Efficient Sublinear-Regret Algorithms for Online Sparse Linear Regression with Limited Observation

Shinji Ito
NEC Corporation
s-ito@me.jp.nec.com

Daisuke Hatano
National Institute of Informatics
hatano@nii.ac.jp

Hanna Sumita
National Institute of Informatics
sumita@nii.ac.jp

Akihiro Yabe
NEC Corporation
a-yabe@cq.jp.nec.com

Takuro Fukunaga
JST, PRESTO
takuro@nii.ac.jp

Naonori Kakimura
Keio University
kakimura@math.keio.ac.jp

Ken-ichi Kawarabayashi
National Institute of Informatics
k-keniti@nii.ac.jp

Abstract

Online sparse linear regression is the task of applying linear regression analysis to examples arriving sequentially subject to a resource constraint that a limited number of features of examples can be observed. Despite its importance in many practical applications, it has been recently shown that there is no polynomial-time sublinear-regret algorithm unless **NP**$\subseteq$**BPP**, and only an exponential-time sublinear-regret algorithm has been found. In this paper, we introduce mild assumptions to solve the problem. Under these assumptions, we present polynomial-time sublinear-regret algorithms for the online sparse linear regression. In addition, thorough experiments with publicly available data demonstrate that our algorithms outperform other known algorithms.

1 Introduction

In online regression, a learner receives examples one by one, and aims to make a good prediction from the features of arriving examples, learning a model in the process. Online regression has attracted attention recently in the research community in managing massive learning data.In real-world scenarios, however, with resource constraints, it is desired to make a prediction with only a limited number of features per example. Such scenarios arise in the context of medical diagnosis of a disease [3] and in generating a ranking of web pages in a search engine, in which it costs to obtain features or only partial features are available in each round. In both these examples, predictions need to be made sequentially because a patient or a search query arrives online.

To resolve the above issue of limited access to features, Kale [8] proposed *online sparse regression*. In this problem, a learner makes a prediction for the labels of examples arriving sequentially over a number of rounds. Each example has d features that can be potentially accessed by the learner. However, in each round, the learner can acquire the values of at most k' features out of the d features, where k' is a parameter set in advance. The learner then makes a prediction for the label of the example. After the prediction, the true label is revealed to the learner, and the learner suffers a loss for making an incorrect prediction. The performance of the prediction is measured here by the standard notion of *regret*, which is the difference between the total loss of the learner and the total

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

Table 1: Computational complexity of online sparse linear regression.

Assumptions (1)	(2)	(a)	(b)	Time complexity
✓	✓			Hard [6]
✓		✓		**Hard (Theorem 1)**
✓	✓	✓		**Polynomial time (Algorithms 1, 2)**
✓	✓		✓	**Polynomial time (Algorithm 3)**

loss of the best predictor. In [8], the best predictor is defined as the best k-sparse linear predictor, i.e., the label is defined as a linear combination of at most k features.

Online sparse regression is a natural online variant of sparse regression; however, its computational complexity was not well known until recently, as Kale [8] raised a question of whether it is possible to achieve sublinear regret in polynomial time for online sparse linear regression. Foster et al. [6] answered the question by proving that no polynomial-time algorithm achieves sublinear regret unless **NP**$\subseteq$**BPP**. Indeed, this hardness result holds even when observing $\Omega(k \log d)$ features per example. On the positive side, they also proposed an exponential-time algorithm with sublinear regret, when we can observe at least $k + 2$ features in each round. However, their algorithm is not expected to work efficiently in practice. In fact, the algorithm enumerates all the $\binom{d}{k'}$ possibilities to determine k' features in each round, which requires exponential time for any instance.

Our contributions. In this paper, we show that online sparse linear regression admits a polynomial-time algorithm with sublinear regret, under mild practical assumptions. First, we assume that the features of examples arriving online are determined by a hidden distribution (Assumption (1)), and the labels of the examples are determined by a weighted average of k features, where the weights are fixed through all rounds (Assumption (2)). These are natural assumptions in the online linear regression. However, Foster et al. [6] showed that no polynomial-time algorithm can achieve sublinear regret unless **NP**$\subseteq$**BPP** even under these two assumptions.[1]

Owing to this hardness, we introduce two types of conditions on the distribution of features, both of which are closely related to the restricted isometry property (RIP) that has been studied in the literature of sparse recovery. The first condition, which we call *linear independence of features* (Assumption (a)), is stronger than RIP. This condition roughly says that all the features are linearly independent. The second condition, which we call *compatibility* (Assumption (b)), is weaker than RIP. Thus, an instance having RIP always satisfies the compatibility condition. Under these assumptions, we propose the following three algorithms. Here, T is the number of rounds.

- Algorithm 1: A polynomial-time algorithm that achieves $O(\frac{d}{k'-k}\sqrt{T})$ regret, under Assumptions (1), (2), and (a), which requires at least $k + 2$ features to be observed per example.
- Algorithm 2: A polynomial-time algorithm that achieves $O(\sqrt{dT} + \frac{d^{16}}{k'^{16}})$ regret, under Assumptions (1), (2), and (a), which requires at least k features to be observed per example.
- Algorithm 3: A polynomial-time algorithm that achieves $O(\sqrt{dT} + \frac{d^{16}}{k'^{16}})$ regret, under Assumptions (1), (2), and (b), which requires at least k features to be observed per example.

We can also construct an algorithm achieving $O(\frac{d}{k'-k}\sqrt{T})$ regret under Assumption (b) for the case where $k' \geq k + 2$, analogous to Algorithm 1, but we omit it due to space limitations.

Assumptions (1)+(2)+(a) or (1)+(2)+(b) seem to be minimal assumptions needed to achieve sublinear regret in polynomial time. Indeed, as listed in Table 1, the problem is hard if any one of the assumptions is violated, where *hard* means that no polynomial-time algorithm can achieve sublinear regret unless **NP**$\subseteq$**BPP**. Note that Assumption (a) is stronger than (b).

In addition to proving theoretical regret bounds of our algorithms, we perform thorough experiments to evaluate the algorithms. We verified that our algorithms outperform the exponential-time algorithm [6] in terms of computational complexity as well as performance of the prediction. Our algorithms also outperform (baseline) heuristic-based algorithms and algorithms proposed in [2, 7]

[1] Although the statement in [6] does not mention the assumptions, its proof indicates that the hardness holds even with these assumptions.

for online learning based on limited observation. Moreover, we observe that our algorithms perform well even for a real dataset, which may not satisfy our assumptions (deciding whether the model satisfies our assumptions is difficult; for example, the RIP parameter cannot be approximated within any constant factor under a reasonable complexity assumption [10]). Thus, we can conclude that our algorithm is applicable in practice.

Overview of our techniques. One naive strategy for choosing a limited number of features is to choose "large-weight" features in terms of estimated ground-truth regression weights. This strategy, however, does not achieve sublinear regret, as it ignores small-weight features. When we have Assumption (a), we show that if we observe two more features chosen uniformly at random, together with the largest k features, we can make a good prediction. More precisely, using the observed features, we output the label that minimizes the least-square loss function, based on the technique using an unbiased estimator of the gradient [2, 7] and the regularized dual averaging (RDA) method (see, e.g., [12, 4]). This idea gives Algorithm 1, and the details are given in Section 4. The reason why we use RDA is that it is efficient in terms of computational time and memory space as pointed out in [12] and, more importantly, we will combine this with the ℓ_1 regularization later. However, this requires at least $k + 2$ features to be observed in each round.

To avoid the requirement of two extra observations, the main idea is to employ Algorithm 1 with a partial dataset. As a by-product of Algorithm 1, we can estimate the ground-truth regression weight vector with high probability, even without observing extra features in each round. We use the ground-truth weight vector estimated by Algorithm 1 to choose k features. Combining this idea with RDA adapted for the sparse regression gives Algorithm 2 (Section 5.1) under Assumption (a).

The compatibility condition (Assumption (b)) is often used in LASSO (Least Absolute Shrinkage and Selection Operator), and it is known that minimization with an ℓ_1 regularizer converges to the sparse solution under the compatibility condition [1]. We introduce ℓ_1 regularization into Algorithm 1 to estimate the ground-truth regression weight vector when we have Assumption (b) instead of Assumption (a). This gives Algorithm 3 (Section 5.2).

Related work. In the online learning problem, a learner aims to predict a model based on the arriving examples. Specifically, in the linear function case, a learner predicts the coefficient $\mathbf{w}_t$ of a linear function $\mathbf{w}_t^\top \mathbf{x}_t$ whenever an example with features $\mathbf{x}_t$ arrives in round t. The learner then suffers a loss $\ell_t(\mathbf{w}_t) = (y_t - \mathbf{w}_t^\top \mathbf{x}_t)^2$. The aim is to minimize the total loss $\sum_{t=1}^{T}(\ell_t(\mathbf{w}_t) - \ell_t(\mathbf{w}))$ for an arbitrary $\mathbf{w}$. It is known that both the gradient descent method [13] and the dual averaging method [12] attain an $O(\sqrt{T})$ regret even for the more general convex function case. However, these methods require access to all features of the examples.

In *linear regression with limited observation*, the limited access to features in regression has been considered [2, 7]. In this problem, a learner can acquire only the values of at most k' features among d features. The purpose here is to estimate a good weight vector, e.g., minimize the loss function $\ell(\mathbf{w})$ or the loss function with ℓ_1 regularizer $\ell(\mathbf{w}) + \|\mathbf{w}\|_1$. Let us note that, even if we obtain a good weight vector $\mathbf{w}$ with small $\ell(\mathbf{w})$, we cannot always compute $\mathbf{w}^\top \mathbf{x}_t$ from limited observation of $\mathbf{x}_t$ and, hence, in our setting the prediction error might not be as small as $\ell(\mathbf{w})$. Thus, our setting uses a different loss function, defined in Section 2, to minimize the prediction error.

Another problem incorporating the limited access is proposed by Zolghadr et al. [14]. Here, instead of observing k' features, one considers the situation where obtaining a feature has an associated cost. In each round, one chooses a set of features to pay some amount of money, and the purpose is to minimize the sum of the regret and the total cost. They designed an exponential-time algorithm for the problem.

Online sparse linear regression has been studied in [6, 8], but only an exponential-time algorithm has been proposed so far. In fact, Foster et al. [6] suggested designing an efficient algorithm for a special class of the problem as future work. The present paper aims to follow this suggestion.

Recently, Kale et al. [9][2] presented computationally efficient algorithms to achieve sublinear regret under the assumption that input features satisfy RIP. Though this study includes similar results to ours, we can realize some differences. Our paper considers the assumption of the compatibility condition without extra observation (i.e., the case of $k' = k$), whereas Kale et al. [9] studies a

[2]The paper [9] was published after our manuscript was submitted.

stronger assumption with extra observation ($k' \geq k+2$) that yields a smaller regret bound than ours. They also studies the agnostic (adversarial) setting.

2 Problem setting

Online sparse linear regression. We suppose that there are T rounds, and an example arrives online in each round. Each example is represented by d features and is associated with a label, where features and labels are all real numbers. We denote the features of the example arriving in round t by $\mathbf{x}_t = (x_{t1}, \ldots, x_{td})^\top \in \{\mathbf{x} \in \mathbb{R}^d \mid \|\mathbf{x}\| \leq 1\}$, where the norm $\|\cdot\|$ without subscripts denotes the ℓ_2 norm. The label of each example is denoted by $y_t \in [-1, 1]$.

The purpose of the online sparse regression is to predict the label $y_t \in \mathbb{R}$ from a partial observation of $\mathbf{x}_t$ in each round $t = 1, \ldots, T$. The prediction is made through the following four steps: (i) we choose a set $S_t \subseteq [d] := \{1, \ldots, d\}$ of features to observe, where $|S_t|$ is restricted to be at most k'; (ii) observe the selected features $\{x_{ti}\}_{i \in S_t}$; (iii) on the basis of observation $\{x_{ti}\}_{i \in S_t}$, estimate a predictor $\hat{y}_t$ of y_t; and (iv) observe the true value of y_t.

From S_t, we define $D_t \in \mathbb{R}^{d \times d}$ to be the diagonal matrix such that its (i,i)th entries are 1 for $i \in S_t$ and the other entries are 0. Then, observing the selected features $\{x_{ti}\}_{i \in S_t}$ in (ii) is equivalent to observing $D_t \mathbf{x}_t$. The predictor $\hat{y}_t$ is computed by $\hat{y}_t = \mathbf{w}_t^\top D_t \mathbf{x}_t$ in (iii).

Throughout the paper, we assume the following conditions, corresponding to Assumptions (1) and (2) in Section 1, respectively.

Assumption (1) There exists a weight vector $\mathbf{w}^* \in \mathbb{R}^d$ such that $\|\mathbf{w}\| \leq 1$ and $y_t = \mathbf{w}^{*\top} \mathbf{x}_t + \epsilon_t$ for all $t = 1, \ldots, T$, where $\epsilon_t \sim \mathcal{D}_\epsilon$, independent and identically distributed (i.i.d.), and $\mathbf{E}[\epsilon_t] = 0$, $\mathbf{E}[{\epsilon_t}^2] = \sigma^2$. There exists a distribution $\mathcal{D}_\mathbf{x}$ on $\mathbb{R}^d$ such that $\mathbf{x}_t \sim \mathcal{D}_\mathbf{x}$, i.i.d. and independent of $\{\epsilon_t\}$.

Assumption (2) The true weight vector $\mathbf{w}^*$ is k-sparse, i.e., $S^* = \mathrm{supp}(\mathbf{w}^*) = \{i \in [d] \mid w_i^* \neq 0\}$ satisfies $|S^*| \leq k$.

Regret. The performance of the prediction is evaluated based on the *regret* $R_T(\mathbf{w})$ defined by

$$R_T(\mathbf{w}) = \sum_{t=1}^{T} (\hat{y}_t - y_t)^2 - \sum_{t=1}^{T} (\mathbf{w}^\top \mathbf{x}_t - y_t)^2. \tag{1}$$

Our goal is to achieve smaller regret $R_T(\mathbf{w})$ for an arbitrary $\mathbf{w} \in \mathbb{R}^d$ such that $\|\mathbf{w}\| \leq 1$ and $\|\mathbf{w}\|_0 \leq k$. For random inputs and randomized algorithms, we consider the expected regret $\max_{\mathbf{w}:\|\mathbf{w}\|_0 \leq k, \|\mathbf{w}\| \leq 1} \mathbf{E}[R_T(\mathbf{w})]$.

Define the loss function $\ell_t(\mathbf{w}) = (\mathbf{w}^\top \mathbf{x}_t - y_t)^2$. If we compute a predictor $\hat{y}_t = \mathbf{w}_t^\top D_t \mathbf{x_t}$ using a weight vector $\mathbf{w}_t = (w_{t1}, \ldots, w_{td})^\top \in \mathbb{R}^d$ in each step, we can rewrite the regret $R_T(\mathbf{w})$ in (1) using D_t and $\mathbf{w}_t$ as

$$R_T(\mathbf{w}) = \sum_{t=1}^{T} (\ell_t(D_t \mathbf{w}_t) - \ell_t(\mathbf{w})) \tag{2}$$

because $(\hat{y}_t - y_t)^2 = (\mathbf{w}_t^\top D_t \mathbf{x}_t - y_t)^2 = \ell_t(D_t \mathbf{w}_t)$. It is worth noting that if our goal is only to construct $\mathbf{w}_t$ that minimizes the loss function $\ell_t(\mathbf{w}_t)$, then the definition of the regret should be

$$R'_T(\mathbf{w}) = \sum_{t=1}^{T} (\ell_t(\mathbf{w}_t) - \ell_t(\mathbf{w})). \tag{3}$$

However, the goal of online sparse regression involves predicting $\mathbf{y}_t$ from the limited observation. Hence, we use (2) to evaluate the performance. In terms of the regret defined by (3), several algorithms based on limited observation have been developed. For example, the algorithms proposed by Cesa-Bianchi et al. [3] and Hazan and Koren [7] achieve $O(\sqrt{T})$ regret of (3).

3 Extra assumptions on features of examples

Foster et al. [6] showed that Assumptions (1) and (2) are not sufficient to achieve sublinear regret. Owing to this observation, we impose extra assumptions.

Let $V := \mathbf{E}[\mathbf{x}_t^\top \mathbf{x}_t] \in \mathbb{R}^{d \times d}$ and let L be the Cholesky decomposition of V (i.e., $V = L^\top L$). Denote the largest and the smallest singular values of L by σ_1 and σ_d, respectively. Under Assumption (1) in Section 2, we have $\sigma_1 \leq 1$ because, for arbitrary unit vector $\mathbf{u} \in \mathbb{R}^d$, it holds that $\mathbf{u}^\top V \mathbf{u} = \mathbf{E}[(\mathbf{u}^\top \mathbf{x})^2] \leq 1$. For a vector $\mathbf{w} \in \mathbb{R}^{[d]}$ and $S \subseteq [d]$, we let $\mathbf{w}_S$ denote the restriction of $\mathbf{w}$ onto S. For $S \subseteq [d]$, S^c denotes $[d] \setminus S$. We assume either one of the following conditions holds.

(a) **Linear independence of features:** $\sigma_d > 0$.

(b) **Compatibility:** There exists a constant $\phi_0 > 0$ that satisfies $\phi_0^2 \|\mathbf{w}_{S^*}\|_1^2 \leq k \mathbf{w}^\top V \mathbf{w}$ for all $\mathbf{w} \in \mathbb{R}^d$ with $\|\mathbf{w}_{(S^*)^c}\|_1 \leq 2\|\mathbf{w}_{S^*}\|_1$.

We assume the linear independence of features in Sections 4 and 5.1, and the compatibility in Section 5.2 to develop efficient algorithms.

Note that condition (a) means that L is non-singular, and so is V. In other words, condition (a) indicates that the features in $\mathbf{x}_t$ are linearly independent. This is the reason why we call condition (a) the "linear independence of features" assumption. Note that the linear independence of features does *not* imply the *stochastic* independence of features.

Conditions (a) and (b) are closely related to RIP. Indeed, condition (b) is a weaker assumption than RIP, and RIP is weaker than condition (a), i.e., (a) linear independence of features $\Longrightarrow$ RIP $\Longrightarrow$ (b) compatibility (see, e.g., [1]). We now clarify how the above two assumptions are connected to the regret. The expectation of the loss function $\ell_t(\mathbf{w})$ is equal to

$$\begin{aligned}\mathbf{E}_{\mathbf{x}_t, y_t}[\ell_t(\mathbf{w})] &= \mathbf{E}_{\mathbf{x}_t \sim \mathcal{D}_\mathbf{x}, \epsilon_t \sim \mathcal{D}_\epsilon}[(\mathbf{w}^\top \mathbf{x}_t - \mathbf{w}^{*\top} \mathbf{x}_t - \epsilon_t)^2] \\ &= \mathbf{E}_{\mathbf{x}_t \sim \mathcal{D}_\mathbf{x}}[((\mathbf{w} - \mathbf{w}^*)^\top \mathbf{x}_t)^2] + \mathbf{E}_{\epsilon_t \sim \mathcal{D}_\epsilon}[\epsilon_t^\top \epsilon_t] = (\mathbf{w} - \mathbf{w}^*)^\top V (\mathbf{w} - \mathbf{w}^*) + \sigma^2\end{aligned}$$

for all t, where the second equality comes from $\mathbf{E}[\epsilon_t] = 0$ and that $\mathbf{x}_t$ and ϵ_t are independent. Denote this function by $\ell(\mathbf{w})$, and then $\ell(\mathbf{w})$ is minimized when $\mathbf{w} = \mathbf{w}^*$. If D_t and $\mathbf{w}_t$ are determined independently of $\mathbf{x}_t$ and y_t, the expectation of the regret $R_T(\mathbf{w})$ satisfies

$$\begin{aligned}\mathbf{E}[R_T(\mathbf{w})] &= \mathbf{E}[\sum_{t=1}^T (\ell(D_t \mathbf{w}_t) - \ell(\mathbf{w}))] \leq \mathbf{E}[\sum_{t=1}^T (\ell(D_t \mathbf{w}_t) - \ell(\mathbf{w}^*))] \\ &= \mathbf{E}[\sum_{t=1}^T (D_t \mathbf{w}_t - \mathbf{w}^*)^\top V (D_t \mathbf{w}_t - \mathbf{w}^*)] = \mathbf{E}[\sum_{t=1}^T \|L(D_t \mathbf{w}_t - \mathbf{w}^*)\|^2]. \qquad (4)\end{aligned}$$

We bound (4) in the analysis.

Hardness result. Similarly to [6], we can show that it remains hard under Assumptions (1), (2), and (a). Refer to Appendix A for the proof.

Theorem 1. *Let D be any positive constant, and let $c_D \in (0, 1)$ be a constant dependent on D. Suppose that Assumptions (1) and (2) hold with $k = O(d^{c_D})$ and $k' = \lfloor kD \ln d \rfloor$. If an algorithm for the online sparse regression problem runs in $\mathrm{poly}(d, T)$ time per iteration and achieves a regret at most $\mathrm{poly}(d, 1/\sigma_d) T^{1-\delta}$ in expectation for some constant $\delta > 0$, then* **NP**$\subseteq$**BPP**.

4 Algorithm with extra observations and linear independence of features

In this section, we present Algorithm 1. Here we assume $k' \geq k + 2$, in addition to the linear independence of features (Assumption (a)). The additional assumption will be removed in Section 5.

As noted in Section 2, our algorithm first computes a weight vector $\mathbf{w}_t$, chooses a set S_t of k' features to be observed, and computes a label $\hat{y}_t$ by $\hat{y}_t = \mathbf{w}_t^\top D_t \mathbf{x}_t$ in each round t. In addition, our algorithm constructs an unbiased estimator $\hat{\mathbf{g}}_t$ of the gradient $\mathbf{g}_t$ of the loss function $\ell_t(\mathbf{w})$ at $\mathbf{w} = \mathbf{w}_t$, i.e., $\mathbf{g}_t = \nabla_\mathbf{w} \ell_t(\mathbf{w}_t) = 2\mathbf{x}_t(\mathbf{x}_t^\top \mathbf{w}_t - y_t)$ at the end of the round. In the following, we describe how to compute $\mathbf{w}_t$, S_t, and $\hat{\mathbf{g}}_t$ in round t, respectively, assuming that $\mathbf{w}_{t'}$, $S_{t'}$, and $\hat{\mathbf{g}}_{t'}$ are computed in the previous rounds $t' = 1, \ldots, t-1$. The entire algorithm is described in Algorithm 1.

Algorithm 1

Input: $\{\mathbf{x}_t, y_t\} \subseteq \mathbb{R}^d \times \mathbb{R}$, $\{\lambda_t\} \subseteq \mathbb{R}_{>0}$, $k' \geq 2$ and $k_1 \geq 0$ such that $k_1 \leq k' - 2$.

1: Set $\hat{\mathbf{h}}_0 = 0$.
2: **for** $t = 1, \ldots, T$ **do**
3: Define $\mathbf{w}_t$ by (5) and define S_t by $\texttt{Observe}(\mathbf{w}_t, k', k_1)$.
4: Observe $D_t\mathbf{x}_t$ and output $\hat{y}_t := \mathbf{w}_t^\top D_t \mathbf{x}_t$.
5: Observe y_t and define $\hat{\mathbf{g}}_t$ by (6) and set $\hat{\mathbf{h}}_t = \hat{\mathbf{h}}_{t-1} + \hat{\mathbf{g}}_t$
6: **end for**

Computing $\mathbf{w}_t$. We use $\hat{\mathbf{g}}_1, \ldots, \hat{\mathbf{g}}_{t-1}$ to estimate $\mathbf{w}_t$ by the dual averaging method as follows. Define $\hat{\mathbf{h}}_{t-1} = \sum_{j=1}^{t-1} \hat{\mathbf{g}}_j$, which is the average of all estimators of gradients computed in the previous rounds. Moreover, let $(\lambda_1, \ldots, \lambda_T)$ be a monotonically non-decreasing sequence of positive numbers. From these, we define $\mathbf{w}_t$ by

$$\mathbf{w}_t = \operatorname*{arg\,min}_{\mathbf{w} \in \mathbb{R}^d, \|\mathbf{w}\| \leq 1} \left\{ \hat{\mathbf{h}}_{t-1}^\top \mathbf{w} + \frac{\lambda_t}{2} \|\mathbf{w}\|^2 \right\} = -\frac{1}{\max\{\lambda_t, \|\hat{\mathbf{h}}_{t-1}\|\}} \hat{\mathbf{h}}_{t-1}, \tag{5}$$

Computing S_t. Let k_1 be an integer such that $k_1 \leq k' - 2$. We define $U_t \subseteq [d]$ as the set of the k_1 largest features with respect to $\mathbf{w}_t$, i.e., choose U_t so that $|U_t| = k_1$ and all $i \in U_t$ and $j \in [d] \setminus U_t$ satisfy $|w_{ti}| \geq |w_{tj}|$. Let V_t be the set of $(k' - k_1)$ elements chosen from $[d] \setminus U_t$ uniformly at random. Then our algorithm observes the set $S_t = U_t \cup V_t$ of the k' features. We call this procedure to obtain S_t $\texttt{Observe}(\mathbf{w}_t, k', k_1)$.

Observation 1. *We observe that $U_t \subseteq S_t$ and $\mathrm{Prob}[i, j \in S_t] \geq \frac{(k'-k_1)(k'-k_1-1)}{d(d-1)} =: C_{d,k',k_1}$. Thus, $\mathrm{Prob}[i, j \in S_t] > 0$ for all $i, j \in [d]$ if $k' \geq k_1 + 2$.*

For simplicity, we use the notation $p_i^{(t)} = \mathrm{Prob}[i \in S_t]$ and $p_{ij}^{(t)} = \mathrm{Prob}[i, j \in S_t]$ for $i, j \in [d]$.

Computing $\hat{\mathbf{g}}_t$. Define $\tilde{X}_t = (\tilde{x}_{tij}) \in \mathbb{R}^{d \times d}$ by $\tilde{X}_t = D_t \mathbf{x}_t^\top \mathbf{x}_t D_t$ and let $X_t \in \mathbb{R}^{d \times d}$ be a matrix whose (i, j)-th entry is $\tilde{x}_{tij} / p_{ij}^{(t)}$. It follows that X_t is an unbiased estimator of $\mathbf{x}_t \mathbf{x}_t^\top$. Similarly, defining $\mathbf{z}_t = (z_{ti}) \in \mathbb{R}^d$ by $z_{ti} = x_{ti} / p_i^{(t)}$ for $i \in S_t$ and $z_{ti} = 0$ for $i \notin S_t$, we see that $\mathbf{z}_t$ is an unbiased estimator of $\mathbf{x}_t$. Using X_t and $\mathbf{z}_t$, we define $\hat{\mathbf{g}}_t$ to be

$$\hat{\mathbf{g}}_t = 2X_t \mathbf{w}_t - 2y_t \mathbf{z}_t. \tag{6}$$

Regret bound of Algorithm 1. Let us show that the regret achieved by Algorithm 1 is $O(\frac{d}{k'-k}\sqrt{T})$ in expectation.

Theorem 2. *Suppose that the linear independence of features is satisfied and $k \leq k' - 2$. Let k_1 be an arbitrary integer such that $k \leq k_1 \leq k' - 2$. Then, for arbitrary $\mathbf{w} \in \mathbb{R}^d$ with $\|\mathbf{w}\| \leq 1$, Algorithm 1 achieves $\mathbf{E}[R_T(\mathbf{w})] \leq \frac{3}{\sigma_d^2} \left(\frac{16}{C_{d,k',k_1}} \sum_{t=1}^T \frac{1}{\lambda_t} + \frac{\lambda_{T+1}}{2} \right)$. By setting $\lambda_t = 8\sqrt{t}/C_{d,k',k_1}$ for each $t = 1, \ldots, T$, we obtain*

$$\mathbf{E}[R_T(\mathbf{w})] \leq \frac{24}{\sigma_d^2} \sqrt{\frac{d(d-1)}{(k'-k_1)(k'-k_1-1)}} \cdot \sqrt{T+1}. \tag{7}$$

The rest of this section is devoted to proving Theorem 2. By (4), it suffices to evaluate $\mathbf{E}[\sum_{t=1}^T \|L(D_t \mathbf{w}_t - \mathbf{w}^*)\|^2]$ instead of $E[R_T(\mathbf{w})]$. The following lemma asserts that each term of (4) can be bounded, assuming the linear independence of features. Proofs of all lemmas are given in the supplementary material.

Lemma 3. *Suppose that the linear independence of features is satisfied. If $S_t \supseteq U_t$,*

$$\|L(D_t \mathbf{w}_t - \mathbf{w}^*)\|^2 \leq \frac{3}{\sigma_d^2} \|L(\mathbf{w}_t - \mathbf{w}^*)\|^2. \tag{8}$$

Proof. We have

$$\|L(D_t\mathbf{w}_t - \mathbf{w}^*)\|^2 \le \sigma_1^2\|D_t\mathbf{w}_t - \mathbf{w}^*\|^2 = \sigma_1^2\left(\sum_{i\in S^*\cap S_t}(w_{ti}-w_i^*)^2 + \sum_{i\in S^*\setminus S_t} w_i^{*2} + \sum_{i\in S_t\setminus S^*} w_{ti}^2\right)$$
$$\le \sigma_1^2\left(\|\mathbf{w}_t - \mathbf{w}^*\|^2 + \sum_{i\in S^*\setminus S_t} w_i^{*2}\right), \tag{9}$$

where the second inequality holds since $w_i^* = 0$ for $i \in [d] \setminus S^*$. It holds that

$$\sum_{i\in S^*\setminus S_t} w_i^{*2} \le \sum_{i\in S^*\setminus U_t} w_i^{*2} \le \sum_{i\in S^*\setminus U_t}\left(2w_{ti}^2 + 2(w_{ti}-w_i^*)^2\right)$$
$$\le 2\sum_{i\in U_t\setminus S^*} w_{ti}^2 + 2\sum_{i\in S^*\setminus U_t}(w_{ti}-w_i^*)^2 \le 2\|\mathbf{w}_t - \mathbf{w}^*\|^2. \tag{10}$$

The first and third inequalities come from $U_t \subseteq S_t$ and the definition of U_t. Putting (10) into (9), we have

$$\|L(D_t\mathbf{w}_t - \mathbf{w}^*)\|^2 \le 3\sigma_1^2\|\mathbf{w}_t - \mathbf{w}^*\|^2 \le \frac{3\sigma_1^2}{\sigma_d^2}\|L(\mathbf{w}_t - \mathbf{w}^*)\|^2.$$

□

It follows from the above lemma that, if $\mathbf{w}_t$ converges to $\mathbf{w}^*$, we have $D_t\mathbf{w}_t = \mathbf{w}^*$, and hence S_t includes the support of $\mathbf{w}^*$. Moreover, it holds that $\sum_{t=1}^T \mathbf{E}[\|L(\mathbf{w}_t - \mathbf{w}^*)\|^2] = \mathbf{E}[\sum_{t=1}^T(\ell_t(\mathbf{w}_t) - \ell_t(\mathbf{w}^*))] = \mathbf{E}[R'_T(\mathbf{w}^*)]$, since $\mathbf{w}_t$ is independent of $\mathbf{x}_t$ and y_t. Thus, to bound $\sum_{t=1}^T \mathbf{E}[\|L(\mathbf{w}_t - \mathbf{w}^*)\|^2]$, we shall evaluate $\mathbf{E}[R'_T(\mathbf{w}^*)]$.

Lemma 4 ([12]). *Suppose that $\mathbf{w}_t$ is defined by* (5) *for each $t = 1, \dots, T$, and $\mathbf{w} \in \mathbb{R}^d$ satisfies $\|\mathbf{w}\| \le 1$. Let $G_t = \mathbf{E}[\|\hat{\mathbf{g}}_t\|^2]$ for $t = 1, \dots, T$. Then,*

$$\mathbf{E}[R'_T(\mathbf{w})] \le \sum_{t=1}^T \frac{1}{\lambda_t}G_t + \frac{\lambda_{T+1}}{2}. \tag{11}$$

If $G_t = O(1)$ and $\lambda_t = \Theta(\sqrt{t})$, the right-hand side of (11) is $O(\sqrt{T})$. The following lemma shows that this is true if $p_{ij}^{(t)} = \Omega(1)$.

Lemma 5. *Suppose that the linear independence of features is satisfied. Let $t \in [T]$, and let q be a positive number such that $q \le \min\{p_i^{(t)}, p_{ij}^{(t)}\}$. Then we have $G_t \le 16/q$.*

We are now ready to prove Theorem 2.

Proof of Theorem 2. The expectation $\mathbf{E}[R_T(\mathbf{w})]$ of the regret is bounded as $\mathbf{E}[R_T(\mathbf{w})] \le \sum_{t=1}^T \mathbf{E}[\|L(D_t\mathbf{w_t} - \mathbf{w}^*)\|^2] \le \frac{3}{\sigma_d^2}\sum_{t=1}^T \mathbf{E}[\|L(\mathbf{w}_t - \mathbf{w}^*)\|^2] = \frac{3}{\sigma_d^2}\mathbf{E}[R'_T(\mathbf{w}^*)]$, where the first inequality comes from (4) and the second comes from Lemma 3. From Lemma 4, $\mathbf{E}[R'_T(\mathbf{w}^*)]$ is bounded by $\mathbf{E}[R'_T(\mathbf{w}^*)] \le H_T := \sum_{t=1}^T \frac{1}{\lambda_t}G_t + \frac{\lambda_{T+1}}{2}$. Lemma 5 and Observation 1 yield $G_t \le 16/C_{d,k',k_1}$. Hence, for $\lambda_t = 8\sqrt{C_{d,k',k_1}t}$, H_T satisfies $H_T \le \sum_{t=1}^T \frac{16}{C_{d,k',k_1}\lambda_t} + \frac{\lambda_{T+1}}{2} = \sum_{t=1}^T \frac{2}{\sqrt{C_{d,k',k_1}t}} + \frac{4}{\sqrt{C_{d,k',k_1}}}\sqrt{T+1} \le 8\frac{1}{\sqrt{C_{d,k',k_1}}}\sqrt{T+1}$. Combining the above three inequalities, we obtain (7). □

5 Algorithms without extra observations

5.1 Algorithm 2: Assuming (a) the linear independence of features

In Section 4, Lemma 3 showed a connection between R_T and R'_T: $\mathbf{E}[R_T(\mathbf{w})] \le \frac{3\sigma_1^2}{\sigma_d 2}\mathbf{E}[R'_T(\mathbf{w}^*)]$ under $U_t \subseteq S_t$. Then, Lemmas 4 and 5 gave an upper bound of $\mathbf{E}[R'_T(\mathbf{w}^*)]$: $\mathbf{E}[R'_T(\mathbf{w}^*)] = O(\sqrt{T})$

under $p_{ij}^{(t)} = \Omega(1)$. In the case of $k' = k$, however, the conditions $U_t \subseteq S_t$ and $p_{ij}^{(t)} = \Omega(1)$ may not be satisfied simultaneously, since, if $U_t \subseteq S_t$ and $|S_t| = k' = k \geq k_1 = |U_t|$, then we have $U_t = S_t$, which means $p_{ij}^{(t)} = 0$ for $i \notin U_t$ or $j \notin U_t$. Thus, we cannot use both relationships for the analysis. In Algorithm 2, we bound $R_T(\mathbf{w})$ without bounding $R'_T(\mathbf{w})$.

Let us describe an idea of Algorithm 2. To achieve the claimed regret, we first define a subset J of $\{1, 2, \ldots, T\}$ by the set of squares, i.e., $J = \{s^2 \mid s = 1, \ldots, \lfloor \sqrt{T} \rfloor\}$. Let t_s denote the s-th smallest number in J for each $s = 1, \ldots, |J|$. In each round t, the algorithm computes S_t, a weight vector $\tilde{\mathbf{w}}_t$, and a vector $D_t\tilde{\mathbf{g}}_t$, where $\tilde{\mathbf{g}}_t$ is the gradient of $\ell_t(\mathbf{w})$ at $\mathbf{w} = D_t\tilde{\mathbf{w}}_t$. In addition, if $t = t_s$, the algorithm computes other weight vectors $\mathbf{w}_s$ and $\bar{\mathbf{w}}_s := \frac{1}{s}\sum_{j=1}^{s} \mathbf{w}_j$, and an unbiased estimator $\hat{\mathbf{g}}_s$ of the gradient of the loss function $\ell_t(\mathbf{w})$ at $\mathbf{w}_s$.

At the beginning of round t, if $t = t_s$, the algorithm first computes $\mathbf{w}_s$, and $\bar{\mathbf{w}}_s$ is defined as the average of $\mathbf{w}_1, \ldots, \mathbf{w}_s$. Roughly speaking, $\mathbf{w}_s$ is the weight vector computed with Algorithm 1 applied to the examples $(x_{t_1}, y_{t_1}), \ldots, (x_{t_s}, y_{t_s})$, setting k_1 to be at most $k - 2$. Then, we can show that $\bar{\mathbf{w}}_s$ is a consistent estimator of $\mathbf{w}^*$. This step is only performed if $t \in J$. Then S_t is defined from $\bar{\mathbf{w}}_s$, where s is the largest number such that $t_s \leq t$. Thus, S_t does not change for any $t \in [t_s, t_{s+1} - 1]$. After this, the algorithm computes $\tilde{\mathbf{w}}_t$ from $D_1\tilde{\mathbf{g}}_1, \ldots, D_{t-1}\tilde{\mathbf{g}}_{t-1}$, and predicts the label of $\mathbf{x}_t$ as $\hat{y}_t := \tilde{\mathbf{w}}_t^\top D_t\mathbf{x}_t$. At the end of the round, the true label y_t is observed, and $D_t\tilde{\mathbf{g}}_t$ is computed from $\mathbf{w}_t$ and $(D_t\mathbf{x}_t, y_t)$. In addition, if $t = t_s$, $\hat{\mathbf{g}}_s$ is computed as in Algorithm 1. We need $\hat{\mathbf{g}}_s$ for computing $\mathbf{w}_{s'}$ with $s' > s$ in the subsequent rounds $t_{s'}$.

The following theorem bounds the regret of Algorithm 2. See the supplementary material for details of the algorithm and the proof of the theorem.

Theorem 6. *Suppose that (a), the linear independence of features, is satisfied and $k \leq k'$. Then, there exists a polynomial-time algorithm such that $\mathbf{E}[R_T(\mathbf{w})]$ is at most*

$$8(1+\sqrt{d})\sqrt{T+1}+12T\sum_{i\in S^*}|w_i^*|\exp(-\frac{C_{d,k',0}^2(T^{\frac{1}{4}}-1)|w_i^*|^2\sigma_d^2}{18432})+4\sum_{i\in S^*}|w_i^*|(\frac{4096}{C_{d,k',0}^2 w_i^{*4}\sigma_d^4}+1)^2,$$

for arbitrary $\mathbf{w} \in \mathbb{R}^d$ with $\|\mathbf{w}\| \leq 1$, where $C_{d,k',0} = \frac{k'(k'-1)}{d(d-1)} = O(\frac{k'^2}{d^2})$.[2]

5.2 Algorithm 3: Assuming (b) the compatibility condition

Algorithm 3 adopts the same strategy as Algorithm 2 except for the procedure for determining $\mathbf{w}_s$ and $\bar{\mathbf{w}}_s$. In the analysis of Algorithm 2, we show that, to achieve the claimed regret, it suffices to generate $\{S_t\}$ that satisfies $\sum_{t=1}^{T} \mathrm{Prob}[i \notin S_t] = O(\sqrt{T})$ for $i \in S^*$. The condition was satisfied by defining S_t as the set of k largest features with respect to a weight vector $\bar{\mathbf{w}}_s = \sum_{j=1}^{s} \mathbf{w}_j/s$. The linear independence of features guarantees that $\bar{\mathbf{w}}_s$ computed in Algorithm 2 converges to $\mathbf{w}^*$, and hence $\{S_t\}$ defined as above possesses the required property. Unfortunately, if the assumption of the independence of features is not satisfied, e.g., if we have almost same features, then $\bar{\mathbf{w}}_s$ does not converge to $\mathbf{w}^*$. However, if we introduce an ℓ_1-regularization to the minimization problem in the definition of $\mathbf{w}_s$ and change the definition of $\bar{\mathbf{w}}_s$ to a weighted average of the modified vectors $\mathbf{w}_1, \ldots, \mathbf{w}_s$, then we can generate a required set $\{S_t\}$ under the compatibility assumption. See the supplementary material for details and the proof of the following theorem.

Theorem 7. *Suppose that (b), the compatibility assumption, is satisfied and $k \leq k'$. Then, there exists a polynomial-time algorithm such that $\mathbf{E}[R_T(\mathbf{w})]$ is at most*

$$8(1+\sqrt{d})\sqrt{T+1}+12T\sum_{i\in S^*}|w_i^*|\exp(-\frac{C_{d,k',0}\sqrt{T^{\frac{1}{4}}-1}|w_i^*|^2\phi_0^2}{5832k})+4\sum_{i\in S^*}|w_i^*|(\frac{64\cdot 36^4k^2}{C_{d,k',0}^2 w_i^{*4}\phi_0^4}+1)^2,$$

for arbitrary $\mathbf{w} \in \mathbb{R}^d$ with $\|\mathbf{w}\| \leq 1$, where $C_{d,k',0} = \frac{k'(k'-1)}{d(d-1)} = O(\frac{k'^2}{d^2})$.[3,4]

[3] The asymptotic regret bound mentioned in Section 1, can be yielded by bounding the second term with the aid of the following: $\max_{T\geq 0} T\exp(-\alpha T^\beta) = (\alpha\beta)^{-\frac{1}{\beta}}\exp(-1/\beta)$ for arbitrary $\alpha > 0,\ \beta > 0$.

[4] Note that ϕ_0 is the constant appearing in Assumption (b) in Section 3.

6 Experiments

In this section, we compare our algorithms with the following four baseline algorithms: (i) a greedy method that chooses the k' largest features with respect to $\mathbf{w}_t$ computed as in Algorithm 1; (ii) a uniform-random method that chooses k' features uniformly at random; (iii) the algorithm of [7] (called AELR); and (iv) the algorithm of [6] (called FKK). Owing to space limitations, we only present typical results here. Other results and the detailed descriptions on experiment settings are provided in the supplementary material.

Synthetic data. First we show results on two kinds of synthetic datasets: instances with (d, k, k') and instances with (d, k_1, k). We set $k_1 = k$ in the setting of (d, k, k') and $k' = k$ in the setting of (d, k_1, k). The instances with (d, k, k') assume that Algorithm 1 can use the ground truth k, while Algorithm 1 cannot use k in the instances with (d, k_1, k). For each (d, k, k') and (d, k_1, k), we executed all algorithms on five instances with $T = 5000$ and computed the averages of regrets and run time, respectively. When $(d, k, k') = (20, 5, 7)$, FKK spent 1176 s on average, while AELR spent 6 s, and the others spent at most 1 s.

Figures 1 and 2 plot the regrets given by (1) over the number of rounds on a typical instance with $(d, k, k') = (20, 5, 7)$. Tables 2 and 3 summarize the average regrets at $T = 5000$, where A1, A2, A3, G, and U denote Algorithm 1, 2, 3, greedy, and uniform random, respectively. We observe that Algorithm 1 achieves smallest regrets in the setting of (d, k, k'), whereas Algorithms 2 and 3 are better than Algorithm 1 in the setting of (d, k_1, k). The results match our theoretical results.

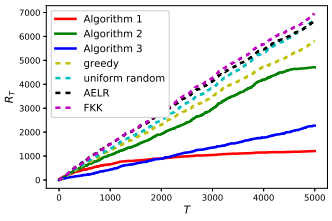

Figure 1: Plot of regrets with $(d, k, k') = (20, 5, 7)$

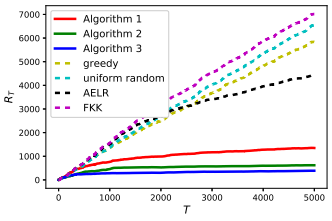

Figure 2: Plot of regrets with $(d, k_1, k) = (20, 5, 7)$

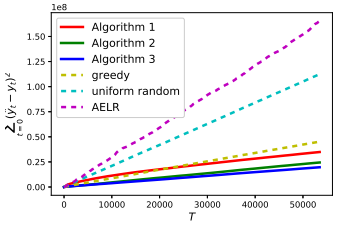

Figure 3: CT-slice datasets

Table 2: Values of $R_T/10^2$ when changing (d, k, k').

(d, k_1, k)	A1	A2	A3	G	U	AELR	FKK
(10,2,4)	**1.53**	2.38	3.60	33.28	25.73	60.76	24.05

Table 3: Values of $R_T/10^2$ when changing (d, k_1, k).

(d, k_1, k)	A1	A2	A3	G	U	AELR	FKK
(10,2,4)	26.88	20.59	**17.19**	43.03	60.02	64.75	58.71

Real data. We next conducted experiments using a CT-slice dataset, which is available online [11]. Each data consists of 384 features retrieved from 53500 CT images associated with a label that denotes the relative position of an image on the axial axis.

We executed all algorithms except FKK, which does not work due to its expensive run time. Since we do not know the ground-truth regression weights, we measure the performance by the first term of (1), i.e., square loss of predictions. Figure 3 plots the losses over the number of rounds. The parameters are $k_1 = 60$ and $k' = 70$. For this instance, the run times of Algorithms 1 and 2, greedy, uniform random, and AELR were 195, 35, 147, 382, and 477 s, respectively.

We observe that Algorithms 2 and 3 are superior to the others, which implies that Algorithm 2 and 3 are suitable for instances where the ground truth k is not known, such as real data-based instances.

Acknowledgement

This work was supported by JST ERATO Grant Number JPMJER1201, Japan.

References

[1] P. Bühlmann and S. van de Geer. *Statistics for high-dimensional data*. 2011.

[2] N. Cesa-Bianchi, S. Shalev-Shwartz, and O. Shamir. Some impossibility results for budgeted learning. In *Joint ICML-COLT workshop on Budgeted Learning*, 2010.

[3] N. Cesa-Bianchi, S. Shalev-Shwartz, and O. Shamir. Efficient learning with partially observed attributes. *Journal of Machine Learning Research*, 12:2857–2878, 2011.

[4] X. Chen, Q. Lin, and J. Pena. Optimal regularized dual averaging methods for stochastic optimization. In *Advances in Neural Information Processing Systems*, pages 395–403, 2012.

[5] I. Dinur and D. Steurer. Analytical approach to parallel repetition. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, pages 624–633. ACM, 2014.

[6] D. Foster, S. Kale, and H. Karloff. Online sparse linear regression. In *29th Annual Conference on Learning Theory*, pages 960–970, 2016.

[7] E. Hazan and T. Koren. Linear regression with limited observation. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 807–814, 2012.

[8] S. Kale. Open problem: Efficient online sparse regression. In *Proceedings of The 27th Conference on Learning Theory*, pages 1299–1301, 2014.

[9] S. Kale, Z. Karnin, T. Liang, and D. Pál. Adaptive feature selection: Computationally efficient online sparse linear regression under rip. In *Proceedings of the 34th International Conference on Machine Learning (ICML-17)*, pages 1780–1788, 2017.

[10] P. Koiran and A. Zouzias. Hidden cliques and the certification of the restricted isometry property. *IEEE Trans. Information Theory*, 60(8):4999–5006, 2014.

[11] M. Lichman. UCI machine learning repository, 2013.

[12] L. Xiao. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11:2543–2596, 2010.

[13] M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 928–936, 2003.

[14] N. Zolghadr, G. Bartók, R. Greiner, A. György, and C. Szepesvári. Online learning with costly features and labels. In *Advances in Neural Information Processing Systems*, pages 1241–1249, 2013.

Appendix

A Proof of Theorem 1

As is the case for Theorem 2 in [6], our reduction starts from the work of Dinur and Steurer [5].

Theorem 8 ([5]). *For any given constant $D > 0$, there is a constant $c_D \in (0,1)$ and a $\mathrm{poly}(n^D)$-time algorithm that takes a* **3CNF** *formula ϕ of size n as input and constructs a* **Set Cover** *instance over a ground set of size $m = \mathrm{poly}(n^D)$ with $d = \mathrm{poly}(n)$ sets, with the following properties:*

1. *if $\phi \in$ **SAT**, then there is a collection of $k = O(d^{c_D})$ sets, which covers each element exactly once; and*
2. *if $\phi \notin$ **SAT**, then no collection of $k' = \lfloor D \ln(d) k \rfloor < d$ sets covers all elements; i.e., at least one element is left uncovered.*

The **Set Cover** instance generated from ϕ can be encoded as a binary matrix $M_\phi \in \{0,1\}^{m \times d}$ with the rows corresponding to the elements of the ground set, and the columns corresponding to the sets such that each column is the indicator vector of the corresponding set. From the definition of M_ϕ and the above theorem, if $\phi \in$ **SAT**, then there exists a k-sparse binary vector $\mathbf{z} \in \{0,1\}^m$ $M_\phi \mathbf{z} = \mathbf{1}$, where $\mathbf{1}$ is the all-ones vector, and if $\phi \notin$ **SAT**, then for arbitrary $S \subseteq [d]$ such that $|S| \leq k'$, there exists at least one row of M_ϕ that is 0 in all the coordinates in S.

Using this reduction, we show that an algorithm **Alg** for online sparse algorithm with properties of Theorem 1 can be used to give a **BPP** algorithm for **SAT**. Algorithm 4 is a randomized algorithm for deciding satisfiability of a given **3CNF** formula ϕ using the algorithm **Alg**. Since Step 3 runs in polynomial time and since T is a polynomial in n and **Alg** runs in $\mathrm{poly}(d, T)$ time per iteration, Algorithm 4 is a polynomial-time algorithm.

We now claim that this algorithm correctly decides satisfiability of ϕ with probability at least $3/4$, and is hence a **BPP** algorithm for **SAT**.

Suppose that $\phi \in$ **SAT**. Then there exists a k-sparse vector $\mathbf{z} \in \{0,1\}^d$ such that $M_\phi \mathbf{z} = \mathbf{1}_m$. Hence, for X and $\mathbf{y}$ defined in Algorithm 4, we have

$$\begin{aligned}\gamma &= \min_{\mathbf{w} \in \mathbb{R}^d} \|X\mathbf{w} - \tilde{\mathbf{y}}\|_2^2 \leq \min_{\mathbf{w} \in \mathbb{R}^d, \|\mathbf{w}\|_0 \leq k'} \|X\mathbf{w} - \tilde{\mathbf{y}}\|_2^2 \\ &\leq \|\frac{1}{\sqrt{d}} X\mathbf{z} - \tilde{\mathbf{y}}\|_2^2 = \|\frac{1}{4\sqrt{d}(m+d)^3}\mathbf{z}\|_2^2 \leq \frac{1}{16(m+d)^6}, \end{aligned} \tag{13}$$

which means that Step 5 is not executed. Since the ℓ_2 norm of each row of X and each entry of $\mathbf{y}$ are at most 1, it holds that $\|\mathbf{x}_t\| \leq 1$ and $|y_t| \leq 1$. Next, let us see that $\mathbf{x}_t, y_t$ satisfies Assumptions (1), (2), and (a). Since $\mathbf{y} = X\tilde{\mathbf{w}}$, it holds for all t that $y_t = \tilde{\mathbf{w}}^\top x_t$, which means that Assumption (1) holds, where $\epsilon_t = 0$. From the definition of $\mathbf{x}_t, \mathbf{x}_1, \ldots, \mathbf{x}_T$ follows a distribution on $\mathbb{R}^d$ independently, and independent of ϵ_t because ϵ_t are constant, and, hence, Assumption (2) holds. Moreover, $V = \mathbf{E}[\mathbf{x}_t \mathbf{x}_t^\top] = \frac{1}{m+d} X^\top X$ is non-singular (i.e., Assumption (a) holds) and has the smallest singular value at least $\frac{1}{(m+d)}$. Indeed, for an arbitrary d-dimensional unit vector $\mathbf{u} \in \mathbb{R}^d$, the ℓ_2 norm of $X\mathbf{u} = [\frac{1}{\sqrt{d}} M_\phi \mathbf{u}; \frac{1}{4(m+d)^3}\mathbf{u}]$ is at least $\frac{1}{4(m+d)^3}$ and, hence, $\mathbf{u}^\top X^\top X \mathbf{u} \geq \frac{1}{(4(m+d)^3)^2}$, which means that σ_d, the smallest singular value of $V = \frac{1}{m+d} X^\top X$, is at least $\frac{1}{16(m+d)^7}$. Let us now show that $\mathbf{E}[\sum_{t=0}^T (y_t - \hat{y}_t)^2] \leq \frac{T}{8(m+d)^5}$. From the assumption on **Alg**, it holds for all k-sparse vectors $\mathbf{w}$ that

$$\mathbf{E}[R_T(\mathbf{w})] \leq p_2(d, 16(m+d)^7) T^{1-\delta} \leq \frac{T}{16(m+d)^5}, \tag{14}$$

Algorithm 4 An algorithm for deciding satisfiability of 3CNF formula

Input: A constant $D > 0$, and an algorithm **Alg** for the (k, k', d)-online sparse regression problem, where k, k', d and c_D are the constants from Theorem 8, that runs in $p_1(d, T)$ time per iteration, with expected regret bounded by $p_2(d, 1/\sigma_d)T^{1-\delta}$ under Assumptions (1), (2), and (4) in Section 2 and Assumption (a) in Section 3. A **3CNF** formula ϕ.

Compute the matrix M_ϕ and the associated parameters k, k', d, m from Theorem 8.

Define $X \in \mathbb{R}^{(m+d)\times d}$ and $\tilde{\mathbf{y}} \in \mathbb{R}^{m+d}$ by

$$X = \begin{bmatrix} \frac{1}{\sqrt{d}} M_\phi \\ \frac{1}{4(m+d)^3} I_d \end{bmatrix}, \quad \tilde{\mathbf{y}} = \begin{bmatrix} \frac{1}{d}\mathbf{1}_m \\ \mathbf{0}_d \end{bmatrix}. \tag{12}$$

Compute $\tilde{\mathbf{w}} \in \arg\min_{\mathbf{w}\in\mathbb{R}^m} \|X\mathbf{w} - \tilde{\mathbf{y}}\|_2^2$ and define $\mathbf{y} = X\tilde{\mathbf{w}}$, $\gamma = \|X\tilde{\mathbf{w}} - \tilde{\mathbf{y}}\|_2^2$.

if $\gamma > \frac{1}{16d^2}$ **then**
  Return "unsatisfiable."
end if

Run **Alg** with the parameters k, k', d for $T := \lceil \max\{(16(m+d)^5 p_1(d, 16(m+d)^7))^{1/\delta}, 256(m+d)^{10}\}\rceil$ iterations.

for $t = 1, \ldots, T$ **do**
  Sample i from $[m+d]$ uniformly at random and set $\mathbf{x}_t$ and y_t to be the i-th row of X and the i-th entry of $\mathbf{y}$, respectively.
  Obtain a set of coordinates S_t of size at most k' by running **Alg**, and provide it with the coordinates $\mathbf{x}_t(S_t)$.
  Obtain the prediction $\hat{y}_t$ from **Alg**, and provide it with the true label y_t.
end for

if $\sum_{t=1}^T (y_t - \hat{y}_t)^2 \le \frac{T}{2(m+d)^5}$ **then**
  Return "satisfiable."
else
  Return "unsatisfiable."
end if

where the second inequality comes from $T \ge (16(m+d)^5 p_1(d, 16(m+d)^7))^{\frac{1}{\delta}}$. Since $\mathbf{z}$ is a k-sparse binary vector, we have

$$\begin{aligned}\mathbf{E}[R_T(\frac{1}{\sqrt{d}}\mathbf{z})] &= \mathbf{E}[\sum_{t=1}^T (y_t - \hat{y}_t)^2] - \mathbf{E}[\sum_{t=1}^T (y_t - \frac{1}{\sqrt{d}}\mathbf{z}^\top \mathbf{x}_t)^2] \\ &= \mathbf{E}[\sum_{t=1}^T (y_t - \hat{y}_t)^2] - \frac{T}{m+d}\|\mathbf{y} - \frac{1}{\sqrt{d}}X\mathbf{z}\|_2^2.\end{aligned}$$

Since we have $\|\mathbf{y} - \frac{1}{\sqrt{d}}X\mathbf{z}\|_2^2 \le \|\tilde{\mathbf{y}} - \frac{1}{\sqrt{d}}X\mathbf{z}\|_2^2 \le \frac{1}{16(m+d)^6}$, we obtain

$$\mathbf{E}[\sum_{t=1}^T (y_t - \tilde{y}_t)^2] \le \frac{T}{16(m+d)^5} + \frac{T}{16(m+d)^7} \le \frac{T}{8(m+d)^5}. \tag{15}$$

Since $\sum_{t=1}^T (y_t - \tilde{y}_t)^2$ is a non-negative random variable, by Markov's inequality we conclude that with probability at least $3/4$, the total loss $\sum_{t=1}^T (y_t - \tilde{y}_t)^2$ is bounded by $\frac{T}{2(m+d)^5}$ from above, and hence Algorithm 4 correctly outputs "satisfiable."

Next, suppose $\phi \notin$ **SAT**. If $\gamma > \frac{1}{16d^2}$, then Algorithm 4 correctly outputs "unsatisfiable." Hence, it suffices to consider the case of $\gamma \le \frac{1}{16d^2}$. Fix any round t and let S_t be the set of coordinates of size at most k' selected by **Alg** to query. Since $\phi \notin$ **SAT**, there is at least one element in the ground set that is not covered by any set among these k' sets. This implies that there is at least one row of M_ϕ that is 0 in all the coordinates in S_t. Let d_1 denote the number of such rows. Further, the number d_2 of rows of I_d that are 0 in all the coordinates in S_t is equal to $d - k'$. Since $\mathbf{x}_t$ is a uniformly random

row of X chosen independently of S_t, we have

$$\mathrm{Prob}[\mathbf{x}_t(S_t) = \mathbf{0}] = \frac{d_1 + d_2}{m + d} \geq \frac{2}{m + d}. \tag{16}$$

The conditional probability that given $\mathbf{x}_t(S_t) = \mathbf{0}$, i in Step 9 is at most m is equal to $\frac{d_1}{d_1+d_2}$.

Now, we claim that $\mathbf{E}[(y_t - \hat{y}_t)^2 \mid \mathbf{x}_t(S_t) = \mathbf{0}] \geq \frac{1}{2(m+d)^4}$. Since y_t and $\hat{y}_t$ are conditionally independent given $\mathbf{x}_t(S_t)$, we have $\mathbf{E}[(y_t - \hat{y}_t)^2 \mid \mathbf{x}_t(S_t) = \mathbf{0}] \geq \mathrm{var}[y_t \mid \mathbf{x}_t(S_t) = 0]$. Let us recall that $\mathrm{Prob}[i \leq m \mid \mathbf{x}_t(S_t) = \mathbf{0}] = \frac{d_1}{d_1+d_2}$ for i in Step 9. If $i \leq m$, then $y_t \geq \frac{3}{4d}$ and otherwise $y_t \leq \frac{1}{4d}$ since the difference between $\mathbf{y}$ and $\tilde{\mathbf{y}}$ is bounded in absolute value by $\|\mathbf{y} - \tilde{\mathbf{y}}\|_2 \leq \sqrt{\gamma} \leq 1/3$ and the ith element of $\tilde{\mathbf{y}}$ is 1 if $i \leq m$ and 0 otherwise. Hence, given $\mathbf{x}_t(S_t) = 0$, $y_t \geq \frac{3}{4d}$ with probability $\frac{d_1}{d_1+d_2}$ and $y_t \leq \frac{1}{4d}$ with probability $\frac{d_2}{d_1+d_2}$. Since $d_1, d_2 \geq 1$ and $d_1 + d_2 \leq m + d$, we have $\mathbf{E}[(y_t - \hat{y}_t)^2 \mid \mathbf{x}_t(S_t) = \mathbf{0}] \geq \frac{2}{(m+d)^2} \cdot \frac{1}{(2d)^2} \geq \frac{1}{2(m+d)^4}$. Further, from (16), we obtain

$$\mathbf{E}[(y_t - \hat{y}_t)^2] \geq \mathbf{E}[(y_t - \hat{y}_t)^2 \mid \mathbf{x}_t(S_t) = \mathbf{0}] \cdot \mathrm{Prob}[\mathbf{x}_t(S_t) = \mathbf{0}] \geq \frac{1}{2(m+d)^4} \cdot \frac{2}{m+d} = \frac{1}{(m+d)^5}. \tag{17}$$

Let $\mathbf{E_t}$ denote the expectation of a random variable conditioned on all randomness prior to round t. Since the choices of $\mathbf{x}_t$ and y_t are independent of previous choices in each round, the same argument also implies that $\mathbf{E}_t[(y_t - \hat{y}_t)^2] \geq \frac{1}{(m+d)^5}$. Applying Azuma's inequality to the martingale difference sequence $\mathbf{E}_t[(y_t - \hat{y}_t)^2] - (y_t - \hat{y}_t)^2$ for $t = 1, \ldots, T$, since each term is bounded in absolute value by 4, we obtain

$$\mathrm{Prob}[\sum_{t=1}^{T} \mathbf{E}_t[(y_t - \hat{y}_t)^2] - (y_t - \hat{y}_t)^2 \geq 8\sqrt{T}] \leq \exp(-\frac{64T}{2 \cdot 16T}) \leq \frac{1}{4}. \tag{18}$$

Thus, with probability at least $3/4$, the total loss $\sum_{t=1}^{T}(y_t - \hat{y}_t)^2$ is greater than $\sum_{t=1}^{T} \mathbf{E}_t[(y_t - \hat{y}_t)^2] - 8\sqrt{T} \geq \frac{T}{(m+d)^5} - 8\sqrt{T} \geq \frac{T}{2(m+d)^5}$ since $T \geq 256(m+d)^{10}$. Thus, in the case of $\phi \notin \mathbf{SAT}$, Algorithm 4 correctly outputs "unsatisfiable" with probability at least $3/4$.

B Preliminary lemmas

Before presenting the statement, let us introduce some notation. In the following, we use the following facts without notice:

- $\|\mathbf{g}_t\| = 2\|\mathbf{x}_t(\mathbf{x}_t^\top \mathbf{w}_t - y_t)\|$ is bounded from above by 4; and
- $|\ell_t(\mathbf{w}) - \ell(\mathbf{w})|$ is bounded from above by 8;

which come from $\mathbf{w} \leq 1$, $\|\mathbf{x}_t\| \leq 1$ and $|y_t| \leq 1$. Further, we define the following numbers for notational convenience:

- $G = 4/C_{d,k',k_1}$, an upper bound of $\|\hat{\mathbf{g}}_t - \mathbf{g}_t\|$;
- $C = 128/C_{d,k',k_1}^2 = 8G^2$.

The following lemma is used for bounding $R'_T(\mathbf{w})$.

Lemma 9. *Let $(\lambda_1, \ldots, \lambda_T)$ be a monotonically non-decreasing sequence of positive numbers, and let $(\zeta_1, \ldots, \zeta_T)$ be a sequence of non-negative numbers. If $\mathbf{w}_t$ is defined by*

$$\mathbf{w}_t = \operatorname*{arg\,min}_{\mathbf{w} \in \mathbb{R}^d, \|\mathbf{w}\| \leq 1} \{\hat{\mathbf{h}}_t^\top \mathbf{w} + \frac{\lambda_t}{2}\|\mathbf{w}\|^2 + \sum_{j=1}^{t} \zeta_j \|\mathbf{w}\|_1\}$$

for all $t = 1, \ldots, T$, then, for any $\mathbf{w} \in \mathbb{R}^d$ satisfying $\|\mathbf{w}\| \leq 1$, it holds that

$$\sum_{t=1}^{T} \left(\hat{\mathbf{g}}_t^\top (\mathbf{w}_t - \mathbf{w}) + \zeta_t(\|\mathbf{w}_t\|_1 - \|\mathbf{w}\|_1)\right) \leq \sum_{t=1}^{T} \frac{1}{\lambda_t}\|\hat{\mathbf{g}}_t\|^2 + \frac{\lambda_{T+1}}{2}.$$

Proof. See, e.g., [12]. □

The following lemma is used for bounding the gap between $\sum_{t=1}^{T} \|L(\mathbf{w}_t - \mathbf{w}^*)\|^2$ and $\sum_{t=1}^{T} \hat{\mathbf{g}}_t^\top (\mathbf{w}_t - \mathbf{w})$, with high probability.

Lemma 10. *For arbitrary $\delta > 0$, we have*

$$\mathrm{Prob}[\sum_{t=1}^{T} \|L(\mathbf{w}_t - \mathbf{w}^*)\|^2 \geq \sum_{t=1}^{T} \hat{\mathbf{g}}_t^\top (\mathbf{w}_t - \mathbf{w}) + 3\delta] \leq 3 \exp\left(-\frac{\delta^2}{Ct}\right). \tag{19}$$

Proof. We have $\sum_{j=1}^{t} \|L(\mathbf{w}_j - \mathbf{w}^*)\|^2 = \sum_{j=1}^{t} (\ell(\mathbf{w}_j) - \ell(\mathbf{w}^*))$, and this can be expanded as

$$\sum_{j=1}^{t} (\ell(\mathbf{w}_j) - \ell(\mathbf{w}^*)) = \sum_{j=1}^{t} (\ell(\mathbf{w}_j) - \ell_j(\mathbf{w}_j)) + \sum_{j=1}^{t} (\ell_j(\mathbf{w}^*) - \ell(\mathbf{w}^*)) + \sum_{j=1}^{t} (\ell_j(\mathbf{w}_j) - \ell_j(\mathbf{w}^*)).$$

From the convexity of ℓ_j, the term $\sum_{j=1}^{t} (\ell_j(\mathbf{w}_j) - \ell_j(\mathbf{w}^*))$ can be bounded as

$$\sum_{j=1}^{t} (\ell_j(\mathbf{w}_j) - \ell_j(\mathbf{w}^*)) \leq \sum_{j=1}^{t} \mathbf{g}_j^\top (\mathbf{w}_j - \mathbf{w}^*) = \sum_{j=1}^{t} \hat{\mathbf{g}}_j^\top (\mathbf{w}_j - \mathbf{w}^*) + \sum_{j=1}^{t} (\mathbf{g}_j - \hat{\mathbf{g}}_j)^\top (\mathbf{w}_j - \mathbf{w}^*).$$

Summarizing the above inequalities, $\sum_{j=1}^{t} \|L(\mathbf{w}_j - \mathbf{w}^*)\|^2 - \sum_{t=1}^{T} \hat{\mathbf{g}}_t^\top (\mathbf{w}_t - \mathbf{w})$ is bounded from above by the sum of (i) $\sum_{j=1}^{t} (\mathbf{g}_j - \hat{\mathbf{g}}_j)^\top (\mathbf{w}_j - \mathbf{w}^*)$, (ii) $\sum_{j=1}^{t} (\ell(\mathbf{w}_j) - \ell_j(\mathbf{w}_j))$, and (iii) $\sum_{j=1}^{t} (\ell_j(\mathbf{w}^*) - \ell(\mathbf{w}^*))$. Let us construct bounds for each term.

First, consider (i). Denote the j-th input data $(\mathbf{x}_j, y_j)$ by z_j. Since we have

$$\mathbf{E}[(\mathbf{g}_j - \hat{\mathbf{g}}_j)(\mathbf{w}_j - \mathbf{w}^*) | z_1, \ldots, z_{j-1}, S_1, \ldots, S_{j-1}] = 0$$

and $|(\mathbf{g}_j - \hat{\mathbf{g}}_j)^\top (\mathbf{w}_j - \mathbf{w}^*)| \leq 2G$, we can apply the Azuma–Hoeffding inequality to (i) to obtain that

$$\mathrm{Prob}[\sum_{j=1}^{t} (\mathbf{g}_j - \hat{\mathbf{g}}_j)^\top (\mathbf{w}_j - \mathbf{w}^*) \geq \delta] \leq \exp(\frac{-\delta^2}{8tG^2}).$$

The value of (ii) also can be bounded by using the Azuma–Hoeffding inequality, since we have

$$\mathbf{E}[\ell(\mathbf{w}_j) - \ell_j(\mathbf{w}_j) | z_1, \ldots, z_{j-1}, S_1, \ldots, S_{j-1}] = 0,$$

and $|\ell(\mathbf{w}_j) - \ell_j(\mathbf{w}_j)| \leq 8$. Accordingly, we obtain

$$\mathrm{Prob}[\sum_{j=1}^{t} \ell(\mathbf{w}_j) - \ell_j(\mathbf{w}_j) \geq \delta] \leq \exp(\frac{-\delta^2}{128t}).$$

Similarly, since $\{\ell_j(\mathbf{w}^*) - \ell(\mathbf{w}^*)\}$ are independent random variables such that $\mathbf{E}[\ell_j(\mathbf{w}^*) - \ell(\mathbf{w}^*)] = 0$ and $|\ell_j(\mathbf{w}^*) - \ell(\mathbf{w}^*)| \leq 8$, the value of (iii) can be bounded by using Hoeffding's inequality, as follows:

$$\mathrm{Prob}[\sum_{j=1}^{t} \ell_j(\mathbf{w}^*) - \ell(\mathbf{w}^*) \geq \delta] \leq \exp(\frac{-\delta^2}{128t}).$$

Summarizing the above inequalities, we have

$$\mathrm{Prob}\left[\sum_{j=1}^{t} \|L(\mathbf{w}_j - \mathbf{w}^*)\|^2 \geq \sum_{t=1}^{T} \hat{\mathbf{g}}_t^\top (\mathbf{w}_t - \mathbf{w}) + 3\delta\right] \leq 2\exp(\frac{-\delta^2}{128t}) + \exp(\frac{-\delta^2}{8tG^2}) \leq 3\exp(\frac{-\delta^2}{Ct}).$$

By substituting this for $\lambda_j = 2G\sqrt{j}$, we obtain (19). □

C Proofs of the lemmas in Section 4

C.1 Proof of Lemma 4

Since $\ell_t(\mathbf{w})$ is a convex function, it holds for arbitrary $\mathbf{w} \in \mathbb{R}^d$ that $\ell_t(\mathbf{w}_t) - \ell_t(\mathbf{w}) \leq \mathbf{g}_t^\top(\mathbf{w}_t - \mathbf{w})$. By taking the expectation, we have $\mathbf{E}[\ell_t(\mathbf{w}_t) - \ell_t(\mathbf{w})] \leq \mathbf{E}[\mathbf{g}_t^\top(\mathbf{w}_t - \mathbf{w})] = \mathbf{E}[\hat{\mathbf{g}}_t^\top(\mathbf{w}_t - \mathbf{w})]$ since $\hat{\mathbf{g}}_t$ is an unbiased estimator of $\mathbf{g}_t$. By taking the sum of this inequality for $t = 1, \ldots, T$, we obtain that

$$\mathbf{E}[R'_T(\mathbf{w}^*)] \leq \mathbf{E}[\sum_{t=1}^{T} \hat{\mathbf{g}}_t^\top(\mathbf{w}_t - \mathbf{w}^*)].$$

From Lemma 9, the right-hand side of this can be bounded as

$$\mathbf{E}[\sum_{t=1}^{T} \hat{\mathbf{g}}_t^\top(\mathbf{w}_t - \mathbf{w}^*)] \leq \mathbf{E}[\sum_{t=1}^{T} \frac{1}{\lambda_t}\|\hat{\mathbf{g}}_t\|^2 + \frac{\lambda_{T+1}}{2}] \leq \sum_{t=1}^{T} \frac{1}{\lambda_t} G_t + \frac{\lambda_{T+1}}{2},$$

where we used $\mathbf{E}[\|\mathbf{g}_t\|^2] = G_t$ in the second inequality.

C.2 Proof of Lemma 5

Proof. First, $\hat{\mathbf{g}}_t$ defined by (6) satisfies

$$\|\hat{\mathbf{g}}_t\|^2 = \|2X\mathbf{w}_t - 2y_t\mathbf{z}\|^2 \leq 2\|2X\mathbf{w}_t\|^2 + 2\|2y_t\mathbf{z}\|^2 \leq 8\|X\|_{\mathrm{F}}^2\|\mathbf{w}_t\|^2 + 8|y_t|\|\mathbf{z}\|^2.$$

The expectation of $\|X\|_{\mathrm{F}}^2$ is bounded as

$$\mathbf{E}[\|X\|_{\mathrm{F}}^2] = \mathbf{E}\left[\sum_{1 \leq i,j \leq d} p_{ij}^{(t)} (\frac{x_{ti}x_{tj}}{p_{ij}^{(t)}})^2\right] \leq \mathbf{E}\left[\frac{1}{q}\sum_{1 \leq i,j \leq d} (x_i x_j)^2\right] \leq \frac{1}{q}.$$

Similarly, we have $\mathbf{E}[\|\mathbf{z}\|^2] \leq 1/q$. By combining these inequalities, we obtain Lemma 5. □

D Details of Algorithm 2

Estimating $\mathbf{w}^*$ Although we assumed $k' \geq k+2$ in the analysis of the regret bound, Algorithm 1 can be defined even if $k' = k$ by setting k_1 to a number at most k. At the end of round t, Algorithm 1 keeps weight vectors $\mathbf{w}_1, \ldots, \mathbf{w}_t$. From these weight vectors, define $\bar{\mathbf{w}}_t$ as $\frac{1}{t}\sum_{j=1}^{t} \mathbf{w}_j$. In the following, we prove that $\bar{\mathbf{w}}_t$ is a consistent estimator of $\mathbf{w}^*$ even if $k' = k$ and $k_1 \leq k-2$. We use this fact in the next section.

Proposition 11. *Let $\bar{\mathbf{w}}_t$ be the average of $\mathbf{w}_1, \ldots, \mathbf{w}_t$ computed by Algorithm 1 with setting $\lambda_t = \frac{8}{C}\sqrt{t}$ for $t = 1, \ldots, T$. Then, for arbitrary $\delta > 0$, $\|L(\bar{\mathbf{w}}_t - \mathbf{w}^*)\|^2 \leq \frac{1}{t}(\frac{8}{C}\sqrt{t+1} + 3\delta)$ holds with probability at least $1 - 3\exp(\frac{-C^2\delta^2}{128t})$. Accordingly, assuming the linear independence of features, $\|\bar{\mathbf{w}}_t - \mathbf{w}^*\|^2 \leq \frac{1}{t\sigma_d^2}(\frac{8}{C}\sqrt{t+1} + 3\delta)$ holds with probability at least $1 - 3\exp(\frac{-C^2\delta^2}{128t})$.*

Proof. From the convexity of the square loss and Jensen's inequality, we have $\|L(\bar{\mathbf{w}}_t - \mathbf{w}^*)\| \leq \frac{1}{t}\sum_{j=1}^{t}\|L(\mathbf{w}_j - \mathbf{w}^*)\|$. Hence, it suffices to show that

$$\mathrm{Prob}\left[\sum_{j=1}^{t}\|L(\mathbf{w}_j - \mathbf{w}^*)\| \geq 2G\sqrt{t+1} + 3\delta\right] \leq 3\exp(\frac{-\delta^2}{Ct}). \tag{20}$$

We have $\sum_{j=1}^{t}\|L(\mathbf{w}_j - \mathbf{w}^*)\|^2 = \sum_{j=1}^{t}(\ell(\mathbf{w}_j) - \ell(\mathbf{w}^*))$ and this can be expanded as

$$\sum_{j=1}^{t}(\ell(\mathbf{w}_j) - \ell(\mathbf{w}^*)) = \sum_{j=1}^{t}(\ell(\mathbf{w}_j) - \ell_j(\mathbf{w}_j)) + \sum_{j=1}^{t}(\ell_j(\mathbf{w}^*) - \ell(\mathbf{w}^*)) + \sum_{j=1}^{t}(\ell_j(\mathbf{w}_j) - \ell_j(\mathbf{w}^*)).$$

From the convexity of ℓ_j, the term $\sum_{j=1}^t(\ell_j(\mathbf{w}_j) - \ell_j(\mathbf{w}^*))$ can be bounded as

$$\sum_{j=1}^t(\ell_j(\mathbf{w}_j) - \ell_j(\mathbf{w}^*)) \le \sum_{j=1}^t \mathbf{g}_j^\top(\mathbf{w}_j - \mathbf{w}^*) = \sum_{j=1}^t \hat{\mathbf{g}}_j^\top(\mathbf{w}_j - \mathbf{w}^*) + \sum_{j=1}^t(\mathbf{g}_j - \hat{\mathbf{g}}_j)^\top(\mathbf{w}_j - \mathbf{w}^*).$$

Summarizing the above inequalities, $\sum_{j=1}^t \|L(\mathbf{w}_j - \mathbf{w}^*)\|^2$ is bounded by the sum of (i) $\sum_{j=1}^t \hat{\mathbf{g}}_j^\top(\mathbf{w}_j - \mathbf{w}^*)$, (ii) $\sum_{j=1}^t(\mathbf{g}_j - \hat{\mathbf{g}}_j)^\top(\mathbf{w}_j - \mathbf{w}^*)$, (iii) $\sum_{j=1}^t(\ell(\mathbf{w}_j) - \ell_j(\mathbf{w}_j))$, and (iv) $\sum_{j=1}^t(\ell_j(\mathbf{w}^*) - \ell(\mathbf{w}^*))$. Next, let us construct bounds for each term.

From Lemma 9, (i) $\sum_{j=1}^t \hat{\mathbf{g}}_j^\top(\mathbf{w}_j - \mathbf{w}^*)$ can be bounded as

$$\sum_{j=1}^t \hat{\mathbf{g}}_j^\top(\mathbf{w}_j - \mathbf{w}^*) \le \sum_{j=1}^t \frac{G^2}{\lambda_j} + \frac{\lambda_{t+1}}{2}$$

with probability 1. Next, consider (ii). Denote the j-th input data $(\mathbf{x}_j, y_j)$ by z_j. Since we have

$$\mathbf{E}[(\mathbf{g}_j - \hat{\mathbf{g}}_j)(\mathbf{w}_j - \mathbf{w}^*)|z_1, \dots, z_{j-1}, S_1, \dots, S_{j-1}] = 0$$

and $|(\mathbf{g}_j - \hat{\mathbf{g}}_j)(\mathbf{w}_j - \mathbf{w}^*)| \le 2G$, we can apply the Azuma–Hoeffding inequality to (ii) to obtain that

$$\mathrm{Prob}[\sum_{j=1}^t(\mathbf{g}_j - \hat{\mathbf{g}}_j)^\top(\mathbf{w}_j - \mathbf{w}^*) \ge \delta] \le \exp(\frac{-\delta^2}{8tG^2}).$$

The value of (iii) can also be bounded by using the Azuma–Hoeffding inequality, since we have

$$\mathbf{E}[\ell(\mathbf{w}_j) - \ell_j(\mathbf{w}_j)|z_1, \dots, z_{j-1}, S_1, \dots, S_{j-1}] = 0,$$

and $|\ell(\mathbf{w}_j) - \ell_j(\mathbf{w}_j)| \le Q$. Accordingly, we obtain

$$\mathrm{Prob}[\sum_{j=1}^t \ell(\mathbf{w}_j) - \ell_j(\mathbf{w}_j) \ge \delta] \le \exp(\frac{-\delta^2}{2tR^Q}).$$

Similarly, since $\{\ell_j(\mathbf{w}^*) - \ell(\mathbf{w}^*)\}$ are independent random variables such that $\mathbf{E}[\ell_j(\mathbf{w}^*) - \ell(\mathbf{w}^*)] = 0$ and $|\ell_j(\mathbf{w}^*) - \ell(\mathbf{w}^*)| \le Q$ the value of (iv) can be bounded by using Hoeffding's inequality, as follows:

$$\mathrm{Prob}[\sum_{j=1}^t \ell_j(\mathbf{w}^*) - \ell(\mathbf{w}^*) \ge \delta] \le \exp(\frac{-\delta^2}{2tQ^2}).$$

Summarizing the above inequalities, we have

$$\mathrm{Prob}\left[\sum_{j=1}^t \|L(\mathbf{w}_j - \mathbf{w}^*)\|^2 \ge \sum_{j=1}^t \frac{G^2}{\lambda_j} + \frac{\lambda_{t+1}}{2} + 3\delta\right] \le 2\exp(\frac{-\delta^2}{2tQ^2}) + \exp(\frac{-\delta^2}{8tG^2}) \le 3\exp(\frac{-\delta^2}{Ct}).$$

By substituting this for $\lambda_j = 2G\sqrt{j}$, we obtain (20). □

We note that the probability claimed in Proposition 11 is over the randomness of both the examples and Algorithm 1.

Computing $\mathbf{w}_s$, $\bar{\mathbf{w}}_s$, and $\hat{\mathbf{g}}_s$. As noted above, $\bar{\mathbf{w}}_s$ is defined as the average of $\mathbf{w}_1, \dots, \mathbf{w}_s$ computed as in Algorithm 1 applied to the examples $\{(\mathbf{x}_{t_1}, y_{t_1}), \dots, (\mathbf{x}_{t_s}, y_{t_s})\}$, setting $k_1 \le k - 2$. Recall that Algorithm 1 computes $\mathbf{w}_s$ from $\hat{\mathbf{g}}_1, \dots, \hat{\mathbf{g}}_{s-1}$ using (5), and computes $\hat{\mathbf{g}}_j$ from $\mathbf{w}_{t_j}$ and $\{(D_{t_1}\mathbf{x}_{t_1}, y_{t_1}), \dots, (D_{t_j}\mathbf{x}_{t_j}, y_{t_j})\}$ using (6) for any $j \in [s]$. We use D_{t_j} defined from S_{t_j} in Algorithm 2 instead of Algorithm 1.

For convenience, we define $\bar{\mathbf{w}}_0$ as the zero vector.

Computing S_t. Let s be the largest number such that $t_s \leq t$. Then S_t is defined as the set of k largest features with respect to $\bar{\mathbf{w}}_s$. Note that S_t is the same for all t with $t_s \leq t < t_{s+1}$. In the following, we show from Proposition 11 that S_t contains S^* with high probability.

Lemma 12. *If $w_i^{*2}\sigma_d^2 - 8Gs^{-\frac{1}{2}} \geq 0$, the following holds for any $i \in S^*$ and $t = t_s, \ldots, t_{s+1} - 1$:*

$$\mathrm{Prob}[i \notin S_t] \leq 3\exp\left(-\frac{C^2 s}{4608}(w_i^{*2}\sigma_d^2 - \frac{32}{C}s^{-\frac{1}{2}})^2\right).$$

Proof. If a feature i satisfies $i \in S^* \setminus S_t$, it holds that $\|\bar{\mathbf{w}}_s - \mathbf{w}^*\|^2 \geq w_i^{*2}/2$, which can be confirmed as follows. Since $|S^*| = |S_t|$, $i \in S^* \setminus S_t$ means that there exist $j \in S_t \setminus S^*$ such that $|\bar{w}_{si}| < |\bar{w}_{sj}|$, which implies that $\|\bar{\mathbf{w}}_s - \mathbf{w}^*\|^2 \geq (\bar{w}_{si} - w_i^*)^2 + (\bar{w}_{sj} - w_j^*)^2 = (\bar{w}_{si} - w_i^*)^2 + \bar{w}_{sj}^2 > (\bar{w}_{si} - w_i^*)^2 + \bar{w}_{si}^2 \geq w_i^{*2}/2$, where the first equality comes from $j \notin S^*$, the second inequality comes from $|\bar{w}_{si}| < |\bar{w}_{sj}|$, and the third inequality holds for arbitrary $\bar{w}_{si} \in \mathbb{R}$. Hence, we have

$$\mathrm{Prob}[i \notin S_t] \leq \mathrm{Prob}[\|\bar{\mathbf{w}}_s - \mathbf{w}^*\|^2 \geq w_i^{*2}/2]. \tag{21}$$

From Proposition 11, if $\delta := \frac{1}{6}(s\sigma_d^2 w_i^{*2} - 4G\sqrt{s+1}) \geq 0$, we have

$$\mathrm{Prob}[i \notin S_t] \leq 3\exp(\frac{-\delta^2}{Cs})$$

for $i \in S^2$. From the inequality $\delta \geq \frac{s}{6}(\sigma_d^2 w_i^{*2} - 8Gs^{-\frac{1}{2}})$, we obtain Lemma 12.

□

Computing $\tilde{\mathbf{w}}_t$ and $\tilde{\mathbf{g}}_t$. We define $\tilde{\mathbf{w}}_1 = 0$. If $t \geq 2$, $\tilde{\mathbf{w}}_t$ is defined as follows. Recall that $D_1\tilde{\mathbf{g}}_1, \ldots, D_{t-1}\tilde{\mathbf{g}}_{t-1}$ are available at the beginning of round t. Let $\tilde{\mathbf{h}}_{t-1} = \sum_{j=1}^{t-1} D_j\tilde{\mathbf{g}}_j$. We prepare a sequence $(\tilde{\lambda}_1, \ldots, \tilde{\lambda}_T)$ of non-negative numbers in advance, and $\tilde{\lambda}_t$ is used in round t. Then, $\tilde{\mathbf{w}}_t$ is defined by

$$\tilde{\mathbf{w}}_t = \operatorname*{arg\,min}_{\mathbf{w}\in\mathbb{R}^d, \|\mathbf{w}\|\leq 1} \left\{\tilde{\mathbf{h}}_{t-1}^\top \mathbf{w} + \frac{\tilde{\lambda}_t}{2}\|\mathbf{w}\|^2\right\}. \tag{22}$$

We define $\tilde{\mathbf{g}}_t$ as the gradient of the loss function $\ell_t(\mathbf{w})$ at $\mathbf{w} = D_t\tilde{\mathbf{w}}_t$, i.e.,

$$\tilde{\mathbf{g}}_t = \nabla_{\mathbf{w}}\ell_t(D_t\tilde{\mathbf{w}}_t) = 2\mathbf{x}_t(\mathbf{x}_t^\top D_t\tilde{\mathbf{w}}_t - y_t). \tag{23}$$

Note that we cannot compute $\tilde{\mathbf{g}}_t$ because all features in $\mathbf{x}_t$ cannot be observed. Nevertheless, we can compute $D_t\tilde{\mathbf{g}}_t$ from available information $D_t\mathbf{x}_t$, y_t, and $\tilde{\mathbf{w}}_t$.

Regret bound of Algorithm 2 We prove that Algorithm 2 achieves $O(\sqrt{dT})$ regret under the independence of features assumption.

Lemma 13. *If $\mathbf{w} \in \mathbb{R}^d$ satisfies $\|\mathbf{w}\| \leq 1$, then we have*

$$R_T(\mathbf{w}) \leq \sum_{t=1}^T \mathbf{w}^\top(D_t - I)\tilde{\mathbf{g}}_t + \sum_{t=1}^T \frac{\|D_t\tilde{\mathbf{g}}_t\|^2}{\tilde{\lambda}_t} + \frac{\tilde{\lambda}_{T+1}}{2}. \tag{24}$$

Proof. Since $\ell_t(\mathbf{w})$ is convex, we have $\ell_t(D_t\tilde{\mathbf{w}}_t) - \ell_t(\mathbf{w}) \leq \tilde{\mathbf{g}}_t^\top(D_t\mathbf{w}_t - \mathbf{w})$. Hence, the regret $R_T(\mathbf{w})$ can be bounded as

$$R_T(\mathbf{w}) \leq \sum_{t=1}^T \tilde{\mathbf{g}}_t^\top(D_t\mathbf{w}_t - \mathbf{w}) = \sum_{t=1}^T \tilde{\mathbf{g}}_t^\top D_t\mathbf{w}_t - \sum_{t=1}^T \tilde{\mathbf{g}}_t^\top \mathbf{w}.$$

From a similar argument to the proof of Lemma 9, we obtain

$$\sum_{t=1}^T \tilde{\mathbf{g}}_t^\top D_t\mathbf{w}_t \leq \tilde{\mathbf{h}}_T^\top\mathbf{w} + \sum_{t=1}^T \frac{\|D_t\tilde{\mathbf{g}}_t\|^2}{\tilde{\lambda}_t} + \frac{\tilde{\lambda}_{T+1}\|\mathbf{w}\|^2}{2}.$$

By combining the above two inequalities, we obtain (24). □

Algorithm 2

Input: $\{(\mathbf{x}_t, y_t)\} \subseteq \mathbb{R}^d \times \mathbb{R}$, $\{\lambda_s\}, \{\tilde{\lambda}_t\} \subseteq \mathbb{R}_{>0}$, $k' \geq 2$ and $k_1 \geq 0$ such that $0 \leq k_1 \leq k' - 2$, $J \subseteq \{1, \ldots, T\}$

Set $\hat{\mathbf{h}}_0 = 0, \tilde{\mathbf{h}}_0 = 0, \bar{\mathbf{w}}_0 = 0, s = 0$.
for $t = 1, \ldots, T$ **do**
  if $t \in J$ **then**
    Set $s = s + 1$.
    Define $\mathbf{w}_s$ by (5), and $\bar{\mathbf{w}}_s = \bar{\mathbf{w}}_{s-1} + \mathbf{w}_s$.
  end if
  Define $\tilde{\mathbf{w}}_t$ by (22).
  Define S_t by $\texttt{Observe}(\bar{\mathbf{w}}_s, k', k_1)$.
  Observe $D_t \mathbf{x}_t$ and output $\hat{y}_t := \tilde{\mathbf{w}}_t^\top D_t \mathbf{x}_t$.
  Observe y_t.
  if $t \in J$ **then**
    Define $\hat{\mathbf{g}}_s$ by (6), and set $\hat{\mathbf{h}}_s = \hat{\mathbf{h}}_{s-1} + \hat{\mathbf{g}}_s$.
  end if
  Compute $D_t \tilde{\mathbf{g}}_t$ ($\tilde{\mathbf{g}}_t$ is defined by (23)).
  Set $\tilde{\mathbf{h}}_t = \tilde{\mathbf{h}}_{t-1} + D_t \tilde{\mathbf{g}}_t$.
end for

Because of (24), if $\|\tilde{\mathbf{g}}_t\| = O(1)$ holds and we set $\tilde{\lambda}_t = \Theta(\sqrt{t})$, we have

$$\mathbf{E}[R_T(\mathbf{w}^*)] = O\left(\sum_{t=1}^T \|(D_t - I)\mathbf{w}^*\| + \sqrt{T}\right).$$

From Lemma 12, we can prove that S_t satisfies $\sum_{t=1}^T \sum_{j \in S^*} \mathrm{Prob}[j \notin S_t] = O(\sqrt{T})$. Combining these two facts, we obtain $O(\sqrt{T})$ regret. A more precise statement is given in the following theorem.

Proof of Theorem 6

Proof. From Lemma 13 and that $\|D_t \tilde{\mathbf{g}}_t\|^2 \leq 4$ and $\tilde{\lambda}_t = 8\sqrt{t}$, we have

$$\begin{aligned} R_T(\mathbf{w}^*) &\leq \sum_{t=1}^T \mathbf{w}^{*\top}(D_t - I)\tilde{\mathbf{g}}_t + \sum_{t=1}^T \frac{\|D_t \tilde{\mathbf{g}}_t\|^2}{\tilde{\lambda}_t} + \frac{\tilde{\lambda}_{T+1}}{2} \\ &= \sum_{t=1}^T \mathbf{w}^{*\top}(D_t - I)\tilde{\mathbf{g}}_t + 8\sqrt{T+1}. \end{aligned}$$

Further, since $((D_t - I)\mathbf{w}^*)_i = -w_i^*$ if $i \in S^* \setminus S_t$ and $((D_t - I)\mathbf{w}^*)_i = 0$ otherwise, the first term can be bounded as

$$\sum_{t=1}^T \mathbf{w}^{*\top}(D_t - I)\tilde{\mathbf{g}}_t = -\sum_{t=1}^T \sum_{i \in S^* \setminus S_t} w_i^* \tilde{g}_{ti} \leq 4 \sum_{t=1}^T \sum_{i \in S^* \setminus S_t} |w_i^*|,$$

where the last inequality comes from $\|\tilde{g}_t\| \leq 4$. From the above two inequalities, by taking the expectation, we obtain

$$\mathbf{E}[R_T(\mathbf{w}^*)] \leq 4 \sum_{t=1}^T \sum_{i \in S^*} |w_i^*| \mathrm{Prob}[i \notin S_t] + 8\sqrt{T+1}. \tag{25}$$

Next, we give a upper bound on $\sum_{t=1}^T \mathrm{Prob}[i \notin S_t]$ for $i \in S^*$ by using Lemma 12. Define $\gamma(s) = \frac{1}{2}\sigma_d^2 w_i^{*2} - 8Gs^{-\frac{1}{2}}$. If s is large enough so that $\gamma(s) \geq 0$, i.e., if $s \geq 256 \frac{G^2}{\sigma_d^4 w_i^{*4}} =: \kappa_i$, then we

have $\sigma_d^2 w_i^{*2} - 8Gs^{-\frac{1}{2}} \geq \frac{1}{2}\sigma_d^2 w_i^{*2}$. From Lemma 12, then, we have $\mathrm{Prob}[i \in S_t] \leq 3\exp(-\frac{s\sigma_d^2 w_i^{*2}}{144C})$. Thus, we have

$$\begin{aligned}
\sum_{t=1}^{T} \mathrm{Prob}[i \notin S_t] &= \sum_{t \in J \cap [T]} \mathrm{Prob}[i \notin S_t] + \sum_{t \in [T] \setminus J} \mathrm{Prob}[i \notin S_t] \\
&\leq \sqrt{T} + \sum_{t \in [T] \setminus J} \mathrm{Prob}[i \notin S_t] \\
&= \sqrt{T} + \sum_{t \in [T] \setminus J, \sqrt{t}-1 \leq \kappa_i} \mathrm{Prob}[i \notin S_t] + \sum_{t \in [T] \setminus J, \sqrt{t}-1 > \kappa_i} \mathrm{Prob}[i \notin S_t] \\
&\leq \sqrt{T} + (\kappa_i + 1)^2 + \sum_{t \in [T] \setminus J, \sqrt{t}-1 > \kappa_i} \mathrm{Prob}[i \notin S_t], \qquad (26)
\end{aligned}$$

where the first inequality comes from $|[T] \cup J|$, and the second inequality comes from $|\{t \geq 1 \mid \sqrt{t} - 1 \leq \kappa_i\}| \leq (\kappa_i + 1)^2$. Note that, since $s \geq \sqrt{t} - 1$ holds in each step t, $\sqrt{t} - 1 > \kappa_i$ implies that $s > \kappa_i$ and, hence, $\mathrm{Prob}[i \notin S_t] \leq 3\exp(-\frac{s\sigma_d^2 w_i^{*2}}{144C}) \leq 3\exp(-\frac{(\sqrt{t}-1)\sigma_d^2 w_i^{*2}}{144C})$ holds. From this, the last term of (26) can be bounded as

$$\begin{aligned}
\sum_{t \in [T] \setminus J, \sqrt{t}-1 > \kappa_i} \mathrm{Prob}[i \notin S_t] &= \sum_{t \in [T] \setminus J, \sqrt{t}-1 > \kappa_i, t \leq \sqrt{T}} \mathrm{Prob}[i \notin S_t] + \sum_{t \in [T] \setminus J, \sqrt{t}-1 > \kappa_i, t > \sqrt{T}} \mathrm{Prob}[i \notin S_t] \\
&\leq \sqrt{T} + \sum_{t \in [T] \setminus J, \sqrt{t}-1 > \kappa_i, t > \sqrt{T}} 3\exp(-\frac{(\sqrt{t}-1)\sigma_d^2 w_i^{*2}}{144C}) \\
&\leq \sqrt{T} + 3T\exp(\frac{-(T^{\frac{1}{4}}-1)\sigma_d^2 w_i^{*2}}{144C})
\end{aligned}$$

By combining the above two inequalities, we obtain

$$\sum_{t=1}^{T} \mathrm{Prob}[i \notin S_t] \leq 2\sqrt{T} + (\kappa_i + 1)^2 + 3T\exp(\frac{-(T^{\frac{1}{4}}-1)\sigma_d^2 w_i^{*2}}{144C}).$$

By substituting this inequality into (25), we have

$$\mathbf{E}[R_T(\mathbf{w}^*)] \leq 8\sum_{i \in S^*} |w_i^*|\sqrt{T} + 8\sqrt{T+1} + 4\sum_{i \in S^*} |w_i^*|((\kappa_i + 1)^2 + 3T\exp(\frac{-(T^{\frac{1}{4}}-1)\sigma_d^2 w_i^{*2}}{144C})).$$

The first term of the right-hand side can be bounded as $4\sum_{i \in S^*} |w_i^*|\sqrt{T} = \|\mathbf{w}^*\|_1 4T \leq \sqrt{d}4\sqrt{T+1}$ because $\|\mathbf{w}^*\| \leq 1$. Further, substituting $\kappa_i = 256\frac{G^2}{\sigma_d^4 w_i^{*4}}$, we obtain Theorem 6. □

E Details of Algorithm 3

Computing $\mathbf{w}_s$ and $\bar{\mathbf{w}}_s$. Let $\{\lambda_j\}, \{\eta_j\} \subseteq \mathbb{R}_{>0}$ be positive monotone increasing sequences. Denote $\zeta_j = \eta_j - \eta_{j-1}$ for $j > 1$ and $\zeta_1 = \eta_1$. Then, we have $\zeta_j > 0$ and $\eta_s = \sum_{j=1}^{s} \zeta_j$.

Recall that $\hat{\mathbf{h}}_{s-1} = \sum_{j=1}^{s-1} \hat{\mathbf{g}}_j$. We define $\mathbf{w}_s$ and $\bar{\mathbf{w}}_s$ by

$$\mathbf{w}_s = \operatorname*{arg\,min}_{\mathbf{w} \in \mathbb{R}^d, \|\mathbf{w}\| \leq 1} \left\{ \hat{\mathbf{h}}_{s-1}^\top \mathbf{w} + \frac{\lambda_s}{2}\|\mathbf{w}\|^2 + \eta_s \|\mathbf{w}\|_1 \right\} = -\frac{1}{\max\{\lambda_t, \|\hat{\mathbf{h}}_{t-1}\|/1\}} \hat{\mathbf{h}}_t,$$

and $\bar{\mathbf{w}}_s = \sum_{j=1}^{s} \zeta_j \mathbf{w}_j / \eta_s$. Then, $\bar{\mathbf{w}}_s$ gets close to $\mathbf{w}^*$ with high probability.

Lemma 14. *Let G and C be as defined in Section B. Set $\lambda_j = \frac{8}{C}\sqrt{j}$ and $\zeta_j = \phi_0\sqrt{4/(Ck)}j^{-\frac{1}{4}}$. Under the compatibility assumption, for arbitrary $\delta > 0$, $4\phi_0(s^{\frac{3}{4}} - 1)\sqrt{\frac{4}{Ck}}\|\bar{\mathbf{w}}_s - \mathbf{w}^*\|_1 \leq \frac{144}{C}\sqrt{s+1} + 27\delta$ holds with probability at least $1 - 3\exp(\frac{-C^2\delta^2}{128s})$ over the randomness of both the examples and the algorithm.*

Proof. From the convexity of the triangle inequality of ℓ_1 norm, we have

$$\eta_t \|\bar{\mathbf{w}}_t - \mathbf{w}^*\|_1 = \|\sum_{j=1}^{t} \zeta_j (\mathbf{w}_t - \mathbf{w}^*)\|_1 \leq \sum_{j=1}^{t} \zeta_j \|\mathbf{w}_t - \mathbf{w}^*\|_1.$$

Hence, it suffices to give a bound on $\sum_{j=1}^{t} \zeta_j \|\mathbf{w}_t - \mathbf{w}^*\|_1$. Define $\gamma_j = \|L(\mathbf{w}_j - \mathbf{w}^*)\|^2 + \zeta_j(\|\mathbf{w}_j\|_1 - \|\mathbf{w}^*\|_1)$, and we shall show the following bound on $\zeta_j \|\mathbf{w}_s - \mathbf{w}^*\|_1$:

$$\zeta_j \|\mathbf{w}_j - \mathbf{w}^*\|_1 \leq 3k\zeta_j^2/\phi_0^2 + 3\gamma_j$$

under the assumption of the compatibility condition, i.e., $\|\mathbf{w}_{[d]\setminus S^*}\|_1 \leq 2\|\mathbf{w}_{S^*}\|_1 \Longrightarrow \phi_0^2 \|\mathbf{w}_{S^*}\|_1^2 \leq k\|L\mathbf{w}\|^2$. In the following, we use the notation $\Delta = \mathbf{w}_j - \mathbf{w}^*$ for convenience. Then, we have

$$\begin{aligned}
\gamma_j &= \|L\Delta\|^2 + \zeta_j(\|\mathbf{w}_j|_{S^*}\|_1 + \|\mathbf{w}_j|_{S^{*c}}\|_1 - \|\mathbf{w}^*|_{S^*}\|_1) \\
&\geq \|L\Delta\|^2 + \zeta_j(\|\Delta|_{S^{*c}}\|_1 - \|\Delta|_{S^*}\|_1) \\
&\geq \zeta_j(\|\Delta|_{S^{*c}}\|_1 - \|\Delta|_{S^*}\|_1). \qquad (27)
\end{aligned}$$

We will bound Δ by considering the following two cases: (i) $\gamma_j \leq \zeta_j \|\Delta|_{S^*}\|_1$ and (ii) $\gamma_j > \zeta_j \|\Delta|_{S^*}\|_1$.

Case (i) $\gamma_j \leq \zeta_j \|\Delta|_{S^*}\|_1$:
From (27) and $\gamma_j \leq \zeta_j \|\Delta|_{S^*}\|_1$, we have

$$\zeta_j \|\Delta|_{S^{*c}}\|_1 \leq \zeta_j \|\Delta|_{S^*}\|_1 + \gamma_j \leq 2\zeta_j \|\Delta|_{S^*}\|_1.$$

From the compatibility condition, we have $\phi_0^2 \|\Delta|_{S^*}\|_1^2 \leq k\|L\Delta\|^2$. Hence, we have

$$\begin{aligned}
\zeta_j \|\Delta\|_1 &= \zeta_j \|\Delta|_{S^*}\|_1 + \zeta_j \|\Delta|_{S^{*c}}\|_1 \\
&\leq 2\zeta_j \|\Delta|_{S^*}\|_1 + \gamma_j - \|L\Delta_j\|^2 \\
&\leq 2\zeta_j \|\Delta|_{S^*}\|_1 + \gamma_j - \phi_0^2 \|\Delta|_{S^*}\|_1^2/k \\
&\leq k\zeta_j^2/\phi_0^2 + \gamma_j \leq 3k\zeta_j^2/\phi_0^2 + 3\gamma_j,
\end{aligned}$$

where the first, second, and third inequalities come from (27), the compatibility condition, and completing the square.

Case (ii) $\gamma_j > \zeta_j \|\Delta|_{S^*}\|_1$:
From (27) and $\gamma_j \leq \zeta_j \|\Delta|_{S^*}\|_1$, we have

$$\zeta_j \|\Delta\|_1 = \zeta_j \|\Delta|_{S^*}\|_1 + \zeta_j \|\Delta|_{S^{*c}}\|_1 \leq 2\zeta_j \|\Delta|_{S^*}\|_1 + \gamma_j \leq 3\gamma_j.$$

From the argument on cases (i) and (ii), we have $\zeta_j \Delta = \zeta_j \|\mathbf{w}_j - \mathbf{w}^*\|_1 \leq 3k\zeta_j^2/\phi_0^2 + 3\gamma_j$. Taking the sum over $j = 1, 2, \ldots, t$, we have

$$\sum_{j=1}^{t} \zeta_j \|\mathbf{w}_j - \mathbf{w}^*\|_1 \leq \frac{3k}{\phi_0^2} \sum_{j=1}^{s} \zeta_j^2 + 3\sum_{j=1}^{s} \gamma_j.$$

Here, from Lemma 10, for all $\delta > 0$, the value $\sum_{j=1}^{s} \gamma_j$ can be bounded as

$$\begin{aligned}
\sum_{j=1}^{s} \gamma_j &= \sum_{j=1}^{s} (\|L(\mathbf{w}_j - \mathbf{w}^*)\|^2 + \zeta_j(\|\mathbf{w}_j\|_1 - \|\mathbf{w}^*\|_1)) \\
&\leq \sum_{j=1}^{s} (\hat{\mathbf{g}}_j^\top (\mathbf{w}_j - \mathbf{w}^*) + \zeta_j(\|\mathbf{w}_j\|_1 - \|\mathbf{w}^*\|_1)) + 3\delta
\end{aligned}$$

with probability at least $1 - 3\exp\left(\frac{-\delta^2}{Cs}\right)$. Further, from Lemma 9 the right-most side can be bounded as

$$\begin{aligned}
&\sum_{j=1}^{s} (\hat{\mathbf{g}}_j^\top (\mathbf{w}_j - \mathbf{w}^*) + \zeta_j(\|\mathbf{w}_j\|_1 - \|\mathbf{w}^*\|_1)) + 3\delta \\
&\leq \sum_{j=1}^{s} \frac{1}{\lambda_j} \|\hat{\mathbf{g}}_j\|^2 + \frac{1\lambda_{s+1}}{2} + 3\delta \leq \sum_{j=1}^{s} \frac{16}{\lambda_j} + \frac{1\lambda_{s+1}}{2} + 3\delta
\end{aligned}$$

with probability one. Summarizing the above argument, it holds that

$$\eta_s \|\bar{\mathbf{w}}_s - \mathbf{w}^*\| \leq \frac{3k}{\phi_0^2} \sum_{j=1}^{s} \zeta_j^2 + 3 \sum_{j=1}^{s} \frac{16}{\lambda_j} + \frac{3\lambda_{s+1}}{2} + 9\delta$$

with probability at least $1 - 3\exp\left(\frac{-\delta^2}{Cs}\right)$. By assigning $\lambda_j = 2G\sqrt{j}$ and $\zeta_j = \phi_0\sqrt{G/k}j^{-\frac{1}{4}}$, we obtain Lemma 14.

□

Regret bound of Algorithm 3 We prove that Algorithm 3 achieves $O(\sqrt{dT})$ regret assuming the compatibility condition. Recall that S_t is the set of the k largest features with respect to $\bar{\mathbf{w}}_s$, From Lemma 14, S_t contains S^* with high probability as follows.

Lemma 15. *Let G and C be constants defined as in Section B. Let $\{\lambda_j\}$ and $\{\zeta_j\}$ be sequences defined as in Lemma 14. For any $i \in S^*$ and $t = t_s, \ldots, t_{s+1}-1$, if $\delta := \frac{1}{27}(4\phi_0(s^{\frac{3}{4}}-1)|w_i^*|\sqrt{\frac{4}{Ck}} - \frac{144}{C}\sqrt{s+1}) > 0$, then we have* $\mathrm{Prob}[i \notin S_t] \leq 3\exp(\frac{-C^2\delta^2}{128s})$.

Proof. If a feature i satisfies $i \in S^* \setminus S_t$, it holds that $\|\bar{\mathbf{w}}_s - \mathbf{w}^*\|_1 \geq |w_i^*|$, which can be confirmed as follows. Since $|S^*| = |S_t|$, $i \in S^* \setminus S_t$ means that there exist $j \in S_t \setminus S^*$ such that $|\bar{w}_{sj}| > |\bar{w}_{si}|$, which implies that $\|\bar{\mathbf{w}}_s - \mathbf{w}^*\|_1 \geq |\bar{w}_{si} - w_i^*| + |\bar{w}_{sj} - w_j^*| = |\bar{w}_{si} - w_i^*| + |\bar{w}_{sj}| \geq |\bar{w}_{si} - w_i^*| + |\bar{w}_{si}| \geq |w_i^*|$, where the first equality comes from $j \notin S^*$, the second inequality comes from $|\bar{w}_{sj}| > |\bar{w}_{si}|$, and the third is the triangle inequality. Hence, we have

$$\mathrm{Prob}[i \notin S_t] \leq \mathrm{Prob}[\|\bar{\mathbf{w}}_s - \mathbf{w}^*\|_1 \geq |w_i^*|].$$

From Lemma 14, if $\delta := \frac{1}{27}(4\phi_0(s^{\frac{3}{4}} - 1)\sqrt{G/k}|w_i^*| - 36G\sqrt{s+1}) \geq 0$, we have

$$\mathrm{Prob}[\|\bar{\mathbf{w}}_s - \mathbf{w}^*\|_1 \geq |w_i^*|] \leq 3\exp(\frac{-\delta^2}{Cs})$$

for $i \in S^*$. The above two inequalities yield Lemma 15. □

Proof of Theorem 7

Proof. The outline of the proof is similar to that of Theorem 6. From Lemma 13, we obtain

$$\mathbf{E}[R_T(\mathbf{w}^*)] \leq 4 \sum_{t=1}^{T} \sum_{i \in S^*} |w_i^*| \mathrm{Prob}[i \notin S_t] + 8\sqrt{T+1}, \tag{28}$$

in a similar way to the proof of Theorem 6.

Next, we give a upper bound on $\sum_{t=1}^{T} \mathrm{Prob}[i \notin S_t]$ for $i \in S^*$ by using Lemma 15. Define $\gamma(s) = 2|w_i^*|\phi_0\sqrt{G/k} - 4|w_i^*|\phi_0\sqrt{G/k}s^{-\frac{3}{4}} - 36Gs^{-\frac{3}{4}}\sqrt{s+1}$. If s is large enough so that $\gamma(s) \geq 0$, from Lemma 12, we have $\mathrm{Prob}[i \in S_t] \leq 3\exp(-\frac{4\sqrt{s}|w_i^*|^2\phi_0^2 G}{27^2 Ck})$. Note that if $s \geq \frac{4 \cdot 36^4 G^2 k^2}{w_i^{*4}\phi_0^4} =: \kappa_i$, it holds that $\gamma(s) > 0$, from the definition of $\gamma(s)$. Further, since $s \geq \sqrt{t} - 1$ in each step t, if $\sqrt{t} - 1 \geq \kappa_i$, we have $\mathrm{Prob}[i \in S_t] \leq 3\exp(-\frac{4\sqrt{\sqrt{t}-1}|w_i^*|^2\phi_0^2 G}{27^2 Ck})$. From this property of S_t, by a similar argument to the proof of Theorem 6, we have

$$\sum_{t=1}^{T} \mathrm{Prob}[i \notin S_t] \leq 2\sqrt{T} + (\kappa_i + 1)^2 + 3T\exp(-\frac{4\sqrt{T^{\frac{1}{4}} - 1}|w_i^*|^2\phi_0^2 G}{27^2 Ck}).$$

By combining this inequality with (28), we obtain

$$\mathbf{E}[R_T(\mathbf{w}^*)] \leq 8 \sum_{i \in S^*} |w_i^*|\sqrt{T} + 8\sqrt{T+1}$$
$$+ \sum_{i \in S^*} |w_i^*|((\kappa_i + 1)^2 + 3T\exp(-\frac{4\sqrt{T^{\frac{1}{4}} - 1}|w_i^*|^2\phi_0^2 G}{27^2 Ck})).$$

The first term of the right-hand side can be bounded as $4\sum_{i \in S^*} |w_i^*|\sqrt{T} = \|\mathbf{w}^*\|_1 4T \leq \sqrt{d}8\sqrt{T+1}$ because $\|\mathbf{w}^*\| \leq 1$. Further, substituting $\kappa_i = \frac{4 \cdot 36^4 G^2 k^2}{w_i^{*4}\phi_0^4}$, we obtain Theorem 6. □

F More Experiments

In this section, we provide supplementary descriptions of our experiments.

Experimental environment. The experiments were performed on a server with Intel Xeon E5-2680 v3 CPUs. All algorithms are implemented in Python.

The generation procedure of synthetic datasets. We first create the ground-truth weight vector $\mathbf{w}_{\text{true}}$ by choosing a set S_k of k features from $[d]$ uniformly at random, and setting $w_{\text{true},i} \in [-1, 1]$ for $i \in S_k$ and $w_{\text{true},i} = 0$ otherwise. For each $t \in T$, we generate $\mathbf{x}_t$ by sampling $x_{t,i}$ $(i \in [d])$ from $\mathcal{N}(0, 1)$ and set $y_t = \mathbf{w}_{\text{true}}^\top \mathbf{x}_t + 0.5z$, where z is sampled from $\mathcal{N}(0, 1)$.

Preprocessing of CT-slice datasets. We deal with features that are outside of an image as those having a value of zero. The sequence of the dataset is randomly shuffled to avoid the effects of biased sequences of images. The maximum number of positive features in one image is 165, the minimum is 9, and the average is 73. Thus, we set k to be 60, 70, or 90 in this paper.

Results on synthetic datasets. Figures 4, 5 show typical results for some instance with $(d, k, k') = (20, 5, 7)$. We remark that Figure 1 in the main body plots the regrets for only the first 5000 iterations. We observe that our algorithms achieve small regrets at the end of iterations. Figure 5 indicates that the increase in the regrets of our algorithms is smaller than baseline algorithms for large T. We remark that FKK is executed for 5000 iterations because the run time is too expensive compared with the others. However, Figure 1, which plots regrets for the first 5000 iterations, shows that the increase in the regrets for FKK is similar to greedy and uniform-random. Thus, we can expect that our algorithms perform much better than all baseline algorithms.

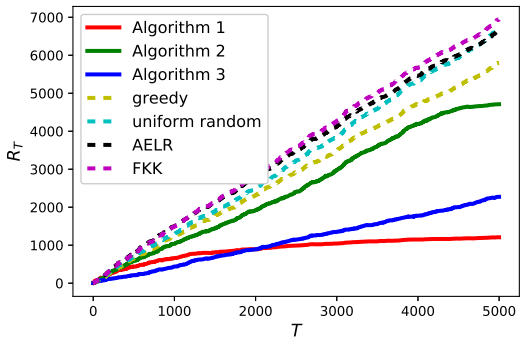

Figure 4: The regrets for a synthetic instance with $(d, k, k') = (20, 5, 7)$.

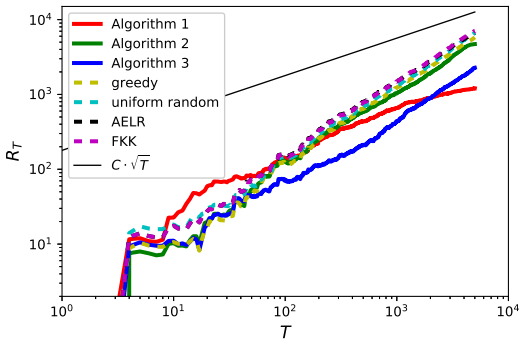

Figure 5: The log-log plots of the regrets for a synthetic instance with $(d, k, k') = (20, 5, 7)$.

We compute the averages of final regrets and execution times for each combination of $d \in \{10, 20, 50, 100\}$, $k \in \{2, 5, 10\}$, and $k' \in \{k + 2, k + 5, k + 10\}$. For each combination of (d, k, k'), we executed all algorithms on five instances with $T = 5000$. Tables 4 and 5 summarize the average regrets and execution times. In the tables, "T" denotes that we do not execute FKK due to the expensive run time. We observe that Algorithm 1 performs best for almost all cases. Our algorithms, greedy, and uniform-random can all process each round about three times as fast as AELR. Thus, we can say that our algorithms outperform the others in terms of both regrets and execution times.

Results on CT-slice datasets. We present results with $(k, k') = \{(60, 70), (70, 80), (90, 95)\}$ in Figures 6, 7, and 8, respectively. We observe that Algorithms 2 and 3 perform best. The increases in the regrets of our algorithms are much smaller than those of uniform-random and AELR. Greedy performs well at the beginning, but the performance degrades at the end.

Table 4: Average regrets $R_T/10^2$ (A1 = Algorithm 1, A2 = Algorithm 2, A3 = Algorithm 3, G = greedy, U = uniform random).

(d, k, k')	A1	A2	A3	G	U	AELR	FKK
(10,2,4)	**1.53**	2.38	3.60	33.28	25.73	60.76	24.05
(10,2,7)	**1.45**	4.73	1.84	39.62	16.45	66.60	5.92
(10,5,7)	**3.35**	46.18	16.74	79.38	44.46	145.87	57.89
(20,2,4)	**6.03**	7.31	13.63	16.52	14.89	37.15	14.90
(20,2,7)	**1.03**	5.99	5.03	8.13	6.07	17.16	6.65
(20,2,12)	**0.81**	3.11	3.13	8.18	4.47	16.49	6.46
(20,5,7)	**15.45**	83.53	49.58	49.37	80.55	138.05	83.50
(20,5,10)	**6.10**	43.15	17.20	16.18	65.08	140.03	T
(20,5,15)	**4.00**	9.81	8.40	7.93	34.23	140.12	T
(20,10,12)	**15.00**	40.86	24.76	21.38	104.63	254.52	T
(20,10,15)	**6.79**	5.14	9.97	10.08	71.74	283.76	T
(100,2,7)	**41.09**	42.13	49.67	49.78	47.42	83.70	T
(100,2,12)	**25.90**	51.34	50.13	50.66	46.82	84.11	T
(100,5,7)	133.54	118.23	119.17	119.51	**113.17**	214.84	T
(100,5,10)	**74.06**	118.95	106.78	120.12	112.51	217.52	T
(100,5,15)	**62.45**	104.57	84.52	122.71	111.60	211.22	T
(100,10,12)	295.95	225.06	**203.04**	227.81	214.09	366.71	T
(100,10,15)	159.95	213.20	**159.35**	198.82	212.39	423.73	T
(100,10,20)	**85.89**	225.77	199.80	193.25	207.16	372.67	T

Table 5: Average run time [s] (A1 = Algorithm 1, A2 = Algorithm 2, A3 = Algorithm 3, G = greedy, U = uniform random).

(d, k, k')	A1	A2	A3	G	U	AELR	FKK
(10,2,4)	0.54	0.48	0.49	0.49	0.57	4.90	3.30
(10,2,7)	0.81	0.50	0.51	0.63	0.89	5.03	3.50
(10,5,7)	0.69	0.50	0.51	0.62	0.87	5.09	18.97
(20,2,4)	0.60	0.53	0.52	0.55	0.64	6.19	13.11
(20,2,7)	0.87	0.54	0.59	0.66	0.93	6.20	13.01
(20,2,12)	1.63	0.57	0.66	1.10	1.76	6.31	14.03
(20,5,7)	0.71	0.48	0.47	0.61	0.87	5.46	1021.51
(20,5,10)	1.08	0.52	0.52	0.81	1.31	5.93	T
(20,5,15)	2.01	0.56	0.56	1.26	2.37	6.04	T
(20,10,12)	1.13	0.56	0.54	1.00	1.73	6.36	T
(20,10,15)	1.73	0.61	0.63	1.34	2.51	6.83	T
(100,2,4)	1.09	0.83	0.82	0.88	1.10	11.59	T
(100,2,7)	1.36	0.92	0.89	1.06	1.46	12.37	T
(100,2,12)	2.12	0.95	0.90	1.35	2.21	12.06	T
(100,5,7)	1.21	0.86	0.84	1.00	1.37	11.52	T
(100,5,10)	1.58	0.88	0.86	1.18	1.80	11.62	T
(100,5,15)	2.46	0.94	0.90	1.59	2.81	11.52	T
(100,10,12)	1.61	0.89	0.88	1.33	2.16	11.98	T
(100,10,15)	2.13	0.93	0.94	1.61	2.81	12.39	T
(100,10,20)	3.29	1.00	1.00	2.16	4.24	12.66	T

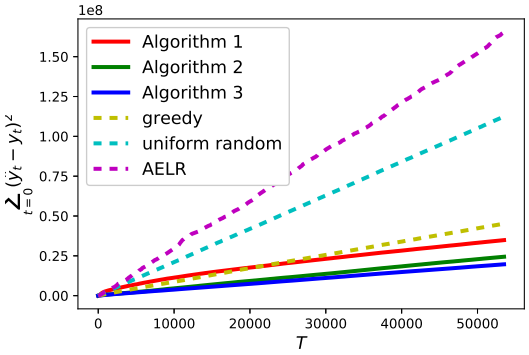

Figure 6: The square loss for CT-slice datasets ($k = 60, k' = 70$).

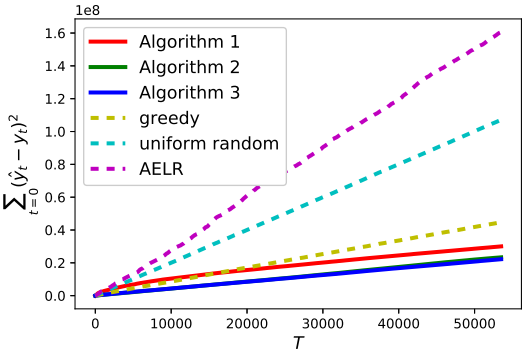

Figure 7: The square loss for CT-slice datasets ($k = 70, k' = 80$).

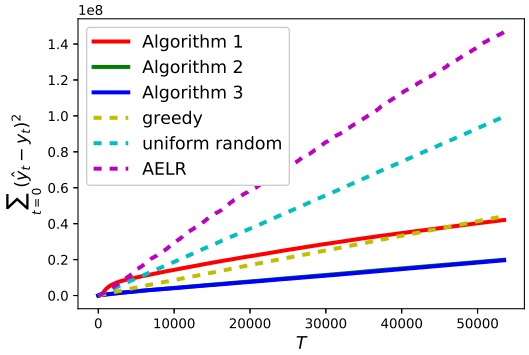

Figure 8: The square loss for CT-slice datasets ($k = 90, k' = 95$).