# PROVABLY AUDITING ORDINARY LEAST SQUARES IN LOW DIMENSIONS

**Ankur Moitra & Dhruv Rohatgi**
Massachusetts Institute of Technology
{moitra, drohatgi}@mit.edu

## ABSTRACT

Auditing the stability of a machine learning model to small changes in the training procedure is critical for engendering trust in practical applications. For example, a model should not be overly sensitive to removing a small fraction of its training data. However, algorithmically validating this property seems computationally challenging, even for the simplest of models: Ordinary Least Squares (OLS) linear regression. Concretely, recent work defines the stability of a regression as the minimum number of samples that need to be removed so that rerunning the analysis overturns the conclusion (Broderick et al., 2020), specifically meaning that the sign of a particular coefficient of the OLS regressor changes. But the only known approach for estimating this metric, besides the obvious exponential-time algorithm, is a greedy heuristic that may produce severe overestimates and therefore cannot certify stability. We show that stability can be efficiently certified in the *low-dimensional* regime: when the number of covariates is a constant but the number of samples is large, there are polynomial-time algorithms for estimating (a fractional version of) stability, with provable approximation guarantees. Applying our algorithms to the Boston Housing dataset, we exhibit regression analyses where our estimator outperforms the greedy heuristic, and can successfully certify stability even in the regime where a constant fraction of the samples are dropped.

## 1 INTRODUCTION

A key facet of interpretability of machine learning models is understanding how different subsets of the training data influence the learned model and its predictions. Computing the influences of individual training points has been shown to be a useful tool for enhancing trust in the model (Zhou et al., 2019), for tracing the origins of model bias (Brunet et al., 2019), and for identifying mislabelled training data and other model debugging (Koh & Liang, 2017). Modelling the influence of groups of training points has applications to measuring fairness (Chen et al., 2018), vulnerability to contamination of multi-source training data (Hayes & Ohrimenko, 2018), and (most relevant to this paper) identification of unstable predictions (Ilyas et al., 2022) and models (Broderick et al., 2020). In a high-stakes machine learning application, it would likely be alarming if some data points were so influential that the removal of, say, $1\%$ of the training data dramatically changed the model. An ideal, trustworthy machine learning pipeline therefore should include validation that this does not happen.

But the obvious algorithm for checking if a model trained on $n$ data points exhibits this instability would require computing the group influences of $\binom{n}{n/100}$ different subsets of the data, which is computationally infeasible even for fairly small $n$. Instead, current methods for estimating the stability of a model simply use the first-order approximation of group influence: namely, the sum of individual influences of data points in the group. With this approximation, vulnerability of a model to dropping $\alpha n$ data points is heuristically estimated by dropping the $\alpha n$ most individually influential data points (Broderick et al., 2020; Ilyas et al., 2022).

This heuristic can be thought of as using "local" stability as a proxy for "global" stability, and it has found substantial anecdotal success in diagnosing *unstable* models. Unfortunately, for correlated groups of data points, the first-order approximation of the group influence is often an underestimate (Koh et al., 2019), so large local stability does not actually certify that a model is provably *stable* to

removing small subsets of data. In fact, stability certification is a challenging and open problem even in the simplest of models: linear regression via Ordinary Least Squares (OLS).

Concretely, given a regression dataset, a natural metric for the stability of the OLS regressor is the minimum number of data points that need to be removed from the dataset to flip the sign of a particular coefficient of the regressor (e.g., in causal inference, the coefficient measuring the treatment effect). Recent work has used the local stability heuristic to diagnose unstable OLS regressions in several prominent economics studies (Broderick et al., 2020), identifying examples where even a statistically significant conclusion can be overturned by removing less than $1\%$ of the data points. However, the converse question of validating stable conclusions remains unaddressed:

*Given a regression dataset, can we efficiently certify non-trivial lower bounds on the stability of the OLS regressor?*

Our work takes steps towards addressing this question, via the following contributions:

- We introduce a natural fractional relaxation of the above notion of OLS stability, where we allow removing fractions of data points, and seek to minimize the total removed weight. We call this *finite-sample stability*, and henceforth refer to the prior notion as "integral" stability.
- We develop approximation algorithms for the finite-sample stability, with **(a)** provable guarantees under reasonable anti-concentration assumptions on the dataset, and **(b)** running time polynomial in the size of the dataset, so long as the dimension of the data is a constant (in contrast, the naive algorithm is exponential in the size of the dataset). Moreover, we prove that (at least for exact algorithms) exponential dependence of the running time on the dimension is unavoidable under standard complexity assumptions.
- We use modifications of our algorithms to compute *assumption-free upper and lower bounds* on the finite-sample stability of several simple synthetic and real datasets, achieving tighter upper bounds than prior work and the first non-trivial lower bounds, i.e. certifications that the OLS regressor is stable.

**Why define stability this way?** The definition of integral stability was introduced in (Broderick et al., 2020), along with several variants (e.g. smallest perturbation which causes the first coordinate to lose significance). We choose the definition based on the sign of the first coordinate, because it has clear practical interpretation—does the first covariate positively or negatively affect the response?—which does not depend on choice of additional parameters such as significance level.

We study the fractional relaxation so that the stability is defined by a continuous optimization problem. Note that certifying a lower bound on fractional stability immediately certifies a lower bound on the integral stability; we will see later (Remark 3.1) that a near-converse holds in low dimensions.

**Why is low-dimensional regression important?** Given that much of machine learning happens in high-dimensional settings, where the number of covariates can even be larger than the number of datapoints, it is natural to wonder why low-dimensional settings are still important. First, in application areas such as econometrics, linear regressions with as few as two to four covariates are very common (Britto et al., 2022; Bianchi & Bigio, 2022; Hopenhayn et al., 2022), often serving as proofs-of-concept for more complex models. Second, even in settings where the number of covariates is larger, it is often the expectation that few covariates are relevant. In such applications, analysis often consists of a variable selection step followed by regression on a much-reduced set of covariates (Cai & Wang, 2011). In all these settings, understanding the stability of an estimator is important, and our work gives some of the first provable guarantees that avoid making strong distributional assumptions. Moreover our lower bounds show that certifying stability of truly high-dimensional models, even linear ones, is intractable.

### 1.1 Formal problem statement

We are given a deterministic and arbitrary set of $n$ samples $(X_i, y_i)_{i=1}^n$, where each $X_i$ is a vector of $d$ real-valued *covariates*, and each $y_i$ is a real-valued *response*. We are interested in a single coefficient of the OLS regressor (without loss of generality, the first coordinate): in an application, the first covariate may be the treatment and the rest may be controls. The sign of this coefficient

is important because it estimates whether the treatment has a positive or negative effect. Thus, we want to determine if it can be changed by dropping a few samples from the regression. Formally, we consider the fractional relaxation, where we allow dropping fractions of samples:

**Definition 1.1.** Fix $(X_i, y_i)_{i=1}^n$ with $X_1, \ldots, X_n \in \mathbb{R}^d$ and $y_1, \ldots, y_n \in \mathbb{R}$. For any $w \in [0,1]^n$, the *weight-$w$ OLS solution set* of $(X_i, y_i)_{i=1}^n$ is

$$\mathsf{OLS}(X, y, w) := \arg\min_{\beta \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^n w_i (\langle X_i, \beta \rangle - y_i)^2.$$

The *finite-sample stability* of $(X_i, y_i)_{i=1}^n$ is

$$\mathrm{Stability}(X, y) := \inf_{w \in [0,1]^n, \beta \in \mathbb{R}^d} \{n - \|w\|_1 : \beta_1 = 0 \text{ and } \beta \in \mathsf{OLS}(X, y, w)\}.$$

This is the minimum number of samples (in a fractional sense) which need to be removed to zero out the first coordinate of the OLS regressor. If the OLS solution set contains multiple regressors, then it suffices if any regressor $\beta$ in the solution set has $\beta_1 = 0$. Our algorithmic goal is to compute $\mathrm{Stability}(X, y)$, or at least to approximate $\mathrm{Stability}(X, y)$ up to an additive $\epsilon n$ error.

### 1.2 Results

By brute-force search, the (integral) stability can be computed in time $2^n \cdot \mathrm{poly}(n)$. However, because the complexity is exponential in the number of samples, it is computationally infeasible even when the dimension $d$ of the data is low, which is a common situation in many scientific applications. Similarly, the fractional stability (Definition 1.1) is the solution to a non-convex optimization problem in more than $n$ variables, which seems no simpler. Can we still hope for a *polynomial-time* algorithm in constant dimensions? We show that the answer is yes.

**Theorem 1.2.** *There is an $n^{O(d^3)}$-time algorithm which, given $n$ arbitrary samples $(X_i, y_i)_{i=1}^n$ with $X_1, \ldots, X_n \in \mathbb{R}^d$ and $y_1, \ldots, y_n \in \mathbb{R}$, and given $k \geq 0$, decides whether $\mathrm{Stability}(X, y) \leq k$.*

We also show that the exponential dependence on dimension $d$ is necessary under standard complexity assumptions:

**Theorem 1.3.** *Under the Exponential Time Hypothesis, there is no $n^{o(d)}$-time algorithm which, given $(X_i, y_i)_{i=1}^n$ and $k \geq 0$, decides whether $\mathrm{Stability}(X, y) \leq k$.*

This theorem in particular rules out fixed-parameter tractability, i.e. algorithms with time complexity $f(d) \cdot \mathrm{poly}(n)$. However, it only applies to *exact* algorithms. In practice, it is unlikely to matter whether $\mathrm{Stability}(X, y) = 0.01n$ or $\mathrm{Stability}(X, y) = 0.02n$; in both cases, the conclusion is sensitive to dropping a very small fraction of the data. This motivates our next two algorithmic results on $\epsilon n$-additive approximation of the stability (where we think of $\epsilon > 0$ as a constant). First, we make a mild anti-concentration assumption, under which the stability can $\epsilon n$-approximated in time roughly $n^{d+O(1)}$. While still not fixed-parameter tractable, this algorithm can now be run on moderate sized problems in low dimensions, unlike the algorithm in Theorem 1.2.

**Assumption A.** *Let $\epsilon, \delta > 0$. We say that samples $(X_i, y_i)_{i=1}^n$ satisfy $(\epsilon, \delta)$-anti-concentration if for every $\beta \in \mathbb{R}^d$, it holds that*

$$\left| \left\{ i \in [n] : |\langle X_i, \beta \rangle - y_i| < \frac{\delta}{\sqrt{n}} \left\| X\beta^{(0)} - y \right\|_2 \right\} \right| \leq \epsilon n,$$

*where $X : n \times d$ is the matrix with rows $X_1, \ldots, X_n$, and $\beta^{(0)} \in \mathsf{OLS}(X, y, \mathbb{1})$ is any unweighted OLS regressor of $y$ against $X$.*

See Appendix F.1 for discussion of when Assumption A holds. Under this assumption, we present an $O(\epsilon n)$-approximation algorithm:

**Theorem 1.4.** *For any $\epsilon, \delta, \eta > 0$, there is an algorithm* PartitionAndApprox *with time complexity*

$$\left( n + \frac{Cd}{\epsilon^2} \log \frac{1}{\delta} \log \frac{1}{\epsilon\eta} \right)^{d+O(1)}$$

*which, given $\epsilon$, $\delta$, $\eta$, and samples $(X_i, y_i)_{i=1}^n$ satisfying $(\epsilon, \delta)$-anti-concentration, returns an estimate $\hat{S}$ such that with probability at least $1 - \eta$,*

$$|\hat{S} - \text{Stability}(X, y)| \leq 12\epsilon n + 1.$$

In fact, PARTITIONANDAPPROX also can detect when Assumption A fails (see Theorem D.6 for a precise statement), so it can be used to compute unconditional lower bounds on stability with high probability (where the lower bound is provably tight if the data satisfies anti-concentration). Moreover, as discussed in Appendix F.1, the required anti-concentration is very mild. If $\epsilon, \eta > 0$ are constants, the algorithm has time complexity $n^{d+O(1)}$, so long as the samples satisfy $(\epsilon, \exp(-\Omega(n)))$-anti-concentration. This is true for arbitrary *smoothed* data. Finally, unlike the exact algorithm, PARTITIONANDAPPROX avoids heavy algorithmic machinery; it only requires solving linear programs.

**Fixed-parameter tractability?** Our final result is that $\epsilon n$-approximation of the stability is in fact fixed-parameter tractable, under a stronger anti-concentration assumption.

**Assumption B.** *Let $\epsilon, \delta > 0$. We say that samples $(X_i, y_i)_{i=1}^n$ satisfy $(\epsilon, \delta)$-strong anti-concentration if for every $\beta \in \mathbb{R}^{d+1}$, it holds that*

$$\left| \left\{ i \in [n] : |\langle \overline{X}_i, \beta \rangle| < \frac{\delta}{\sqrt{n}} \left\| \overline{X}\beta \right\|_2 \right\} \right| \leq \epsilon n$$

*where $\overline{X} : n \times (d+1)$ is the matrix with columns $(X^T)_1, \ldots, (X^T)_d, y$.*

This assumption holds with constant $\epsilon, \delta > 0$ under certain distributional assumptions on $(X_i, y_i)_{i=1}^n$, e.g. centered Gaussian mixtures with uniformly bounded condition number (Appendix F.2).

**Theorem 1.5.** *For any $\epsilon, \delta > 0$, there is a $(\sqrt{d}/(\epsilon\delta^2))^d \cdot \text{poly}(n)$-time algorithm* NETAPPROX *which, given $\epsilon$,$\delta$, and samples $(X_i, y_i)_{i=1}^n$ satisfying $(\epsilon, \delta)$-strong anti-concentration, returns an estimate $\hat{S}$ satisfying $\text{Stability}(X, y) \leq \hat{S} \leq \text{Stability}(X, y) + 3\epsilon n + 1$. Moreover, $\text{Stability}(X, y) \leq \hat{S}$ holds for arbitrary $(X_i, y_i)_{i=1}^n$.*

**Extensions.** Another model, frequently used in causal inference and econometrics, is *instrumental variables (IV) linear regression.* When the noise $\eta$ in a hypothesized causal relationship $y = \langle X, \beta^* \rangle + \eta$ is believed to be endogenous (i.e. correlated with $X$), a common approach (Sargan, 1958; Angrist et al., 1996; Card, 2001) is to find a $p$-dimensional variable $Z$ (the *instrument*) for which domain knowledge suggests that $\mathbb{E}[\eta | Z] = 0$. Positing that $\beta^*$ is identified by the moment condition $\mathbb{E}[Z(y - \langle X, \beta \rangle)] = 0$, the IV estimator set given samples $(X_i, y_i, Z_i)_{i=1}^n$ is then

$$\mathsf{IV}(X, y, Z) = \{\beta \in \mathbb{R}^d : Z^T(w \star (X\beta - y) = 0\}$$

where $a \star b$ denotes elementwise product, and $Z : n \times p$ and $X : n \times d$ are the matrices of instruments and covariates respectively. Stability can be defined as in Definition 1.1. Although for simplicity we state all of our results for OLS (i.e. the special case $Z = X$), it can be seen that Theorem 1.2 and Theorem 1.5 both extend directly to the IV regression setting. See Appendix G for further discussion.

**Experiments.** We implement modifications of NETAPPROX and PARTITIONANDAPPROX which give *unconditional*, *exact* upper and lower bounds on stability, respectively. We use these algorithms to obtain tight data-dependent bounds on stability of isotropic Gaussian datasets for a broad range of signal-to-noise ratios, and we demonstrate heterogeneous synthetic datasets where our algorithms' upper bounds are an order of magnitude better than upper bounds obtained by the prior heuristic. On the Boston Housing dataset (Harrison Jr & Rubinfeld, 1978), we regress house values against all pairs of features. For the majority of these regressions, we bound the stability within a factor of two. On the one hand, we detect many sensitive conclusions (including some which the greedy heuristic claims are stable); on the other hand, we certify that some conclusions are stable to dropping as much as half the dataset.

### 1.3 ORGANIZATION

In Section 2 we review related work. In Section 3 we collect notation and formulas that will be useful later. In Section 4 we sketch the intuition behind our algorithmic results. Section 5 covers our experiments. In Appendices B, C, D, and E we prove Theorems 1.2, 1.3, 1.4, and 1.5 respectively.

## 2 RELATED WORK

There is a rich literature on topics related to finite-sample stability, including sensitivity analysis and robustness to distribution shift and contamination. Due to space constraints, here we only discuss the works most relevant to ours, and we postpone broader discussion to Appendix A.

Most directly related is the prior work on heuristics for the (integral) stability (Broderick et al., 2020; Kuschnig et al., 2021). The heuristic given by Broderick et al. (2020) (to approximate the most-influential $k$ samples) is simply the *local* approximation: compute the local influence of each sample at $w = \mathbb{1}$, sort the samples from largest to smallest influence, and output the top $k$ samples. Subsequent work (Kuschnig et al., 2021) refines this heuristic by recomputing the influences after removing each sample, which alleviates issues such as masking (Chatterjee & Hadi, 1986). But this is still just a greedy heuristic, and it may fail when samples are jointly but not individually influential. Except under the strong assumption that the sample covariance remains nearly constant when we remove any $\epsilon n$ samples (see Theorem 1 in Broderick et al. (2020), which relies on Condition 1 in Giordano et al. (2019)), the local influence approach can upper bound the finite-sample stability but cannot provably lower bound it. In fact, in Section 5 we provide examples where the greedy heuristic of Kuschnig et al. (2021) is very inaccurate due to instability in the sample covariance.

Closely related to finite-sample stability, the $s$-value (Gupta & Rothenhäusler, 2021) is the minimum Kullback-Leibler divergence $D(P||P_0)$ over all distributions $P$ for which the conclusion is null, where $P_0$ is the empirical distribution of the samples. Unfortunately, while the $s$-value is an interesting and well-motivated metric, computing the $s$-value for OLS estimation appears to be computationally intractable, and the algorithms given by Gupta & Rothenhäusler (2021) lack provable guarantees.

## 3 PRELIMINARIES

For vectors $u, v \in \mathbb{R}^m$, we let $u \star v$ denote the elementwise product $(u \star v)_i = u_i v_i$. Throughout the paper, we will frequently use the closed-form expression for the (weighted) OLS solution set

$$\mathsf{OLS}(X, y, w) = \{\beta \in \mathbb{R}^d : X^T(w \star (X\beta - y)) = 0\}$$

where $X : n \times d$ is the matrix with rows $X_1, \ldots, X_n$. In particular, setting $\lambda = \beta_{2:d}$, this means that the finite-sample stability can be rewritten as

$$\text{Stability}(X, y) = \inf_{w \in [0,1]^n, \lambda \in \mathbb{R}^{d-1}} \{n - \|w\|_1 : X^T(w \star (\tilde{X}\lambda - y)) = 0\} \tag{1}$$

where (here and throughout the paper) $\tilde{X} : n \times (d-1)$ is the matrix with columns $(X^T)_2, \ldots, (X^T)_d$.

**Remark 3.1.** As previously mentioned, the finite-sample stability always lower bounds the integral stability (the minimum number of samples that need to be removed to make the first coordinate of the regressor change sign), by continuity of the OLS solution set in $w$. Additionally, it can be seen from Equation 1 that a partial converse holds in low dimensions. For any feasible $(w, \lambda)$, the set of $w'$ such that **(a)** $(w', \lambda)$ is feasible, and **(b)** $\|w\|_1 = \|w'\|_1$, has the form $[0,1]^n \cap V$ for some subspace $V \subseteq \mathbb{R}^n$ of codimension at most $d+1$. Thus, there is some $w' \in [0,1]^n \cap V$ with at most $d+1$ non-integral weights. If $\text{Stability}(X, y) = \alpha n$, then $w'$ witnesses that the first coordinate of the OLS regressor can be zeroed out by downweighting at most $\alpha n + d + 1$ samples.

## 4 OVERVIEW OF ALGORITHMS

**An exact algorithm.** Our main tool for Theorem 1.2 is the following special case of an important result due to Renegar (1992) on solving quantified polynomial systems of inequalities:

**Theorem 4.1** (Renegar (1992))**.** *Given an expression*

$$\forall x \in \mathbb{R}^{n_1} : \exists y \in \mathbb{R}^{n_2} : P(x, y),$$

*where $P(x, y)$ is a system of $m$ polynomial inequalities with maximum degree $d$, the truth value of the expression can be decided in time $(md)^{O(n_1 n_2)}$.*[1]

[1] This is in the real number model; a similar statement can be made in the bit complexity model.

Roughly, for a constant number of quantifier alternations, a quantified polynomial system can be decided in time exponential in the number of variables. Unfortunately, a naive formulation of the expression $\mathrm{Stability}(X, y) \leq k$, by direct evaluation of Equation 1, has $n + d - 1$ variables:

$$\exists \lambda \in \mathbb{R}^{d-1}, w \in [0,1]^n : \sum_{i=1}^{n} w_i \geq n - k \text{ and } X^T(w \star (\tilde{X}\lambda - y)) = 0.$$

Intuitively, it may not be necessary to search over all $w \in [0,1]^n$; for fixed $\lambda$, the maximum-weight $w$ is described by a simple linear program. Formally, the linear program can be rewritten (Lemma B.1) by the separating hyperplane theorem, so that the overall expression becomes:

$$\exists \lambda \in \mathbb{R}^{d-1} : \forall u \in \mathbb{R}^d : \exists w \in [0,1]^n : \|w\|_1 \geq n - k \text{ and } \sum_{i=1}^{n} w_i(\langle \tilde{X}_i, \lambda \rangle - y_i)\langle X_i, u \rangle \geq 0. \quad (2)$$

Now, for fixed $\lambda$ and $u$, the maximum-weight $w$ has very simple description: it only depends on the *relative ordering* of the $n$ summands $(\langle \tilde{X}, \lambda \rangle - y_i)\langle X_i, u \rangle$. By classical results on connected components of varieties, since the summands have only $2d - 1$ variables, the number of achievable orderings is only $n^{\Omega(d)}$ rather than $n!$, and the orderings can be enumerated efficiently (Milnor, 1964; Renegar, 1992). This allows the quantifier over $w \in [0,1]^n$ to be replaced by a quantifier over the $n^{\Omega(d)}$ achievable orderings, after which Theorem 4.1 implies that the overall expression can be decided in time $n^{\Omega(d^3)}$. See Appendix B for details.

**Approximation via partitioning.** Next, we show how to avoid the heavy algorithmic machinery used in the previous result. For Theorem 1.4, the strategy of PARTITIONANDAPPROX is to partition the OLS solution space $\mathbb{R}^{d-1}$ into roughly $n^d$ regions, such that if we restrict $\lambda$ to any one region, the bilinear program which defines the stability can be approximated by a linear program.

Concretely, we start by writing the formulation (1) as

$$n - \mathrm{Stability}(x, y) = \sup_{w \in [0,1]^n, \lambda \in \mathbb{R}^{d-1}} \left\{ \sum_{i \in [n]} w_i \,\middle|\, X^T(w \star (\tilde{X}\lambda - y)) = 0 \right\}. \quad (3)$$

This has a nonlinear (and nonconvex) constraint due to the pointwise product between $w$ and the residual vector $\tilde{X}\lambda - y$. Thus, we can introduce the change of variables $g_i = w_i(\langle \tilde{X}_i, \lambda \rangle - y_i)$ for $i \in [n]$. This causes two issues. First, the constraint $0 \leq w_i \leq 1$ becomes $0 \leq g_i/(\langle X_i, \lambda \rangle - y_i) \leq 1$, which is no longer linear. To fix this, suppose that instead of maximizing over all $\lambda \in \mathbb{R}^{d-1}$, we maximize over a region $R \subseteq \mathbb{R}^{d-1}$ where each residual $\langle \tilde{X}, \lambda \rangle - y_i$ has constant sign $\sigma_i$. The constraint $0 \leq w_i \leq 1$ then becomes one of two linear constraints, depending on $\sigma_i$. Let $V_R$ denote the value of Program 3 restricted to $\lambda \in R$. Then with the change of variables, we have that

$$V_R = \sup_{g \in \mathbb{R}^n, \lambda \in R} \left\{ \sum_{i \in [n]} \frac{g_i}{\langle X_i, \lambda \rangle - y_i} \,\middle|\, \begin{array}{ll} X^T g = 0 & \\ 0 \leq g_i \leq \langle X_i, \lambda \rangle - y_i & \forall i \in [n] : \sigma_i = 1 \\ \langle X_i, \lambda \rangle - y_i \leq g_i \leq 0 & \forall i \in [n] : \sigma_i = -1 \end{array} \right\},$$

with the convention that $0/0 = 1$. Now the constraints are linear. Unfortunately, (and this is the second issue), the objective is no longer linear. The solution is to partition the region $R$ further: if the region were small enough that every residual $\langle X_i, \lambda \rangle - y_i$ had at most $(1 \pm \epsilon)$-multiplicative variation, then the objective could be approximated to within $1 \pm \epsilon$ by a linear objective.

How many regions do we need? Let $M = \left\| X\beta^{(0)} - y \right\|_2$ be the unweighted OLS error. If all the residuals were bounded between $\delta M/\sqrt{n}$ and $M$ in magnitude, for all $\lambda \in \mathbb{R}^{d-1}$, then the regions could be demarcated by $O(n \log_{1+\epsilon}(n/\delta))$ hyperplanes, for a total of $O(n \log_{1+\epsilon}(n/\delta))^d$ regions. Of course, for some $\lambda$, some residuals may be very small or very large. But $(\epsilon, \delta)$-anti-concentration implies that for every $\lambda$, at most $\epsilon n$ residuals are very small, and it can be shown that if $\lambda$ is a weighted OLS solution, the total weight on samples with large residuals is low. Thus, for any region, we can exclude from the objective function the samples with residuals that are not well-approximated within the region, and this only affects the objective by $O(\epsilon n)$.

This gives an algorithm with time complexity $(n\epsilon^{-1} \log(1/\delta))^{d+O(1)}$. To achieve the time complexity in Theorem 1.4, where the $\log(n/\delta)$ is additive rather than multiplicative, we use subsampling. Every

residual is still partitioned by sign, but we multiplicatively partition only a random $\tilde{O}(d/\epsilon)$-size subset of the residuals. Intuitively, most residuals will still be well-approximated in any given region. This can roughly be formalized via a VC dimension argument, albeit with some technical complications. See Section D for details and Appendix J for formal pseudocode of the algorithm.

**Net-based approximation.** The algorithm NETAPPROX for Theorem 1.5 is intuitively the simplest. For any fixed $\lambda \in \mathbb{R}^{d-1}$, Program 1 reduces to a linear program with value denoted $S(\lambda)$. Thus, an obvious approach is to construct a net $\mathcal{N} \subseteq \mathbb{R}^{d-1}$ in some appropriate metric, and compute $\min_{\lambda \in \mathcal{N}} S(\lambda)$. This always upper bounds the stability, but to prove that it's an approximate lower bound, we need $S(\lambda)$ to be Lipschitz under the metric.

The right metric turns out to be

$$d(\lambda, \lambda') = \left\| \frac{\tilde{X}\lambda - y}{\left\|\tilde{X}\lambda - y\right\|_2} - \frac{\tilde{X}\lambda' - y}{\left\|\tilde{X}\lambda' - y\right\|_2} \right\|_2 .$$

Under this metric, $\mathbb{R}^{d-1}$ essentially embeds into a $d$-dimensional subspace of the Euclidean sphere $\mathcal{S}^{n-1}$, and therefore has a $\gamma$-net of size $O(1/\gamma)^d$. Why is $S(\lambda)$ Lipschitz under $d$? First, if $\tilde{X}\lambda - y$ equals $\tilde{X}\lambda' - y$ up to rescaling, then it can be seen from Program 3 that $S(\lambda) = S(\lambda')$. More generally, if the residuals are close up to rescaling, we apply the dual formulation of $S(\lambda)$ from expression (2):

$$n - S(\lambda) = \inf_{u \in \mathbb{R}^d} \sup_{w \in [0,1]^n} \|w\|_1 : \sum_{i=1}^{n} w_i(\langle \tilde{X}_i, \lambda \rangle - y_i)\langle X_i, u \rangle \geq 0.$$

For any $u$, the optimal $w$ for $\lambda$ and $u$ can be rounded to some feasible $w'$ for $\lambda'$ and $u$ without decreasing the $\ell_1$ norm too much, under strong anti-concentration. This shows that $S(\lambda)$ and $S(\lambda')$ are close. See Appendix E for details and Appendix J for formal pseudocode of NETAPPROX.

## 5 EXPERIMENTS

In this section, we apply (modifications of) NETAPPROX and PARTITIONANDAPPROX to several datasets in two and three dimensions. Due to space constraints, we defer detailed discussion of the algorithmic modifications to Appendix I.1; we simply note that the modifications are made to improve practical efficiency and usability. Most saliently, the modified algorithms do not rely on Assumptions A and B: the modified NETAPPROX provides an unconditional *upper bound* on stability (referred to henceforth as "net upper bound"), and the modified PARTITIONANDAPPROX provides an unconditional *lower bound* ("LP lower bound"). As a result, we are able to experimentally verify that our algorithms provide tight (and unconditional) bounds on stability for a variety of datasets.

As a baseline upper bound, we implement the greedy heuristic of Kuschnig et al. (2021) which refines Broderick et al. (2020). We are not aware of any prior work on lower bounding stability, so we implement a simplification of our full lower bound algorithm as a baseline. See Appendix I for implementation details, hyperparameter choices, and discussion of error bars.

### 5.1 SYNTHETIC DATA

**Heterogeneous data.** We start with a simple two-dimensional dataset with two disparate subpopulations, where the greedy baseline fails to estimate the stability but our algorithms give tight estimates. For parameters $n$, $k$, and $\sigma$, we generate $k$ independent samples $(X_i, y_i)$, where $X_i \in \mathbb{R}^2$ has independent coordinates $X_{i1} \sim N(-1, 0.01)$ and $X_{i2} \sim N(0, 1)$, and $y_i = X_{i1}$. Then, we generate $n - k$ independent samples $(X_i, y_i)$ where $X_{i1} = 0$ and $X_{i2} \sim N(0, 1)$, and $y \sim N(0, 1)$. It always suffices to remove the first subpopulation, so the stability is at most $k$. However, the first subpopulation has small individual influences, because the OLS regressor on the whole dataset can nearly interpolate the first subpopulation. Thus, we expect that the greedy algorithm will fail to notice the first subpopulation, and therefore remove far more than $k$ samples.

Indeed, this is what happens. For $n = 1000$ and $k$ varying from 10 to 500, we compare our net upper bound and LP lower bound with the baselines. As seen in Figure 1, our methods are always better

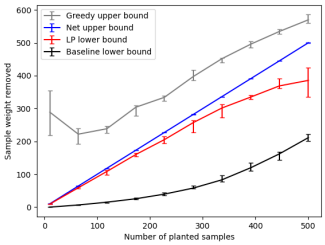
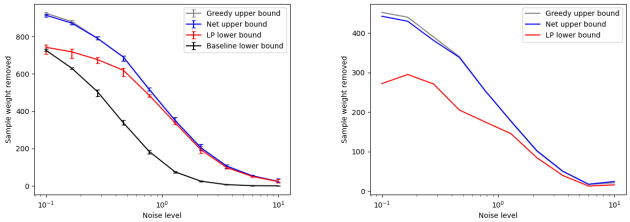



Figure 1: Heterogeneous data




(a) $d = 2$ and $n = 1000$; median (b) $d = 3$ and $n = 500$; one trial of 10 trials for each noise level and for each noise level and algorithm algorithm

Figure 2: Isotropic Gaussian data

than the baselines, and certifiably approximate the stability within a small constant factor. In the regime where $k$ is small, our upper bound outperforms the greedy upper bound by a factor of 30.

**Covariance shift.** In the previous example, removing $k$ samples caused a pathological change in the sample covariance; it became singular. However, even modest, constant-factor instability in the sample covariance can cause the greedy algorithm to fail; see Appendix I.5 for details.

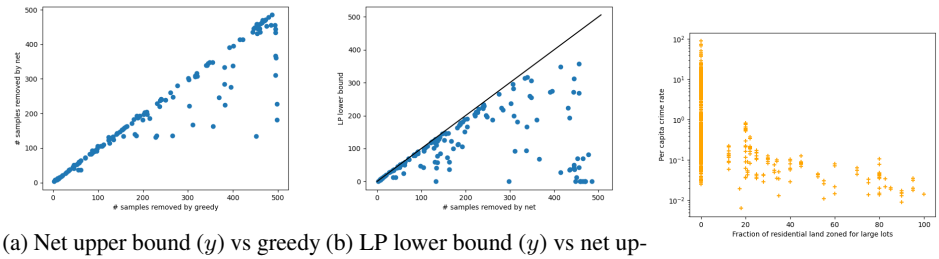
**Isotropic Gaussian data.** Instability can arise even in homogeneous data, as a result of a low signal-to-noise ratio (Broderick et al., 2020). But when the noise level is low, can we certify stability? For a broad range of noise levels, we experimentally show that this is the case. Specifically, for $d \in \{2, 3\}$ and noise parameter $\sigma$ ranging from 0.1 to 10, we generate $n$ independent samples $(X_i, y_i)_{i=1}^n$ where $X_i \sim N(0, I_d)$ and $y_i = \langle X_i, \mathbb{1} \rangle + N(0, \sigma^2)$. For $d = 2$ and $n = 1000$ (Figure 2a), our LP lower bound is nearly tight with the upper bounds, particularly as the noise level increases (in comparison, the baseline lower bound quickly degenerates towards zero). For $d = 3$ and $n = 500$ (Figure 2b), the bounds are looser for small noise levels but still always within a small constant factor.

### 5.2 Boston Housing dataset

The Boston Housing dataset (Harrison Jr & Rubinfeld, 1978; Gilley et al., 1996) consists of data from 506 census tracts of Greater Boston in the 1970 Census. There are 14 real-valued features, one of which—the median house value in USD 1000s—we designate as the response. Unfortunately the entire set of features is too large for our algorithms, so for our experiments we pick various subsets of two or three features to use as covariates.

**A Tale of Two Datasets.** We exemplify our results with two particular feature subsets. First, we investigate the effect of zn (percentage of residential land zoned for large lots) on house values, controlling for rm (average number of rooms per home) and rad (highway accessibility index) but no bias term. On the entire dataset, we find a modest positive effect: the estimated coefficient of zn is roughly 0.06. Both the greedy heuristic and our net algorithm find subsets of just 8% of the data (38-40 samples) which, if removed, would nullify the effect. But is this tight, or could there be a much smaller subset with the same effect? Our LP lower bound **certifies that removing at least** 22.4 **samples is necessary**.

Second, we investigate the effect of zn on house values, this time controlling only for crim (per capita crime rate). Our net algorithm finds a subset of just 27% of the data which was driving the effect, and the LP lower bound certifies that the stability is at least 8%. But this time, the greedy algorithm removes 90% of the samples, a clear failure. What happened? Plotting zn against crim reveals a striking heterogeneity in the data: 73% of the samples have zn $= 0$, and the remaining 27% of the samples (precisely those removed by the net algorithm) have crim $< 0.83$, i.e. very low crime rates. As in the synthetic example, this heterogeneity explains the greedy algorithm's failure. But heterogeneity is very common in real data: in this case, it's between the city proper and the suburbs, and in fact the OLS regressors of these two subpopulations on all 13 features are markedly different




(a) Net upper bound ($y$) vs greedy upper bound ($x$)

(b) LP lower bound ($y$) vs net upper bound ($x$)

(c) crim ($y$) vs zn ($x$)

Figure 3: Results from Boston Housing dataset. Figure (a) plots the net upper bounds on the $y$-axis against the greedy upper bounds on the $x$-axis; Figure (b) plots the LP lower bounds on the $y$-axis against the net upper bounds on the $x$-axis. In both (a) and (b), each mark corresponds to one of the 156 feature pairs. Figure (c) plots the feature zn against the feature crim (on log scale); each mark is one of the 506 datapoints.

(Appendix I.6). Thus, it's important to have algorithms with provable guarantees for detecting when heterogeneity causes (or doesn't cause) unstable conclusions.

**All-feature-pairs analysis.** To be thorough, we also apply our algorithms to all 156 ordered pairs of features. For each pair, we regress the response (i.e. median house value) against the two features by Ordinary Least Squares, and we use our algorithms on this 2-dimensional dataset to estimate how many samples need to be removed to nullify the effect of the first feature on the response. We also compare to the greedy upper bound. See Figure 3 for a perspective on the results. In each figure, each point corresponds to the results of one dataset. The left figure plots the net upper bound against the greedy upper bound: we can see that our net algorithm substantially outperforms the greedy heuristic on some datasets (i.e. finds a much smaller upper bound) and never performs much worse. The right figure plots the LP lower bound against the net upper bound (along with the line $y = x$). For a majority of the datasets, the upper bound and lower bound are close. Concretely, for 116 of the 156 datasets, **we certifiably estimate the stability up to a factor of two** – some are sensitive to removing less than 10 samples, and some are stable to removing even a majority of the samples.

## 6 Conclusions

In this work, we studied efficient estimation of the stability of OLS regressions to removing subsets of the training data. We showed that in low dimensions the problem is both theoretically and experimentally tractable, whereas in high dimensions exact computation of the stability likely requires exponential time. However, this is only the beginning of the story. Most immediately, since our lower bound algorithm takes time $n^{\Omega(d)}$, our experiments were limited to no more than three dimensions. Certifying stability of OLS regressions from e.g. recent econometric studies may require additional heuristics or insights (e.g. developing a fixed-parameter tractable lower bound algorithm). Beyond that, identifying reasonable assumptions under which exponential dependence on dimension can be entirely circumvented is another valuable direction for future work.

Of course, machine learning extends far beyond linear regression, and for more and more complex and opaque models, stability certification is all the more crucial as a tool for enhancing trustworthiness. Certainly, OLS is important in its own right, but inasmuch as it is a key building block in more complex machine learning systems (from regression trees (Loh, 2011) to generative adversarial networks (Mao et al., 2017) and policy iteration in linear MDPs (Lagoudakis & Parr, 2003)), our work on estimating stability of OLS is also a first step towards estimating stability for these systems.

Finally, we remark that care must be taken when interpreting stability in practice. Large stability may increase trust in a model's parameters or predictions, but it does not mean that conclusions drawn from the model are "correct." Conversely, even if the stability is small, the conclusions may still be useful, with the caveat that they may be driven by a small sub-population. Understanding whether this heterogeneity is problematic or not is context-dependent, and is a separate but important issue.