Communication-Efficient Distributed Maximum Likelihood Estimation with the Optimal Weighted Average

Abstract

We present a distributed algorithm for maximum likelihood estimation (MLE) with low communication cost. In our algorithm, each machine in a cluster first independently solves the MLE problem on a subset of the data; a second round of optimization then finds the optimal weighted average of the parameter vectors. We show analytically that this method reduces both the bias and variance of the estimator when compared to the estimator trained by a single machine. (The straight average is known to reduce only the variance.) Our analysis relies on simple properties of sub-Gaussian random vectors. It is therefore simpler and more general than the analysis of similar distributed algorithms. Notably, we do not assume (as do all previous analyses) that the likelihood function is concave or that any quantities are bounded. A major practical advantage of our method is that it is robust to the amount of regularization, which speeds up model selection.

1. INTRODUCTION

Many modern datasets are too large to fit in the memory of a single machine, so they must be partitioned onto many machines. To analyze these datasets, we need distributed algorithms. Existing distributed algorithms can be classified as either interactive or non-interactive depending on their communication complexity. In this paper we propose an algorithm that exhibits the benefits of both types.

Interactive algorithms require many rounds of communication between machines. These algorithms often resemble standard iterative algorithms where each iteration is followed by a communication step. The appeal of interactive algorithms is that they enjoy the same statistical regret bounds as standard sequential algorithms. But, there are two downsides. First, these algorithms can be too slow in practice because communication is the main bottleneck in modern distributed architectures. Second, these algorithms require special implementations and do not work with off-the-shelf statistics libraries provided by (for example) Python, R, and Matlab.

Non-interactive algorithms require only a single round of communication. They are significantly faster than interactive algorithms and easily implemented with standard libraries. The downside is worse regret bounds. Recent work (discussed in Section 3.2) has shown that no non-interactive algorithm can achieve regret bounds comparable to an interactive one.

In this paper, we propose a *semi-interactive* distributed algorithm called *optimal weighted averaging* (OWA). Our algorithm performs two rounds of communication, so it is not subject to the existing regret bounds of non-interactive algorithms. The algorithm has two tunable parameters that let the user trade better statistical performance for worse communication complexity. The OWA algorithm is easily implemented in a MapReduce architecture with standard packages.

In the next section, we formally describe our OWA algorithm. In Section 3, we compare OWA to existing distributed algorithms. We highlight how the analysis of existing algorithms requires more limiting assumptions than our own, and show in detail why existing non-interactive regret bounds do not apply to OWA. Section 4 shows that OWA's regret bounds interpolate between the averaging estimator's regret and the optimal regret. As part of the analysis, we provide novel, more general regret bounds for the averaging estimator. Section 5 shows experimentally that our algorithm performs well. We emphasize that our algorithm is robust to the strength of regularization, which is one of the reasons it performs well in practice.

2. THE OWA ALGORITHM

This section first formally introduces the problem of communication efficient distributed estimation, then describes our proposed OWA distributed estimator.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

2.1. Problem Setting

Let $\mathcal{Y} \subseteq \mathbb{R}$ be the space of response variables, $X \subseteq \mathbb{R}^d$ be the space of covariates, and $\Theta \subseteq \mathbb{R}^d$ be the parameter space. We assume a linear model. The log-likelihood of data point $(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$ given the model's true parameter $\theta^* \in \Theta$ is denoted by $f(y, \mathbf{x}^\mathsf{T}\theta^*)$. Our analysis in Section 4 places very mild restrictions on f. In particular, f need not be concave with respect to θ. Let $Z \subset \mathcal{X} \times \mathcal{Y}$ be a dataset of mn i.i.d. observations. Finally, let $R : \Theta \to \mathbb{R}$ be a regularization function (typically the L1 or L2 norm) and $\lambda \in \mathbb{R}$ be the regularization strength. Then the regularized maximum likelihood estimator (MLE) is

$$\hat\theta^{mle} = \arg\max_\theta \sum_{(\mathbf{x},y)\in Z} f(y, \mathbf{x}^\mathsf{T}\theta) + \lambda R(\theta). \tag{1}$$

In the remainder of this paper, it should be understood that all MLEs are regularized.

Assume that Z has been partitioned onto m machines so that each machine i has dataset Z_i of size n, and all the Z_i are disjoint. Then each machine calculates the local MLE

$$\hat\theta_i^{mle} = \arg\max_\theta \sum_{(\mathbf{x},y)\in Z_i} f(y, \mathbf{x}^\mathsf{T}\theta) + \lambda R(\theta). \tag{2}$$

Solving for $\hat\theta_i^{mle}$ requires no communication with other machines. Our goal is to merge the $\hat\theta_i^{mle}$ into a single improved estimate. A baseline merging procedure is the averaging estimator

$$\hat\theta^{ave} = \frac{1}{m}\sum_{i=1}^m \hat\theta_i^{mle}. \tag{3}$$

This estimator is well studied, and in Section 3 we compare this previous work to our own. Here we briefly recall that an estimator's error $\|\theta^* - \hat\theta^{ave}\|$ can be decomposed into bias $\|\theta^* - \mathbb{E}\hat\theta^{ave}\|$ and variance $\|\mathbb{E}\hat\theta^{ave} - \hat\theta^{ave}\|$ components. The $\hat\theta^{ave}$ estimator is known to have lower variance than the estimator $\hat\theta_i^{mle}$ trained on a single machine, but the same bias. Our goal is to design an estimator that reduces both variance and bias.

2.2. Proposed Solution

We propose a modification to the averaging estimator called the *optimal weighted average* (OWA) that reduces both variance and bias. OWA uses a second round of optimization to calculate the optimal linear combination of the $\hat\theta_i^{mle}$s. This second optimization occurs over a small fraction of the dataset, so its computational and communication overhead is negligible.

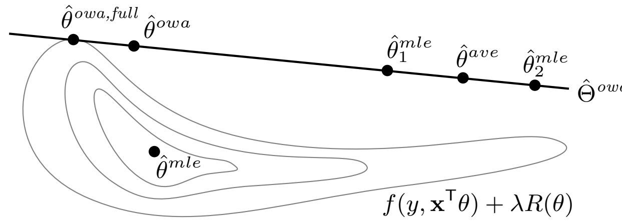

Figure 1. Our method performs a second round of optimization to find the best parameter vector in $\hat\Theta^{owa}$. Since $\hat\Theta^{owa}$ has low dimension, we can use relatively few data points in the second round of optimization to ensure that with high probability $\hat\theta^{owa}$ has higher empirical likelihood than $\hat\theta^{ave}$.

To motivate our estimator, we first present an estimator that uses the entire dataset for the second round of optimization. Define the matrix $\hat W : \mathbb{R}^{d\times m}$ to have ith column equal to $\hat\theta_i^{mle}$. Now consider the estimator

$$\hat\theta^{owa,full} = \hat W \hat\alpha^{full}, \tag{4}$$

where

$$\hat\alpha^{full} = \arg\max_\alpha \sum_{(\mathbf{x},y)\in Z} f\left(y, \mathbf{x}^\mathsf{T}\hat W\alpha\right) + \lambda R(\hat W\alpha). \tag{5}$$

Notice that $\hat\theta^{owa,full}$ is the maximum likelihood estimator when the parameter space Θ is restricted to $\hat\Theta^{owa} = \operatorname{span}\{\hat\theta_i^{mle}\}_{i=1}^m$. In other words, the $\hat\alpha^{full}$ vector contains the optimal weights to apply to each $\hat\theta_i^{mle}$ when averaging. Figure 1 shows graphically that no other estimator in $\hat\Theta^{owa}$ can have higher regularized empirical likelihood than $\hat\theta^{owa,full}$.

Calculating the weights $\hat\alpha^{full}$ directly is infeasible because it requires access to the full dataset. Fortunately, we do not need to consider all the data points for an accurate estimator. The parameter space $\hat\Theta^{owa}$ is m-dimensional. So intuitively, we only need $O(m)$ data points to solve the second optimization to our desired accuracy. This intuition motivates the OWA estimator. Let $Z_i^{owa} \subset Z_i$ be a set of n^{owa} data points uniformly sampled from Z_i without replacement, and let Z^{owa} be the union of the Z_i^{owa}s. Then the OWA estimator is defined as

$$\hat\theta^{owa} = \hat W\hat\alpha, \tag{6}$$

where

$$\hat\alpha = \arg\max_\alpha \sum_{(\mathbf{x},y)\in Z^{owa}} f\left(y, \mathbf{x}^\mathsf{T}\hat W\alpha\right) + \lambda R(\hat W\alpha). \tag{7}$$

Algorithm **??** shows the steps for efficiently computing $\hat\theta^{owa}$. In the first round, each machine calculates $\hat\theta_i^{mle}$

independently and broadcasts the result to every other machine. Since the parameter vector has d dimensions and there are m machines, a total of $O(md)$ bits are transmitted. In the second round, each machine projects its local dataset Z_i^{owa} onto the space $\hat{\Theta}^{owa}$. These projected data points are then transmitted to a predesignated master machine. The projected data points each have dimension m, so a total of $O(m^2 n^{owa})$ bits are transmitted. The master machine now has all of the information to complete the optimization. In total, $O(md + m^2 n^{owa})$ bits were transmitted. When $d < mn^{owa}$, this is the same order of bits as the averaging estimator.

Equations 5 and 7 cannot be solved directly using off the shelf optimizers because existing optimizers do not support the non-standard regularization term $R(\hat{W}\alpha)$. In practice, it is sufficient to approximate this regularization by L2 regularization directly on the α vector:

$$\lambda R(\hat{W}\alpha) \approx \lambda_2 \|\alpha\|. \tag{8}$$

Intuitively, this is because even when we want the parameter vector θ to be sparse (and so are regularizing by $R =$ the L1 norm), we have no reason to believe that the α vector should be sparse. The desired sparsity is induced by the regularization when solving for $\hat{\theta}_i^{mle}$s and maintained in any linear combination of the $\hat{\theta}_i^{mle}$s. The new λ_2 regularization parameter should be set by cross validation. This will be a fast procedure, however, because there are few data points to optimize over, and the L2 regularized problem is much easier to solve than the L1 problem. With this minor modification, our distributed estimator can be implemented using any existing optimizer.

3. RELATED WORK

Related work can be divided into two categories: alternative estimators, and bounds on the communication complexity of distributed learning.

3.1. Alternative estimators

The simplest and most popular communication-efficient estimator is the averaging estimator

$$\hat{\theta}^{ave} = \frac{1}{m} \sum_{i=1}^{m} \hat{\theta}_i^{mle}. \tag{9}$$

Previous analysis of $\hat{\theta}^{ave}$ makes a number of limiting assumptions. (McDonald et al., 2009) analyze $\hat{\theta}^{ave}$ in the special case of L2 regularized maximum entropy models. They provide tail bounds on $\|\theta^* - \hat{\theta}^{ave}\|$, showing that the variance $\|\mathbb{E}\hat{\theta}^{ave} - \hat{\theta}^{ave}\|$ reduces as $O((nm)^{-1/2})$, but they do not show a reduction in bias. Their analysis uses a martingale technique that requires the radius of the dataset be independent of the size of the dataset. This is a particularly limiting assumption as even the simple case of normally-distributed data does not satisfy it. (Zhang et al., 2012) provide a more general analysis showing that the mean squared error (MSE) $\mathbb{E}\|\theta^* - \hat{\theta}^{ave}\|^2$ decays as $O((nm)^{-1} + n^{-2})$. This matches the optimal MSE of $\hat{\theta}^{mle}$ whenever $m < n$. Their analysis still requires a number of technical assumptions. For example, they assume the parameter space Θ is bounded. This assumption does not hold under the standard Bayesian interpretation of L2 regularization as a Gaussian prior of the parameter space. They further make strong convexity and $8th$ order smoothness assumptions which guarantee that $\hat{\theta}_i^{mle}$ is a "nearly unbiased estimator" of θ^*. In Section 4, we provide a simpler analysis of $\hat{\theta}^{ave}$ that relaxes these assumptions. In particular, our analysis does not require any quantities to be bounded or the likelihood to be concave.

Other research has focused on modifications to the $\hat{\theta}^{ave}$ estimator to reduce bias. (Zinkevich et al., 2010) show that if the training sets partially overlap each other (instead of being disjoint), then the resulting estimator will have lower bias. (Zhang et al., 2012) also provide a bootstrap average estimator, which works as follows. Let $r \in (0,1)$, and Z_i^r be a bootstrap sample of Z_i of size rn. Then the bootstrap average estimator is

$$\hat{\theta}^{boot} = \frac{\hat{\theta}^{ave} - r\hat{\theta}^{ave,r}}{1-r}, \tag{10}$$

where

$$\begin{aligned} \hat{\theta}^{ave,r} &= \frac{1}{m}\sum_{i=1}^{m} \hat{\theta}_i^{mle,r}, \\ \hat{\theta}_i^{mle,r} &= \arg\max_{\theta} \sum_{(\mathbf{x},y)\in Z_i^r} f(y, \mathbf{x}^\mathsf{T}\theta) + \lambda R(\theta). \end{aligned} \tag{11}$$

The intuition behind this estimator is to use the bootstrap sample to directly estimate and correct for the bias. This estimator enjoys a MSE that decays as $O((nm)^{-1} + n^{-3})$ under similar assumptions as their analysis of $\hat{\theta}^{ave}$. There are two main limitations to $\hat{\theta}^{boot}$. First, the optimal value of r is not obvious and setting the parameter requires cross validation on the entire data set. Our proposed $\hat{\theta}^{owa}$ estimator has a similar parameter λ_2 that needs tuning, but this tuning happens on a small fraction of the data and always with the L2 regularizer. So properly tuning λ_2 is more efficient than r. Second, performing a bootstrap on an unbiased estimator increases the variance. This means that $\hat{\theta}^{boot}$ could perform worse than $\hat{\theta}^{ave}$ on unbiased estimators. Our $\hat{\theta}^{owa}$ estimator, in contrast, will perform at least as well as $\hat{\theta}^{ave}$ with high probability as seen in Figure 1. In Section 5, we show that our estimator has better empirical performance.

(Liu and Ihler, 2014) propose a more Bayesian approach inspired by (Merugu and Ghosh, 2003). Instead of averaging the model's parameters, they directly "average the models" with the following KL-average estimator:

$$\hat{\theta}^{kl} = \arg\min_{\theta} \sum_{i=1}^{m} \text{KL}\left(p(\cdot;\hat{\theta}_i^{mle}) \,\middle\|\, p(\cdot;\theta)\right). \tag{12}$$

The minimization is performed via a bootstrap sample from the smaller models. This method has two main advantages. First, it is robust to reparameterizations of the model. Second, it is statistically optimal for the class of non-interactive optimization methods. (We show in the next section that this optimality bound does not apply to our $\hat{\theta}^{owa}$ estimator due to our semi-interactive setting.) The main downside of the KL-average is that the minimization has a prohibitively high computational cost. Let n^{kl} be the size of the bootstrap sample. Then the original implementation's MSE shrinks as $O((nm)^{-1} + (nn^{kl})^{-1})$. This implies that the bootstrap procedure requires as many samples as the original problem to get a MSE that shrinks at the same rate as the averaging estimator. (Han and Liu, 2016) show a method to reduce this rate to $O((nm)^{-1} + (n^2n^{kl})^{-1})$ using control variates, but the procedure remains prohibitively expensive. Their experiments show the procedure scaling only to datasets of size $nm \approx 10^4$, whereas our experiments involve a dataset of size $nm \approx 10^8$.

Surprisingly, (Zhang et al., 2013b) show that in the special case of kernel ridge regression, a reduction in bias is not needed to have the MSE of $\hat{\theta}^{ave}$ decay at the optimal sequential rate. By a careful choice of regularization parameter, they cause $\hat{\theta}_i^{mle}$ to have lower bias but higher variance, so that the final estimate of $\hat{\theta}^{ave}$ has both reduced bias and variance. This suggests that a merging procedure that reduces bias is not crucial to good performance if we set the regularization parameter correctly. Typically there is a narrow range of good regularization parameters, and finding a λ in this range is expensive computationally. We show experimentally in Section 5 that our method has significantly reduced sensitivity to λ. Therefore, it is computationally cheaper to find a good λ for our method than for the other methods discussed in this section.

3.2. Performance bounds

Performance bounds come in two flavors: statistical and information theoretic. On the statistical side, (Liu and Ihler, 2014) show that for any non-interactive distributed estimator $\hat{\theta}^q$, the quantity $\|\hat{\theta}^q - \hat{\theta}^{mle}\|^2$ decays as $\Omega(\gamma_{\theta^*}^2 \mathcal{I}_{\theta^*}^{-1}/n^2)$. Here γ_{θ^*} is the statistical curvature of the model and $\mathcal{I}_{\theta^*}$ is the Fisher information. Furthermore, they show that $\hat{\theta}^{kl}$ matches this bound. This bound is not relevant for our $\hat{\theta}^{owa}$ estimator because of our semi-interactive setting. A crucial assumption of Liu and Ihler's analysis is that the merge function not depend on the data.

(Shamir, 2014), (Zhang et al., 2013a), and (Garg et al., 2014) all provide information theoretic lower bounds on the sample complexity of non-interactive learning problems. As above, however, their results are not applicable in our semi-interactive setting. There is one information theoretic lower bound that does apply to us. Let the true parameter vector θ^* be k-sparse. That is, $\|\theta^*\|_0 \le k$. Surprisingly, (Braverman et al., 2015) show that the minimax optimal error rate for least squares regression requires $\Omega(m \cdot \min\{n, d\})$ bits of communication (independent of k) even in the fully interactive setting. This is important because sparsity does not reduce the amount of communication required, and this bound does apply in our setting.

4. ANALYSIS

In this section, we analyze the statistical performance of our $\hat{\theta}^{owa}$ estimator. As part of the analysis, we provide a novel proof of the statistical performance of $\hat{\theta}^{ave}$ with an easier to interpret constant factor. Our analysis is simpler and more general than previous analyses of distributed estimators. In particular, we do not assume that any random variables are bounded or that our likelihood function is concave.

4.1. Conditions

The only condition of our analysis is that the estimation error has sub-Gaussian tails. We first state this condition formally, then explain why this condition is mild.

Definition 1. *We say that a random vector* $\mathbf{x}$ *is* sub-Gaussian *with variance proxy* σ^2 *if it obeys the following concentration bound. Let* $t > 0$*, then with probability at least* $1 - \exp(-\sigma^2 t^2/2)$*,* $\|\mathbf{x}\| < t$*.*

Note in particular that if $\mathbf{x}$ is a Gaussian random vector with mean μ and covariance Σ, then $\Sigma^{1/2}(\mathbf{x} - \mu)$ is sub-Gaussian with $\sigma^2 = 1$. Sub-Gaussian random vectors have recently become an important tool in the analysis of high dimensional statistics. (Vershynin, 2012) provides an accessible tutorial of these results. We now state our condition.

The Sub-Gaussian Tail (SGT) Condition. *Let θ be an estimator trained on n data points. Then the random vector*

$$\Delta_\theta = \sqrt{n}\mathcal{I}_{\theta^*}^{1/2}(\theta - \mathbb{E}\theta) \tag{13}$$

is sub-Gaussian for some σ^2. Above, $\mathcal{I}_{\theta^*}$ is the positive definite Fisher information matrix at the parameter vector's true value θ^*.

The SGT condition is mild and known to hold in many situations of interest. In the asymptotic regime when $n \to \infty$, very strong results are known. Theorem 7.5.2 of (Lehmann, 1999) is a simple example that shows Δ_θ is an isotropic centered Gaussian (and hence sub-Gaussian). Lehman's theorem requires only that f be three times differentiable, the data points be i.i.d., and some mild identifiability conditions. More sophisticated analyses show that the SGT Condition holds very generally. For example (Spokoiny, 2012) shows that Δ_θ is normally distributed even when the data points are correlated.

Similar results hold in the non-asymptotic case $n < \infty$, but known results are less strong. Most non-asymptotic results of this form require that the data points also be sub-Gaussian. For example, (Negahban et al., 2009) considers the case when the data points are sub-Gaussian, the likelihood satisfies a "restricted strong convexity condition," and the regularizer is decomposable. More recently, (Sivakumar et al., 2015) showed the SGT Condition holds when the data are only sub-exponential. The strongest non-asymptotic results on the estimation error known to the authors are due to (Spokoiny, 2012). Spokoiny does not require any conditions on the data, but shows that the SGT Condition is satisfied only up to $t < O(n)$.

Our analysis below requires that $\hat\theta_i^{mle}$ satisfy the SGT Condition. This is strictly more general than the conditions in previous work. (Zhang et al., 2012) for example require that the parameter space Θ be bounded (in addition to other moment conditions). A bounded parameter space automatically implies that $\hat\theta_i^{mle}$ satisfies the SGT Condition because every bounded random variable is sub-Gaussian by definition.

4.2. Analysis of $\hat\theta^{ave}$

We provide a simple bound that shows that averaging improves the variance, but not the bias of an estimator. Similar bounds are well known (see Section 3.1), but our analysis has the following advantages: It requires fewer assumptions, has a simpler proof, and has an easy to interpret constant factor.

Theorem 1. *Assume $\hat\theta_i^{mle}$ satisfies the SGT Condition. Let $t > 0$. Then with probability at least $1 - \exp(-t)$,*

$$\|\theta^* - \hat\theta^{ave}\| \le \|\theta^* - \mathbb{E}\hat\theta_i^{mle}\| + \sqrt{\frac{v_t}{mn}} \tag{14}$$

where

$$v_t = \sigma^2\left(\operatorname{tr}\mathcal{I}_{\theta^*}^{-1} + 2\sqrt{\operatorname{tr}\left(\mathcal{I}_{\theta^*}^{-2}\right)t} + 2\|\mathcal{I}_{\theta^*}^{-1}\|t\right) \tag{15}$$

Note that this reduction in variance is essentially optimal. If we assume that $\hat\theta^{mle}$ satisfies the SGT condition (in most cases this follows from $\hat\theta_i^{mle}$ satisfying the SGT condition), then we are assuming that the variance of $\hat\theta^{mle}$ reduces at the same $O(1/\sqrt{mn})$ rate.

4.3. Analysis of $\hat\theta^{owa}$

Unlike the $\hat\theta^{ave}$ estimator, $\hat\theta^{owa}$ reduces both bias and variance. The m and n^{owa} parameters act as "knobs" that let us trade off lower statistical error for higher communication cost. Before we present the formal analysis, we present a motivating informal analysis.

4.3.1. Informal Analysis

Our main result is captured in the following theorem.

Theorem 2 (Informal). *With high probability, we have that*

$$\|\theta^* - \hat\theta^{owa}\| \le O\left(\sqrt{\frac{1}{mn^{owa}}} + \sqrt{\left(1 - \frac{m}{d}\right)}\|\theta^* - \hat\theta^{ave}\|\right)$$

The rightmost term above captures the error of the optimal parameter vector constrained to $\hat\Theta^{owa}$. That is, if

$$\hat\theta^{owa,*} = \operatorname*{arg\,min}_{\theta\in\hat\Theta^{owa}} \int_{\mathcal{X}\times\mathcal{Y}} \left(f(y, \mathbf{x}^\top\theta) + \lambda R(\theta)\right) d(\mathbf{x}, y), \tag{16}$$

then the rightmost term is the error of $\|\theta^* - \hat\theta^{owa,*}\|$. Clearly, as $m \to d$, the space $\hat\Theta^{owa} \to \Theta$, so $\hat\theta^{owa,*} \to \theta^*$, and $\|\theta^* - \hat\theta^{owa,*}\| \to 0$.

The leftmost term above captures the error due to using only n^{owa} data points in the second round of optimization. That is, the leftmost term captures the error $\|\hat\theta^{owa} - \hat\theta^{owa,*}\|$. As $n^{owa} \to n$, this error approaches the error of the oracle estimator trained on all the data $\|\theta^* - \hat\theta^{mle}\| = O(1/\sqrt{mn})$. By increasing m and n^{owa} our estimator is closer to the sequential oracle at the expense of a higher communication cost.

4.3.2. Formal Analysis

In this section we give a version of Theorem 2 with explicit constant factors. The proof is divided into two lemmas, which we present first because they introduce important notation. In the first lemma we bound the distance $\|\theta^* - \pi_{\hat{\Theta}^{owa}}\theta^*\|$, where $\pi_{\hat{\Theta}^{owa}}$ denotes the orthogonal projection onto $\hat{\Theta}^{owa}$. In the second lemma, we show that $\pi_{\hat{\Theta}^{owa}}\theta^* \approx \hat{\theta}^{owa}$.

Lemma 1. *Assume $\hat{\theta}_i^{mle}$ satisfies the SGT condition. Let $t > 0$ and*

$$\delta = 1 - \exp((d-m)(-t + \ln(t+1))). \tag{17}$$

Then we have that with probability at least δ,

$$\|\theta^* - \pi_{\hat{\Theta}^{owa}}\theta^*\| \leq (t+1)\sqrt{\frac{s_{min}}{s_{max}}\left(1 - \frac{m}{d}\right)}\|\theta^* - \hat{\theta}^{ave}\| \tag{18}$$

where s_{min} and s_{max} are the minimum and maximum eigenvalues of $\mathcal{I}_{\theta^*}^{-1}$.

The proof uses standard properties of projection matrices with sub-Gaussian columns. Due to space limitations, it is included in Appendix A of the supplement. Our next lemma shows that $\pi_{\hat{\Theta}^{owa}}\theta^* \approx \hat{\theta}^{owa}$.

Lemma 2. *Let $q_{hi}, q_{lo} : \mathbb{R}^+ \to \mathbb{R}^+$ be monotonically increasing functions such that for all points $\theta \in \Theta$,*

$$F(\theta^*) - F(\theta) \geq q_{lo}\left(\|\theta^* - \theta\|\right), \tag{19}$$

$$F(\theta^*) - F(\theta) \leq q_{hi}\left(\|\theta^* - \theta\|\right). \tag{20}$$

Then we have that

$$\begin{aligned}\|\theta^* - \hat{\theta}^{owa}\| \leq \|\hat{\theta}^{owa,*} - \hat{\theta}^{owa}\| \\ + q_{lo}^{-1}\left(q_{hi}\left(\|\theta^* - \pi_{\hat{\Theta}^{owa}}\theta^*\|\right)\right).\end{aligned} \tag{21}$$

The proof of Lemma 2 is a straightforward application of the triangle inequality. It is contained in Appendix B of the supplement. We are now ready to state a formal version of the informal Theorem 2.

Theorem 2 (Formal). *Assume that $\hat{\theta}_i^{mle}$ and α satisfy the SGT condition. Let $t > 0$ and*

$$\delta = (1 - \exp((d-m)(-t + \ln(t+1))))(1 - \exp(-t)). \tag{22}$$

Then, with probability at least δ,

$$\begin{aligned}&\|\theta^* - \hat{\theta}^{owa}\| \\ &\leq \|\hat{\theta}^{owa,*} - \mathbb{E}\hat{\theta}^{owa}\| + \sqrt{\frac{v_t}{mn^{owa}}} \\ &+ q_{lo}^{-1}\left(q_{hi}\left((t+1)\sqrt{\frac{s_{min}}{s_{max}}\left(1 - \frac{m}{d}\right)}\|\theta^* - \hat{\theta}^{ave}\|\right)\right)\end{aligned} \tag{23}$$

Proof. Bound the left term of Equation 21 using the procedure we used for Theorem 1. Then use Lemma 1 to bound $\|\theta^* - \pi_{\hat{\Theta}^{owa}}\theta^*\|$ in the right term. □

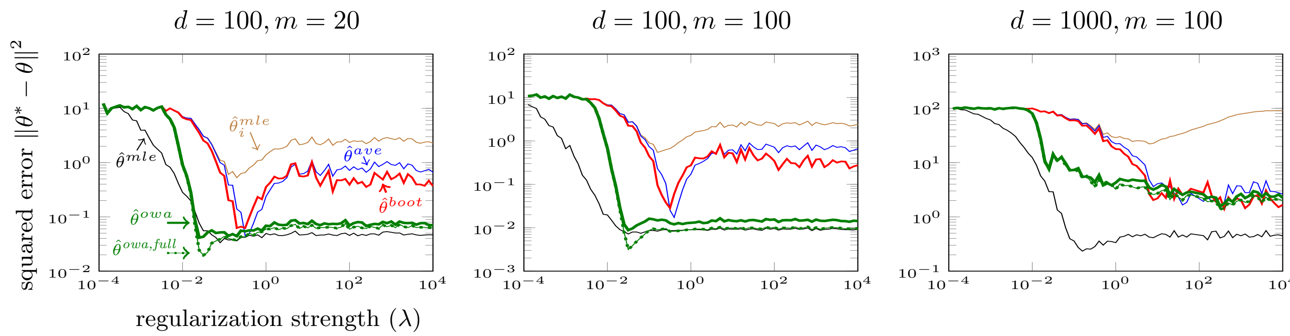

Figure 2. OWA is robust to the regularization strength. Surprisingly, additional regularization introduced by OWA lets it outperform the oracle estimator $\hat\theta^{mle}$ in some cases. Our theory states that as $m \to d$, $\hat\theta^{owa} \to \hat\theta^{mle}$. This is confirmed in the middle experiment. In the leftmost experiment, $m < d$, but $\hat\theta^{owa}$ still behaves similarly to $\hat\theta^{mle}$. In the rightmost experiment, $\hat\theta^{owa}$ has similar performance as $\hat\theta^{ave}$ and $\hat\theta^{boot}$ but is less sensitive to λ.

Notice that the formal version of Theorem 2 contains the term $\|\hat\theta^{owa,*} - \mathbb{E}\hat\theta^{owa}\|$, which does not appear in the informal version. This is the bias of the second round of optimization in the subspace $\hat\Theta^{owa}$ due to n^{owa} being finite. This bias will be less than the bias of the full problem because $\hat\Theta^{owa} \subset \Theta$. As $m \to \infty$, $\hat\theta^{owa,*} \to \theta^*$ and $\mathbb{E}\hat\theta^{owa} \to \hat\Theta^{owa}\hat\theta^{mle}$, so $\|\hat\theta^{owa,*} - \mathbb{E}\hat\theta^{owa}\| \to \|\theta^* - \mathbb{E}\hat\theta^{mle}\|$.

5. EXPERIMENTS

We evaluate OWA on two logistic regression tasks. The first task uses synthetic data. The second task uses real world ad-click data from the Tencent search engine. In each experiment, we compare our $\hat\theta^{owa}$ estimator with four baseline estimators: the naive estimator using the data from only a single machine $\hat\theta_i^{mle}$; the averaging estimator $\hat\theta^{ave}$; the bootstrap estimator $\hat\theta^{boot}$; and the oracle estimator of all data trained on a single machine $\hat\theta^{mle}$. The $\hat\theta^{boot}$ estimator has a parameter r that needs to be tuned. In all experiments we evaluate $\hat\theta^{boot}$ with $r \in \{0.005, 0.01, 0.02, 0.04, 0.1, 0.2\}$, which is a set recommended in the original paper (Zhang et al., 2012), and then report only the value of r with highest true likelihood. Thus we are reporting an overly optimistic estimate of the performance of $\hat\theta^{boot}$, and as we shall see our estimator $\hat\theta^{owa}$ still tends to perform better.

5.1. Synthetic Data

We generate the data according to the following sparse logistic regression model. Each component of the true parameter vector θ^* is sampled i.i.d. from a spike and slab distribution. With probability 0.9, the component is 0; with probability 0.1, the component is sampled from a standard normal distribution. The data points are then sampled as

$$\mathbf{x}_i \sim \mathcal{N}(0, I), \quad y_i = \left(1 + \exp(-\mathbf{x}_i^\mathsf{T}\theta^*)\right)^{-1}. \tag{24}$$

In all experiments, we use the L1 regularizer to induce sparsity in our estimates of θ^*. Our estimators will be biased because the model is misspecified. The true model for L1 regularization has a Laplace prior on θ^*.

Our first experiment shows the sensitivity of the estimators to the strength of regularization λ. In this experiment, λ was allowed to vary from 10^{-4} to 10^4. For each value of λ, we randomly generated 50 datasets (with $n = 1000$) and then calculated the corresponding estimators. Our $\hat\theta^{owa}$ estimator was trained with $n^{owa} = 128$. Figure 2 shows the average of the results for three choices of m and d. Our $\hat\theta^{owa}$ estimator is significantly less sensitive to the choice of λ than the other distributed estimators. Surprisingly, $\hat\theta^{owa}$ even outperforms the oracle $\hat\theta^{mle}$ in some regimes. This is likely due to additional regularization induced by the approximation in Equation 8.

Our second experiment shows how the parallel algorithms scale as the number of machines m increases. We fix $n = 1000$ data points per machine, so the size of the dataset mn grows as we add more machines. This simulates the typical "big data" regime where data is abundant, but processing resources are scarce. Each machine independently uses cross validation to select the λ that best fits the data locally. There are three advantages to this model selection procedure. First, there is no additional communication because model selection is a completely local task. Second, existing optimizers have built-in model selection routines which make the process easy to implement. We used the default model selection procedure from Python's SciKit-Learn (Pedregosa et al., 2011). Third, the data may be best fit using different regularization strengths for

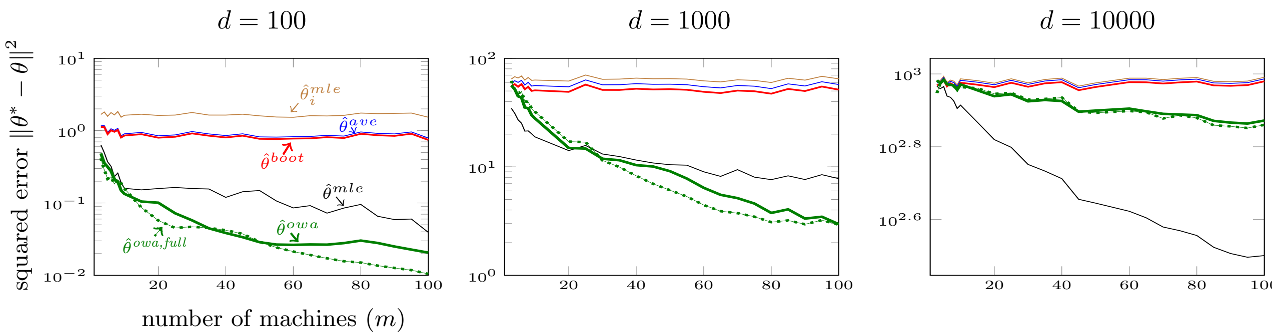

Figure 3. $\hat\theta^{owa}$ scales well with the number of machines. Surprisingly, it outperforms the oracle estimator trained on all of the data $\hat\theta^{mle}$ in some situations. This is likely due to the additional regularization introduced by the OWA algorithm, as seen in Figure 2.

each machine. The results are shown in Figure 3. The performance of $\hat\theta^{owa}$ scales much better than $\hat\theta^{ave}$ and $\hat\theta^{boot}$. Because of the regularization induced by Equation 8 observed in the previous experiment, $\hat\theta^{owa}$ even scales better than $\hat\theta^{mle}$ in some regimes.

5.2. Real World Advertising Data

We now evaluate our estimator on real world data from the KDD 2012 Cup (Niu et al., 2012). The goal is to predict whether a user will click on an ad from the Tencent internet search engine. This dataset was previously used to evaluate the performance of $\hat\theta^{boot}$ (Zhang et al., 2012). This dataset is too large to fit on a single machine. There are 235,582,879 distinct data points, each of dimension 741,725. The data points are sparse, so we use the L1 norm to encourage sparsity in our final solution. The regularization strength was set using cross validation in the same manner as for the synthetic data. For each test, we split the data into 80 percent training data and 20 percent test data. The training data is further subdivided into 128 partitions, one for each of the machines used.

Our first experiment tests the sensitivity of the n^{owa} parameter on large datasets. We fix $m = 128$, and allow n^{owa} to vary from 2^0 to 2^{20}, which is approximately the size of the full dataset. We repeated the experiment 50 times, each time using a different randomly selected set Z^{owa} for the second optimization. Figure 4 shows the results. Our $\hat\theta^{owa}$ estimator has lower loss than the $\hat\theta^{ave}$ using only 16 data points per machine (approximately 4×10^{-8} percent of the full training set) and $\hat\theta^{owa}$ has converged to its final loss value with only 1024 data points per machine (approximately 2.7×10^{-6} percent of the full training set). This justifies our claim that only a small number of data points are needed for the second round of optimization, and so the communication complexity of $\hat\theta^{owa}$ is essentially the same as $\hat\theta^{ave}$.

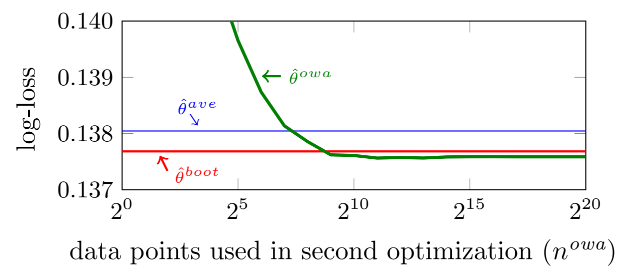

Figure 4. Relatively few data points are needed in the second round of optimization for $\hat\theta^{owa}$ to converge.

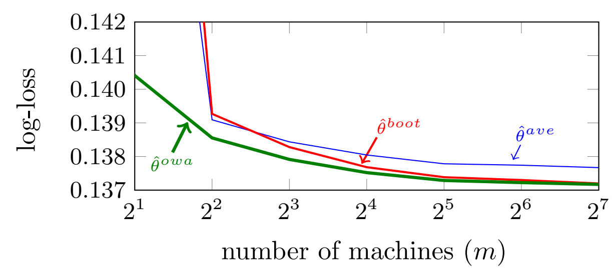

Figure 5. Performance of the parallel estimators on advertising data as the number of machines m increases.

Our last experiment shows the performance as we scale the number of machines m. The results are shown in Figure 5. Here, our $\hat\theta^{owa}$ performs especially well in the low m setting. For large m, $\hat\theta^{owa}$ continues to slightly outperform $\hat\theta^{boot}$ without the need for an expensive model selection procedure to determine the r parameter.

6. CONCLUSION

We introduced a new distributed estimation algorithm called OWA. OWA has the speed advantages of non-interactive distributed estimators, but has better accuracy due to a (cheap) second round of optimization. Unlike other algorithms, OWA does not require expensive hyperparameter tuning. Furthermore, our analysis is more general than the analysis of similar algorithms. As part of our analysis, we also provided a more general analysis of the averaging estimator.

References

Mark Braverman, Ankit Garg, Tengyu Ma, Huy L Nguyen, and David P Woodruff. Communication lower bounds for statistical estimation problems via a distributed data processing inequality. *arXiv preprint arXiv:1506.07216*, 2015.

Sanjoy Dasgupta and Anupam Gupta. An elementary proof of a theorem of Johnson and Lindenstrauss. *Random Structures & Algorithms*, 22(1): 60–65, 2003.

Ankit Garg, Tengyu Ma, and Huy Nguyen. On communication cost of distributed statistical estimation and dimensionality. In *Advances in Neural Information Processing Systems*, pages 2726–2734, 2014.

Jun Han and Qiang Liu. Bootstrap model aggregation for distributed statistical learning. *arXiv preprint arXiv:1607.01036*, 2016.

Daniel Hsu, Sham M Kakade, Tong Zhang, et al. A tail inequality for quadratic forms of subgaussian random vectors. *Electron. Commun. Probab*, 17(52): 1–6, 2012.

Erich Leo Lehmann. *Elements of large-sample theory.* Springer Science & Business Media, 1999.

Qiang Liu and Alexander T Ihler. Distributed estimation, information loss and exponential families. In *Advances in Neural Information Processing Systems*, pages 1098–1106, 2014.

Ryan McDonald, Mehryar Mohri, Nathan Silberman, Dan Walker, and Gideon S Mann. Efficient large-scale distributed training of conditional maximum entropy models. In *Advances in Neural Information Processing Systems*, pages 1231–1239, 2009.

Srujana Merugu and Joydeep Ghosh. Privacy-preserving distributed clustering using generative models. In *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, pages 211–218. IEEE, 2003.

Sahand Negahban, Bin Yu, Martin J Wainwright, and Pradeep K Ravikumar. A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. In *Advances in Neural Information Processing Systems*, pages 1348–1356, 2009.

Yanzhi Niu, Yi Wang, Gordon Sun, Aden Yue, Brian Dalessandro, Claudia Perlich, and Ben Hamner. The tencent dataset and kdd-cup'12. 2012.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830, 2011.

Ohad Shamir. Fundamental limits of online and distributed algorithms for statistical learning and estimation. In *Advances in Neural Information Processing Systems 27*, pages 163–171, 2014.

Vidyashankar Sivakumar, Arindam Banerjee, and Pradeep K Ravikumar. Beyond sub-gaussian measurements: High-dimensional structured estimation with sub-exponential designs. In *Advances in Neural Information Processing Systems*, pages 2206–2214, 2015.

Vladimir Spokoiny. Parametric estimation. finite sample theory. *The Annals of Statistics*, 40(6):2877–2909, 2012.

Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. In Y. Eldar and G. Kutyniok, editors, *Compressed Sensing, Theory and Applications*, chapter 5. 2012.

Yuchen Zhang, Martin J Wainwright, and John C Duchi. Communication-efficient algorithms for statistical optimization. In *Advances in Neural Information Processing Systems*, pages 1502–1510, 2012.

Yuchen Zhang, John Duchi, Michael I Jordan, and Martin J Wainwright. Information-theoretic lower bounds for distributed statistical estimation with communication constraints. In *Advances in Neural Information Processing Systems*, pages 2328–2336, 2013a.

Yuchen Zhang, John C Duchi, and Martin J Wainwright. Divide and conquer kernel ridge regression. In *COLT*, 2013b.

Martin Zinkevich, Markus Weimer, Lihong Li, and Alex J Smola. Parallelized stochastic gradient descent. In *Advances in neural information processing systems*, pages 2595–2603, 2010.

Appendix: Proof of Theorem 1

We have by the triangle inequality that

$$\|\theta^* - \hat\theta^{ave}\| \le \|\theta^* - \mathbb{E}\hat\theta^{ave}\| + \|\mathbb{E}\hat\theta^{ave} - \hat\theta^{ave}\|. \quad (25)$$

The left term above captures the estimator's bias and the right term the variance. First we consider the bias. By the linearity of expectation, we have that

$$\mathbb{E}\hat\theta^{ave} = \mathbb{E}\frac{1}{m}\sum_{i=1}^m \hat\theta_i^{mle} = \frac{1}{m}\sum_{i=1}^m \mathbb{E}\hat\theta_i^{mle} = \mathbb{E}\hat\theta_i^{mle}, \quad (26)$$

and so the bias term $\|\theta^* - \mathbb{E}\hat\theta^{ave}\| = \|\theta^* - \mathbb{E}\hat\theta_i^{mle}\|$.

Now we consider the variance. We have that

$$\begin{aligned}
&\left\|\hat\theta^{ave} - \mathbb{E}\hat\theta^{ave}\right\| && (27)\\
&= \left\|\frac{1}{m}\sum_{i=1}^m \hat\theta_i^{mle} - \mathbb{E}\hat\theta^{ave}\right\| && (28)\\
&= \frac{1}{m}\left\|\sum_{i=1}^m \left(\hat\theta_i^{mle} - \mathbb{E}\hat\theta_i^{mle}\right)\right\| && (29)\\
&= \frac{1}{m\sqrt{n}}\left\|\mathcal{I}_{\theta^*}^{-1/2}\sum_{i=1}^m \sqrt{n}\mathcal{I}_{\theta^*}^{1/2}\left(\hat\theta_i^{mle} - \mathbb{E}\hat\theta_i^{mle}\right)\right\| && (30)\\
&= \frac{1}{m\sqrt{n}}\left\|\mathcal{I}_{\theta^*}^{-1/2}\sum_{i=1}^m \Delta_{\hat\theta_i^{mle}}\right\|. && (31)
\end{aligned}$$

Notice that each of the $\Delta_{\hat\theta_i^{mle}}$ are i.i.d. sub-Gaussian random vectors. Sub-Gaussian random vectors inherit the following basic property from centered Gaussian random vectors: The sum of m i.i.d. sub-Gaussians has the same distribution as $\sqrt{m}$ times the sub-Gaussian. That is, $\sum_{i=1}^m \Delta_{\hat\theta_i^{mle}} \sim \sqrt{m}\Delta_{\hat\theta_1^{mle}}$. Substituting into Equation 31 gives

$$\left\|\hat\theta^{ave} - \mathbb{E}\hat\theta^{ave}\right\| \sim \frac{1}{\sqrt{mn}}\left\|\mathcal{I}_{\theta^*}^{1/2}\Delta_{\hat\theta_1^{mle}}\right\|. \quad (32)$$

Our last step is to bound the norm on the right hand side of Equation 32. Theorem 2.1 from (Hsu et al., 2012) gives a bound on the norm of a positive semidefinite matrix times a sub-Gaussian random variable. Applying the theorem gives that

$$\Pr\left[\left\|\mathcal{I}_{\theta^*}^{-1/2}\Delta_1\right\| < \sqrt{v_t}\right] \ge 1 - \exp(-t), \quad (33)$$

where v_t is defined as in Equation 15. And so the variance of $\hat\theta^{ave}$ satisfies

$$\Pr\left[\left\|\hat\theta^{ave} - \mathbb{E}\hat\theta^{ave}\right\| < \sqrt{\frac{v_t}{mn}}\right] \ge 1 - \exp(-t). \quad (34)$$

Substituting Equations 26 and 34 into Equation 25 gives the stated result.

Appendix A: Proof of Lemma 1

Define the space $\Gamma = \operatorname{span}\{\hat\theta_i^{mle} - \mathbb{E}\hat\theta^{ave}\}_{i=1}^m$. Our proof strategy is to bound the distance $\|\theta^* - \pi_\Gamma\theta^*\|$ and then show that $\Gamma \approx \hat\Theta^{owa}$. Specifically, we have

$$\begin{aligned}
&\|\theta^* - \pi_{\hat\Theta^{owa}}\theta^*\| \\
&\quad= \|(\theta^* - \hat\theta^{ave}) - (\pi_{\hat\Theta^{owa}}\theta^* - \hat\theta^{ave})\| && (35)\\
&\quad= \|(\theta^* - \hat\theta^{ave}) - \pi_{\hat\Theta^{owa}}(\theta^* - \hat\theta^{ave})\| && (36)\\
&\quad\le \|(\theta^* - \hat\theta^{ave}) - \pi_{\hat\Theta^{owa}}\pi_\Gamma(\theta^* - \hat\theta^{ave})\| && (37)\\
&\quad\le \|(\theta^* - \hat\theta^{ave}) - \pi_\Gamma(\theta^* - \hat\theta^{ave})\| \\
&\qquad + \|\pi_\Gamma(\theta^* - \hat\theta^{ave}) - \pi_{\hat\Theta^{owa}}\pi_\Gamma(\theta^* - \hat\theta^{ave})\|. && (38)
\end{aligned}$$

Equation 36 follows because $\hat\theta^{ave} \in \hat\Theta^{owa}$; Equation 37 by the definition of $\pi_{\hat\Theta^{owa}}$; and Equation 38 by the triangle inequality. We now bound each of the terms in Equation 38 separately, beginning with the leftmost.

Define the matrix G with ith column equal to the vector $(\hat\theta_i^{mle} - \mathbb{E}\hat\theta_i^{mle})$. The column space of G is Γ, and we have that

$$\begin{aligned}
\pi_\Gamma &= GG^\dagger && (39)\\
&= \mathcal{I}_{\theta^*}^{-1/2}\left(\sqrt{n}\mathcal{I}_{\theta^*}^{1/2}G\right)\left(\sqrt{n}\mathcal{I}_{\theta^*}^{1/2}G\right)^\dagger\mathcal{I}_{\theta^*}^{1/2}, && (40)
\end{aligned}$$

where the superscript † denotes the Moore-Penrose pseudoinverse. The ith column of matrix $\sqrt{n}\mathcal{I}_{\theta^*}^{1/2}G$ is equal to $\Delta_{\hat\theta_i^{mle}}$ by definition. Let $U\Sigma V^\top$ be the singular value decomposition of $\sqrt{n}\mathcal{I}_{\theta^*}^{1/2}G$, where U and V are orthogonal matrices and Σ is the diagonal matrix of singular values. Substituting into Equation 40 gives

$$\begin{aligned}
\pi_\Gamma &= \mathcal{I}^{-1/2}(U\Sigma V^\top)(U\Sigma V^\top)^\dagger\mathcal{I}^{1/2} && (41)\\
&= \mathcal{I}^{-1/2}(U\Sigma V^\top)(V\Sigma^\dagger U^\top)\mathcal{I}^{1/2} && (42)\\
&= \mathcal{I}^{-1/2}U\Sigma\Sigma^\dagger U^\top\mathcal{I}^{1/2}. && (43)
\end{aligned}$$

This implies that

$$\begin{aligned}
&\|(I - \pi_\Gamma)(\theta^* - \hat\theta^{ave})\| \\
&\quad= \|(I - \mathcal{I}^{-1/2}U\Sigma\Sigma^\dagger U^\top\mathcal{I}^{1/2})(\theta^* - \hat\theta^{ave})\| && (44)\\
&\quad= \|\mathcal{I}^{-1/2}U(I - \Sigma\Sigma^\dagger)U^\top\mathcal{I}^{1/2}(\theta^* - \hat\theta^{ave})\| && (45)\\
&\quad= \sqrt{s_{\min}}\|U(I - \Sigma\Sigma^\dagger)U^\top\mathcal{I}^{1/2}(\theta^* - \hat\theta^{ave})\| && (46)\\
&\quad= \sqrt{s_{\min}}\|(I - \Sigma\Sigma^\dagger)U^\top\mathcal{I}^{1/2}(\theta^* - \hat\theta^{ave})\|. && (47)
\end{aligned}$$

Equation 46 follows from the definition of $s_{\min}$ as the smallest eigenvector of $\mathcal{I}$, and Equation 47 follows from the rotational invariance of the L2 norm. The matrix $(I - \Sigma\Sigma^\dagger)$ contains 1s in the first $d-m$ diagonal entries, and zeros everywhere else. By the rotational invariance of the sub-Gaussian distribution (Lemma 5.9 of Vershynin, 2011), U and V are distributed uniformly over the orthogonal group of matrices. Therefore the operator $(I - \Sigma\Sigma^{\dagger})U^{\mathsf{T}}$ projects a vector onto a subspace of dimension $d-m$ chosen uniformly at random. Lemma 2.2 from (Dasgupta and Gupta, 2003) provides a bound for this situation. The lemma states that for any vector $\mathbf{x}$, for all $t > 0$, with probability at least $1 - \exp((d-m)(-t + \ln(t+1)))$,

$$\|(I - \Sigma\Sigma^{\dagger})U^{\mathsf{T}}\mathbf{x}\| \leq (t+1)\sqrt{1 - \frac{m}{d}}. \tag{48}$$

Setting $\mathbf{x} = \mathcal{I}^{1/2}(\theta^* - \hat{\theta}^{ave})$ gives

$$\begin{aligned} &\|(I - \pi_{\Gamma})(\theta^* - \hat{\theta}^{ave})\| \\ &\quad\leq (t+1)\sqrt{s_{\min}\left(1 - \frac{m}{d}\right)}\|\mathcal{I}^{1/2}(\theta^* - \hat{\theta}^{ave})\| \quad (49) \\ &\quad\leq (t+1)\sqrt{\frac{s_{\min}}{s_{\max}}\left(1 - \frac{m}{d}\right)}\|\theta^* - \hat{\theta}^{ave}\|, \quad (50) \end{aligned}$$

which bounds the leftmost term of Equation 38.

Appendix B: Proof of Lemma 2

By the triangle inequality, we have that

$$\|\theta^* - \hat{\theta}^{owa}\| \leq \|\hat{\theta}^{owa,*} - \hat{\theta}^{owa}\| + \|\theta^* - \hat{\theta}^{owa,*}\|. \tag{51}$$

We bound the right hand term as follows. By the definition of q_{hi} and q_{lo}, we have

$$\begin{aligned} q_{\text{lo}}\left(\|\theta^* - \hat{\theta}^{owa,*}\|\right) &\leq F(\theta^*) - F(\hat{\theta}^{owa,*}) && (52) \\ &\leq F(\theta^*) - F(\pi_{\hat{\Theta}^{owa}}\theta^*) && (53) \\ &\leq q_{\text{hi}}\left(\|\theta^* - \pi_{\hat{\Theta}^{owa}}\theta^*\|\right). && (54) \end{aligned}$$

Applying q_{lo}^{-1} to both sides and substituting into Equation 51 gives the stated result.