

# A Provably Accurate Randomized Sampling Algorithm for Logistic Regression

Agniva Chowdhury<sup>1</sup>, Pradeep Ramuhalli<sup>2</sup>

<sup>1</sup>Computer Science and Mathematics Division, Oak Ridge National Laboratory, TN, USA

<sup>2</sup>Nuclear Energy and Fuel Cycle Division, Oak Ridge National Laboratory, TN, USA  
{chowdhurya, ramuhallip}@ornl.gov

## Abstract

In statistics and machine learning, logistic regression is a widely-used supervised learning technique primarily employed for binary classification tasks. When the number of observations greatly exceeds the number of predictor variables, we present a simple, randomized sampling-based algorithm for logistic regression problem that guarantees high-quality approximations to both the estimated probabilities and the overall discrepancy of the model. Our analysis builds upon two simple structural conditions that boil down to randomized matrix multiplication, a fundamental and well-understood primitive of randomized numerical linear algebra. We analyze the properties of estimated probabilities of logistic regression when leverage scores are used to sample observations, and prove that accurate approximations can be achieved with a sample whose size is much smaller than the total number of observations. To further validate our theoretical findings, we conduct comprehensive empirical evaluations. Overall, our work sheds light on the potential of using randomized sampling approaches to efficiently approximate the estimated probabilities in logistic regression, offering a practical and computationally efficient solution for large-scale datasets.

## 1 Introduction

In statistics and machine learning, logistic regression (Hosmer Jr, Lemeshow, and Sturdivant 2013) is a widely-used supervised learning technique applied to binary classification tasks. It is a statistical method that predicts one of two possible outcomes based on the input features. More specifically, the goal is to model the probability of one of the binary outcomes based on the predictor variables. In machine learning and various scientific applications, logistic regression appears in numerous settings, including online learning (Zhang et al. 2012), feature selection (Koh, Kim, and Boyd 2007), anomaly detection (Hendrycks, Mazeika, and Dietterich 2019; Feng et al. 2014), disease classification (Liao and Chin 2007; Chai et al. 2018), image & signal processing (Dong, Zhu, and Gong 2019; Rosario 2004), probability calibration (Kull et al. 2019) and many more.

Formally, given the data matrix  $\mathbf{X} \in \mathbb{R}^{n \times d}$  and the binary response vector  $\mathbf{y} \in \{0, 1\}^n$ , logistic regression models the

following

$$\mathbb{P}(y_i = 1 \mid \mathbf{X}_{i*}) := p_i(\boldsymbol{\beta}) = \frac{\exp(\mathbf{X}_{i*}\boldsymbol{\beta})}{1 + \exp(\mathbf{X}_{i*}\boldsymbol{\beta})}, \quad (1)$$

for  $i = 1, 2, \dots, n$ . Here,  $y_i \in \{0, 1\}$  is the  $i$ -th component of  $\mathbf{y}$ ,  $\mathbf{X}_{i*}$  is  $i$ -th row (as a row vector) of  $\mathbf{X}$ , and  $\boldsymbol{\beta} \in \mathbb{R}^d$  is the vector of unknown regression coefficients, which is often estimated by the maximum likelihood estimator (MLE) *i.e.*, through maximizing the log-likelihood function with respect to  $\boldsymbol{\beta}$ , which is given by

$$\begin{aligned} \ell(\boldsymbol{\beta}) &= \sum_{i=1}^n (y_i \log p_i(\boldsymbol{\beta}) + (1 - y_i) \log(1 - p_i(\boldsymbol{\beta}))) \\ &= \sum_{i=1}^n (y_i \mathbf{X}_{i*}\boldsymbol{\beta} - \log(1 + \exp(\mathbf{X}_{i*}\boldsymbol{\beta}))), \end{aligned} \quad (2)$$

where we get eqn. (2) by using eqn. (1) in the previous step. The MLE of the coefficients vector  $\boldsymbol{\beta}$  can be written as  $\boldsymbol{\beta}^* = \operatorname{argmax}_{\boldsymbol{\beta}} \ell(\boldsymbol{\beta})$ . Equivalently, eqn. (2) can be written in the following compact form:

$$\ell(\boldsymbol{\beta}) = \mathbf{y}^\top \mathbf{X}\boldsymbol{\beta} - \mathbf{1}^\top \mathbf{g}(\boldsymbol{\beta}) \quad (3)$$

where  $\mathbf{g}(\boldsymbol{\beta})$  is an  $n \times 1$  vector with the  $i$ -th entry  $g_i(\boldsymbol{\beta}) = \log(1 + \exp(\mathbf{X}_{i*}\boldsymbol{\beta}))$ , for  $i = 1, \dots, n$ . The MLE  $\boldsymbol{\beta}^*$  satisfies the following condition

$$\left. \frac{\partial \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} \right|_{\boldsymbol{\beta}=\boldsymbol{\beta}^*} = \mathbf{0} \Rightarrow \mathbf{X}^\top (\mathbf{y} - \mathbf{p}(\boldsymbol{\beta}^*)) = \mathbf{0}. \quad (4)$$

Here  $\mathbf{p}(\boldsymbol{\beta}^*)$  is an  $n$ -dimensional vector of *estimated probabilities*<sup>1</sup>, with the  $i$ -th entry corresponds to  $p_i(\boldsymbol{\beta}^*)$ , for  $i = 1, \dots, n$ . Unfortunately, due to the non-linearity of  $\ell(\boldsymbol{\beta})$ , there is no closed-form analytical solution to eqn. (4). As a result, a variant of Newton’s method, namely, *iteratively reweighted least squares* (IRLS) (Green 1984) is commonly used to find  $\boldsymbol{\beta}^*$  from eqn. (4), that maximizes  $\ell(\boldsymbol{\beta})$ . The IRLS algorithm iteratively computes the MLE of the parameter vector, by solving a weighted least squares problem at every iteration. Therefore, the *per iteration* cost of the algorithm is dominated by the cost of solving the aforementioned weighted least squares problem at each iteration.

<sup>1</sup>The term “predicted probabilities” is used in some literature, but throughout our paper, we consistently refer to it as the vector of “estimated probabilities.”

In our work, we will focus on the data matrix  $\mathbf{X} \in \mathbb{R}^{n \times d}$  with  $n \gg d$  i.e., the number of observations greatly exceeds the number of predictor variables. For simplicity of exposition, we will also assume  $\mathbf{X}$  is of full rank i.e.,  $\text{rank}(\mathbf{X}) = d$ . Now, in such  $n \gg d$  setting, solving the weighted least-squares problem at each iteration of the IRLS algorithm is expensive, taking  $\mathcal{O}(nd^2)$  time, which essentially is the cost of computing the inverse of the Hessian matrix  $\left[ \frac{\partial^2 \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^\top} \right]^{-1} = -(\mathbf{X}^\top \mathbf{W} \mathbf{X})^{-1}$  at each iteration, where  $\mathbf{W} \in \mathbb{R}^{n \times n}$  is a diagonal weight matrix with the diagonal entries  $p_i(\boldsymbol{\beta})(1 - p_i(\boldsymbol{\beta}))$ , for  $i = 1, \dots, n$ . Moreover, in many practical scenarios, obtaining labels for all  $n$  observations of the response variable can be challenging, often involving expensive and lengthy experiments. Therefore, if we can only afford to obtain the responses for a small subset of the data points, a couple of natural question arises: *First, if we estimate the parameter vector  $\boldsymbol{\beta}$  using only this limited subset of data, is it possible to use that estimate to accurately approximate the probabilities of a given class for all the  $n$  instances? Second, what is the minimum sample size required to yield meaningful results?*

### 1.1 Our Contribution

We introduce a randomized sampling-based algorithm for logistic regression with a novel analysis of it, which ensures accurate solutions in terms of the estimated probabilities. Our analysis relies on simple structural conditions that can be reduced to randomized matrix multiplication, a fundamental and well-understood primitive of randomized numerical linear algebra. Our main algorithm (see Algorithm 2) is analyzed in light of the following two structural conditions, which constructs a sampling-based sketching matrix  $\mathbf{S} \in \mathbb{R}^{s \times n}$  (for an appropriate choice of the sketching dimension  $s \ll n$ ), such that for any given vector  $\mathbf{x} \in \mathbb{R}^n$  and accuracy parameter  $0 < \varepsilon < 1$ ,

$$\left| \|\mathbf{U}^\top \mathbf{S}^\top \mathbf{S} \mathbf{x}\|_2 - \|\mathbf{U}^\top \mathbf{x}\|_2 \right| \leq \frac{\varepsilon}{2} \|\mathbf{x}\|_2, \quad (5)$$

$$\|\mathbf{U}^\top \mathbf{S}^\top \mathbf{S} (\mathbf{y} - \mathbf{p}(\boldsymbol{\beta}^*))\|_2 \leq \frac{\varepsilon}{2} \|\mathbf{y} - \mathbf{p}(\boldsymbol{\beta}^*)\|_2 \quad (6)$$

Here,  $\mathbf{U} \in \mathbb{R}^{n \times d}$  contains the left singular vectors of  $\mathbf{X}$ . Indeed, one can use the (exact or approximate) row leverage scores (Mahoney 2011; Mahoney and Drineas 2009) of the matrix  $\mathbf{X}$  (cf. Section 1.3) to satisfy the aforementioned constraints by sampling  $\mathcal{O}(d/\varepsilon^2)$  observations from  $\mathbf{X}$ , in which case  $\mathbf{S}$  is a *sampling-and-rescaling* matrix. Under these structural conditions, the output of Algorithm 2 satisfies

$$\|\mathbf{p}(\hat{\boldsymbol{\beta}}) - \mathbf{p}(\boldsymbol{\beta}^*)\|_2 \leq \varepsilon \|\mathbf{y} - \mathbf{p}(\boldsymbol{\beta}^*)\|_2. \quad (7)$$

In words, our algorithm achieves an approximation bound on the estimated probabilities compared to the estimated probabilities obtained from  $\boldsymbol{\beta}^*$ , which is the MLE based on the full data. Specifically, eqn. (7) can be satisfied by sampling-and-rescaling  $\mathcal{O}\left(\frac{d}{\varepsilon^2}\right)$  rows of  $\mathbf{X}$ . The bound in eqn. (7) depends on  $\|\mathbf{y} - \mathbf{p}(\boldsymbol{\beta}^*)\|_2$  i.e., the goodness-of-fit of the full data model. It measures the overall discrepancy between the actual class labels and the probabilities

assigned by the logistic regression model. A smaller value of  $\|\mathbf{y} - \mathbf{p}(\boldsymbol{\beta}^*)\|_2$  indicates a better fit of the model to the true data labels. Therefore, our bound suggests that our sub-sampled MLE  $\hat{\boldsymbol{\beta}}$  provides better approximations of the estimated probabilities when the full data model is well-suited to the data. This bound in eqn. (7) is highly desirable as it depends on the ability of the full data model to accurately distinguish between different classes. See Section 2 for an important remark on the tightness of our bound. Additionally, Our main result straightforwardly translates into the following relative-error bound in terms of the the overall discrepancy measure:

$$\left| \|\mathbf{y} - \mathbf{p}(\hat{\boldsymbol{\beta}})\|_2 - \|\mathbf{y} - \mathbf{p}(\boldsymbol{\beta}^*)\|_2 \right| \leq \varepsilon \|\mathbf{y} - \mathbf{p}(\boldsymbol{\beta}^*)\|_2. \quad (8)$$

Finally, we summarize our key contributions below:

- The bound in terms of the estimated probabilities constitutes one of our primary contributions (Theorem 1). In addition, it is important not only because it provides a precise approximation of the estimated probabilities, but also because it translates into a relative-error bound in terms of the overall discrepancy (Corollary 2).
- Our second contribution is the obtained sampling complexity due to Algorithm 2. The sampling complexities of the relevant methods, namely, (Munteanu et al. 2018) and (Mai, Musco, and Rao 2021) are  $\tilde{\mathcal{O}}(d^3 \cdot \mu_{\mathbf{y}}(\mathbf{X})^2 / \varepsilon^4)$  and  $\tilde{\mathcal{O}}(d \cdot \mu_{\mathbf{y}}(\mathbf{X})^2 / \varepsilon^2)$  respectively, which depend on the so-called complexity measure  $\mu_{\mathbf{y}}(\mathbf{X})$ , quantifying the difficulty of compressing a dataset for logistic regression. The value of  $\mu_{\mathbf{y}}(\mathbf{X})$  can be substantially large depending on the data. In contrast, our novel analysis eliminates the  $\mu_{\mathbf{y}}(\mathbf{X})^2$  factor and our sampling complexity is  $\mathcal{O}(d/\varepsilon^2)$ , independent of  $\mu_{\mathbf{y}}(\mathbf{X})$ .
- Finally, note that (Munteanu et al. 2018) proposed the so-called L2S method, which is a sampling scheme from a mixture distribution with one component proportional to the *square root of the leverage scores*, a method that significantly differs from conventional leverage score sampling approaches. Similarly, (Mai, Musco, and Rao 2021) utilized a more carefully constructed probability distribution, namely,  $\ell_1$ -*Lewis weights*. Interpreting the impact of a data point based on the above sampling schemes might necessitate additional context and explanation, making them relatively less intuitive for practitioners. In contrast, standard leverage scores are widely used as a sampling tool due to their ease of interpretability. They provide direct, visual, and statistical insights into the importance of individual data points in the model. In this context, our third contribution is that we are the first to employ standard leverage scores sampling in logistic regression and provide strong accuracy guarantees with improved sampling complexity. In addition, the use of leverage score sampling makes our analyses simpler and cleaner as compared to the prior works.

Additionally, we evaluate our algorithm on a variety of real datasets in order to practically assess its performance. In terms of accuracy, Algorithm 2 performs comparably to both (Munteanu et al. 2018; Mai, Musco, and Rao 2021)

and the full-data model. Regarding runtime, we can easily align our method with (Munteanu et al. 2018) by utilizing the fast computation of leverage scores from (Clarkson and Woodruff 2017). See Section 5 for details.

## 1.2 Prior Work

Over the past few decades, randomized numerical linear algebra has strongly advocated for the adoption of sketching and sampling techniques to compress data matrices, providing provable guarantees in various optimization problems, including linear regression (Drineas, Mahoney, and Muthukrishnan 2006), ridge regression (Alaoui and Mahoney 2015; Chowdhury, Yang, and Drineas 2018), low-rank approximation (Sarlos 2006), k-means clustering (Cohen et al. 2015), principal components analysis (Boutsidis, Mahoney, and Drineas 2008), Fisher’s discriminant analysis (Chowdhury, Yang, and Drineas 2019), linear programming (Song and Yu 2021; Chowdhury et al. 2020, 2022; Dexter et al. 2022) and many others. Very recently, there has been a growing interest in applying such sketching techniques to logistic regression problems. In this section, we highlight our contributions in the context of this rapidly growing field of sketching-based algorithms for logistic regression.

Recent works have explored subsampling for logistic regression from statistical viewpoints, employing schemes such as a two-step subsampling approach (Wang, Zhu, and Ma 2018), Poisson subsampling (Wang 2019), and an information-based sampling strategy (Cheng, Wang, and Yang 2020). Similarly, in addressing extreme class imbalance, (Wang 2020; Wang, Zhang, and Wang 2021) examined the randomized undersampling strategy, enhancing model performance by selecting a smaller subset from the majority class. Notably, these efforts focus on optimal strategies in asymptotic scenarios (*i.e.*,  $n \rightarrow \infty$ ) under standard assumptions. In contrast, our work assesses performance in finite data regimes. Importantly, these strategies lack finite sample guarantees and, in most cases, necessitate solving the full-data logistic regression problem for implementation.

The work more closely related to ours in terms of the accuracy bound is (Song and Dai 2022). If  $n$  is extremely large, (Song and Dai 2022) proposed a hybrid sampling scheme based on both randomized and deterministic strategies, and provided non-asymptotic accuracy bounds in terms of the estimated probabilities of the logistic regression. While the deterministic scheme is based on leverage score sampling and primarily follows the two-step algorithm of (Wang, Zhu, and Ma 2018), the randomized strategy relies on the sampling probabilities that implicitly depend on  $\beta^*$ , the full data MLE of the model, which is not very effective in practice. Furthermore, the theoretical bounds provided by (Song and Dai 2022) contain the condition number of the data matrix  $\mathbf{X}$  in the numerator. Therefore, when  $\mathbf{X}$  is ill-conditioned, the condition number can become exceedingly large, resulting in bounds that are relatively imprecise. Finally, the minimum sample size of their sampling methods is determined by  $\beta^*$ , which again poses practical challenges.

In another closely related line of research, as already mentioned in Section 1.1, (Munteanu et al. 2018; Mai, Musco, and Rao 2021) studied the so-called *coresets* for logistic re-

gression and came up with provable bounds using a smaller (and weighted) subset of the original data points of  $\mathbf{X}$  sampled according to carefully constructed probability distributions, such as the so-called  $\ell_1$ -Lewis weights (Cohen and Peng 2015). In particular, (Mai, Musco, and Rao 2021) established the current state-of-the-art  $\varepsilon$ -relative error bounds with  $\tilde{O}(d \cdot \mu_{\mathbf{y}}(\mathbf{X})^2 / \varepsilon^2)$  points, where  $\mu_{\mathbf{y}}(\mathbf{X})$  is a complexity measure of the data matrix  $\mathbf{X}$  and response vector  $\mathbf{y} \in \{-1, 1\}^n$ . Similarly, in more recent works, (Munteanu, Omor, and Woodruff 2021, 2023) introduced data-oblivious, random projection-based sketching methods designed for logistic regression, that came with probabilistic guarantees on the sketched estimate. In  $n \ll d$  regime, (Dexter et al. 2023) recently presented new bounds for coresets construction and dimensionality reduction for logistic regression problem by sketching the feature space. However, it is important to note that in all the aforementioned efforts related to the coresets of logistic regression, the theoretical bounds are defined in terms of the logistic loss function and not directly in relation to the estimated probabilities based on the MLEs of the logistic regression model. In addition, the underlying sampling complexities rely on the previously mentioned complexity measure  $\mu_{\mathbf{y}}(\mathbf{X})$ , a quantity that is contingent on the distribution of the data.

Finally, we refer the reader to the surveys (Woodruff 2014; Mahoney 2011; Drineas and Mahoney 2016, 2018; Martinsson and Tropp 2020) for more background on Randomized Numerical Linear Algebra and its applications.

## 1.3 Notations

We use  $\mathbf{x}, \mathbf{y}, \dots$  to denote vectors and  $\mathbf{X}, \mathbf{Y}, \dots$  to denote matrices. For a matrix  $\mathbf{X}$ ,  $\mathbf{X}_{*i}$  ( $\mathbf{X}_{i*}$ ) denotes the  $i$ -th column (row) of  $\mathbf{X}$  as a column (row) vector and  $\mathbf{X}_{ij}$  is the  $(i, j)$ -th entry of  $\mathbf{X}$ . For a vector  $\mathbf{x}$ ,  $x_i$  is its  $i$ -th entry and  $\|\mathbf{x}\|_2$  denotes its Euclidean norm; for a matrix  $\mathbf{X}$ ,  $\|\mathbf{X}\|_2$  denotes its spectral norm and  $\|\mathbf{X}\|_F$  denotes its Frobenius norm. We refer the reader to (Golub and Van Loan 2012) for properties of norms that will be quite useful in our work. For a matrix  $\mathbf{X} \in \mathbb{R}^{n \times d}$  with  $n > d$  and  $\text{rank}(\mathbf{X}) = d$ , its (*thin*) Singular Value Decomposition (SVD) is the product  $\mathbf{U}\Sigma\mathbf{V}^T$ , with  $\mathbf{U} \in \mathbb{R}^{n \times d}$  (the matrix of the left singular vectors),  $\mathbf{V} \in \mathbb{R}^{d \times d}$  (the matrix of the right singular vectors), and  $\Sigma \in \mathbb{R}^{d \times d}$  a diagonal matrix whose diagonal entries are the non-zero singular values of  $\mathbf{X}$  arranged in non-increasing order. Computation of the SVD takes, in this setting,  $\mathcal{O}(nd^2)$  time. Finally, the row *leverage scores* of  $\mathbf{X}$  are given by  $\|\mathbf{U}_{i*}\|_2^2$  for  $i = 1, 2, \dots, n$ . Additional notation will be introduced as needed.

## 2 Our Approach

### 2.1 Constructing the Sketching Matrix $\mathbf{S}$

We construct the sampling-based sketching matrix  $\mathbf{S}$  using Algorithm 1, which has previously appeared in several prior literature including randomized matrix multiplication (Drineas, Kannan, and Mahoney 2006), linear regression (Drineas, Mahoney, and Muthukrishnan 2006), and many

**Algorithm 1: Construct  $\mathbf{S}$** 

**Input:** Sampling probabilities  $\pi_i$ ,  $i = 1, \dots, n$ , number of sampled indices  $s \ll n$ ;

**Output:** Sampling-and-rescaling matrix  $\mathbf{S} \in \mathbb{R}^{s \times n}$ ;

**Initialize:**  $\mathbf{S} \leftarrow \mathbf{0}_{s \times n}$ ;

- 1: **for**  $i = 1$  **to**  $s$  **do**
- 2:   Pick  $j_i \in \{1, \dots, n\}$  with  $\mathbb{P}(j_i = k) = \pi_k$ ;
- 3:    $\mathbf{S}_{ij_i} \leftarrow (s \pi_{j_i})^{-\frac{1}{2}}$ ;
- 4: **end for**
- 5: **return**  $\mathbf{S}$

**Algorithm 2: Sketched logistic regression**

**Input:** data matrix  $\mathbf{X} \in \mathbb{R}^{n \times d}$ , response vector  $\mathbf{y} \in \{0, 1\}^n$ , sampling-and-rescaling matrix  $\mathbf{S} \in \mathbb{R}^{s \times n}$ ;

**Output:**  $\hat{\beta} \in \mathbb{R}^d$ ,  $\mathbf{p}(\hat{\beta}) \in (0, 1)^n$ ;

- 1: Compute  $\hat{\beta} = \underset{\beta}{\operatorname{argmax}} (\mathbf{y}^\top \mathbf{S}^\top \mathbf{S} \mathbf{X} \beta - \mathbf{1}^\top \mathbf{S}^\top \mathbf{S} \mathbf{g}(\beta))$ ;  
where  $\mathbf{g}(\beta)$  is defined in eqn. (3).
- 2: Compute  $\mathbf{p}(\hat{\beta})$ , with  $p_i(\hat{\beta}) = \frac{\exp(\mathbf{X}_{i*} \hat{\beta})}{1 + \exp(\mathbf{X}_{i*} \hat{\beta})}$
- 3: **return**  $\hat{\beta}$ ,  $\mathbf{p}(\hat{\beta})$ ;

other. In this work, we utilize it in the context of logistic regression, which is why we provide a very brief explanation suitable for general reader.

In Algorithm 1, we construct the sampling-and-rescaling matrix  $\mathbf{S}$  by independently selecting  $s$  elements ( $s \ll n$ ), with replacement, from a set of indices  $\{1, 2, \dots, n\}$ , based on a pre-specified probability distribution  $\{\pi_1, \pi_2, \dots, \pi_n\}$ , where  $0 < \pi_i < 1$  for  $i = 1, 2, \dots, n$ , and  $\sum_{i=1}^n \pi_i = 1$ . If the  $k$ -th independent random trial results in the index  $\ell$ , we assign  $\mathbf{S}_{k\ell} = 1/\sqrt{s\pi_\ell}$ ; otherwise,  $\mathbf{S}_{k\ell} = 0$ , for  $k = 1, \dots, s$  and  $\ell = 1, \dots, n$ .

Now, we outline three key observations about  $\mathbf{S}$  constructed this way using Algorithm 1. First, note that  $\mathbf{S}$  is very sparse, having only one non-zero entry per row, resulting in a total of  $s$  non-zero entries. Second, computing  $\mathbf{S}\mathbf{X}$  is equivalent to selecting  $s$  rescaled rows of  $\mathbf{X}$ , independently and with replacement, according to the same probability distribution  $\{\pi_1, \pi_2, \dots, \pi_n\}$ . Third,  $\mathbf{S}^\top \mathbf{S} \in \mathbb{R}^{n \times n}$  is a diagonal matrix and the  $\ell$ -th diagonal entry of  $\mathbf{S}^\top \mathbf{S}$  is given by  $(\mathbf{S}^\top \mathbf{S})_{\ell\ell} = \frac{L}{s\pi_\ell}$ , where  $L = 0, \dots, s$  denotes the number of times index  $\ell$  is picked up in the sample of size  $s$ . Now, we proceed to our main sampling-based algorithm for logistic regression.

**2.2 Main Algorithm**

Given the sketching matrix  $\mathbf{S}$  constructed using Algorithm 1, our main algorithm (Algorithm 2) is conceptually simple. We first modify the full data log-likelihood function in eqn. (3) by sampling and rescaling  $s$  data points, and the resulting subsampled log-likelihood can be written as

$$\bar{\ell}(\beta) = \mathbf{y}^\top \mathbf{S}^\top \mathbf{S} \mathbf{X} \beta - \mathbf{1}^\top \mathbf{S}^\top \mathbf{S} \mathbf{g}(\beta). \quad (9)$$

Algorithm 2 then maximizes  $\bar{\ell}(\beta)$  and computes the corresponding vector of estimated probabilities with respect to the maximizer  $\hat{\beta}$ . Since,  $\mathbf{S}^\top \mathbf{S}$  is a diagonal matrix (see Section 2.1), we can rewrite  $\bar{\ell}(\beta)$  as  $\bar{\ell}(\beta) = \sum_{i=1}^n (y_i (\mathbf{S}^\top \mathbf{S})_{ii} \mathbf{X}_{i*} \beta - (\mathbf{S}^\top \mathbf{S})_{ii} g_i(\beta))$ . Recall that  $g_i(\beta) = \log(1 + \exp(\mathbf{X}_{i*} \beta))$  is the  $i$ -th entry of  $\mathbf{g}(\beta)$ . Therefore, from the optimality condition  $\frac{\partial \bar{\ell}(\beta)}{\partial \beta} \Big|_{\beta=\hat{\beta}} = \mathbf{0}$ , we have

$$\mathbf{X}^\top \mathbf{S}^\top \mathbf{S} (\mathbf{y} - \mathbf{p}(\hat{\beta})) = \mathbf{0}. \quad (10)$$

Theorem 1 presents our approximation guarantee under the assumption that the sketching matrix  $\mathbf{S}$  satisfies the constraints of eqns. (5) and (6).

**Theorem 1.** *Let  $\mathbf{X} \in \mathbb{R}^{n \times d}$  and  $\mathbf{y} \in \{0, 1\}^n$  be the inputs of the logistic regression problem. Assume that for some constant  $0 < \varepsilon < 1$ , the sketching matrix  $\mathbf{S} \in \mathbb{R}^{s \times n}$  satisfies the structural conditions of eqns. (5) and (6). Then, the estimator  $\hat{\beta}$  returned by Algorithm 2 satisfies*

$$\|\mathbf{p}(\hat{\beta}) - \mathbf{p}(\beta^*)\|_2 \leq \varepsilon \|\mathbf{y} - \mathbf{p}(\beta^*)\|_2.$$

Recall that  $\mathbf{p}(\beta^*)$  is the vector of estimated probabilities from the full data MLE of the logistic regression coefficients.

Further insights are required to better understand the above bound.

*Remark 1.* As mentioned in Section 1.1, the tightness of our bound depends on the performance of the full data model based on  $\beta^*$ . A smaller value of the residual  $\|\mathbf{y} - \mathbf{p}(\beta^*)\|_2$  indicates a better fit of the full data model to the true labels and our bound becomes tighter. In fact, when there is no misclassification in the model with respect to the full data MLE  $\beta^*$ , our bound is even tighter than the relative-error bound. To illustrate this, if the logistic regression model with coefficient vector  $\beta^*$  finds a decision boundary that perfectly separates the two classes, then for all observations with  $y_i = 1$ , we have  $(y_i - p_i(\beta^*)) \leq p_i(\beta^*)$ , and for all  $y_i = 0$ , we trivially have  $(y_i - p_i(\beta^*)) = -p_i(\beta^*)$ . This implies  $\|\mathbf{y} - \mathbf{p}(\beta^*)\|_2^2 \leq \|\mathbf{p}(\beta^*)\|_2^2$ . Therefore, our bound becomes tighter than  $\|\mathbf{p}(\hat{\beta}) - \mathbf{p}(\beta^*)\|_2 \leq \varepsilon \|\mathbf{p}(\beta^*)\|_2$ .

As mentioned in Section 1.1, Theorem 1 further translates into the following relative-error bound in terms of the discrepancy measure.

**Corollary 2.** *Let  $\mathbf{X}, \mathbf{y}, \mathbf{S}$  and  $\varepsilon$  are as defined in Theorem 1. Then, the estimator  $\hat{\beta}$  returned by Algorithm 2 satisfies*

$$\left| \|\mathbf{y} - \mathbf{p}(\hat{\beta})\|_2 - \|\mathbf{y} - \mathbf{p}(\beta^*)\|_2 \right| \leq \varepsilon \|\mathbf{y} - \mathbf{p}(\beta^*)\|_2.$$

*Proof.* We use the lower bound of the triangle inequality:

$$\|\mathbf{p}(\hat{\beta}) - \mathbf{p}(\beta^*)\|_2 \geq \left| \|\mathbf{y} - \mathbf{p}(\hat{\beta})\|_2 - \|\mathbf{y} - \mathbf{p}(\beta^*)\|_2 \right|.$$

Applying this to Theorem 1 yields the desired result.  $\square$

Clearly, our bound serves two purposes simultaneously: it ensures that both our estimated probabilities (Theorem 1) and the degree of misclassification (Corollary 2) are comparable to those of the full data model.

### 3 Proof of Theorem 1

In this section, we will prove Theorem 1. In the proofs, we will use the abbreviations  $\mathbf{p}^*$  for  $\mathbf{p}(\beta^*)$  and  $\hat{\mathbf{p}}$  for  $\mathbf{p}(\hat{\beta})$  to simplify the notation and make it more concise. We remind the reader that  $\mathbf{U} \in \mathbb{R}^{n \times d}$ ,  $\mathbf{V} \in \mathbb{R}^{d \times d}$  and  $\Sigma \in \mathbb{R}^{d \times d}$  are, respectively, the matrices of the left singular vectors, right singular vectors and singular values of  $\mathbf{X}$  in a thin SVD representation. Our first result provides an important identity that will be crucial in proving the final bound.

**Lemma 3.** *Prove that*

$$\mathbf{U}^T \mathbf{S}^T \mathbf{S} (\mathbf{y} - \mathbf{p}(\beta^*)) = \mathbf{U}^T \mathbf{S}^T \mathbf{S} (\mathbf{p}(\hat{\beta}) - \mathbf{p}(\beta^*))$$

*Proof.* We start with the following

$$\begin{aligned} \mathbf{X}^T \mathbf{S}^T \mathbf{S} (\mathbf{y} - \mathbf{p}^*) &= \mathbf{X}^T \mathbf{S}^T \mathbf{S} (\mathbf{y} - \mathbf{p}^* - \hat{\mathbf{p}} + \hat{\mathbf{p}}) \\ &= \mathbf{X}^T \mathbf{S}^T \mathbf{S} (\mathbf{y} - \hat{\mathbf{p}}) + \mathbf{X}^T \mathbf{S}^T \mathbf{S} (\hat{\mathbf{p}} - \mathbf{p}^*) \\ &= \mathbf{X}^T \mathbf{S}^T \mathbf{S} (\hat{\mathbf{p}} - \mathbf{p}^*), \end{aligned} \quad (11)$$

where the last equality directly follows from the fact that  $\mathbf{X}^T \mathbf{S}^T \mathbf{S} (\mathbf{y} - \hat{\mathbf{p}}) = \mathbf{0}$  (from eqn. (10)). Using the thin SVD of  $\mathbf{X}$ , we rewrite eqn. (11) as

$$\mathbf{V} \Sigma \mathbf{U}^T \mathbf{S}^T \mathbf{S} (\mathbf{y} - \mathbf{p}^*) = \mathbf{V} \Sigma \mathbf{U}^T \mathbf{S}^T \mathbf{S} (\hat{\mathbf{p}} - \mathbf{p}^*) \quad (12)$$

The proof follows from pre-multiplying both side of eqn. (12) by  $\Sigma^{-1} \mathbf{V}^T$ , and the fact that  $\mathbf{V}^T \mathbf{V} = \mathbf{I}_d$ .  $\square$

Our next result provides a critical lower bound that is instrumental in bounding  $\|\mathbf{p}(\hat{\beta}) - \mathbf{p}(\beta^*)\|_2$ .

**Lemma 4.** *If the condition in eqn. (5) is satisfied, then*

$$\|\mathbf{U}^T \mathbf{S}^T \mathbf{S} (\mathbf{p}(\hat{\beta}) - \mathbf{p}(\beta^*))\|_2 \geq (1 - \varepsilon/2) \|\mathbf{p}(\hat{\beta}) - \mathbf{p}(\beta^*)\|_2.$$

*Proof.* Assuming  $\hat{\mathbf{p}} \neq \mathbf{p}^*$  (otherwise, we have nothing to prove), we rewrite the left hand side as

$$\|\mathbf{U}^T \mathbf{S}^T \mathbf{S} (\hat{\mathbf{p}} - \mathbf{p}^*)\|_2 = \frac{\|\mathbf{U}^T \mathbf{S}^T \mathbf{S} (\hat{\mathbf{p}} - \mathbf{p}^*)\|_2}{\|\hat{\mathbf{p}} - \mathbf{p}^*\|_2} \cdot \|\hat{\mathbf{p}} - \mathbf{p}^*\|_2 \quad (13)$$

We work on the first term on the right hand side of eqn. (13),

$$\frac{\|\mathbf{U}^T \mathbf{S}^T \mathbf{S} (\hat{\mathbf{p}} - \mathbf{p}^*)\|_2}{\|\hat{\mathbf{p}} - \mathbf{p}^*\|_2} \geq \min_{\mathbf{z} \neq \mathbf{0}} \frac{\|\mathbf{U}^T \mathbf{S}^T \mathbf{S} \mathbf{z}\|_2}{\|\mathbf{z}\|_2} = \frac{\|\mathbf{U}^T \mathbf{S}^T \mathbf{S} \mathbf{z}^*\|_2}{\|\mathbf{z}^*\|_2}, \quad (14)$$

where  $\mathbf{z}^* = \operatorname{argmin}_{\mathbf{z} \neq \mathbf{0}} \frac{\|\mathbf{U}^T \mathbf{S}^T \mathbf{S} \mathbf{z}\|_2}{\|\mathbf{z}\|_2}$ . From eqn. (5), we have

$$\begin{aligned} \|\mathbf{U}^T \mathbf{S}^T \mathbf{S} \mathbf{z}^*\|_2 &\geq \|\mathbf{U}^T \mathbf{z}^*\|_2 - \varepsilon/2 \|\mathbf{z}^*\|_2 \\ \Rightarrow \frac{\|\mathbf{U}^T \mathbf{S}^T \mathbf{S} \mathbf{z}^*\|_2}{\|\mathbf{z}^*\|_2} &\geq \frac{\|\mathbf{U}^T \mathbf{z}^*\|_2}{\|\mathbf{z}^*\|_2} - \frac{\varepsilon}{2} \geq \min_{\mathbf{z} \neq \mathbf{0}} \frac{\|\mathbf{U}^T \mathbf{z}\|_2}{\|\mathbf{z}\|_2} - \frac{\varepsilon}{2} \end{aligned} \quad (15)$$

Combining eqns. (13),(14) and (15), we further have

$$\begin{aligned} \|\mathbf{U}^T \mathbf{S}^T \mathbf{S} (\hat{\mathbf{p}} - \mathbf{p}^*)\|_2 &\geq \left( \min_{\mathbf{z} \neq \mathbf{0}} \frac{\|\mathbf{U}^T \mathbf{z}\|_2}{\|\mathbf{z}\|_2} - \frac{\varepsilon}{2} \right) \|\hat{\mathbf{p}} - \mathbf{p}^*\|_2 \\ &= (\sigma_{\min}(\mathbf{U}^T) - \varepsilon/2) \|\hat{\mathbf{p}} - \mathbf{p}^*\|_2 \\ &= (1 - \varepsilon/2) \|\hat{\mathbf{p}} - \mathbf{p}^*\|_2, \end{aligned}$$

where the first equality follows from the definition of minimum singular value of a matrix and  $\sigma_{\min}(\mathbf{U}^T)$  denotes the minimum singular value of  $\mathbf{U}^T$ , which is equal to one as  $\mathbf{U}$  has orthonormal columns. The proof is now complete.  $\square$

**Proof of Theorem 1.** Combining Lemma 3, Lemma 4, and eqn. (6), we directly have

$$\begin{aligned} \|\hat{\mathbf{p}} - \mathbf{p}^*\|_2 &\leq \frac{\|\mathbf{U}^T \mathbf{S}^T \mathbf{S} (\hat{\mathbf{p}} - \mathbf{p}^*)\|_2}{1 - \varepsilon/2} = \frac{\|\mathbf{U}^T \mathbf{S}^T \mathbf{S} (\mathbf{y} - \mathbf{p}^*)\|_2}{1 - \varepsilon/2} \\ &\leq \frac{\frac{\varepsilon}{2} \|\mathbf{y} - \mathbf{p}^*\|_2}{1 - \varepsilon/2} \leq \varepsilon \|\mathbf{y} - \mathbf{p}^*\|_2, \end{aligned}$$

where the last inequality is due to the fact that  $1 - \varepsilon/2 > 1/2$  as  $0 < \varepsilon < 1$ . This concludes the proof.  $\square$

### 4 Satisfying the Structural Conditions

In this section, we demonstrate how to satisfy the constraints in eqns.(5) and (6) using the sampling-based sketching matrix  $\mathbf{S}$  constructed via Algorithm 1. As space is limited, some of our proofs are deferred to the Appendix. Nevertheless, to offer insights into the mathematical derivations supporting our contributions, we outline the proofs as follows. Also, similar to Section 3, we frequently write  $\mathbf{p}^*$  to denote  $\mathbf{p}(\beta^*)$ . First, we state a fundamental result from the randomized matrix multiplication literature.

**Lemma 5.** *Let  $\mathbf{U} \in \mathbb{R}^{n \times d}$  be the matrix of the left singular vectors of  $\mathbf{X}$ , and  $\mathbf{x} \in \mathbb{R}^n$  be any vector. Furthermore, let  $\mathbf{S} \in \mathbb{R}^{s \times n}$  is constructed using Algorithm 1. Then,*

$$\mathbb{E} \left( \|\mathbf{U}^T \mathbf{S}^T \mathbf{S} \mathbf{x} - \mathbf{U}^T \mathbf{x}\|_2^2 \right) \leq \sum_{i=1}^n \frac{\|\mathbf{U}_{i*}\|_2^2 \cdot x_i^2}{s \pi_i}$$

Lemma 5, with a more general formulation, was originally introduced in (Drineas, Kannan, and Mahoney 2006) where  $\mathbf{U}$  can be any matrix, the vector  $\mathbf{x}$  can be replaced with another matrix, and the left-hand side represents the expectation of squared Frobenius norm. However, for our specific purpose, we narrow it down to  $\mathbf{U}$  being the matrix of the left singular vectors of  $\mathbf{X}$ , and  $\mathbf{x}$  being a vector. For completeness, we prove it in the Appendix<sup>2</sup>.

Next result is a special case of Lemma 5 where the  $\pi_i$ 's i.e., the sampling probabilities are proportional to the row leverage scores of  $\mathbf{X}$ .

**Lemma 6.** *Let matrix  $\mathbf{U}$  and vector  $\mathbf{x}$  are as defined in Lemma 5. If the sketching matrix  $\mathbf{S} \in \mathbb{R}^{s \times n}$  is constructed using Algorithm 1, with sampling probabilities  $\pi_i = \|\mathbf{U}_{i*}\|_2^2 / \|\mathbf{U}\|_F^2$ , for  $i = 1, \dots, n$ . Then,*

$$\mathbb{E} \left( \|\mathbf{U}^T \mathbf{S}^T \mathbf{S} \mathbf{x} - \mathbf{U}^T \mathbf{x}\|_2^2 \right) \leq \frac{d}{s} \|\mathbf{x}\|_2^2$$

Proof of Lemma 6 is immediate from Lemma 5 with  $\pi_i = \frac{\|\mathbf{U}_{i*}\|_2^2}{\|\mathbf{U}\|_F^2}$ , and the fact that  $\|\mathbf{U}\|_F^2 = d$  (because  $\mathbf{U}^T \mathbf{U} = \mathbf{I}_d$ ).

#### 4.1 Sample Complexity

For the condition in eqn. (6), we apply the Markov's inequality with Lemma 6 and use the fact that  $\mathbf{U}^T (\mathbf{y} - \mathbf{p}(\beta^*)) = \mathbf{0}$  (this can be directly derived by applying thin SVD of  $\mathbf{X}$

<sup>2</sup><https://arxiv.org/abs/2402.16326>

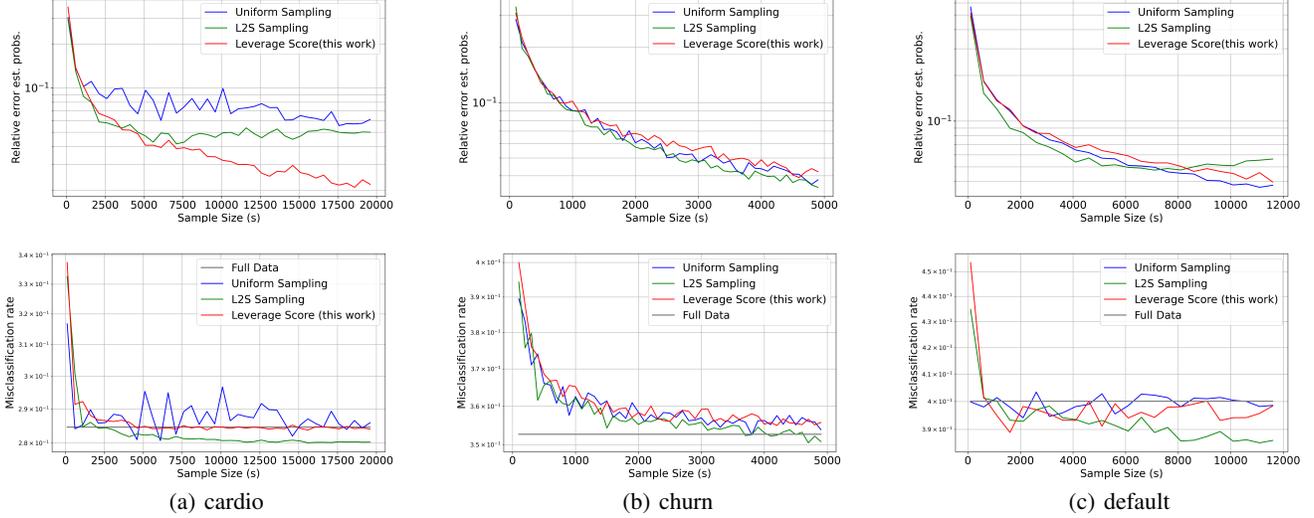


Figure 1: Experiment results on real data: The top row of plots illustrates the relative errors in estimated probabilities and the bottom row shows misclassification rates. Errors are in log-scale.

on eqn. (4) and pre-multiplying the resulting equation by  $\Sigma^{-1}\mathbf{V}^T$ ,

$$\begin{aligned}
 & \mathbb{P}\left(\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}(\mathbf{y} - \mathbf{p}^*)\|_2 \geq \frac{\varepsilon}{2}\|\mathbf{y} - \mathbf{p}^*\|_2\right) \\
 = & \mathbb{P}\left(\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}(\mathbf{y} - \mathbf{p}^*) - \mathbf{U}^T(\mathbf{y} - \mathbf{p}^*)\|_2 \geq \frac{\varepsilon}{2}\|\mathbf{y} - \mathbf{p}^*\|_2\right) \\
 \leq & \frac{4\mathbb{E}\left(\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}(\mathbf{y} - \mathbf{p}^*) - \mathbf{U}^T(\mathbf{y} - \mathbf{p}^*)\|_2^2\right)}{\varepsilon^2\|\mathbf{y} - \mathbf{p}^*\|_2^2} \\
 \leq & \frac{4d\|\mathbf{y} - \mathbf{p}^*\|_2^2}{s\varepsilon^2\|\mathbf{y} - \mathbf{p}^*\|_2^2} = \frac{4d}{s\varepsilon^2}. \quad (16)
 \end{aligned}$$

For the condition in eqn (5), we have from the lower bound of triangle inequality,

$$\begin{aligned}
 & \left|\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{x}\|_2 - \|\mathbf{U}^T\mathbf{x}\|_2\right| \leq \|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{x} - \mathbf{U}^T\mathbf{x}\|_2 \\
 \text{i.e.,} & \left(\left|\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{x}\|_2 - \|\mathbf{U}^T\mathbf{x}\|_2\right| \geq \frac{\varepsilon}{2}\|\mathbf{x}\|_2\right) \text{ implies} \\
 & \left(\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{x} - \mathbf{U}^T\mathbf{x}\|_2 \geq \frac{\varepsilon}{2}\|\mathbf{x}\|_2\right), \\
 \text{i.e.,} & \mathbb{P}\left(\left|\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{x}\|_2 - \|\mathbf{U}^T\mathbf{x}\|_2\right| \geq \frac{\varepsilon}{2}\|\mathbf{x}\|_2\right) \\
 & \leq \mathbb{P}\left(\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{x} - \mathbf{U}^T\mathbf{x}\|_2 \geq \frac{\varepsilon}{2}\|\mathbf{x}\|_2\right) \quad (17)
 \end{aligned}$$

Similar to eqn. (16), applying Markov's inequality on the right hand side of eqn. (17) and using Lemma 6, we have

$$\mathbb{P}\left(\left|\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{x}\|_2 - \|\mathbf{U}^T\mathbf{x}\|_2\right| \geq \frac{\varepsilon}{2}\|\mathbf{x}\|_2\right) \leq \frac{4d}{s\varepsilon^2}. \quad (18)$$

Now, for a failure probability  $0 < \delta < 1$ , if we set the sample size  $s \geq \frac{8d}{\delta\varepsilon^2}$ , eqns. (16) and (18) boil down to

$$\mathbb{P}\left(\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}(\mathbf{y} - \mathbf{p}^*)\|_2 \geq \frac{\varepsilon}{2}\|\mathbf{y} - \mathbf{p}^*\|_2\right) \leq \frac{\delta}{2} \quad (19)$$

$$\mathbb{P}\left(\left|\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{x}\|_2 - \|\mathbf{U}^T\mathbf{x}\|_2\right| \geq \frac{\varepsilon}{2}\|\mathbf{x}\|_2\right) \leq \frac{\delta}{2} \quad (20)$$

Finally, applying the union bound to eqns. (19) and (20), we conclude that if the number of sampled rows  $s$  satisfies

$$s \geq \frac{8d}{\delta\varepsilon^2},$$

then both structural conditions of Theorem 1, namely eqns. (5) and (6) hold with probability at least  $1 - \delta$ .

*Remark 2.* Here, it's worth highlighting that only a single structural condition, namely  $\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}\mathbf{x} - \mathbf{U}^T\mathbf{x}\|_2 \leq \varepsilon\|\mathbf{x}\|_2$ , suffices to establish our bound. Indeed, employing the lower bound of the triangle inequality to the aforementioned constraint leads to the condition presented in eqn. (5). Similarly, the condition described in eqn. (6) can be readily deduced by simply taking  $\mathbf{x} = \mathbf{y} - \mathbf{p}^*$  and utilizing the fact that  $\mathbf{U}^T(\mathbf{y} - \mathbf{p}^*) = \mathbf{0}$  in this context. This way, we can further reduce the constant in the lower bound on sample size  $s$  mentioned above. However, for the sake of clarity for general readers, we break this down into two separate conditions: one for the upper bound and one for the lower bound on  $\|\mathbf{U}^T\mathbf{S}^T\mathbf{S}(\hat{\mathbf{p}} - \mathbf{p}^*)\|_2$ .

**Running Time.** As discussed in Section 1, cost of computing the full data MLE  $\beta^*$  is dominated by the cost of computing the inverse of the Hessian matrix,  $-(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}$ , at each iteration of the IRLS algorithm, which takes  $\mathcal{O}(nd^2)$  time. In contrast, our proposed Algorithm 2 offers a more efficient approach. In our setting, the inverse of the Hessian matrix of the subsampled log-likelihood function  $\bar{\ell}(\beta)$  is given by,  $\left[\frac{\partial^2 \bar{\ell}(\beta)}{\partial \beta \partial \beta^T}\right]^{-1} = -(\mathbf{X}^T\mathbf{W}^{1/2}\mathbf{S}^T\mathbf{S}\mathbf{W}^{1/2}\mathbf{X})^{-1}$ . Recall that  $\mathbf{W} \in \mathbb{R}^{n \times n}$  is a diagonal matrix. Therefore, with our chosen sample size  $s = \mathcal{O}(d/\varepsilon^2)$ , this can be computed in  $\mathcal{O}(\text{nnz}(\mathbf{X}) + d^3/\varepsilon^2)$  time, where  $\text{nnz}(\mathbf{X})$  represents the number of non-zero elements in matrix  $\mathbf{X}$ . Additionally, it's also worth mentioning that approximate leverage scores are sufficient for our purpose. Their computation can be efficiently done without the need to compute  $\mathbf{U}$ , achieving a time com-

plexity of  $\mathcal{O}(\text{nnz}(\mathbf{X}) \log n + d^3 \log^2 d + d^2 \log n)$  due to (Clarkson and Woodruff 2017).

## 5 Empirical Evaluation

**Datasets.** First, we provide a brief introduction to the datasets used in our empirical evaluations. We have applied our algorithm to three distinct real-world datasets. The first dataset, sourced from Kaggle, is the *Cardiovascular disease dataset* (Halder 2020), featuring  $70,000 \times 12$  patient records with a 50% positive case occurrence. This dataset aims to predict the presence of cardiovascular disease. The second dataset, also from Kaggle, is the *Bank customer churn prediction dataset*, containing  $10,000 \times 10$  records with a 20% positive case prevalence, focusing on the classification of customer departure likelihood. The third and final dataset, named the *Default of credit card clients dataset*, is sourced from the UCI ML Repository (Yeh 2016). It consists of  $30,000 \times 24$  records with a 22% positive case ratio and aims to predict the probability of credit card default in the future.

**Comparisons and metrics.** In our experiments, we compare three different sampling schemes: selecting rows (*i*) uniformly at random, (*ii*) proportional to their row leverage scores (this work), and (*iii*) using the L2S method of (Munteanu et al. 2018). For each sampling method, we run Algorithm 2 with varying sample sizes and measure two key metrics which are the most relevant to our analysis, namely, (*i*) the relative error of the estimated probabilities *i.e.*  $\|\hat{\mathbf{p}} - \mathbf{p}^*\|_2 / \|\mathbf{p}^*\|_2$  and (*ii*) the misclassification rates. Each experiment is run 20 times and we report the means of the aforementioned metrics. Notably, we exclude the method from (Mai, Musco, and Rao 2021) for comparison, as they already extensively compared their work with (Munteanu et al. 2018). The performance of (Mai, Musco, and Rao 2021) closely aligns with (Munteanu et al. 2018), with very marginal variations observed for logistic regression. Therefore, we include only (Munteanu et al. 2018) as we do not anticipate any significant differences in performance between ours and (Mai, Musco, and Rao 2021).

**Results.** The first set of results are presented in Figure 1. In the top row, we present relative errors in terms of estimated probabilities. For the first dataset (cardiovascular disease), our sampling approach based on row leverage scores (red) consistently outperforms both L2S (green) and uniform sampling (blue) and our method gets better as the sample size  $s$  increases. For the remaining two datasets (last two columns of Figure 1), both the L2S and uniform sampling methods demonstrate marginally better performance in general, except for the fact that our method gets better than L2S for larger  $s$  in the third column. However, it is noteworthy that the errors in all three methods are consistently very close to each other and become smaller as  $s$  increases. Therefore, the crucial point to note is that our leverage score-based approach indeed works well in practice and demonstrates very comparable results to the other two methods, thereby validating our theoretical bound. In the bottom row of Figure 1, we present a comparison of the misclassification rates. For the first and third datasets with moderate sample sizes, our leverage score-based approach achieves misclassification rates that are either lower than or very close to the

misclassification rate of the full-data model (gray). In contrast, L2S performs slightly better, while uniform sampling performs slightly worse than ours. As for the second dataset (middle column), all three approaches perform comparably, with their respective misclassification rates decreasing and converging to that of the full-data model as  $s$  increases. Overall, we would like to emphasize that our plots are on log scale and if we look at actual numbers on the  $y$ -axis, the difference with (Munteanu et al. 2018) is indeed very small.

For completeness, we also compare our method with respect to the relative-error negative log-likelihoods and due to space constraints, the plots are given in the Appendix<sup>3</sup>. For the same reason, we also postpone some additional experiments in Appendix (*e.g.*, the plots for the standard deviations from the 20 runs for each of the experiments conducted).

Finally, we want to highlight that our experiments are preliminary proof-of-concept showing that our leverage score-based sampling scheme for logistic regression works well in practice and performs very comparably to the prior work. While we use the `numpy.linalg.svd` routine to compute our leverage score-based sampling probabilities, (Munteanu et al. 2018) employed a fast, randomized sketching-based implementation to compute the leverage scores which were subsequently used to calculate the L2S sampling probabilities. This method can also be seamlessly applied in our context for the leverage score computation. Therefore, given the architecture and specific optimization method, running time of our algorithm will be highly comparable to that of (Munteanu et al. 2018).

## 6 Conclusion and Future Work

We have presented a simple structural result to analyze a randomized sampling-based algorithm for the logistic regression problem, guaranteeing highly accurate solutions in terms of both the estimated probabilities and overall discrepancy. There are several immediate future directions that can be explored further. In terms of future research, it is important to explore whether similar bounds can be derived using random projection-based oblivious sketching matrices. This includes exploring techniques like sparse subspace embeddings as presented in (Cohen 2016), very sparse subspace embeddings of (Clarkson and Woodruff 2017), Gaussian sketching matrices, or even combinations of both approaches as outlined in (Cohen, Nelson, and Woodruff 2016). The key challenge is when  $\mathbf{S}$  is a general sketching matrix,  $\mathbf{S}^T \mathbf{S}$  is not necessarily a diagonal matrix. Therefore it is not clear how to formulate an identity similar to eqn. (10) with general sketching matrices. Secondly, it’s worth noting that we have employed the IRLS method in our algorithm as a black-box. Therefore, an obvious future direction would be to further investigate how the errors stemming from the IRLS solver propagate and affect our bound. Lastly, logistic regression finds numerous applications in high-dimensional data scenarios including genomics and bioinformatics, medical diagnostics, image analysis and many more. Hence, exploring similar bounds in high dimensions, *i.e.*, when  $n \ll d$ , would be intriguing.

<sup>3</sup><https://github.com/AgnivaC/SubsampledLogisticRegression>

## Acknowledgments

We would like to thank the anonymous reviewers for their helpful comments. Special thanks to Petros Drineas for the insightful discussions. This work was supported by the Artificial Intelligence Initiative as part of the Laboratory Directed Research and Development (LDRD) Program at the Oak Ridge National Laboratory (ORNL). ORNL is operated by UT-Battelle, LLC., for the U.S. Department of Energy under Contract DE-AC05-00OR22725. This manuscript has been authored by UT-Battelle, LLC. The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (<http://energy.gov/downloads/doe-public-access-plan>).

## References

- Alaoui, A.; and Mahoney, M. W. 2015. Fast randomized kernel ridge regression with statistical guarantees. In *Advances in Neural Information Processing Systems*, 775–783.
- Boutsidis, C.; Mahoney, M. W.; and Drineas, P. 2008. Unsupervised Feature Selection for Principal Components Analysis. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 61–69.
- Chai, H.; Liang, Y.; Wang, S.; and Shen, H.-w. 2018. A novel logistic regression model combining semi-supervised learning and active learning for disease classification. *Scientific reports*, 8(1): 13009.
- Cheng, Q.; Wang, H.; and Yang, M. 2020. Information-based optimal subdata selection for big data logistic regression. *Journal of Statistical Planning and Inference*, 209: 112–122.
- Chowdhury, A.; Dexter, G.; London, P.; Avron, H.; and Drineas, P. 2022. Faster randomized interior point methods for tall/wide linear programs. *The Journal of Machine Learning Research*, 23(1): 15156–15203.
- Chowdhury, A.; London, P.; Avron, H.; and Drineas, P. 2020. Faster randomized infeasible interior point methods for tall/wide linear programs. *Advances in Neural Information Processing Systems*, 33: 8704–8715.
- Chowdhury, A.; Yang, J.; and Drineas, P. 2018. An Iterative, Sketching-based Framework for Ridge Regression. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, 988–997.
- Chowdhury, A.; Yang, J.; and Drineas, P. 2019. Randomized Iterative Algorithms for Fisher Discriminant Analysis. In *Proceedings of The 35th Uncertainty in Artificial Intelligence Conference*, volume 115, 239–249.
- Clarkson, K. L.; and Woodruff, D. P. 2017. Low-rank approximation and regression in input sparsity time. *Journal of the ACM (JACM)*, 63(6): 54.
- Cohen, M. B. 2016. Nearly tight oblivious subspace embeddings by trace inequalities. In *Proceedings of the 27th Annual ACM-SIAM Symposium on Discrete Algorithms*, 278–287.
- Cohen, M. B.; Elder, S.; Musco, C.; Musco, C.; and Persu, M. 2015. Dimensionality Reduction for  $k$ -Means Clustering and Low Rank Approximation. In *Proceedings of the 47th Annual ACM Symposium on Theory of Computing*, 163–172.
- Cohen, M. B.; Nelson, J.; and Woodruff, D. P. 2016. Optimal Approximate Matrix Product in Terms of Stable Rank. In *43rd International Colloquium on Automata, Languages, and Programming*, 11:1–11:14.
- Cohen, M. B.; and Peng, R. 2015. Lp row sampling by lewis weights. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, 183–192.
- Dexter, G.; Chowdhury, A.; Avron, H.; and Drineas, P. 2022. On the convergence of inexact predictor-corrector methods for linear programming. In *International Conference on Machine Learning*, 5007–5038. PMLR.
- Dexter, G.; Khanna, R.; Raheel, J.; and Drineas, P. 2023. Feature Space Sketching for Logistic Regression. *arXiv preprint arXiv:2303.14284*.
- Dong, Q.; Zhu, X.; and Gong, S. 2019. Single-label multi-class image classification by deep logistic regression. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 3486–3493.
- Drineas, P.; Kannan, R.; and Mahoney, M. W. 2006. Fast Monte Carlo Algorithms for Matrices I: Approximating Matrix Multiplication. *SIAM Journal on Computing*, 36(1): 132–157.
- Drineas, P.; and Mahoney, M. W. 2016. RandNLA: Randomized Numerical Linear Algebra. *Communications of the ACM*, 59: 80–90.
- Drineas, P.; and Mahoney, M. W. 2018. *Lectures on randomized numerical linear algebra*, volume 25 of *The Mathematics of Data, IAS/Park City Mathematics Series*. American Mathematical Society.
- Drineas, P.; Mahoney, M. W.; and Muthukrishnan, S. 2006. Sampling Algorithms for  $\ell_2$  Regression and Applications. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms*, 1127–1136.
- Feng, J.; Xu, H.; Mannor, S.; and Yan, S. 2014. Robust logistic regression and classification. *Advances in neural information processing systems*, 27.
- Golub, G. H.; and Van Loan, C. F. 2012. *Matrix computations*, volume 3. Johns Hopkins University Press.
- Green, P. J. 1984. Iteratively Reweighted Least Squares for Maximum Likelihood Estimation, and Some Robust and Resistant Alternatives. *Journal of the Royal Statistical Society: Series B (Methodological)*, 46(2): 149–170.
- Halder, R. K. 2020. Cardiovascular Disease Dataset. Kaggle Dataset. DOI: <https://dx.doi.org/10.21227/7qm5-dz13>.
- Hendrycks, D.; Mazeika, M.; and Dietterich, T. 2019. Deep Anomaly Detection with Outlier Exposure. In *International Conference on Learning Representations*.

- Hosmer Jr, D. W.; Lemeshow, S.; and Sturdivant, R. X. 2013. *Applied logistic regression*, volume 398. John Wiley & Sons.
- Koh, K.; Kim, S.-J.; and Boyd, S. 2007. A Method for large-scale  $\ell_1$ -regularized logistic regression. In *AAAI*, 565–571.
- Kull, M.; Perello Nieto, M.; Kängsepp, M.; Silva Filho, T.; Song, H.; and Flach, P. 2019. Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration. *Advances in neural information processing systems*, 32.
- Liao, J.; and Chin, K.-V. 2007. Logistic regression for disease classification using microarray data: model selection in a large p and small n case. *Bioinformatics*, 23(15): 1945–1951.
- Mahoney, M. W. 2011. Randomized Algorithms for Matrices and Data. *Foundations and Trends in Machine Learning*, 3(2): 123–224.
- Mahoney, M. W.; and Drineas, P. 2009. CUR matrix decompositions for improved data analysis. *Proceedings of the National Academy of Sciences of the United States of America*, 106(3): 697–702.
- Mai, T.; Musco, C.; and Rao, A. 2021. Coresets for classification—simplified and strengthened. *Advances in Neural Information Processing Systems*, 34: 11643–11654.
- Martinsson, P.-G.; and Tropp, J. A. 2020. Randomized numerical linear algebra: Foundations and algorithms. *Acta Numerica*, 29: 403–572.
- Munteanu, A.; Omlor, S.; and Woodruff, D. 2021. Oblivious sketching for logistic regression. In *International Conference on Machine Learning*, 7861–7871. PMLR.
- Munteanu, A.; Omlor, S.; and Woodruff, D. 2023. Almost Linear Constant-Factor Sketching for  $\ell_1$  and Logistic Regression. In *The Eleventh International Conference on Learning Representations*.
- Munteanu, A.; Schwiegelshohn, C.; Sohler, C.; and Woodruff, D. 2018. On coresets for logistic regression. *Advances in Neural Information Processing Systems*, 31.
- Rosario, D. S. 2004. Highly effective logistic regression model for signal (anomaly) detection. In *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 5, V–817. IEEE.
- Sarlos, T. 2006. Improved approximation algorithms for large matrices via random projections. In *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*, 143–152. IEEE.
- Song, Y.; and Dai, W. 2022. Deterministic subsampling for logistic regression with massive data. *Computational Statistics*, 1–24.
- Song, Z.; and Yu, Z. 2021. Oblivious Sketching-based Central Path Method for Linear Programming. In *International Conference on Machine Learning*, 9835–9847. PMLR.
- Wang, H. 2019. More efficient estimation for logistic regression with optimal subsamples. *Journal of machine learning research*, 20.
- Wang, H. 2020. Logistic regression for massive data with rare events. In *International Conference on Machine Learning*, 9829–9836. PMLR.
- Wang, H.; Zhang, A.; and Wang, C. 2021. Nonuniform Negative Sampling and Log Odds Correction with Rare Events Data. *Advances in Neural Information Processing Systems*, 34.
- Wang, H.; Zhu, R.; and Ma, P. 2018. Optimal subsampling for large sample logistic regression. *Journal of the American Statistical Association*, 113(522): 829–844.
- Woodruff, D. P. 2014. Sketching as a Tool for Numerical Linear Algebra. *Foundations and Trends in Theoretical Computer Science*, 10(1-2).
- Yeh, I.-C. 2016. Default of credit card clients. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C55S3H>.
- Zhang, L.; Jin, R.; Chen, C.; Bu, J.; and He, X. 2012. Efficient online learning for large-scale sparse kernel logistic regression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 26, 1219–1225.