcal{Q}|}(\mathcal{F}, \delta) \right\},$$

  where β is defined in Equation (13).
  Uncertainty $\Delta_t = \sup_{f_1, f_2 \in \mathcal{F}_t} \left| f_1(x_t) - f_2(x_t) \right|^2$.
  With probability $\min\{1, \alpha\Delta_t\}$, set $q_t = 1$; otherwise set $q_t = 0$.
  if $q_t = 1$ **then**
    Query y_t. $\mathcal{Q} \leftarrow \mathcal{Q} \bigcup \{t\}$.

Same as Algorithm 1, Algorithm 3 queries with probability $\min\{1, \alpha\Delta_t\}$, where Δ_t is an uncertainty measure of the algorithm on example x_t. To compute the uncertainty measure, it constructs a confidence set $\mathcal{F}_t$, so that with high probability, $\mathcal{F}_t$ contains the ground truth f^*. The uncertainty measure Δ_t is the squared maximum disagreement on x_t between two hypotheses in $\mathcal{F}_t$. It can be shown that with high probability, the regret and query complexity are bounded by $O(\sum_{t=1}^T \Delta_t)$ and $O(\sum_{t=1}^T \min\{1, \alpha\Delta_t\})$, respectively.

We bound the regret of the algorithm on examples from domain u in terms of domain complexity measure $R_u = \tilde{O}(\tilde{\eta}^2 d'_u \log \mathcal{N}(\mathcal{F}, T^{-2}, \|\cdot\|_\infty))$, where $\mathcal{N}(\mathcal{F}, \epsilon, \|\cdot\|_\infty)$ is the ϵ-*covering number* of $\mathcal{F}$ with respect to $\|\cdot\|_\infty$. Specifically, we prove the following theorem.

Theorem 4. *Suppose the example sequence $\{x_t\}_{t=1}^T$ has the following structure: $[T]$ has an admissible partition $\left\{I_u : u \in [m]\right\}$, where for each u, $|I_u| = T_u$, and the eluder dimension of $\mathcal{F}$ w.r.t. $\{x_t\}_{t \in I_u}$ is d'_u. Then, given label budget $B \le \tilde{O}(\sum_u \sqrt{R_u T_u} \min_u \sqrt{R_u/T_u})$, Algorithm 4 satisfies:*
1. It has query complexity $Q \le B$;
2. With probability $1 - \delta$, its regret $R = \tilde{O}((\sum_u \sqrt{R_u T_u})^2 / B)$.

The proof of the theorem can be found in Appendix A.3. Specializing the theorem to linear hypothesis class $\mathcal{F} = \{\langle x, \theta \rangle : \theta \in \mathbb{R}^d, \|\theta\|_2 \le 1\}$, if $\mathcal{X}_u$ is a subset of a d_u-dimensional subspace of $\mathbb{R}^d$, we have $\dim_u^E(\mathcal{F}, 1/T_u^2) = \tilde{O}(d_u)$ and $\log \mathcal{N}(\mathcal{F}, 1/T_u^2, \|\cdot\|_\infty) = \tilde{O}(d)$, implying $R_u = \tilde{O}(\tilde{\eta}^2 d_u d)$, which in turn implies that $R = \tilde{O}(\tilde{\eta}^2 d (\sum_u \sqrt{d_u T_u})^2 / B)$. Compared to Theorem 2, we conjecture that the additional factor d is due to the increased difficulty with adaptive adversaries.

6 Experiments

We evaluate the query-regret tradeoffs of QuFUR, the uniform query baseline (Section 3.2), and naive greedy query (i.e., always query until labeling budget is exhausted) on two linear regression and two classification tasks. Although QuFUR is designed for regression, experiments show that the same query strategy also achieves competitive performance on high-dimensional multi-class classifications tasks. See Appendix H for more details.

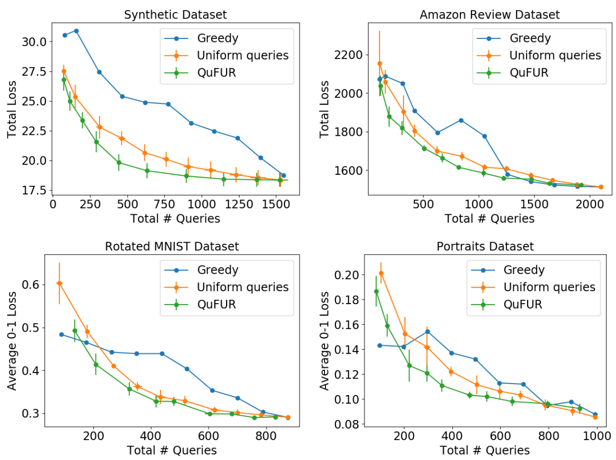

Figure 1: Total squared loss (for regression tasks) / average 0-1 error (for classification tasks) vs. total number of queries in synthetic, Amazon reviews, rotated MNIST, and portraits datasets. Error bars show stddev across 5 runs. QuFUR has the best tradeoffs.

Synthetic dataset is a regression task where the label is generated via $y_t = x_t^\top \theta^* + \xi_t$. Inputs x_i's come from 20 domains that are orthogonal linear subspaces with $d = 88$. Each domain u has either $T_u = 100$ and $d_u = 6$, or $T_u = 50$ and $d_u = 3$. θ^* is a random vector on the unit sphere in $\mathbb{R}^d$. For any x_i in domain u, $x_i = V_u z_i$ where V_u is an orthonormal basis of $\mathcal{X}_u$, and z_i is drawn from the unit sphere in $\mathbb{R}^{d_u}$. Noise ξ_t's are iid zero-mean Gaussian with variance $\eta^2 = 0.1$.

Amazon review dataset (McAuley et al., 2015) is a regression task where we predict ratings from 1 to 5 based on review text. Reviews come from 3 topics / domains: automotive, grocery, video games. We assume that the domains come in succession, with durations $300, 600, 1200$. Each review is encoded as a 768-dimensional vector — the average BERT embedding (Devlin et al., 2018) of each word in the review. Each domain uses a subset of the vocabulary, so the embeddings within the domain reside in a subspace (Reif et al., 2019). The sub-vocabularies are of smaller size (and largely disjoint), motivating our low-dimensional subspace structure for linear models. To check that realizability is a reasonable assumption, we verify that offline linear regression on all domains achieve an MSE of 0.62.

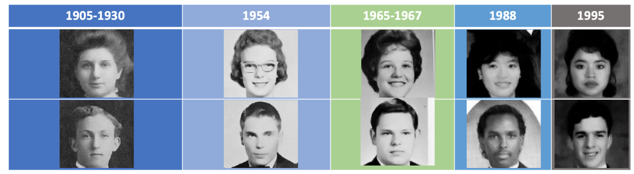

Figure 2: Illustration of modified portraits dataset. Block lengths (not drawn to scale) indicate domain durations, which are unknown to the learner. Facial features for each gender shift over time.

Rotated MNIST dataset (LeCun et al., 2010) is a 10-way classification task. We create 3 domains via rotating the images 60, 30, and 0 degrees. Domain durations are $500, 250$, and 125 in Figure 1. We check that a linear classifier trained on all domains obtain 100% training accuracy.

Portraits dataset (Ginosar et al., 2015; Kumar et al., 2020) contains photos of high school seniors taken across 1900s-2010s, labeled by gender. We sort the photos chronologically, and divide into 5 periods with 8000 photos each. We pick the first $\{512, 256, 128, 64, 32\}$ photos from each period to obtain 5 domains (Figure 2). We check that a linear classifier trained on all domains obtain 99% training accuracy.

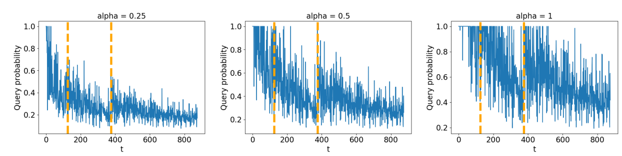

Figure 3: QuFUR's query probability in the rotated MNIST experiment with domain durations 125 / 250 / 500, when alpha is set to 0.25 (**left**), 0.5 (**middle**), 1 (**right**). QuFUR queries more frequently upon domain shift.

Results. We run QuFUR(α) for α sweeping an appropriate range for each dataset, and uniform queries with probability $\mu \in [0.05, 1]$. Figure 1 shows that QuFUR achieves the lowest total regret under the same labeling budget across all datasets. Figure 3 shows that QuFUR's query probability abruptly rises upon domain shifts. We choose highly heterogeneous domain durations since our theory predicts that QuFUR is likely to have the most savings in such situations. We show in Appendix H that in other setups, QuFUR still has competitive performance.

7 Conclusion

We formulate a novel task of active online learning with latent domain structure. We propose a surprisingly simple algorithm that adapts to domain shifts, and give matching upper and lower bounds for a wide range of domain structure specifications for linear regression. The strategy can be generalized to other problems, as we did for non-linear regression, relying on a suitable uncertainty estimate for unlabeled data. We believe that our problem and solution can spur future work on making online learning more practical.

Acknowledgements

We would like to thank Alina Beygelzimer, Akshay Krishnamurthy, Rob Schapire and Steven Wu for helpful discussions on active learning in the online setting and thank Chelsea Finn and Sergey Levine for discussing the motivation of the proposed questions. YC is supported by Stanford Graduate Fellowship. HL is supported by NSF Awards IIS-1755781 and IIS-1943607. TM is partially supported by the Google Faculty Award, Stanford Data Science Initiative, Stanford Artificial Intelligence Laboratory, and Lam Research. CZ acknowledges the startup funding from the University of Arizona for support.

References

Alekh Agarwal. Selective sampling algorithms for cost-sensitive multiclass prediction. In *International Conference on Machine Learning*, pages 1220–1228, 2013.

Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.

Katy S Azoury and Manfred K Warmuth. Relative loss bounds for on-line density estimation with the exponential family of distributions. *Machine Learning*, 43(3): 211–246, 2001.

Maria-Florina Balcan, Alina Beygelzimer, and John Langford. Agnostic active learning. *Journal of Computer and System Sciences*, 75(1):78–89, 2009.

Peter L Bartlett, Varsha Dani, Thomas Hayes, Sham Kakade, Alexander Rakhlin, and Ambuj Tewari. High-probability regret bounds for bandit online linear optimization. 2008.

Alina Beygelzimer, Daniel J Hsu, John Langford, and Tong Zhang. Agnostic active learning without constraints. In *Advances in neural information processing systems*, pages 199–207, 2010.

Olivier Bousquet and Manfred K Warmuth. Tracking a small set of experts by mixing past posteriors. *Journal of Machine Learning Research*, 3(Nov):363–396, 2002.

Giovanni Cavallanti, Nicolò Cesa-Bianchi, and Claudio Gentile. Learning noisy linear classifiers via adaptive and selective sampling. *Machine learning*, 83(1):71–102, 2011.

Nicolo Cesa-Bianchi, Alex Conconi, and Claudio Gentile. On the generalization ability of on-line learning algorithms. *IEEE Transactions on Information Theory*, 50(9): 2050–2057, 2004a.

Nicolo Cesa-Bianchi, Gábor Lugosi, and Gilles Stoltz. Minimizing regret with label efficient prediction. In *International Conference on Computational Learning Theory*, pages 77–92. Springer, 2004b.

Nicolo Cesa-Bianchi, Claudio Gentile, and Luca Zaniboni. Worst-case analysis of selective sampling for linear classification. *Journal of Machine Learning Research*, 7(Jul): 1205–1230, 2006.

Nicolo Cesa-Bianchi, Claudio Gentile, and Francesco Orabona. Robust bounds for classification via selective sampling. In *Proceedings of the 26th annual international conference on machine learning*, pages 121–128, 2009.

Kamalika Chaudhuri, Sham M Kakade, Praneeth Netrapalli, and Sujay Sanghavi. Convergence rates of active learning for maximum likelihood estimation. In *Advances in Neural Information Processing Systems*, pages 1090–1098, 2015.

Kamalika Chaudhuri, Prateek Jain, and Nagarajan Natarajan. Active heteroscedastic regression. In *International Conference on Machine Learning*, pages 694–702, 2017.

David Cohn, Les Atlas, and Richard Ladner. Improving generalization with active learning. *Machine learning*, 15 (2):201–221, 1994.

Amit Daniely, Alon Gonen, and Shai Shalev-Shwartz. Strongly adaptive online learning. In *International Conference on Machine Learning*, pages 1405–1411, 2015.

Sanjoy Dasgupta and Daniel Hsu. Hierarchical sampling for active learning. In *Proceedings of the 25th international conference on Machine learning*, pages 208–215, 2008.

Sanjoy Dasgupta, Daniel J Hsu, and Claire Monteleoni. A general agnostic active learning algorithm. In *Advances in neural information processing systems*, pages 353–360, 2008.

Ofer Dekel, Claudio Gentile, and Karthik Sridharan. Robust selective sampling from single and multiple teachers. In *COLT*, pages 346–358, 2010.

Michał Dereziński and Manfred K Warmuth. Reverse iterative volume sampling for linear regression. *The Journal of Machine Learning Research*, 19(1):853–891, 2018.

Michal Derezinski, Manfred KK Warmuth, and Daniel J Hsu. Leveraged volume sampling for linear regression. In *Advances in Neural Information Processing Systems*, pages 2505–2514, 2018.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Petros Drineas, Michael W Mahoney, and Shan Muthukrishnan. Sampling algorithms for l 2 regression and applications. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 1127–1136. Society for Industrial and Applied Mathematics, 2006.

Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Jacob Steinhardt, and Aleksander Madry. Identifying statistical bias in dataset replication. *arXiv preprint arXiv:2005.09619*, 2020.

Valerii V Fedorov and Peter Hackl. *Model-oriented design of experiments*, volume 125. Springer Science & Business Media, 2012.

Xavier Fontaine, Pierre Perrault, and Vianney Perchet. Active linear regression. *arXiv preprint arXiv:1906.08509*, 2019.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

Shiry Ginosar, Kate Rakelly, Sarah Sachs, Brian Yin, and Alexei A Efros. A century of portraits: A visual historical record of american high school yearbooks. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 1–7, 2015.

Steve Hanneke. Rates of convergence in active learning. *The Annals of Statistics*, 39(1):333–361, 2011.

Steve Hanneke. Theory of disagreement-based active learning. *Foundations and Trends® in Machine Learning*, 7 (2-3):131–309, 2014.

Elad Hazan. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

Elad Hazan and Comandur Seshadhri. Adaptive algorithms for online decision problems. In *Electronic colloquium on computational complexity (ECCC)*, volume 14, 2007.

Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2-3):169–192, 2007.

Samory Kpotufe, Ruth Urner, and Shai Ben-David. Hierarchical label queries with data-dependent partitions. In *Conference on Learning Theory*, pages 1176–1189, 2015.

Ananya Kumar, Tengyu Ma, and Percy Liang. Understanding self-training for gradual domain adaptation. In *International Conference on Machine Learning*, pages 5468–5479. PMLR, 2020.

Tor Lattimore and Csaba Szepesvári. Bandit algorithms. *preprint*, 2018.

Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann. lecun. com/exdb/mnist*, 2, 2010.

Lihong Li, Michael L Littman, Thomas J Walsh, and Alexander L Strehl. Knows what it knows: a framework for self-aware learning. *Machine learning*, 82(3):399–443, 2011.

Andrea Locatelli, Alexandra Carpentier, and Samory Kpotufe. Adaptivity to noise parameters in nonparametric active learning. *Proceedings of Machine Learning Research vol*, 65:1–34, 2017.

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pages 43–52, 2015.

Stanislav Minsker. Plug-in approach to active learning. *The Journal of Machine Learning Research*, 13(1):67–90, 2012.

Piyush Rai, Avishek Saha, Hal Daumé III, and Suresh Venkatasubramanian. Domain adaptation meets active learning. In *Proceedings of the NAACL HLT 2010 Workshop on Active Learning for Natural Language Processing*, pages 27–32. Association for Computational Linguistics, 2010.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do cifar-10 classifiers generalize to cifar-10? *arXiv preprint arXiv:1806.00451*, 2018.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? *arXiv preprint arXiv:1902.10811*, 2019.

Emily Reif, Ann Yuan, Martin Wattenberg, Fernanda B Viegas, Andy Coenen, Adam Pearce, and Been Kim. Visualizing and measuring the geometry of bert. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8594–8603. Curran Associates, Inc., 2019.

Daniel Russo and Benjamin Van Roy. Eluder dimension and the sample complexity of optimistic exploration. In *Advances in Neural Information Processing Systems*, pages 2256–2264, 2013.

Sivan Sabato and Remi Munos. Active regression by stratification. In *Advances in Neural Information Processing Systems*, pages 469–477, 2014.

Avishek Saha, Piyush Rai, Hal Daumé, Suresh Venkatasubramanian, and Scott L DuVall. Active supervised domain adaptation. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 97–112. Springer, 2011.

Amin Sayedi, Morteza Zadimoghaddam, and Avrim Blum. Trading off mistakes and don't-know predictions. In *Advances in Neural Information Processing Systems*, pages 2092–2100, 2010.

Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.

Eliza Strickland. Ai-human partnerships tackle "fake news": Machine learning can get you only so far-then human judgment is required - [news]. *IEEE Spectrum*, 55:12–13, 2018.

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

Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert MÃžller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8(May):985–1005, 2007.

István Szita and Csaba Szepesvári. Agnostic kwik learning and efficient approximate reinforcement learning. In *Proceedings of the 24th Annual Conference on Learning Theory*, pages 739–772, 2011.

Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *CVPR 2011*, pages 1521–1528. IEEE, 2011.

Michal Valko, Nathaniel Korda, Rémi Munos, Ilias Flaounas, and Nelo Cristianini. Finite-time analysis of kernelised contextual bandits. *arXiv preprint arXiv:1309.6869*, 2013.

Volodya Vovk. Competitive on-line statistics. *International Statistical Review*, 69(2):213–248, 2001.

Min Xiao and Yuhong Guo. Online active learning for cost sensitive domain adaptation. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 1–9, 2013.

Liu Yang. Active learning with a drifting distribution. In *Advances in Neural Information Processing Systems*, pages 2079–2087, 2011.

Chicheng Zhang and Kamalika Chaudhuri. The extended littlestone's dimension for learning with mistakes and abstentions. In *Conference on Learning Theory*, pages 1584–1616, 2016.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Kai Zheng, Haipeng Luo, Ilias Diakonikolas, and Liwei Wang. Equipping experts/bandits with long-term memory. In *Advances in Neural Information Processing Systems*, pages 5927–5937, 2019.